SnapDrive® 3.0 for UNIX®
Installation and Administration Guide (IBM® AIX®, HP-UX®, Linux®, Solaris™)

Network Appliance, Inc.
495 East Java Drive
Sunnyvale, CA 94089 USA
Telephone: +1 (408) 822-6000
Fax: +1 (408) 822-4501
Support telephone: +1 (888) 4-NETAPP
Documentation comments: doccomments@netapp.com
Information Web: http://www.netapp.com

Part number 210-03726_B0
June 2007

Copyright and trademark information

Copyright information

Copyright © 1994–2007 Network Appliance, Inc. All rights reserved. Printed in the U.S.A.

No part of this document covered by copyright may be reproduced in any form or by any means—graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system—without prior written permission of the copyright owner.

Software derived from copyrighted Network Appliance material is subject to the following license and disclaimer:

THIS SOFTWARE IS PROVIDED BY NETWORK APPLIANCE "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL NETWORK APPLIANCE BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Network Appliance reserves the right to change any products described herein at any time, and without notice. Network Appliance assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by Network Appliance. The use or purchase of this product does not convey a license under any patent rights, trademark rights, or any other intellectual property rights of Network Appliance.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

RESTRICTED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.277-7103 (October 1988) and FAR 52-227-19 (June 1987).

Trademark information

NetApp, the Network Appliance logo, the bolt design, NetApp–the Network Appliance Company, DataFabric, Data ONTAP, FAServer, FilerView, FlexVol, Manage ONTAP, MultiStore, NearStore, NetCache, SecureShare, SnapDrive, SnapLock, SnapManager, SnapMirror, SnapMover, SnapRestore, SnapValidator, SnapVault, Spinnaker Networks, SpinCluster, SpinFS, SpinHA, SpinMove, SpinServer, SyncMirror, Topio, VFM, and WAFL are registered trademarks of Network Appliance, Inc. in the U.S.A. and/or other countries. Cryptainer, Cryptoshred, Datafort, and Decru are registered trademarks, and Lifetime Key Management and OpenKey are trademarks, of Decru, a Network Appliance, Inc. company, in the U.S.A. and/or other countries. gFiler, Network Appliance, SnapCopy, Snapshot, and The evolution of storage are trademarks of Network Appliance, Inc. in the U.S.A. and/or other countries and registered trademarks in some other countries. ApplianceWatch, BareMetal, Camera-to-Viewer, ComplianceClock, ComplianceJournal, ContentDirector, ContentFabric, EdgeFiler, FlexClone, FlexShare, FPolicy, HyperSAN, InfoFabric, LockVault, NOW, NOW NetApp on the Web, ONTAPI, RAID-DP, RoboCache, RoboFiler, SecureAdmin, Serving Data by Design, SharedStorage, Simplicore, Simulate ONTAP, Smart SAN, SnapCache, SnapDirector, SnapFilter, SnapMigrator, SnapSuite, SohoFiler, SpinMirror, SpinRestore, SpinShot, SpinStor,

StoreVault, vFiler, Virtual File Manager, VPolicy, and Web Filer are trademarks of Network Appliance, Inc. in the United States and other countries. NetApp Availability Assurance and NetApp ProTech Expert are service marks of Network Appliance, Inc. in the U.S.A.

Apple is a registered trademark and QuickTime is a trademark of Apple Computer, Inc. in the United States and/or other countries. Microsoft is a registered trademark and Windows Media is a trademark of Microsoft Corporation in the United States and/or other countries. RealAudio, RealNetworks, RealPlayer, RealSystem, RealText, and RealVideo are registered trademarks and RealMedia, RealProxy, and SureStream are trademarks of RealNetworks, Inc. in the United States and/or other countries.

All other brands or products are trademarks or registered trademarks of their respective holders and should be treated as such.

Network Appliance is a licensee of the CompactFlash and CF Logo trademarks.

Network Appliance NetCache is certified RealSystem compatible.

Table of Contents

Preface ix

Chapter 1 **Overview of SnapDrive for UNIX** 1

How SnapDrive for UNIX works. 2

Comparing SnapDrive for UNIX on different host platforms 9

Where to go for more information 12

Chapter 2 **Preparing to Install SnapDrive for UNIX** 15

Prerequisites for using SnapDrive for UNIX 16

Preparing storage systems 18

Preparing hosts 23

Methods for executing SnapDrive for UNIX. 27

Chapter 3 **Installing and Upgrading SnapDrive for UNIX** 29

Installing SnapDrive for UNIX on an AIX host 30

Uninstalling SnapDrive for UNIX from an AIX host 40

Installing SnapDrive for UNIX on an HP-UX host 46

Uninstalling SnapDrive for UNIX from an HP-UX host 52

Installing SnapDrive for UNIX on a Linux host 54

Uninstalling SnapDrive for UNIX from a Linux host 60

Installing SnapDrive for UNIX on a Solaris host 61

Uninstalling SnapDrive for UNIX from a Solaris host 72

Completing the installation 73

Understanding the files installed by SnapDrive for UNIX on the host. 74

Upgrading to a new version of SnapDrive for UNIX 79

Verifying the Veritas stack configuration 81

Chapter 4 **Configuring and Using SnapDrive for UNIX** 83
Setting configuration information 84
Preparing hosts for adding LUNs. 120
Setting up audit, recovery, and trace logging. 122
Setting up AutoSupport. 130
Setting up multipathing 133
Setting up thin provisioning 140
General steps for executing commands. 142

Chapter 5 **Setting Up Security Features** 145
Overview of SnapDrive for UNIX security features 146
Setting up access control 147
Viewing the current access control settings 152
Specifying the current login information for storage systems 154
Disabling SSL encryption 157

Chapter 6 **Provisioning and Managing Storage** 159
Overview of storage provisioning 160
Creating storage. 163
Displaying information about storage 177
Increasing the size of the storage 192
Connecting LUNs and storage entities to the host 198
Disconnecting LUN mappings from the host. 207
Connecting only the host side of storage 216
Disconnecting only the host side of storage 221
Deleting storage from the host and storage system. 228

Chapter 7 **Creating and Using Snapshot Copies** 235
Overview of Snapshot operations 236
Creating Snapshot copies 238

Displaying information about Snapshot copies 249
Renaming a Snapshot copy 257
Restoring a Snapshot copy 259
Connecting to a Snapshot copy 272
Disconnecting a Snapshot copy 288
Deleting a Snapshot Copy 294

Chapter 8 **Troubleshooting** 297
Data collection utility 298
Understanding error messages 301
Common error messages 303
Standard exit status values 326

Chapter 9 **Command Reference** 339
Collecting information needed by SnapDrive for UNIX commands 340
Summary of the SnapDrive for UNIX commands 341
SnapDrive for UNIX options, keywords, and arguments 348

Glossary 367

Index 373

Preface

About this guide

This document describes how to install, configure, and operate SnapDrive® 3.0 software for UNIX® servers. It does not cover basic system or network administration topics, such as IP addressing, routing, and network topology. It also does not cover topics that are handled in the Network Appliance™ FCP Host Utilities (Attach Kit) or iSCSI Host Utilities (Support Kit) documentation.

Note

The products FCP Host Attach Kit and iSCSI Support Kit are named as FCP Host Utilities and iSCSI Host Utilities. The latest information about SnapDrive for UNIX and its requirements is in the SnapDrive for UNIX *Compatibility Matrix* on the NOW™ NetApp on the Web™ site (http://now.netapp.com/NOW/knowledge/docs/docs.shtml).

Audience

This guide is for system administrators who possess working knowledge of NetApp® storage systems.

This guide assumes that you are familiar with the following topics:

- Fibre Channel Protocol (FCP)
- Internet Small Computer System Interface (iSCSI) Protocol
- Basic network functions and operations
- UNIX servers
- UNIX security
- Data storage array administration concepts
- NetApp storage system management
- Logical volume manager on the system you are using

Command conventions

You can use a Telnet session to enter the storage system commands on the system console or from any client that can obtain access to the storage system. In examples that illustrate commands executed on a UNIX workstation, the command syntax and output might differ, depending on your version of UNIX. The examples might be from any of the hosts supported for SnapDrive for UNIX.

Terminology

This guide uses the following terms:

- *LUN* (logical unit number) refers to a logical unit of storage identified by a number.
- *LUN ID* refers to the numerical identifier for a LUN.
- *Filer* refers to a NetApp storage system that incorporates the CompactFlash™ unit. Storage systems can support FCP and iSCSI.
- Some UNIX operating systems use the term *disk group* while others use the term *volume group*. This guide uses the term *disk group* to refer to both disk groups and volume groups.
- Some operating systems refer to *host volumes* while others refer to *logical volumes*. This guide uses the term *host volume* to refer to both host volumes and logical volumes.

Formatting conventions

The following table lists different character formats used in this guide to set off special information.

Formatting convention	Type of information
Italic type	◆ Words or characters that require special attention. ◆ Placeholders for information you must supply. For example, if the guide requires you to enter the `fctest` *adaptername* command, you enter the characters "fctest" followed by the actual name of the adapter. ◆ Book titles in cross-references.
`Monospaced` font	◆ Command and daemon names. ◆ Information displayed on the system console or other computer monitors. ◆ The contents of files.
`Bold monospaced` font	Words or characters you type. What you type is always shown in lowercase letters, unless your program is case-sensitive and uppercase letters are necessary for it to work properly.

Keyboard conventions

This guide uses capitalization and some abbreviations to refer to the keys on the keyboard. The keys on your keyboard might not be labeled exactly as they are in this guide.

What is in this guide…	What it is used to mean…
hyphen (-)	Separates individual keys. For example, Ctrl-D means holding down the Ctrl key while pressing the D key.
Enter	Refers to the key that generates a carriage return; the key is named Return on some keyboards.
type	Pressing one or more keys on the keyboard.
enter	Pressing one or more keys and then pressing the Enter key.

Special messages

This guide contains special messages that are described as follows:

Note

A note contains important information that helps you install or operate the system efficiently.

CAUTION

A caution contains instructions that you must follow to avoid damage to the equipment, a system crash, or loss of data.

WARNING

A warning contains instructions that you must follow to avoid personal injury.

Overview of SnapDrive for UNIX 1

About this chapter

This chapter provides information about how SnapDrive for UNIX works and what it does.

Topics in this chapter

This chapter includes the following topics:

- "How SnapDrive for UNIX works" on page 2
- "Comparing SnapDrive for UNIX on different host platforms" on page 9
- "Where to go for more information" on page 12

How SnapDrive for UNIX works

What SnapDrive for UNIX does

SnapDrive for UNIX is a tool that simplifies the backup of data so that you can recover it should it be accidentally deleted or modified. SnapDrive for UNIX uses Snapshot™ technology to create an image (that is, a Snapshot copy) of the data on a shared and unshared storage system attached to a UNIX host at a specific time. If the need arises later, you can restore the data to the storage system. When you restore a Snapshot copy, it replaces the current data on the storage system with the image of the data in the Snapshot copy.

In addition, SnapDrive for UNIX lets you automate storage provisioning tasks on the storage system to manage both node-local file systems (or disk groups or LUNs) and cluster-wide shared file systems, in a Veritas™ Storage Foundation for Oracle Real Application Clusters (SFRAC) 4.1 environment. SnapDrive for UNIX provides a number of storage features that enable you to manage the entire storage hierarchy, from the host-side application-visible file, to the volume manager, to the storage-system-side logical unit numbers (LUNs) providing the actual repository.

On nonclustered UNIX systems: With SnapDrive for UNIX installed on nonclustered UNIX systems, you can perform the following tasks:

- Create a Snapshot copy of one or more volume groups on a storage system. The Snapshot copy can contain file systems, logical volumes, disk groups, LUNs, and NFS directory trees. After you create a Snapshot copy, you can rename it, restore it, or delete it. You can also connect it to a different location on the same host or to a different host. After you connect it, you can view and modify the content of the Snapshot copy, or you can disconnect the Snapshot copy. In addition, SnapDrive for UNIX lets you display information about Snapshot copies that you created.
- Create storage that includes LUNs, file systems, logical volumes, and disk groups. After it is created, you can increase the storage or delete it. You can also connect the storage to a host or disconnect it. In addition, SnapDrive for UNIX lets you display information about the storage that you created.

On clustered UNIX systems: With SnapDrive for UNIX installed on clustered UNIX systems, you can perform the following tasks:

- Create storage for a cluster-wide shared storage system that includes disk groups, file systems, and LUNs that are visible from all nodes in the cluster.

- Conduct Snapshot operations on a cluster-wide shared storage system that includes disk groups and file systems. The Snapshot operations include create, rename, restore, connect, disconnect, display, and delete.

Note

The SnapDrive for UNIX operations for clustered systems are available only for Veritas SFRAC 4.1 on a Solaris host. For more information on setting up a cluster environment, see Chapter 3, "Setting up an SFRAC 4.1 I/O fencing environment on a storage system," on page 64.

How SnapDrive for UNIX works on a clustered UNIX system

SnapDrive for UNIX provides storage provisioning and Snapshot copy management options to manage a cluster-wide shared storage system that includes disk groups and file systems in an SFRAC 4.1 environment on a Solaris host. All SnapDrive for UNIX operations are allowed from any node in the cluster. The SnapDrive for UNIX operations can be executed from the following modes:

- The cluster master node, where the command is executed locally on the cluster-master node.
- The noncluster-master node, where the command is sent to the master node and executed.

For this, you have to ensure that the `rsh` or `ssh` access-without-password-prompt for the root user is configured for all nodes in the cluster. For more information on enabling access-without-password-prompt, see Chapter 4, "Determining options and their default values," on page 86.

How SnapDrive for UNIX works on a vFiler unit

SnapDrive for UNIX does not distinguish between a physical storage system and a vFiler™ unit. Therefore, there will not be any change in the I/O parameters of Snapshot and storage operations.

When working on a vFiler unit, consider the following:

- SnapDrive for UNIX does not support access to vFiler units through FCP, because this process is not supported by Data ONTAP.
- SnapDrive for UNIX provides storage provisioning operations, Snapshot operations, host operations, and configuration operations on a vFiler unit only if the vFiler unit is created on a FlexVol volume. These operations are not supported on a vFiler unit that is created on a qtree, because Snapshot operations are disallowed unless the vFiler unit owns the entire storage volume.

- Application data should not be stored in the root volume of the vFiler unit.
- Snapshot operations are not supported on a vFiler unit, if the root of the vFiler unit is a qtree.
- You have to set the value of the Data ONTAP 7.2.2 configuration option `vfiler.vol_clone_zapi_allow` to "on" in order to connect to a Snapshot copy of a volume or LUN in a vFiler unit.

How SnapDrive for UNIX manages storage

SnapDrive for UNIX storage commands help you provision and manage NetApp storage when you create storage entities.

Managing LVM entities: If you request a storage operation that provisions an LVM entity, like a disk group that includes host volumes or file systems, the `snapdrive storage` commands work with the LVM to create the LVM objects and file systems that use the storage. During the storage provision operation, the following actions occur:

- The host LVM combines LUNs from a storage system into disk or volume groups. This storage is then divided into logical volumes, which are used as if they were raw disk devices to hold file systems or raw data.
- SnapDrive for UNIX integrates with the host LVM to determine which NetApp LUNs make up each disk group, host volume, and file system requested for Snapshot copy. Because data from any given host volume can be distributed across all disks in the disk group, Snapshot copies can be made and restored only for whole disk groups.

Managing raw entities: If you request a storage operation for a raw entity, like a LUN, or a file system that is created directly on a LUN, SnapDrive for UNIX performs the storage operation without using the host system LVM. SnapDrive for UNIX allows you to create, delete, connect, and disconnect LUNs, and the file systems that they contain, without activating the LVM.

How SnapDrive for UNIX creates Snapshot copies

SnapDrive for UNIX software provides commands that you enter on the host that create, restore, and manage Snapshot copies of NetApp storage entities. You can use SnapDrive for UNIX commands to create, restore, and manage Snapshot copies of the following:

- Logical Volume Manager (LVM) entities— Disk groups with host volumes and file systems that you create using the host LVM.
- Raw entities— LUNs or LUNs that contain file systems without creating any volumes or disk groups. These are mapped directly to the host.
- Network File System (NFS) entities— NFS files and directory trees.

Note

NFS entities are not supported on clustered systems.

The Snapshot copy that you create can span multiple storage systems and storage system volumes. SnapDrive for UNIX checks the read/writes against the storage entities in the Snapshot copy to ensure that all Snapshot data is crash-consistent. SnapDrive for UNIX will not create a Snapshot copy unless the data is crash-consistent. For detailed information about crash consistency, see "Crash-consistent Snapshot copies" on page 238.

Host communications

SnapDrive for UNIX communicates with the storage system using the host IP interface that you specified when you set up the storage system.

Security considerations

You must log in as a root user to use the SnapDrive for UNIX commands.

To enable SnapDrive for UNIX to access the storage systems connected to the host, you must configure it to use the login names and passwords assigned to the storage systems when you set them up. If you do not provide this information, SnapDrive for UNIX cannot communicate with the storage system. For information on supplying the login names and passwords, see "Specifying the current login information for storage systems" on page 154.

SnapDrive for UNIX stores this information on the host in an encrypted file. On AIX, Solaris, and Linux hosts, by default, SnapDrive for UNIX encrypts the password information it sends out across the network. It communicates using HTTPS over the standard IP connection. For more information about the security features, see "Overview of SnapDrive for UNIX security features" on page 146.

Attention

On HP-UX hosts, SnapDrive 3.0 for UNIX uses HTTP protocol to communicate to storage systems.

In a Solaris SFRAC 4.1 cluster environment, you have to configure `rsh` or `ssh` access-without-password-prompt-for-root among all nodes in the cluster.

Using access permissions on a storage system

SnapDrive for UNIX lets you specify access permissions for each host in a file that resides on the storage system. These permissions indicate whether a host can perform certain Snapshot copy and storage operations. They do not affect any of the show or list operations. For more information on types of access permissions, see "Setting up access control" on page 147.

You can also control what action SnapDrive for UNIX takes when it does not find a permission file for a given host, by the value you set in the snapdrive.conf configuration file for the `all-access-if-rbac-unspecified` variable. The options are to allow all access to that storage system or to disable all access to it. For more information on the configuration file, see "Setting configuration information" on page 84.

SnapDrive for UNIX stack

SnapDrive for UNIX requires the following stack:

- Host operating system and appropriate patches
  - AIX
  - HP-UX
  - Linux

  Note

  Linux includes Red Hat® Enterprise Linux, SUSE® Linux, and Oracle® Enterprise Linux.

  - Solaris
- Host file systems
  - AIX: JFS/JFS2 or VxFS
  - HP-UX: VxFS
  - Linux: Ext3
  - Solaris: VxFS or UNIX File System (UFS)
- NFS
- Volume manager
  - AIX: LVM or VxVM
  - HP-UX: LVM or VxVM
  - Linux: LVM1 or LVM2
  - Solaris: VxVM
- FCP Host Utilities or iSCSI Host Utilities required software

  For example, if you are using SnapDrive for UNIX with an AIX host, you must set up the features required by the FCP Host Utilities for that host when

you are using multipathing. If your configuration uses multipathing, it means you must install the Dot Hill® SANpath® software, which the FCP Host Utilities uses to handle multipathing.

- Storage system licenses
  - FCP, iSCSI, or NFS license, depending on your configuration
  - FlexClone® license (NFS configurations only)
  - SnapRestore® license on the storage system
- Data ONTAP® software on your storage system
- Network Appliance™ MultiStore® software on your storage system for vFiler unit setup.
- Internet Protocol (IP) access between the host and storage system

NetApp adds new attach utilities and components on an ongoing basis. To keep up with these changes, NetApp has set up matrices that contain the most up-to-date information for using NetApp products in a SAN environment. The following matrices contain information about the SnapDrive for UNIX system requirements:

- SnapDrive for UNIX Compatibility Matrix
- Compatibility and Configuration Guide for NetApp's FCP and iSCSI Products

  The preceding information is available on the NOW site (http://now.netapp.com/NOW/knowledge/docs/docs.shtml).

Considerations when using SnapDrive for UNIX

When using SnapDrive for UNIX, consider the following:

- A LUN managed by SnapDrive for UNIX cannot serve as either of the following:
  - A boot disk or a system disk
  - A location for the system paging file or memory dump files (or swap disk)
- Solaris and Linux hosts have operating system limits on how many LUNs you can create. To avoid having a problem when you create LUNs on these hosts, use the `snapdrive config check luns` command. For more information, see “Preparing hosts for adding LUNs” on page 120.
- SnapDrive for UNIX does not support the colon symbol (:) in the long forms of the names for LUNs and Snapshot copies. The only place SnapDrive for UNIX accepts colons is between the components of a long Snapshot copy name or between the storage system name and the storage system volume name of a LUN. For example, `toaster:/vol/vol1:snap1`

would be a typical long Snapshot copy name, while `toaster:/vol/vol1/lunA` would be a typical long LUN name.

NFS files or directory trees: The following limitations apply to SnapDrive for UNIX operations that involve NFS files or directory trees:

- SnapDrive for UNIX does not provide storage provisioning commands for NFS files or directory trees.
- SnapDrive for UNIX supports the `snapdrive snap create` and `snapdrive snap restore` commands on versions of Data ONTAP 6.5 and later. However, `snapdrive snap connect` and `snapdrive snap disconnect` commands that involve NFS use the Data ONTAP FlexVol® volumes feature for read and write access, and therefore require Data ONTAP 7.0 or later and FlexVol volumes. Configurations with Data ONTAP 6.5 or later and traditional volumes can create and restore Snapshot copies, but the Snapshot connect operation is restricted to read-only access.

Cluster environment: The SnapDrive for UNIX operations for clustered systems are available only for Veritas SFRAC 4.1 on a Solaris host. Before you use SnapDrive for UNIX in a cluster environment, see the unsupported configurations on "Unsupported configurations" on page 70.

FCP/iSCSI configuration: FCP and iSCSI configurations are not supported on the same host.

Multipathing: Multipathing is not supported on the following hosts:

- AIX host using iSCSI protocol
- Linux

Veritas stack: On AIX hosts, the JFS file system type is supported only for Snapshot operations and not for storage operations.

Thin provisioning: SnapDrive for UNIX does not support the following Data ONTAP features:

- Fractional reserve
- Space reclamation (Hole punching)
- Snapshot reserve
- Space monitoring command-line interface (CLI)
- LUN fill on demand
- Volume fill on demand
- Volume autogrow/Snapshot autodelete

Comparing SnapDrive for UNIX on different host platforms

About SnapDrive for UNIX on multiple platforms

Under most circumstances, SnapDrive for UNIX executes the same way on multiple host operating system platforms. This section describes the differences in how SnapDrive for UNIX executes across host operating systems. Apart from these points, this document describes how to use SnapDrive for UNIX regardless of the host operating system. Differences in SnapDrive for UNIX behavior between operating systems include the following:

- Installation instructions
- Pathnames
- Terminology
- Output from examples
- Cross-volume Snapshot support
- Originating-host support of Snapshot connect operation

This document contains separate installation instructions for each operating system. In addition, it provides a list of files that each operating system installs so you can see the pathnames used by that operating system.

In an effort to make this document easier to read, this guide uses one term for an item regardless of the operating system. For example, the terms *disk group* and *volume group* in this documentation can refer to either a disk group or a volume group. The term *host volume* refers to either a host volume or a logical volume; both are distinguished from storage system volumes.

Wherever possible, this document uses generic examples. In some cases, this document provides multiple examples to show the output from different operating systems. In most cases, the differences in output have to do with the differences in path names and terminology on an operating system, so they are clear to users familiar with UNIX operating systems. For example, an HP-UX host uses /dev/rdsk/c22t0d0 as a device name, while a Solaris host uses /dev/vx/dmp/c3t0d1s2, an AIX host uses /dev/hdisk2, and a Linux host uses /dev/sdd.

Note

To avoid making the examples overly long, this document sometimes uses one operating system for one example and a different operating system for the next example.

Differences between host platforms

There are the following three main differences in supported SnapDrive for UNIX functionality across different host operating systems:

- Snapshot support for storage entities spanning multiple storage system volumes or multiple storage systems is limited on hosts or configurations that do not permit a freeze operation in the software stack. Refer to "Crash-consistent Snapshot copies" on page 238 for additional information.
- Using the `snap connect` command to connect LUNs to the originating host is supported on AIX, HP-UX, and Solaris hosts.

  On Linux hosts, SnapDrive 3.0 for UNIX supports `snapdrive snap connect` operation on the originating host, unless the LUN or a LUN with a file system is part of the Linux LVM1 volume manager. For additional information, see "Connecting to a Snapshot copy" on page 272.
- You must prepare Linux and Solaris hosts before you add LUNs. See "Preparing hosts for adding LUNs" on page 120.
- On AIX, Solaris, and Linux hosts, by default, SnapDrive for UNIX encrypts the password information it sends out across the network. It communicates using HTTPS over the standard IP connection. Whereas, on HP-UX hosts, SnapDrive 3.0 for UNIX uses HTTP protocol to communicate to storage systems.

For more information on these differences, see "Considerations when using SnapDrive for UNIX" on page 7.

SnapDrive for UNIX terms for volume managers

The following table summarizes some of the differences in terms when referring to volume managers on different host platforms.

Host	Volume manager	Volume or disk groups	Location of logical volumes	Location of multipathing devices
AIX	Native LVM	Volume groups (*vg*)	/dev/*lvol* All logical volumes share the same namespace.	/dev/hdisk (FCP only) Multipathing is not supported with iSCSI.
	Veritas Volume Manager (VxVM)	Volume groups (*vg*)	/dev/vx/dsk/*dg/lvol*	/dev/vx/dmp/Disk_1

Host	Volume manager	Volume or disk groups	Location of logical volumes	Location of multipathing devices
HP-UX	Native LVM	Volume groups (*vg*)	/dev/*dg/lvol*	/dev/dsk/c15t0d2
	VxVM	Volume groups (*vg*)	/dev/vx/dsk/*dg/lvol*	/dev/vx/dmp/c15t0d2
Linux	Native LVM1	Volume groups (*vg*)	/dev/*dg-name /lvol-name*	Multipathing is not supported.
	Native LVM2	Volume groups (*vg*)	/dev/mapper/*dgname-lvolname*	
Solaris	VxVM	Disk groups (*dg*)	/dev/vx/dsk/dg/lvol	/dev/vx/dmp/c15t0d2

Note
The multipathing device location depends on the multipathing software you have on the host.

Where to go for more information

Relevant documentation

This guide provides information about the basic tasks involved in installing SnapDrive for UNIX on your host, and working with Snapshot copies. The following documentation might also be useful to you:

- SnapDrive for UNIX *Release Notes (IBM AIX, HP-UX, Linux, Solaris)*

  This document comes with SnapDrive for UNIX. You can also download a copy from http://now.netapp.com. It contains any last-minute information that you need to get your configuration up and running smoothly. It also contains late-breaking problems and their work around.

- SnapDrive for UNIX *Quick Start Guide (IBM AIX, HP-UX, Linux, Solaris)*

  This document comes with SnapDrive for UNIX. You can also download a copy from http://now.netapp.com. It provides high-level steps so that you can quickly start using SnapDrive for UNIX to create Snapshot copies and manage storage.

- SnapDrive for UNIX *Compatibility Matrix*

  This document is available at http://now.netapp.com/NOW/knowledge/docs/docs.shtml. It is a dynamic, online document that contains the most up-to-date information specific to SnapDrive for UNIX and its platform requirements.

- SnapDrive for UNIX man page

  This online document comes with the product. It contains descriptions of the SnapDrive for UNIX commands and covers issues such as using initiator groups and internal name generation.

- *Compatibility and Configuration Guide for NetApp's FCP and iSCSI Products*

  This document is available at http://now.netapp.com/NOW/knowledge/docs/san/fcp_iscsi_config/. It is a dynamic, online document that contains the most up-to-date information about the requirements for setting up a system in a NetApp SAN environment. It provides the most current details about storage systems and host platforms, cabling issues, switch issues, and configurations.

- *File Access Management Protocols Guide*

  This document is available at http://now.netapp.com/NOW/knowledge/docs/ontap/rel71rc/. It describes storage system operations, and how to manage NFS, CIFS, HTTP, FTP, webDAV and DAFS protocols.

- The FCP Host Utilities documentation

  This documentation comes with the FCP Host Utilities product. It includes of the installation guide for your host and the release notes for that host utility.
- The iSCSI Host Utilities documentation

  This documentation comes with the iSCSI Host Utilities product. It includes of the installation guide for your host and the release notes for that host utility.
- *System Configuration Guide*

  This document is available at http://now.netapp.com/NOW/knowledge/docs/docs.shtml/. This is an online document that contains information about the supported storage system models for Data ONTAP.
- Data ONTAP *Block Access Management Guide*

  This document is available from the Data ONTAP library on the NOW site at http://now.netapp.com/NOW/knowledge/docs.shtml/. It provides information about using Data ONTAP and setting up your storage system to work with Data ONTAP.
- Host operating system and host bus adapter (HBA) information

  NetApp does not provide these documents. See the Readme files and other documentation that you received with your host operating system.

Preparing to Install SnapDrive for UNIX 2

The following topics explain the tasks you must complete before installing the SnapDrive for UNIX application:

- "Prerequisites for using SnapDrive for UNIX" on page 16
- "Preparing storage systems" on page 18
- "Preparing hosts" on page 23
- "Methods for executing SnapDrive for UNIX" on page 27

Prerequisites for using SnapDrive for UNIX

Prerequisites for SnapDrive for UNIX differ depending on whether you have an FCP configuration, an iSCSI configuration, or a configuration that uses NFS directory trees.

Supported FCP, iSCSI, or NFS configurations

SnapDrive for UNIX supports the following host cluster and storage cluster topologies:

- A nonclustered configuration in which a single host is connected to a single storage system
- Any of the topologies involving NetApp storage system cluster failover
- Any of the topologies involving host clusters supported by NetApp

FCP or iSCSI configurations support the same host cluster and storage system cluster configurations that the FCP Host Utilities or iSCSI Host Utilities supports. See the utilities documentation for more information about the recommended configurations for your host and the storage systems you are using. The host utilities documentation is available at http://now.netapp.com/NOW/knowledge/docs/san/ .

Note

If you need a SnapDrive for UNIX configuration that is not mentioned in the utilities documentation, consult your NetApp technical support representative.

FCP or iSCSI configurations

If you have a configuration that uses FCP or iSCSI, you must do the following before you install SnapDrive for UNIX:

- For FCP configurations, install the FCP Host Utilities for your host.

  SnapDrive for UNIX works with the following host utilities:

  - FCP IBM AIX Host Utilities
  - FCP HP-UX Host Utilities
  - FCP Solaris Host Utilities
  - FCP Linux Host Utilities
- For iSCSI configurations, check that the configuration conforms to requirements defined in the iSCSI Host Utilities documentation for your host.

  SnapDrive for UNIX works with the following host utilities:

- iSCSI IBM AIX Host Utilities
- iSCSI HP-UX Host Utilities
- iSCSI Solaris Host Utilities
- iSCSI Linux Host Utilities

- Set up your host and storage systems.

  Follow the instructions provided with the utilities to set up your storage systems to work with the host. Configurations that include multipathing or volume manager software must use the software that is supported by the FCP Host Utilities and SnapDrive for UNIX.

Note

The latest information about SnapDrive for UNIX and its requirements is in the SnapDrive for UNIX *Compatibility Matrix* on the NOW™ NetApp on the Web™ site (http://now.netapp.com/NOW/knowledge/docs/docs.shtml).

NFS configurations

For configurations that use NFS, you must complete the following:

- Check that NFS clients are operating properly. See the *File Access Management Protocols Guide* for detailed information. This document is available at http://now.netapp.com/NOW/knowledge/docs/ontap/rel71rc/. It describes storage system operations, and how to manage NFS, CIFS, HTTP, FTP, webDAV and DAFS protocols.
- Set up your host and storage systems.

  To use SnapDrive for UNIX with NFS-mounted directories on the storage systems, you should ensure that the storage system directories are exported correctly to the host. If your host has multiple IP interfaces to the storage system, ensure that the directory is exported correctly to all of them. SnapDrive for UNIX issues warnings unless all such interfaces have read or write permission, or in the case of the `snapdrive snap connect` command with the `-readonly` option, at least read-only permissions. The `snapdrive snap restore` and `snapdrive snap connect` commands fails if none of those interfaces has permission to access the directory.

Preparing storage systems

Verify storage system readiness

Verify the storage systems are ready by performing the following tasks:

- The storage systems are online.
- The storage systems meet the minimum system requirements for SnapDrive for UNIX. See "Requirements for storage systems" on page 18.
- The HBAs and/or network interface cards (NICs) in your storage systems meet the requirements for your host operating system. For more information on HBA cards, see the *Compatibility and Configuration Guide for NetApp FCP and iSCSI Products* at http://now.netapp.com/NOW/knowledge/docs/san/fcp_iscsi_config/ and the *Setup Guide* at http://now.netapp.com/NOW/knowledge/docs/san/#iscsi_host
- The hosts and the storage systems can communicate using an IP interface. (You should have set this up when you set up storage system.)
- Licenses for the following:
  - SnapRestore
  - MultiStore software
  - Secure HTTP access to the storage system

Requirements for storage systems

Each storage system in your SnapDrive for UNIX configuration must meet the requirements in the following table.

Component	Minimum requirement
Operating system	Data ONTAP 6.5.2 or later ◆ SnapDrive for UNIX supports FlexVol volumes, but does not take advantage of all FlexVol volume features. ◆ Configurations that use NFS must use Data ONTAP 7.0 or later and FlexVol volumes to use `snapdrive snap connect` to read and write to a connected NFS file or directory tree. Configurations with traditional volumes are provided with read-only access to NFS files and directory trees.

Component	Minimum requirement
Storage system setup	You must specify the partner IP address in the storage system cluster that can be used if a storage system failover occurs. **Note** You specify this address when you run the setup program on the storage system. See "Confirm storage system has partner IP address" on page 19.
Licenses	◆ FCP, iSCSI, or NFS, depending on the host platform ◆ FlexClone license, for NFS configurations **Note** You must have the correct protocols running on the storage system for SnapDrive for UNIX to execute. ◆ SnapRestore software ◆ MultiStore software You should set the SnapRestore and MultiStore licenses when you set up the storage system. ◆ Secure HTTP access to the storage system For detailed information about administering the storage system, see the Data ONTAP *Storage Management Guide.*

Note

For the latest SnapDrive for UNIX requirements, see the online SnapDrive for UNIX *Compatibility Matrix* and the *Compatibility and Configuration Guide for NetApp's FCP and iSCSI Products.* These matrices are at http://now.netapp.com/NOW/knowledge/docs/docs.shtml/.

Confirm storage system has partner IP address

When you ran the setup program on your storage system, it prompted you for an IP address for a partner storage system to use in case of a failover. Make sure that you supplied this address. If you did not supply it, SnapDrive for UNIX cannot inquire about the storage entities on a storage system that was taken over.

Example: The following is the portion of the storage system setup script that requests the IP address. This example uses the IP address 10.2.21.35.

```
filer_A> setup
...
Should interface e0 take over a partner IP address during failover?
[n]: y
Please enter the IP address or interface name to be taken over by e0
[]: 10.2.21.35
...
filer_A> reboot -t 0
```

NFS considerations

The following considerations apply to configurations that use SnapDrive for UNIX in an NFS environment:

- The NFS service must be running on the storage system.
- The NFS client must have permissions to export (mount) resources from the storage system.

  If you are using SnapDrive for UNIX to restore or connect to NFS-mounted directories on the storage systems, you should ensure that the storage system directories are exported correctly to the host. If your host has multiple IP interfaces that can access the storage system, ensure that the directory is exported correctly to all of them. SnapDrive for UNIX issues warnings unless all such interfaces have read-write permissions, or in the case of `snapdrive snap connect` with the `-readonly` option, at least read-only permissions. The `snapdrive snap restore` and `snapdrive snap connect` commands fails if none of these interfaces has permissions to access the directory.

See the *File Access Management Protocols Guide* for detailed information. This document is available at: http://now.netapp.com/NOW/knowledge/docs/ontap/rel71rc/. It describes storage system operations, and how to manage NFS, CIFS, HTTP, FTP, webDAV and DAFS protocols.

Cautions for using SnapDrive for UNIX

NetApp strongly recommends that you read the following cautions:

- Use the default value for the space reservation setting for any LUN managed by SnapDrive for UNIX.
- In FCP or iSCSI configurations, set the `snap reserve` option on the storage system to 0 percent for each volume.

- Place all LUNs connected to the same host on a dedicated storage system volume accessible by only that host.
- If you use Snapshot copies, you cannot use the entire space on a storage system volume to store your LUNs.

  The storage system volume hosting the LUNs should be at least twice the combined size of all the LUNs on the storage system volume.
- Data ONTAP uses /vol/vol0 (root volume) to administer the storage system. Do not use this volume to store data. Also, if you have configured any other volume (other than /vol/vol0) as root volume to administer the storage system, do not use it to store data.

Preparing a storage system volume

You need to perform the following tasks on the storage system to create a volume that can hold the SnapDrive for UNIX LUNs or NFS entities attached to a single host:

- Create a storage system volume.

  Note

  You can use either the command-line prompt on the storage system or FilerView to create a storage system volume dedicated to SnapDrive for UNIX.

  For more information about the following procedures, see the *Data ONTAP Block Access Management Guide.*

- If you are in an FCP or iSCSI environment, reset the `snapdrive snap reserve` option to 0 percent on the storage system volume holding all the LUNs attached to the host (optional, but highly recommended). See "Resetting the snap reserve option" on page 22.

When you create a volume on a storage system to hold LUNs or NFS directory trees, remember the following:

- You can create multiple LUNs or NFS directory trees on a storage system volume.
- You should not store user data in the root volume on the storage system or vFiler unit.

Optimizing storage system volumes in an FCP or iSCSI environment: You can optimize your storage system volumes in the following ways:

- When multiple hosts share the same storage system, each host should have its own dedicated storage system volume to hold all the LUNs connected to that host.

- NetApp recommends that, when multiple LUNs exist on a storage system volume, the dedicated volume on which the LUNs reside contain only the LUNs for a single host. It must not contain any other files or directories.

Resetting the snap reserve option

By default, the `snap reserve` option for Data ONTAP 6.5.*x* is 20 percent. When you use Data ONTAP in an FCP or iSCSI environment, NetApp strongly recommends that you reset the `snap reserve` option to 0 percent on all storage system volumes holding SnapDrive for UNIX LUNs. To reset the `snap reserve` option on the storage system, complete the following steps.

Step	Action
1	Access the storage system either by using a command such as `telnet` from the host or by going to the storage system console.
2	Enter the following command: **`# snap reserve vol_name 0`** *vol_name* is the name of the volume on which you want to set the `snap reserve` option.

To reset the `snap reserve` option using FilerView, complete the following steps.

Step	Action
1	Open a FilerView session to the storage system holding the volume whose `snap reserve` setting is to be changed.
2	From the main FilerView menu, navigate to Volumes > snapshot > Configure.
3	In the Volume field, select the volume whose `snap reserve` setting is to be changed.
4	In the Snapshot Reserve field, enter “0.”
5	Click Apply at the bottom of the panel.

Preparing hosts

Installing the utilities

Before you install SnapDrive for UNIX, you must ensure that your host and storage system are configured properly:

- If your configuration requires an FCP Host Utilities, you must install it and get it working.
- If your configuration uses a iSCSI Host Utilities, you must refer iSCSI Host Utilities documentation to ensure that the system is set up properly.

Use the documentation that came with the FCP Host Utilities or iSCSI Host Utilities. It contains information about volume managers, multipathing, and other features you need to set up before you install SnapDrive for UNIX.

Migrating the transport protocol

To migrate from FCP to iSCSI or from iSCSI to FCP, complete the following steps.

Step	Action
1	Disconnect the host from the required storage system on which you want to migrate the transport protocol, by using the `snapdrive storage disconnect` command.
2	Stop the transport protocol service.
3	Configure or start the new transport protocol service. For more information on installation and configuration of the transport protocol, see the FCP or iSCSI Host Utilities *Installation and Setup Guide*.
4	Connect the host to the required storage system by using the `snapdrive storage connect` command.

Verify that the hosts are ready

Verify that the hosts are ready by performing the following tasks:

- Confirm that the host and storage system can communicate by completing the following steps.

Step	Action
1	Test whether the host is connected to the storage system by entering the following command: **`ping filername`**
2	Check for the storage system name in the Domain Name System (DNS) server by entering the following command: **`nslookup filername`**
3	If Steps 1 and 2 fail, complete the following steps: **a.** Set the filername-IP address pair in the /etc/hosts file. **b.** In the following file, enter the order of lookup as host local files (/etc/hosts), and then the DNS server: ❖ /etc/nsswitch.conf file for Linux, Solaris, and HP-UX hosts ❖ /etc/netsvc.conf file for AIX hosts

- Confirm that you have set up the host and storage system correctly according to the instructions in the FCP or iSCSI Host Utilities for the host.
- If you have a configuration that uses NFS, configure the exports file. Refer to the *File Access and Protocols Management Guide* on http://now.netapp.com/NOW/knowledge/docs/ontap/ontap_index.shtml for additional information.
- Verify that the host meets the minimum requirements for SnapDrive for UNIX, including the required operating system patches.

Note

For the latest information about the SnapDrive for UNIX requirements and the FCP Host Utilities that NetApp provides, see the online SnapDrive for UNIX *Compatibility Matrix* and the *Compatibility and Configuration Guide for NetApp's FCP and iSCSI Products.* These matrices are at http://now.netapp.com/NOW/knowledge/docs/docs.shtml/.

- Before installing the Veritas stack on the host, install the NTAPasl library. For complete installation instructions, see information about Veritas and the Array Support Library in the FCP Host Utilities for Native OS and Veritas *Installation and Setup Guide.*

Note

If you have installed the Veritas stack without installing the NTAPasl library, install the NTAPasl library and execute the `vxinstall` command to bring the LUNs and disk groups online.

Get a copy of the SnapDrive for UNIX software package

You can obtain the SnapDrive for UNIX software package in two ways:

- Download the package from http://now.netapp.com. The SnapDrive for UNIX software is bundled into a single compressed file. For information about getting this file, see "Downloading the SnapDrive for UNIX software from NOW" on page 25.
- Get the package from the SnapDrive for UNIX CD-ROM. For information about getting the software package from the CD-ROM, see "Getting SnapDrive for UNIX software from the CD-ROM" on page 26.

Note

To verify that you have the latest software package, check the online *S*napDrive for UNIX *Compatibility Matrix* and the *Compatibility and Configuration Guide for NetApp's FCP and iSCSI Products*. They are both available at http://now.netapp.com/NOW/knowledge/docs/docs.shtml.

Downloading the SnapDrive for UNIX software from NOW

To download the SnapDrive for UNIX software from NOW, complete the following steps.

Step	Action
1	Log in to http://now.netapp.com. In the fields on the right side of the page, complete the following steps: **a.** Enter your user name and password. **b.** From the Select Start Page list box, click Download Software. **c.** Click the Login button.
2	Go to the SnapDrive for UNIX product row of the Software Download table and select your host operating system from the Select Platform drop-down list.

Step	Action
3	Click Go!
4	Follow the prompts to reach the Software Download page.
5	Download the software file to a local directory. If you did not download the file to your host machine, move it to that machine.
6	Go to the instructions for installing the SnapDrive for UNIX software on your host operating system.

Getting SnapDrive for UNIX software from the CD-ROM

To get the SnapDrive for UNIX software from the CD-ROM, complete the following steps.

Step	Action
1	Insert the CD-ROM containing the version of SnapDrive for UNIX for your host operating system into the CD-ROM drive.
2	Change to the directory containing the SnapDrive for UNIX software.

Methods for executing SnapDrive for UNIX

You execute SnapDrive for UNIX from the UNIX host. SnapDrive for UNIX manages Snapshot copies and storage provisioning by using a command-line interface. SnapDrive for UNIX allows you to perform the following actions:

- Enter individual commands at the command line prompt
- Run scripts you create that contain embedded SnapDrive for UNIX commands

Installing and Upgrading SnapDrive for UNIX 3

About this chapter

This chapter explains the installation procedure you must follow to install or upgrade SnapDrive for UNIX.

Topics in this chapter

This chapter discusses the following topics:

- "Installing SnapDrive for UNIX on an AIX host" on page 30
- "Uninstalling SnapDrive for UNIX from an AIX host" on page 40
- "Installing SnapDrive for UNIX on an HP-UX host" on page 46
- "Uninstalling SnapDrive for UNIX from an HP-UX host" on page 52
- "Installing SnapDrive for UNIX on a Linux host" on page 54
- "Uninstalling SnapDrive for UNIX from a Linux host" on page 60
- "Installing SnapDrive for UNIX on a Solaris host" on page 61
- "Uninstalling SnapDrive for UNIX from a Solaris host" on page 72
- "Completing the installation" on page 73
- "Understanding the files installed by SnapDrive for UNIX on the host" on page 74
- "Upgrading to a new version of SnapDrive for UNIX" on page 79
- "Verifying the Veritas stack configuration" on page 81

Installing SnapDrive for UNIX on an AIX host

System requirements for FCP or iSCSI configurations

The following table lists the minimum requirements for using SnapDrive for UNIX with an AIX host in an FCP or iSCSI environment.

Note

SnapDrive for UNIX does not support FCP and iSCSI configurations simultaneously on a same host.

Component	Requirement
NetApp iSCSI AIX Host Utilities or FCP AIX Host Utilities	To make sure you have the correct version of the utility, see the online SnapDrive for UNIX *Compatibility Matrix* and the *Compatibility and Configuration Guide for NetApp FCP and iSCSI Products.* Set up the host and storage system according to the instructions in the *Setup Guide* for the iSCSI or FCP AIX utility. You must do this before you install SnapDrive for UNIX.
Additional disk space	SnapDrive for UNIX maintains three log files. While SnapDrive for UNIX rotates the files when they reach a maximum size, you should make sure you have sufficient disk space for them. For more information about the log files, see "Setting up audit, recovery, and trace logging" on page 122. Based on the default settings for the audit and trace log files, you need at least 1.1 MB of space. There is no default size for the recovery log because it rotates only after an operation completes, not when it reaches a specific size.

To obtain the SnapDrive for UNIX software package, see Chapter 2, "Get a copy of the SnapDrive for UNIX software package," on page 25.

Uncompressing the downloaded software

If you downloaded the software package from NOW, you can uncompress it by completing the following steps.

Step	Action
1	Uncompress the file and extract the software by entering the following command: `# uncompress NetApp.snapdrive_aix_3_0.Z` **Result:** This command extracts the installation package NetApp.snapdrive_aix_3_0.
2	You are now ready to install SnapDrive for UNIX. Follow the steps in the section "Installing SnapDrive for UNIX on an AIX host" on page 31.

Installing SnapDrive for UNIX on an AIX host

Use the System Management Interface Tool (SMIT) to install the AIX package. Complete the following steps.

Step	Action
1	Make sure you are logged in as root.

<table>
<tr><th>Step</th><th colspan="2">Action</th></tr>
<tr><td rowspan="3">2</td><td>If you are installing...</td><td>Then...</td></tr>
<tr><td>Software from the CD-ROM</td><td>a. Insert the CD-ROM into the CD-ROM drive.
b. Change to the root directory, which contains the software package NetApp.snapdrive_aix_3_0.
Note
This package is in a standard AIX Licensed Program Product (LPP) format.
c. Optional: Copy the package from the CD-ROM's root directory to a temporary location on the host. (You can also install the package from the CD-ROM.)
Example: If you mount the CD-ROM on /mnt and want to place the package in the /tmp directory, you might enter commands similar to the following:
mount -v cdrfs -o ro /dev/cd0 /mnt
cp /mnt/install /tmp
cd /tmp</td></tr>
<tr><td>Software from NOW</td><td>Change to the directory on which you placed the uncompressed file NetApp.snapdrive_aix_3_0.</td></tr>
<tr><td>3</td><td colspan="2">Start SMIT by entering the following command:
smit</td></tr>
</table>

Step	Action
4	Select the “Software Installation and Maintenance” option. **Example:** When you start SMIT, it displays the following screen. On this screen, the “Software Installation and Maintenance” option is the first menu option. +--+ root> # smit System Management Move cursor to desired item and press Enter. Software Installation and Maintenance Software License Management Devices System Storage Management (Physical & Logical Storage) Security & Users Communications Applications and Services Print Spooling Problem Determination Performance & Resource Scheduling System Environments Processes & Subsystems Applications Cluster System Management Using SMIT (information only) F1=Help F2=Refresh F3=Cancel Esc+8=Image Esc+9=Shell Esc+0=Exit Enter=Do +--+

Step	Action

5	At the screen that appears, select the Install and Update Software menu option.
	
	Example: The following is an example of the Software Installation and Maintenance screen.
	
	`+-----------------------------------------------------------------------------+`
	`Software Installation and Maintenance`
	
	`Move cursor to desired item and press Enter.`
	
	`  Install and Update Software`
	`  List Software and Related Information`
	`  Software Maintenance and Utilities`
	`  Network Installation Management`
	`  System Backup Manager`
	
	`F1=Help  F2=Refresh  F3=Cancel`
	`Esc+8=Image`
	`Esc+9=Shell  Esc+0=Exit  Enter=Do`
	
	`+-----------------------------------------------------------------------------+`

6	At the next screen, select the Install Software menu option.
	
	Example: The following is an example of the Install and Update Software screen.
	
	`+-----------------------------------------------------------------------------+`
	`Install and Update Software`
	
	`Move cursor to desired item and press Enter.`
	
	`  Install Software`
	`  Update Installed Software to Latest Level (Update All)`
	`  Install Software Bundle`
	`  Update Software by Fix (APAR)`
	`  Install and Update from ALL Available Software`
	
	`F1=Help  F2=Refresh  F3=Cancel`
	`Esc+8=Image`
	`Esc+9=Shell  Esc+0=Exit  Enter=Do`
	`+-----------------------------------------------------------------------------+`

Step	Action

7	At the Install Software screen, specify the location of the software in one of the following ways:
	◆ Manually enter the location by providing the following information:
	❖ If you are installing from the CD-ROM, enter the CD-ROM drive.
	❖ If you are installing from the host machine, enter the path to the software package (for example, /tmp/NetApp.snapdrive_aix_3_0).
	◆ Press F4 to display a list of options.
	If you want to use the F4 method, complete the following steps:
	a. Press F4.
	b. At the prompt asking which software you want to install, enter **`NetApp.snapdrive`**.
	c. At the prompt asking whether you want to continue or cancel, press Enter to complete the installation.
	The installation process checks the version of AIX and displays a warning message before it completes if you are not running a version of AIX that SnapDrive for UNIX supports.
	Example 1: The following is an example of entering the path to the software package when you are at the Install Software screen.
	`+--------------------------------------------------------------------------+`
	`Install Software`
	
	`Type or select a value for the entry field.`
	`Press Enter AFTER making all desired changes.`
	
	`[Entry Fields]`
	`* INPUT device / directory for software [/tmp/aix/NetApp.snapdrive_aix_3_0]+`
	
	`F1=Help F2=Refresh F3=Cancel F4=List`
	`Esc+5=Reset Esc+6=Command Esc+7=Edit`
	`Esc+8=Image`
	`Esc+9=Shell Esc+0=Exit Enter=Do`
	`+--------------------------------------------------------------------------+`

Step	Action
	Example 2: After you enter the path to the software package, SMIT displays the following screen. This is the screen on which you enter the name of the software package, **`NetApp.snapdrive`**.

```
+--------------------------------------------------------------------------+
Install Software

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
* INPUT device / directory for software
/tmp/aix/NetApp.snapdrive_aix_3_0
* SOFTWARE to install                                [NetApp.snapdrive]       +
  PREVIEW only? (install operation will NOT occur)    no                     +
  COMMIT software updates?                            yes                    +
  SAVE replaced files?                                no                     +
  AUTOMATICALLY install requisite software?           yes                    +
  EXTEND file systems if space needed?                yes                    +
  OVERWRITE same or newer versions?                   no                     +
  VERIFY install and check file sizes?                no                     +
  Include corresponding LANGUAGE filesets?            yes                    +
  DETAILED output?                                    no                     +
  Process multiple volumes?                           yes                    +
  ACCEPT new license agreements?                      no                     +
  Preview new LICENSE agreements?                     no                     +



F1=Help                 F2=Refresh              F3=Cancel               F4=List
Esc+5=Reset              Esc+6=Command            Esc+7=Edit
Esc+8=Image
Esc+9=Shell              Esc+0=Exit               Enter=Do
+--------------------------------------------------------------------------+
```

Step	Action

	Example of a successful installation: The following is an example of the output you might see when an installation successfully completes:
	
	`File:`
	`    I:NetApp.snapdrive             3.0`
	
	`+-------------------------------------------------------------------------------+`
	`                    Pre-installation Verification...`
	`+-------------------------------------------------------------------------------+`
	`Verifying selections...done`
	`Verifying requisites...done`
	`Results...`
	
	`SUCCESSES`
	`---------`
	`  Filesets listed in this section passed pre-installation verification`
	`  and will be installed.`
	
	`  Selected Filesets`
	`  -----------------`
	`  NetApp.snapdrive 3.0 # Network Appliance Snapdrive`
	
	`  << End of Success Section >>`
	
	`FILESET STATISTICS`
	`------------------`
	`    1  Selected to be installed, of which:`
	`        1  Passed pre-installation verification`
	`  ----`
	`    1  Total to be installed`

Step	Action
	+--+ Installing Software... +--+ installp: APPLYING software for: NetApp.snapdrive 3.0 . . . . . << Copyright notice for NetApp.snapdrive >> . . . . . . Copyright (c) 2007 Network Appliance, Inc. All Rights Reserved. . . . . . << End of copyright notice for NetApp.snapdrive >>. . . . Finished processing all filesets. (Total time: 2 secs). +--+ Summaries: +--+ Installation Summary -------------------- Name Level Part Event Result --- NetApp.snapdrive 3.0 USR APPLY SUCCESS
8	Verify your installation of the package by entering the **`lslpp -l NetApp.snapdrive`** command line. It should produce the following output: # lslpp -l NetApp.snapdrive Fileset Level State Description -- Path: /usr/lib/objrepos NetApp.snapdrive 3.0 COMMITTED Network Appliance Snapdrive You can also check the SMIT log file (smit.log and smit.script) if your installation fails. These files are in the SMIT log directory ($HOME) and contains installation errors.

Step	Action
9	Complete the setup by configuring SnapDrive for UNIX for the system. Most of this information is set by default; however, you need to specify the following information: ◆ The login information for the storage system. See "Specifying the current login information for storage systems" on page 154. ◆ The AutoSupport settings (AutoSupport is an optional feature, but NetApp recommends you enable it). See "Setting up AutoSupport" on page 130. ◆ The correct configuration value for the following options based on whether you are using the FCP protocol or the iSCSI protocol: ❖ `default-transport`: Set this equal to your protocol. The acceptable values are "fcp" and "iscsi". ❖ `multipathing-type`: For FCP, set it to any one of the following values: "NativeMPIO", "SANPath", or "DMP". For iSCSI, set this to "none". Multipathing is not available with iSCSI for this release. **Note** For more information about configuration settings, see "Setting configuration information" on page 84.

Uninstalling SnapDrive for UNIX from an AIX host

Use SMIT to uninstall SnapDrive for UNIX from an AIX system. Complete the following steps.

Step	Action
1	Ensure that you are logged in as root.
2	Start SMIT by entering the following command: **# smit**
3	Select the Software Installation and Maintenance menu option. **Example:** When you start SMIT, it displays the following screen: +--+ root> # smit System Management Move cursor to desired item and press Enter. Software Installation and Maintenance Software License Management Devices System Storage Management (Physical & Logical Storage) Security & Users Communications Applications and Services Print Spooling Problem Determination Performance & Resource Scheduling System Environments Processes & Subsystems Applications Cluster System Management Using SMIT (information only) F1=Help F2=Refresh F3=Cancel Esc+8=Image Esc+9=Shell Esc+0=Exit Enter=Do +--+

Step	Action
4	At the screen that appears, select the Software Maintenance and Utilities menu option. **Example:** The following is an example of the Software Installation and Maintenance screen: +--+ Software Installation and Maintenance Move cursor to desired item and press Enter. Install and Update Software List Software and Related Information Software Maintenance and Utilities Network Installation Management System Backup Manager F1=Help F2=Refresh F3=Cancel Esc+8=Image Esc+9=Shell Esc+0=Exit Enter=Do +--+

Step	Action
5	At the next screen, select the Remove Installed Software menu option. **Example:** The following is an example of the Software Maintenance and Utilities screen: +---+ Software Maintenance and Utilities Move cursor to desired item and press Enter. Commit Applied Software Updates (Remove Saved Files) Reject Applied Software Updates (Use Previous Version) Remove Installed Software Copy Software to Hard Disk for Future Installation Check Software File Sizes After Installation Verify Software Installation and Requisites Clean Up After Failed or Interrupted Installation F1=Help F2=Refresh F3=Cancel Esc+8=Image Esc+9=Shell Esc+0=Exit Enter=Do +---+

Step	Action
6	Remove the software in one of the following ways: ◆ Enter the package name (NetApp.snapdrive). (Make sure that the “Preview only” option is set to “no”.) ◆ Press F4 to display a list of names. If you use F4, complete the following steps: **a.** Press F4. **b.** Scroll down the list of names until you reach NetApp.snapdrive. **c.** Select NetApp.snapdrive and press Enter. **d.** At the prompt asking whether you want to continue or cancel, press Enter to complete the uninstall. The uninstall process places the log file in /tmp/snapdrive_uninstall. **Note** When you uninstall the software, the log files are not removed. You must go to the /var/log/sd-trace.log files directory and manually remove them. **Example:** The following is an example of the Remove Installed Software screen. **Note** By default, the "PREVIEW only" field is set to “yes”. You must change it to “no” if you want to uninstall.

```
+--------------------------------------------------------------------------+
Remove Installed Software
Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
* SOFTWARE name                                     [NetApp.snapdrive]        +
  PREVIEW only? (remove operation will NOT occur)     no                      +
  REMOVE dependent software?                          no                      +
  EXTEND file systems if space needed?                no                      +
  DETAILED output?                                    no                      +

F1=Help                 F2=Refresh              F3=Cancel               F4=List
Esc+5=Reset              Esc+6=Command            Esc+7=Edit
Esc+8=Image
Esc+9=Shell              Esc+0=Exit               Enter=Do
+--------------------------------------------------------------------------+
```

Step	Action

	Example of successful uninstall: The following output appears when you successfully uninstall the software:
	
	`File:`
	`NetApp.snapdrive`
	
	`+-----------------------------------------------------------------------------+`
	`Pre-deinstall Verification...`
	`+-----------------------------------------------------------------------------+`
	`Verifying selections...done`
	`Verifying requisites...done`
	`Results...`
	
	`SUCCESSES`
	`---------`
	`Filesets listed in this section passed pre-deinstall verification`
	`and will be removed.`
	
	`Selected Filesets`
	`-----------------`
	`NetApp.snapdrive 3.0 # Network Appliance Snapdrive`
	
	`<< End of Success Section >>`
	
	`FILESET STATISTICS`
	`------------------`
	`1 Selected to be deinstalled, of which:`
	`1 Passed pre-deinstall verification`
	`----`
	`1 Total to be deinstalled`

Step	Action
	+--+ Deinstalling Software... +--+ installp: DEINSTALLING software for: NetApp.snapdrive 3.0 Finished processing all filesets. (Total time: 1 secs). +--+ Summaries: +--+ Installation Summary -------------------- Name Level Part Event Result --- NetApp.snapdrive 3.0 USR DEINSTALL SUCCESS **Example of warning:** If you are trying to uninstall a product that is not currently installed, you see messages such as the following: Not Installed ------------- No software could be found on the system that could be deinstalled for the following requests: NetAppSnapdrive (The fileset may not be currently installed, or you may have made a typographical error.)

Installing SnapDrive for UNIX on an HP-UX host

System requirements for FCP or iSCSI configurations

The following table lists the minimum requirements for using SnapDrive for UNIX with an HP-UX host in an FCP or iSCSI environment.

Note

SnapDrive for UNIX does not support FCP and iSCSI configurations simultaneously on a same host.

Component	Requirement
NetApp iSCSI HP-UX Host Utilities or FCP HP-UX Host Utilities	To make sure you have the correct version of the utility, see the online SnapDrive for UNIX *Compatibility Matrix* and the *Compatibility and Configuration Guide for NetApp FCP and iSCSI Products.* Set up the host and storage system according to the instructions in the *Setup Guide* for the FCP or iSCSI HP-UX utility. You must do this before you install SnapDrive for UNIX.
Additional disk space	SnapDrive for UNIX maintains the audit, recovery, and trace files. While SnapDrive for UNIX rotates the files when they reach a maximum size, you should make sure you have sufficient disk space for them. For more information about the log files, see "Setting up audit, recovery, and trace logging" on page 122. Based on the default settings for the audit and trace log files, you need at least 1.1 MB of space. There is no default size for the recovery log because it rotates only after an operation completes, not when it reaches a specific size.

System requirements for the Veritas stack

Before installing the Veritas stack on the host, install the NTAPasl library. If you have installed the Veritas stack without installing the NTAPasl library, complete the following steps.

Step	Action
1	Install the NTAPasl library.
2	Execute the `vxinstall` command to bring the LUNs and disk groups online.
3	**Conditional:** For the previous versions of the HP-UX 11i host, execute the `insf -d mm` or `mknod/dev/zero c 3 4` command, so that the `vxdiskupsetup` command can clear the disk private regions properly.

To obtain the SnapDrive for UNIX software package, Chapter 2, "Get a copy of the SnapDrive for UNIX software package," on page 25.

Moving the downloaded file to a local directory

If you downloaded the file and did not place it on your HP-UX machine, you must move it to that machine. Complete the following steps.

Step	Action

1	Use commands similar to the following ones to move the file you downloaded from the NOW site to your HP-UX machine:
	`# mkdir /tmp/hpux`
	`# cd /tmp/hpux`
	`# cp /u/hpux/HP-pkgs/NTAPsnapdrive_hpux_3_0.depot.`
	Make sure you include the period (.) at the end of the copy command line.

2	You are now ready to install SnapDrive for UNIX. Follow the steps in the section "Installing SnapDrive for UNIX on an HP-UX host" on page 48.

Installing SnapDrive for UNIX on an HP-UX host

To install SnapDrive for UNIX on an HP-UX host, complete the following steps.

Step	Action
1	Make sure you are logged in as either root or an authorized software distributor (SD) user for that HP-UX host. **Note** The installation file is a standard HP-UX depot file, so any user on your system with software installation privileges can perform the installation. For more information on authorized users, see the HP-UX `swacl` man page.

<table>
<tr><th>Step</th><th colspan="2">Action</th></tr>
<tr><td rowspan="3">2</td><td>If you are installing...</td><td>Then...</td></tr>
<tr><td>Software from the CD-ROM</td><td>a. Mount the CD-ROM on the host system by entering a command similar to the following:

mount /dev/cdrom /mnt/cdrom

b. Change to the directory on which your CD-ROM is mounted. For example, change to /mnt/cdrom:

cd /mnt/cdrom

Note
You can install the software directly from the CD-ROM. NetApp, however, recommends that you copy the file to the local disk before installing.

c. If you want to install the software from a local directory on the host, copy the file to that directory. For example, you might need to enter commands similar to the following:

mkdir /tmp/hpux
mount /dev/cdrom /mnt/cdrom
cp /mnt/cdrom/NTAPsnapdrive_hpux_3_0.depot /tmp/hpux
cd /tmp/hpux</td></tr>
<tr><td>Downloaded software</td><td>Change to the directory where you put the software you downloaded from the NOW site.</td></tr>
</table>

Step	Action

3	Enter the following command to install the software:
	# swinstall -s <*pathname*>/NTAPsnapdrive_hpux_3_0.depot snapdrive
	pathname is the full path to the NTAPsnapdrive file. This path must begin with a slash (/). The file needs to be on an NFS mounted directory. The script installs SnapDrive for UNIX in the /opt/NetApp/snapdrive/ directory.
	Example: The `swinstall` command installs the SnapDrive for UNIX software without a problem. It writes installation information to a log file.
	`# swinstall -s /tmp/hpux/NTAPsnapdrive_hpux_3_0.depot snapdrive`
	`=======  03/11/04 15:49:28 EST  BEGIN swinstall SESSION (non-interactive) (jobid=netapp2-0168)`
	`* Session started for user "root@netapp2".`
	`* Beginning Selection`
	`* Target connection succeeded for "netapp2:/".`
	`* Source: /tmp/hpux/NTAPsnapdrive_hpux_3_0.depot`
	`* Targets: netapp2:/`
	`* Software selections:`
	`snapdrive.command_parisc,r=1,fr=3.0,fa=HP-UX_B.11.00_32/64`
	`snapdrive.snapdrive_diag_scripts,r=1`
	`snapdrive.snapdrive_docs,r=1`
	`* Selection succeeded.`
	`* Beginning Analysis and Execution`
	`* Session selections have been saved in the file "/.sw/sessions/swinstall.last".`
	`* The analysis phase succeeded for "netapp2:/".`
	`* The execution phase succeeded for "netapp2:/".`
	`* Analysis and Execution succeeded.`
	`NOTE: More information may be found in the agent logfile using the command "swjob -a log netapp2-0168 @ netapp2:/".`
	`=======  03/11/04 15:49:48 EST  END swinstall SESSION (non-interactive) (jobid=netapp2-0168)`

Step	Action

4	Verify your installation using the `swverify` command.
	
	Example: The following example uses the `swverify` command to confirm that the SnapDrive for UNIX software installed:
	
	`# swverify snapdrive`
	
	`=======  03/12/04 11:55:49 EST  BEGIN swverify SESSION (non-interactive) (jobid=netappp2-0172)`
	
	`* Session started for user "root@netapp2".`
	
	`* Beginning Selection`
	`* Target connection succeeded for "netapp2:/".`
	`* Software selections:`
	`snapdrive.command_parisc,l=/opt/NetApp/snapdrive,r=2,fr=3.0,fa=HP-UX_B.11.00_32/64`
	`snapdrive.snapdrive_diag_scripts,l=/opt/NetApp/snapdrive,r=3.0`
	`snapdrive.snapdrive_docs,l=/opt/NetApp/snapdrive,r=1`
	`* Selection succeeded.`
	`* Beginning Analysis`
	`* Session selections have been saved in the file "/.sw/sessions/swverify.last".`
	`* The analysis phase succeeded for "netapp2:/".`
	`* Verification succeeded.`
	
	`NOTE:  More information may be found in the agent logfile using the command "swjob -a log netapp2-0172 @ netapp2:/".`
	
	`=======  03/12/04 11:55:54 EST  END swverify SESSION (non-interactive) (jobid=netapp2-0172)`

5	Complete the setup by configuring SnapDrive for UNIX for the system. Most of this information is set by default; however, you need to specify the following information: ◆ The login information for the storage system. See “Specifying the current login information for storage systems” on page 154. ◆ The AutoSupport settings (AutoSupport is an optional feature, but NetApp recommends you enable it). See “Setting up AutoSupport” on page 130. **Note** For general information about configuration settings, see “Setting configuration information” on page 84.

Uninstalling SnapDrive for UNIX from an HP-UX host

To uninstall SnapDrive for UNIX from an HP-UX system, complete the following steps.

Step	Action
1	Ensure that you are logged in as either root or an authorized SD user for that HP-UX host.
2	Use the `swremove` command to remove the software: **`# swremove snapdrive`** **Note** This command does not remove the log files. You must go to the /var/snapdrive directory and manually remove them.

Step	Action

	Example: The following example uninstalls SnapDrive for UNIX on an HP-UX system:
	
	`# swremove snapdrive`
	
	`=======  03/11/04 16:22:44 EST  BEGIN swremove SESSION (non-interactive) (jobid=netapp2-0170)`
	
	`* Session started for user "root@netapp2".`
	`* Beginning Selection`
	`* Target connection succeeded for "netapp2:/".`
	`* Software selections:`
	`snapdrive.command_parisc,l=/opt/NetApp/snapdrive,r=2,fr=3.0,fa=HP-UX_B.11.00_32/64`
	`snapdrive.snapdrive_diag_scripts,l=/opt/NetApp/snapdrive,r=1`
	`snapdrive.snapdrive_docs,l=/opt/NetApp/snapdrive,r=1`
	`* Selection succeeded.`
	`* Beginning Analysis`
	`* Session selections have been saved in the file "/.sw/sessions/swremove.last".`
	`* The analysis phase succeeded for "netapp2:/".`
	`* Analysis succeeded.`
	`* Beginning Execution`
	`* The execution phase succeeded for "netapp2:/".`
	`* Execution succeeded.`
	`NOTE: More information may be found in the agent logfile using the command "swjob -a log netapp2-0170 @ netapp2:/".`
	
	`=======  03/11/04 16:22:51 EST  END swremove SESSION (non-interactive) (jobid=netapp2-0170)`

Installing SnapDrive for UNIX on a Linux host

System requirements for FCP or iSCSI configurations

The following table lists the minimum requirements for using SnapDrive for UNIX on a Linux host in an FCP or iSCSI environment.

Note

SnapDrive for UNIX does not support FCP and iSCSI configurations simultaneously on a same host.

Component	Requirement
NetApp iSCSI Linux Host Utilities (Support Kit)	To make sure you have the correct version of the utility, see the online SnapDrive for UNIX *Compatibility Matrix* and the *Compatibility and Configuration Guide for NetApp FCP and iSCSI Products.* Set up the host and storage system according to the instructions in the *Setup Guide* for the FCP or iSCSI Linux Host Utilities. You must do this before you install SnapDrive for UNIX.
Additional disk space	SnapDrive for UNIX maintains the audit, recovery, and trace log files. While SnapDrive for UNIX rotates the files when they reach a maximum size, ensure you have sufficient disk space for them. For more information about the log files, see "Setting up audit, recovery, and trace logging" on page 122. Based on the default settings for the audit and trace log files, you need at least 1.1 MB of space. There is no default size for the recovery log because it rotates only after an operation completes, not when it reaches a specific size.

To obtain the SnapDrive for UNIX software package, see Chapter 2, "Get a copy of the SnapDrive for UNIX software package," on page 25.

Moving the downloaded file to a local directory

If you downloaded the file and did not place it on the Linux host, you must move it to that host. Complete the following steps.

Step	Action

1	Copy the downloaded file to the Linux host. You can place it in any directory on the host.
	
	Example: You can use commands similar to the following ones to move the file you downloaded from the NOW site to the Linux machine:
	
	`# mkdir /tmp/linux`
	`# cd /tmp/linux`
	`# cp /u/linux/netapp.snapdrive.linux_3_0.rpm .`
	
	Make sure you include the period (.) at the end of the copy command line.

2	You are now ready to install SnapDrive for UNIX. Follow the steps in the section "Installing SnapDrive for UNIX on a Linux host" on page 56.

Note

Ensure that all the supported service packs are installed on the host before installing SnapDrive 3.0 for UNIX. For more information on the support service packs, see SnapDrive for UNIX *Compatibility Matrix* on the NOW site (http://now.netapp.com/NOW/knowledge/docs/docs.shtml).

Installing SnapDrive for UNIX on a Linux host

To install SnapDrive for UNIX on a Linux host, complete the following steps.

<table>
<tr><th>Step</th><th colspan="2">Action</th></tr>
<tr><td>1</td><td colspan="2">Make sure you are logged in as root.

If you are executing this file remotely and the system configuration does not allow you to log in as root, use the <code>su</code> command to become root.

Note
The installation file is a standard Linux.rpm file.</td></tr>
<tr><td rowspan="3">2</td><td>If you are installing...</td><td>Then...</td></tr>
<tr><td>Software from the CD-ROM</td><td>a. Mount the CD-ROM on the host system by entering a command similar to the following:

<code># mount /dev/cdrom /mnt/cdrom</code>

b. Change to the directory where your CD-ROM is mounted. For example, change to /mnt/cdrom:

<code>cd /mnt/cdrom</code>

Note
You can install the software directly from the CD-ROM or you can copy it to a local disk.

c. If you want to install the software from a local directory on the host, copy the file to that directory. For example, you might need to enter commands similar to the following:

<code># mkdir /tmp/linux
mount /dev/cdrom /mnt/cdrom
cp /mnt/cdrom/netapp.snapdrive.linux_3_0.rpm /tmp/linux
cd /tmp/linux</code></td></tr>
<tr><td>Downloaded software</td><td>Change to the directory on your Linux host where you put the software you downloaded from the NOW site.</td></tr>
</table>

Step	Action

3	Use the `rpm` command to install the software.
	
	`# rpm -U -v <pathname>/netapp.snapdrive.linux_3_0.rpm`
	
	Example 1: The `rpm` command installs the SnapDrive for UNIX software without a problem. It writes installation information to a log file.
	
	`# rpm -U -v netapp.snapdrive.linux_3_0.rpm`
	`  Preparing packages for installation...`
	`  netapp.snapdrive-3.0`
	
	CAUTION
	Do not erase the sg3_utils package that is installed by default. If you remove it, you can hit intermittent device discovery failure.

Step	Action

4	Verify the installation.
	
	Example 1: The following example uses the `rpm` command with the `-qai` option to verify the installation. The `-qai` option gives you detailed information about the SnapDrive for UNIX installation package:
	
	`# rpm -qai netapp.snapdrive`
	
	`Name : netapp.snapdrive Relocations: (not relocatable)`
	`Version : 2.3 Vendor: Network Appliance`
	`Release : 1 Build Date: Saturday 10,March,2007 05:25:48 PM IST`
	`Install Date: Saturday 10,March,2007 08:02:50 PM IST BuildHost: bldl17-fe.eng.netapp.com`
	`Group : Applications Source RPM: netapp.snapdrive-2.3-1.src.rpm`
	`Size : 9025104 License: netapp`
	`Signature : (none)`
	`Packager : Network Appliance`
	`URL : http://now.netapp.com/`
	`Summary : SnapDrive for Linux`
	`Description :`
	`SnapDrive is a SAN storage management utility.`
	`It provides an easy to use interface that allows the user to create snapshots of LVM objects (i.e. volume groups) and restore from those snapshots. SnapDrive also provides a simple interface to allow for the provisioning of LUNs for mapping LVM objects to them.`
	
	Example 2: The following example uses the `rpm` and `ls` commands to verify that the SnapDrive for UNIX software installed:
	
	`# rpm -qa netapp.snapdrive`
	`netapp.snapdrive-3.0`
	`# ls -R /opt/NetApp`
	`/opt/NetApp:`
	`snapdrive`
	
	`/opt/NetApp/snapdrive:`
	`bin diag docs snapdrive.conf`
	
	`/opt/NetApp/snapdrive/bin:`
	`snapdrive`

Step	Action

	`/opt/NetApp/snapdrive/diag:`
	`filer_info  linux_info  SHsupport.pm  snapdrive.dc  Telnet.pm`
	
	`/opt/NetApp/snapdrive/docs:`
	`man1`
	
	`/optNetApp/snapdrive/docs/man1:`
	`brocade_info.1  filer_info.1  linux_info.1  snapdrive.1  snapdrive.dc.1`
	`# ls -l /usr/sbin/snapdrive`
	`-rwxr-xr-x  1 root  root ../../opt/NetApp/snapdrive/bin/snapdrive`

5	Complete the setup by configuring SnapDrive for UNIX for the system. Most of this information is set by default; however, you need to specify the following information: ◆ The login information for the storage system. See "Specifying the current login information for storage systems" on page 154. ◆ The AutoSupport settings (AutoSupport is an optional feature, but NetApp recommends you enable it). See "Setting up AutoSupport" on page 130. **Note** For general information about configuration settings, see "Setting configuration information" on page 84.

Uninstalling SnapDrive for UNIX from a Linux host

To uninstall SnapDrive for UNIX from a Linux system, complete the following steps.

Step	Action
1	Ensure that you are logged in as root.
2	Use the `rpm` command to remove the software. **Example:** The following example uses the `rpm` command with the `-e` option to uninstall the SnapDrive for UNIX software: `# rpm -e netapp.snapdrive` **Note** This command does not remove the log files. You must go to the /var/log directory and manually remove them.
3	Verify that the package is uninstalled. **Example:** The following example verifies that SnapDrive for UNIX is no longer installed: `# rpm -qa netapp.snapdrive`

Installing SnapDrive for UNIX on a Solaris host

System requirements for FCP or iSCSI configurations

The following table lists the minimum requirements for using SnapDrive for UNIX with a Solaris host in an FCP or iSCSI environment.

Note

SnapDrive for UNIX does not support FCP and iSCSI configurations simultaneously on a same host.

Note

On Solaris 9 update 8 hosts, you have to install SANsurfer CLI iSCSI HBA Manager for Solaris, for hardware iSCSI to work.

Component	Requirement
NetApp iSCSI Solaris Host Utilities or FCP Solaris Host Utilities	To make sure you have the correct version of the utility, see the online SnapDrive for UNIX *Compatibility Matrix* and the *Compatibility and Configuration Guide for NetApp FCP and iSCSI Products.* Set up the host and storage system according to the instructions in the *Setup Guide* for the iSCSI or FCP Solaris utility. You must do this before you install SnapDrive for UNIX.
Additional disk space	SnapDrive for UNIX maintains the audit, recovery, and trace log files. While SnapDrive for UNIX rotates the files when they reach a maximum size, you should make sure you have sufficient disk space for them. For more information about the log files, see "Setting up audit, recovery, and trace logging" on page 122. Based on the default settings for the audit and trace log files, you need at least 1.1 MB of space. There is no default size for the recovery log because it rotates only after an operation completes, not when it reaches a specific size.

To obtain the SnapDrive for UNIX software package, see Chapter 2, "Get a copy of the SnapDrive for UNIX software package," on page 25.

Uncompressing the software package you downloaded

If you downloaded the software package from NOW, you can uncompress it by completing the following steps.

Step	Action
1	Change to the directory to which you downloaded the compressed file. For example, change to the /tmp directory: `# cd /tmp`
2	Enter the following command to uncompress the file: `# uncompress NTAPsnapdrive.tar.Z`
3	Enter the following command to extract the software: `# tar -xvf NTAPsnapdrive.tar`
4	You are now ready to install SnapDrive for UNIX. Follow the steps in the section "Installing SnapDrive for UNIX on a Solaris host" on page 63.

Installing SnapDrive for UNIX on a Solaris host

To install SnapDrive for UNIX on a Solaris host, complete the following steps.

<table>
<tr><th>Step</th><th colspan="2">Action</th></tr>
<tr><td>1</td><td colspan="2">Ensure that you are logged in as root.

If you are executing this file remotely and your system configuration does not allow you to log in as root, use the <code>su</code> command to become root.

Note
The installation file is a standard Solaris.pkg file.</td></tr>
<tr><td rowspan="3">2</td><td>If you are installing...</td><td>Then...</td></tr>
<tr><td>Software from the CD-ROM</td><td>a. Mount the CD-ROM on the host system.

b. Change to the directory where your CD-ROM is mounted.

For example, change to /cdrom/cdrom0:

<code>cd /cdrom/cdrom0</code></td></tr>
<tr><td>Downloaded software from NOW</td><td>Change to the directory to which you extracted the software.</td></tr>
<tr><td>3</td><td colspan="2">Enter the following command to install the software:

<code># ./install</code>

You should have both the attach kit and the ASL installed before you install SnapDrive for UNIX.

Example: The script installs the SnapDrive for UNIX software without a problem. It writes installation information to a log file.

<code># ./install
Installing NTAPsnapdrive now ...
NTAPsnapdrive install completed successfully.
snapdrive Installation complete. Log is in
/tmp/snapdrive_install_log.23752.</code></td></tr>
</table>

Step	Action
4	Complete the setup by configuring SnapDrive for UNIX for the system. Most of this information is set by default; however, you need to specify the following information: ◆ The login information for the storage system. See Chapter 5, "Specifying the current login information for storage systems," on page 154. ◆ The AutoSupport settings (AutoSupport is an optional feature, but NetApp recommends you enable it). See Chapter 4, "Setting up AutoSupport," on page 130. For general information about configuration settings, see Chapter 4, "Setting configuration information," on page 84.

Setting up an SFRAC 4.1 I/O fencing environment on a storage system

SnapDrive for UNIX provides storage provisioning and Snapshot management options to manage both cluster-wide shared and node-local disk groups, and file systems in an SFRAC 4.1 environment. To set up the SFRAC 4.1 environment on a storage system, complete the following steps.

Prerequisites: You need to check for the following before setting up the SFRAC 4.1 I/O fencing environment on a storage system.

Step	Action
1	Set up `rsh` or `ssh` manually to use access-without-password-prompt for the root user between all cluster nodes. For setup instructions, see *Veritas Cluster Server 4.1 Installation Guide for Solaris*.
2	Install SnapDrive for UNIX on all the nodes in the cluster. If different versions of SnapDrive for UNIX are installed on different nodes, then SnapDrive for Unix operations fail. For installation instructions, see "Installing SnapDrive for UNIX on a Solaris host" on page 63.
3	Check the FCP connectivity between the storage systems. To learn about hardware requirements for hosts, see the SFRAC *Release Notes*. **Note** The /opt/NTAPsnapdrive/snapdrive.conf file on all the nodes must have the `default-transport` configuration variable set to FCP. For more information about this configuration variable, see Chapter 4, "Determining options and their default values," on page 86.

Step	Action
4	Set a value for the `secure-communication-among-cluster-nodes` configuration variable, to ensure that the `rsh` or `ssh` access-without-password-prompt for the root user is configured for all nodes in the cluster. This is necessary because, if you initiate the SnapDrive for UNIX commands from any node (master or nonmaster) in the cluster, SnapDrive for UNIX carries out operations on other nodes in the cluster. For more information about this configuration variable, see Chapter 4, "Determining options and their default values," on page 86.

5	Check for device discovery on the cluster nodes. To do this, execute the `snapdrive storage create` command on each node in the cluster:
	
	`snapdrive storage create -lun long_lun_name [lun_name ...] -lunsize size [{-reserve | -noreserve}] [-igroup ig_name [ig_name ...]]`
	
	For detailed information about the SnapDrive for UNIX commands and their descriptions, see Chapter 9, "Command Reference," on page 339.
	
	Example:
	`# snapdrive storage create -lun f270-197-109:/vol/vol2/luntest -lunsize 20m`
	
	`LUN f270-197-109:/vol/vol2/luntest ... created`
	
	`mapping new lun(s) ... done`
	`discovering new lun(s) ... done`
	
	`LUN to device file mappings:`
	`-f270-197-109:/vol/vol2/luntest => /dev/vx/dmp/c5t0d6s2`
	
	`# snapdrive storage delete -lun f270-197-109:/vol/vol2/luntest -lunsize 20m`
	`- LUN f270-197-109:/vol/vol2/luntest ... deleted`

After verifying the success of device discovery on all nodes in the cluster, complete the following steps to set up an SFRAC 4.1 I/O fencing environment.

Step	Action
1	Install the SFRAC 4.1 package on all the nodes in the cluster without setting up shared storage and I/O fencing for SFRAC. For the minimum size of the shared I/O fencing LUNs, refer *Veritas Cluster Server 4.1 Installation Guide for Solaris*. At this point, the /sbin/gabconfig -a shows that Cluster File System and Cluster Volume Manager are not up.

2	Set up shared storage and I/O fencing for SFRAC for Cluster Volume Manager (CVM) by completing the following steps:
	
	a. Create at least three LUNs from any one node in the cluster. Execute the `snapdrive storage create` command from any one node in the cluster:
	
	`snapdrive storage create -lun long_lun_name [lun_name ...] -lunsize size [{-reserve | -noreserve}] [-igroup ig_name [ig_name ...]]`
	
	For detailed information on creating storage, see Chapter 6, "Provisioning and Managing Storage," on page 159.
	
	Example:
	`# snapdrive storage create -lun f270-197-109:/vol/vol2/VXFEN_COORDLUN0 VXFEN_COORDLUN1 VXFEN_COORDLUN2 -lunsize 5g`
	
	`LUN f270-197-109:/vol/vol2/VXFEN_COORDLUN0 ... created`
	`LUN f270-197-109:/vol/vol2/VXFEN_COORDLUN1 ... created`
	`LUN f270-197-109:/vol/vol2/VXFEN_COORDLUN2 ... created`
	
	`mapping new lun(s) ... done`
	`discovering new lun(s) ... done`
	
	`LUN to device file mappings:`
	`- f270-197-109:/vol/vol2/VXFEN_COORDLUN0 => /dev/vx/dmp/c3t0d3s2`
	`- f270-197-109:/vol/vol2/VXFEN_COORDLUN1 => /dev/vx/dmp/c3t0d4s2`
	`- f270-197-109:/vol/vol2/VXFEN_COORDLUN2 => /dev/vx/dmp/c3t0d8s2`

Step	Action

	Note
	Ensure that you do not delete any of these LUNs at any point while setting up the SFRAC cluster environment. These LUNs are required throughout the life of the SFRAC cluster environment and not during the setup phase alone.
	If you delete the LUNs, SnapDrive for UNIX does not alert you with an error message, and the SFRAC setup fails.
	
	b. Execute the `snapdrive storage connect` command from all other nodes in the cluster:
	`snapdrive storage connect -lun long_lun_name [lun_name ...] [-igroup ig_name [ig_name ...]]`
	For detailed information on connecting to a storage entity, see Chapter 6, "Provisioning and Managing Storage," on page 159.
	Example:
	`# snapdrive storage connect -lun f270-197-109:/vol/vol2/VXFEN_COORDLUN0 VXFEN_COORDLUN1 VXFEN_COORDLUN2`
	`mapping lun(s) ... done`
	`discovering lun(s) ... done`
	`LUN f270-197-109:/vol/vol2/VXFEN_COORDLUN0 connected`
	`- device filename(s): /dev/vx/dmp/c5t0d3s2`
	`LUN f270-197-109:/vol/vol2/VXFEN_COORDLUN1 connected`
	`- device filename(s): /dev/vx/dmp/c5t0d5s2`
	`LUN f270-197-109:/vol/vol2/VXFEN_COORDLUN2 connected`
	`- device filename(s): /dev/vx/dmp/c5t0d9s2`
	c. Verify that the shared LUNs are visible across all nodes in the cluster. To verify, execute the `snapdrive storage show -all` command on each node in the cluster:
	`snapdrive storage show -all`

Step	Action

	Example:
	
	`# snapdrive storage show -all`
	
	`Connected LUNs and devices:`
	
	`device filename adapter path size proto state clone lun path backing snapshot`
	`---------- ------- ---- ---- ----- ----- ----- --- ----- ----------`
	`/dev/vx/dmp/c3t0d0s2 - P 5g fcp online No f270-197-109:/vol/vol1/VXFEN_COORD_LUN1 -`
	`/dev/vx/dmp/c3t0d1s2 - P 5g fcp online No f270-197-109:/vol/vol1/VXFEN_COORD_LUN2 -`
	`/dev/vx/dmp/c3t0d2s2 - P 5g fcp online No f270-197-109:/vol/vol1/VXFEN_COORD_LUN3 -`

3	Set up the shared storage and I/O fencing for SFRAC. For instructions, see *Veritas Cluster Server 4.1 Installation Guide for Solaris*. For setting up shared storage and I/O fencing for SFRAC, you have to use the disks that you created in Step 2.

Step	Action
4	Verify the Group Membership Services/Atomic Broadcast (GAB) port membership on each node in the cluster by using the `snapdrive config check cluster` command. For example, following is the output of a 2-node SFRAC cluster:

```
# /sbin/gabconfig -a
GAB Port Memberships
===============================================================
Port a gen   344303 membership 01
Port b gen   344306 membership 01
Port d gen   344305 membership 01
Port f gen   344313 membership 01
Port h gen   344321 membership 01
Port o gen   344309 membership 01
Port q gen   344311 membership 01
Port v gen   34430d membership 01
Port w gen   34430f membership 01

# hastatus -summary

-- SYSTEM STATE
-- System               State                Frozen

A  sfrac-57             RUNNING              0
A  sfrac-58             RUNNING              0

-- GROUP STATE
-- Group           System               Probed     AutoDisabled    State

B  cvm             sfrac-57             Y          N               ONLINE
B  cvm             sfrac-58             Y          N               ONLINE
```

Adding a node to a cluster

To add a new node to a cluster, follow the instructions given in the *Veritas Cluster Server 4.1 Installation Guide for Solaris*. After configuring Low Latency Transport (LLT) and GAB, you have to complete the following steps.

Note

NetApp recommends to use the `snapdrive config check cluster` option before using any of the snapdrive commands to ensure that the cluster is set up properly. For more information on this option, see "Command-line options" on page 348.

Step	Action
1	Map all the LUNs for all the shared disk groups in the cluster. To do this, use the `snapdrive storage connect` command. For more information on storage management operations, see Chapter 6, "Provisioning and Managing Storage," on page 159.
2	Start CVM on the new node. For instructions, see to *Veritas Cluster Server 4.1 Installation Guide for Solaris*. **Result:** The shared disk group and file system will automatically be visible on the new node.

Removing a node from a cluster

To remove a new node from a cluster, follow the instructions given in the *Veritas Cluster Server 4.1 Installation Guide for Solaris*. After configuring LLT and GAB, you have to unmap all the LUNs for all the shared storage entities in the cluster. To do this, use the `snapdrive storage disconnect` command. For more information on storage management operations, see Chapter 6, "Provisioning and Managing Storage," on page 159.

Unsupported configurations

Following are the unsupported configurations in a cluster environment on the Solaris host:

- The `-igroup` option is unsupported in command line that contains a storage entity on a raw LUN or NFS entities.
- Manually created igroups are unsupported for shared LUNs in the cluster.
- SnapDrive for UNIX operations fail, if any of the following conditions are not met:
  - The SnapDrive for UNIX access permissions must be set uniformly across all the nodes in the cluster.
  - The LUNs for shared file systems or disk groups in the cluster should not be mapped to a node outside of the cluster.
  - The value of the configuration variables must be set uniformly across all the nodes in the cluster.
- The storage system-root password pairs are stored locally on each node in the cluster. You have to ensure that all the storage systems are accessible from each node in the cluster.

- Manually changing the VCS configuration, which can affect the file systems and disk groups that are created by SnapDrive for UNIX.
- SnapDrive for UNIX attempts to recover from configuration failures or device mapping error conditions. This attempt might fail, depending on the configuration settings on the cluster.
- SnapDrive for UNIX operations can time out in the cluster depending on your request. In this case, the operations have to be restricted.

Uninstalling SnapDrive for UNIX from a Solaris host

To uninstall SnapDrive for UNIX from a Solaris system, complete the following steps.

Step	Action
1	Ensure that you are logged in as root.

2	Use the `uninstall` command to remove the software. Answer `y` (or `yes`) when the command asks whether you are sure you want to remove the attach kit.
	If you answer `n` (or `no`), the command does not uninstall the attach kit.
	`# ./uninstall`
	`Are you sure you want to uninstall the snapdrive package? [y,n,?,q] y`
	`Removing NTAPsnapdrive ...`
	`NTAPsnapdrive successfully removed.`
	`Remove process complete. Log is in /tmp/snapdrive_remove_log.23789`
	To avoid getting the confirmation query, enter the **`./uninstall`** command with the **`-f`** option, which prevents the uninstall script from running in interactive mode.
	Note
	The uninstall process might fail if the package has been partially uninstalled. If that happens, you can uninstall SnapDrive for UNIX by inserting the CD-ROM and executing the `./uninstall` command from the top-level directory.

Completing the installation

Completing the installation

After you have installed SnapDrive for UNIX on the host, you need to complete the installation by performing the following tasks:

- Verify that the installation program installed all the necessary files on your host. For more information, see "Understanding the files installed by SnapDrive for UNIX on the host" on page 74.
- Confirm that the configuration variables in the snapdrive.conf file have the correct settings. For the majority of these variables, the default values should be correct. In a few cases, you might need to modify these values. See "Setting configuration information" on page 84.
- Supply SnapDrive for UNIX with the current storage system login information. When you set up your storage system, you supplied a user login for it. SnapDrive for UNIX needs this login information in order to work with the storage system. For information on supplying the login information, see "Specifying the current login information for storage systems" on page 154.
- Specify the access control permissions for each host on each storage system. If you do not specify access control permissions, SnapDrive for UNIX checks the value of the variable `all-access-if-rbac-unspecified` in the snapdrive.conf file. If the variable is set to "on" (the default), it allows the host to perform all SnapDrive for UNIX operations on that storage system. If the value is set to "off", it prevents the host from performing any operations other than the show or list operations. For information on setting permissions, see "Setting up access control" on page 147.

Understanding the files installed by SnapDrive for UNIX on the host

Files installed by the software

When you install SnapDrive for UNIX, it installs a number of files on the host. These files serve different purposes.

Note

There are slight differences in the files, based on your host operating system. In most cases, these differences have to do with the path names used by that host. Also, some hosts use symbolic links (symlinks) to files, while other hosts simply install the file in two locations.

Executables: Following are the executables used by SnapDrive for UNIX.

Host operating system	Path
AIX	/opt/NetApp/snapdrive/bin/snapdrive
HP-UX	◆ /opt/NetApp/snapdrive/bin/snapdrive ◆ /opt/NetApp/snapdrive/bin/snapdrive_parisc
Linux	/opt/NetApp/snapdrive/bin/snapdrive
Solaris	/opt/NTAPsnapdrive/bin/snapdrive

Configuration file: SnapDrive for UNIX stores configuration information in this file. You should modify it for your system. If you upgrade your version of SnapDrive for UNIX, it maintains your current snapdrive.conf file. For more information on setting configuration variables, see "Setting configuration information" on page 84. For information on upgrading your version of SnapDrive for UNIX and how it handles the snapdrive.conf file, see "Upgrading to a new version of SnapDrive for UNIX" on page 79.

Host operating system	Path
AIX, HP-UX, and Linux	/opt/NetApp/snapdrive/snapdrive.conf
Solaris	/opt/NTAPsnapdrive/snapdrive.conf

Uninstall files: If you decide to remove SnapDrive for UNIX, it uses these files. These files are also in the top-level directory on the SnapDrive for UNIX CD-ROM. See "Installing SnapDrive for UNIX on a Solaris host" on page 63 for more information.

Host operating system	Path
AIX	You use SMIT to uninstall SnapDrive for UNIX from an AIX host, so there is no uninstall file.
HP-UX	You use `swremove` command to uninstall SnapDrive for UNIX from a HP-UX host, so there is no uninstall file.
Linux	You use `rpm` command to uninstall SnapDrive for UNIX from a Linux host, so there is not an uninstall file.
Solaris	◆ /opt/NTAPsnapdrive/bin/snapdrive.admin ◆ /opt/NTAPsnapdrive/bin/uninstall

Diagnostic files: You can run the following files if you have a problem with SnapDrive. For more information about these files, see "Data collection utility" on page 298.

Host operating system	Path
AIX	◆ /opt/NetApp/snapdrive/diag/snapdrive.dc ◆ /opt/NetApp/snapdrive/diag/filer_info ◆ /opt/NetApp/snapdrive/diag/brocade_info ◆ /opt/NetApp/snapdrive/diag/cisco_info ◆ /opt/NetApp/snapdrive/diag/mcdata_info ◆ /opt/NetApp/snapdrive/diag/SHsupport.pm ◆ /opt/NetApp/snapdrive/diag/Telnet.pm ◆ /opt/NetApp/snapdrive/diag/aix_info
HP-UX	◆ /opt/NetApp/snapdrive/diag/snapdrive.dc ◆ /opt/NetApp/snapdrive/diag/filer_info ◆ /opt/NetApp/snapdrive/diag/brocade_info ◆ /opt/NetApp/snapdrive/diag/cisco_info ◆ /opt/NetApp/snapdrive/diag/mcdata_info ◆ /opt/NetApp/snapdrive/diag/SHsupport.pm ◆ /opt/NetApp/snapdrive/diag/Telnet.pm ◆ /opt/NetApp/snapdrive/diag/hpux_info
Linux	◆ /opt/NetApp/snapdrive/diag/snapdrive.dc ◆ /opt/NetApp/snapdrive/diag/filer_info ◆ /opt/NetApp/snapdrive/diag/linux_info ◆ /opt/NetApp/snapdrive/diag/SHsupport.pm ◆ /opt/NetApp/snapdrive/diag/Telnet.pm
Solaris	◆ /opt/NTAPsnapdrive/diag/snapdrive.dc ◆ /opt/NTAPsnapdrive/diag/solaris_info ◆ /opt/NTAPsnapdrive/diag/filer_info ◆ /opt/NTAPsnapdrive/diag/brocade_info ◆ /opt/NTAPsnapdrive/diag/cisco_info ◆ /opt/NTAPsnapdrive/diag/mcdata_info ◆ /opt/NTAPsnapdrive/diag/SHsupport.pm ◆ /opt/NTAPsnapdrive/diag/Telnet.pm

Log files: SnapDrive for UNIX creates several log files.

Host operating system	Path
AIX, Linux, and Solaris	◆ /var/log/sd-audit.log ◆ /var/log/sd-recovery.log ◆ /var/log/sd-trace.log
HP-UX	◆ /var/snapdrive/sd-audit.log ◆ /var/snapdrive/sd-recovery.log ◆ /var/snapdrive/sd-trace.log

Man pages: SnapDrive for UNIX provides man pages in several formats.

Host operating system	Path
AIX	◆ /opt/NetApp/snapdrive/docs/snapdrive.dc.1 ◆ /opt/NetApp/snapdrive/docs/snapdrive.1 ◆ /opt/NetApp/snapdrive/docs/snapdrive.1.html ◆ /opt/NetApp/snapdrive/docs/brocade_info.1 ◆ /opt/NetApp/snapdrive/docs/mcdata_info.1 ◆ /opt/NetApp/snapdrive/docs/cisco_info.1 ◆ /opt/NetApp/snapdrive/docs/filer_info.1 ◆ /opt/NetApp/snapdrive/docs/aix_info.1
HP-UX	◆ /opt/NetApp/snapdrive/man/man1/snapdrive.dc.1 ◆ /opt/NetApp/snapdrive/man/man1/snapdrive.1 ◆ /opt/NetApp/snapdrive/docs/snapdrive.1.html ◆ /opt/NetApp/snapdrive/docs/brocade_info.1 ◆ /opt/NetApp/snapdrive/docs/mcdata_info.1 ◆ /opt/NetApp/snapdrive/docs/cisco_info.1 ◆ /opt/NetApp/snapdrive/docs/filer_info.1 ◆ /opt/NetApp/snapdrive/docs/hpux_info.1

Host operating system	Path
Linux	◆ /opt/NetApp/snapdrive/docs/man1/snapdrive.dc.1 ◆ /opt/NetApp/snapdrive/docs/man1/snapdrive.1 ◆ /opt/NetApp/snapdrive/docs/man1/filer_info.1 ◆ /opt/NetApp/snapdrive/docs/man1/linux_info.1 ◆ /opt/NetApp/snapdrive/docs/snapdrive.1.html
Solaris	◆ /usr/share/man/man1/snapdrive.1 ◆ /opt/NTAPsnapdrive/docs/snapdrive.1.man ◆ /opt/NTAPsnapdrive/docs/snapdrive.1.html ◆ /opt/NTAPsnapdrive/docs/solaris_info.1 ◆ /opt/NTAPsnapdrive/docs/brocade_info.1 ◆ /opt/NTAPsnapdrive/docs/mcdata_info.1 ◆ /opt/NTAPsnapdrive/docs/cisco_info.1 ◆ /opt/NTAPsnapdrive/docs/filer_info.1 ◆ /opt/NTAPsnapdrive/docs/snapdrive.dc.1

Upgrading to a new version of SnapDrive for UNIX

Upgrading your version of SnapDrive for UNIX

SnapDrive for UNIX makes it easy for you to upgrade from an earlier version, such as 2.1 or 2.2, to the current version.

You do not have to uninstall SnapDrive for UNIX. Instead, install the latest version of the software on top of the current version.

When you install a new version, SnapDrive for UNIX checks to see if you already have a version installed. If you do, it preserves the current snapdrive.conf file and renames the version of the file it is installing to snapdrive.conf.3.0. This way it avoids overwriting your existing snapdrive.conf file, so you do not lose any changes you made if you customized your settings in the file.

By default, SnapDrive for UNIX comments out the variables in the snapdrive.conf file. This means it automatically uses the default values for all variables except the ones you customize. As a result, SnapDrive for UNIX uses the default values for variables that are new even if you have an earlier version of the snapdrive.conf file. If you want to change these values, you must add the variables to your current snapdrive.conf file and specify the values you want.

For more information on the snapdrive.conf file and its variables, see “Setting configuration information” on page 84.

Changed variable in the snapdrive.conf file

The default value of the following configuration variables is changed in this release:

- `use-https-to-filer=on`

  In the previous releases of SnapDrive 3.0 for UNIX, SnapDrive for UNIX automatically used HTTP protocol to communicate to storage systems on all host platforms. In the SnapDrive 3.0 for UNIX release, by default, SnapDrive for UNIX uses HTTPS protocol to communicate to storage systems on AIX, Solaris, and Linux hosts. If you do not want to use HTTPS on these hosts, set the value of the `use-https-to-filer` variable to "off".

  Attention

  On HP-UX hosts, SnapDrive 3.0 for UNIX uses HTTP protocol to communicate to storage systems.

Note

If you have any previous version of SnapDrive for UNIX installed on your AIX, Solaris, and Linux hosts (which uses HTTP protocol), after you upgrade to SnapDrive 3.0 for UNIX, you should enable SSL on the storage system for SnapDrive 3.0 for UNIX to work. For more information on enabling SSL, see the chapter on managing SSL for SecureAdmin in the *Data ONTAP 7.2 System Administration Guide*.

- `enable-implicit-host-preparation=on`

  The default value of this variable is changed to “on”.

New variables in the snapdrive.conf file

In this release, the following new configuration variables are added to the snapdrive.conf file:

- `enable-split-clone="off"`

  To enable to split the cloned volumes or LUNs during Snapshot connect and Snapshot disconnect operations.

- `fstype="vxfs"`

  To specify the default file system type you want to use for SnapDrive for UNIX operations.

- `space-reservations-enabled=on`

  To enable space reservation when creating LUNs.

- `vmtype="vxvm"`

  To specify the default volume manager type you want to use for SnapDrive for UNIX operations.

For more information on the configuration variables, see Chapter 4, “Determining options and their default values,” on page 86.

Verifying the Veritas stack configuration

To verify the Veritas stack configuration, first ensure you have installed the following in the mentioned sequence:

1. NTAPasl library
2. Veritas licenses
3. Veritas stack (storage foundation)
4. Multipathing licenses
5. SnapDrive for UNIX software

Then perform the following task.

Step	Action
1	Execute the `snapdrive storage connect -lun` *`filername`* to connect to an OS specific device.
2	Execute the `vxdisk list` command to get the device information.

Result: You will get one of the following results:

- If the Veritas configuration on the host is correct, the expected output for the device status will be `online invalid`.
- If the Veritas configuration on the host is incorrect, the expected output for the device status will be `error`.

  This happens when you have installed the Veritas stack without installing the NTAPasl library. To rectify this error, install the NTAPasl library and execute the `vxinstall` command to bring the LUNs and disk groups online.

Configuring and Using SnapDrive for UNIX 4

About this chapter

This chapter provides details about setting your SnapDrive for UNIX configuration options and general information about using SnapDrive for UNIX.

Topics in this chapter

This chapter discusses the following topics:

- "Setting configuration information" on page 84
- "Preparing hosts for adding LUNs" on page 120
- "Setting up audit, recovery, and trace logging" on page 122
- "Setting up AutoSupport" on page 130
- "Setting up multipathing" on page 133
- "Setting up thin provisioning" on page 140
- "General steps for executing commands" on page 142

Setting configuration information

Using the snapdrive.conf file

The snapdrive.conf file controls the configurable variables available in SnapDrive for UNIX. This file contains a name-value pair for each configurable variable.

Use a text editor to modify this file. SnapDrive for UNIX automatically checks the information in this file each time it starts.

The snapdrive.conf file is in the SnapDrive for UNIX installation directory (see the installation instructions for your operating system for the complete path to this directory and also, see Chapter 3, "Understanding the files installed by SnapDrive for UNIX on the host," on page 74).

SnapDrive for UNIX also provides some commands you can use to work with this file, such as the `snapdrive config show` command, which displays this file.

Verify the settings in snapdrive.conf

The snapdrive.conf file comes with most of the variables set to default values. In most cases, these are the values you should use when you run SnapDrive for UNIX. To complete the SnapDrive for UNIX installation, you should verify the default values in the snapdrive.conf file. In particular, you should check the values for the following variables:

- Logging
- AutoSupport
- Accurate path information for the system

To view the current settings, execute the `snapdrive config show` command.

Some operations available with the snapdrive config command

The `snapdrive config` command has several formats. These commands allow you to perform a number of configuration tasks, including working with the snapdrive.conf file, and setting the user and password information that SnapDrive for UNIX needs to access the storage systems, setting access control permissions, and determining LUN configuration information for your system. The following is a brief overview of the `snapdrive config` command formats.

Command	Description
`snapdrive config show [host_file_name]`	This command displays the snapdrive.conf current name-value pairs and their defaults. If you include the *host_file_name* argument, it writes information to the file that you specify on the host.
◆ `snapdrive config list` ◆ `snapdrive config set user_name filername [filename . . .]` ◆ `snapdrive config delete filename [filename . . .]`	These versions of the `config` command handle user logins for the storage systems. Use the `snapdrive config list` command to display the user names that have been configured in SnapDrive for UNIX for accessing storage systems. The `set` version of the command lets you tell SnapDrive for UNIX which user name-password pair to use to access a storage system. The `delete` command removes the specified user name-password pair from SnapDrive for UNIX. For more information, see "Specifying the current login information for storage systems" on page 154.
`snapdrive config access {list | show} filername`	This operation lets you display the access control settings for a host trying to access a storage system. For more information, see "Setting up access control" on page 147.
◆ `snapdrive config prepare luns -count count` ◆ `snapdrive config check luns`	These commands let you prepare the host for creating a specific number of LUNs as well as to determine how many LUNs you can create. For more information, see "Preparing hosts for adding LUNs" on page 120. **Note** You only need these commands on Linux and Solaris hosts. AIX and HP-UX hosts do not require any preparation prior to creating LUNs.

Command	Description
`snapdrive config check cluster`	This operation lets you check for the following in the SFRAC cluster environment on a Solaris host: ◆ SnapDrive for UNIX version ◆ GAB configuration ◆ Cluster status ◆ CVM status ◆ Usage of `rsh` or `ssh` for a secure communication within the cluster nodes ◆ Differences in setting the following configuration variable values in the snapdrive.conf file: ❖ `default-transport`= "fcp" ❖ `multipathing-type`="DMP"

Determining options and their default values

The supported configurable items and their default settings can vary across host operating systems and the different versions of SnapDrive for UNIX. An easy way to determine current configurable items and their settings is to execute the `snapdrive config show` command. The following example shows output that the `snapdrive config show` option can produce. You should run the `snapdrive config show` command to get a current copy of the file for your host platform.

Note

If you are running SnapDrive for UNIX on a different host operating system, some of the defaults might be different. For example, on an HP-UX host, the default path for a log file is /var/snapdrive/... instead of the AIX, Linux or Solaris default path of /var/log/... .

The following is an example of a snapdrive.conf file:

```
# Snapdrive Configuration
#       file: /opt/NetApp/snapdrive/snapdrive.conf
#       Version 3.0     (Change 666767 Built Tue May 15 10:22:26 IST 2007)
#
#
# Default values are shown by lines which are commented-out in this file.
# If there is no un-commented-out line in this file relating to a particular value, then
# the default value represented in the commented-out line is what SnapDrive will use.
#
# To change a value:
#
#     -- copy the line that is commented out to another line
#     -- Leave the commented-out line
#     -- Modify the new line to remove the '#' and to set the new value.
#     -- Save the file and exit
#
audit-log-file="/var/log/sd-audit.log"  # audit log file
trace-log-file="/var/log/sd-trace.log"  # trace log file
recovery-log-file="/var/log/sd-recovery.log"  # recovery log file
autosupport-enabled=off  # Enable autosupport (requires autosupport-filer be set)
autosupport-filer=""  # Filer to use for autosupport (filer must be configured for
autosupport)
audit-log-max-size=20480  # Maximum size (in bytes) of audit log file
audit-log-save=2  # Number of old copies of audit log file to save
available-lun-reserve=8  # Number of LUNs for which to reserve host resources
cluster-operation-timeout-secs=600  # Cluster Operation timeout in seconds
contact-http-port=80  # HTTP port to contact to access the filer
contact-ssl-port=443  # SSL port to contact to access the filer
device-retries=3  # Number of retries on Ontap filer LUN device inquiry
device-retry-sleep-secs=1  # Number of seconds between Ontap filer LUN device inquiry
retries
enable-implicit-host-preparation=on  # Enable implicit host preparation for LUN
creation
filer-restore-retries=140  # Number of retries while doing lun restore
filer-restore-retry-sleep-secs=15  # Number of secs between retries while restoring lun
filesystem-freeze-timeout-secs=300  # File system freeze timeout in seconds
default-noprompt=off  # A default value for -noprompt option in the command line
mgmt-retries=3  # Number of retries on ManageONTAP control channel
mgmt-retry-sleep-secs=2  # Number of seconds between retries on ManageONTAP control
channel
mgmt-retry-sleep-long-secs=90  # Number of seconds between retries on ManageONTAP
control channel (failover error)
prepare-lun-count=16  # Number of LUNs for which to request host preparation
PATH="/sbin:/usr/sbin:/bin:/usr/bin:/opt/NTAP/SANToolkit/bin:/opt/sanlun/bin"  #
toolset search path
```

```
password-file="/opt/NetApp/snapdrive/.pwfile"  # location of password file
prefix-filer-lun=""  # Prefix for all filer LUN names internally generated by storage
create
recovery-log-save=20  # Number of old copies of recovery log file to save
snapcreate-consistency-retries=3  # Number of retries on best-effort snapshot
consistency check failure
snapcreate-consistency-retry-sleep=1  # Number of seconds between best-effort snapshot
consistency retries
snapcreate-must-make-snapinfo-on-qtree=off  # snap create must be able to create
snapinfo on qtree
snapcreate-cg-timeout="relaxed"  # Timeout type used in snapshot creation with
Consitency Groups. Possible values are "urgent", "medium" or "relaxed".
snapcreate-check-nonpersistent-nfs=off  # Check that entries exist in /etc/fstab for
specified nfs fs.
enable-split-clone="off"  # Enable split clone volume or lun during connnect/disconnect
snapconnect-nfs-removedirectories=off  # NFS snap connect cleaup unwanted dirs;
snapdelete-delete-rollback-with-snap=off  # Delete all rollback snapshots related to
specified snapshot
snaprestore-snapmirror-check=on  # Enable snapmirror destination volume check in snap
restore
snaprestore-delete-rollback-after-restore=on  # Delete rollback snapshot after a
successfull restore
snaprestore-make-rollback=on  # Create snap rollback before restore
snaprestore-must-make-rollback=on  # Do not continue 'snap restore' if rollback
creation fails
space-reservations-enabled=on  # Enable space reservations when creating new luns
snapmirror-dest-multiple-filervolumes-enabled=off  # Enable snap restore and snap
connect commands to deal with snapshots moved to another filer volume (e.g. via
SnapMirror) where snapshot spans multiple filers or volumes
default-transport="fcp"  # Transport type to use for storage provisioning, when a
decision is needed
multipathing-type="none"  # Multipathing software to use when more than one
multipathing solution is available
fstype="ext3"  # Default File System type to be used
vmtype="lvm"  # Default Volume Manager type to be used
trace-enabled=on  # Enable trace
secure-communication-among-cluster-nodes=off  # Enable Secure Communication
trace-level=7  # Trace levels: 1=FatalError; 2=AdminError; 3=CommandError; 4=warning,
5=info, 6=verbose
trace-log-max-size=0  # Maximum size of trace log file in bytes; 0 means one trace log
file per command
trace-log-save=100  # Number of old copies of trace log file to save
all-access-if-rbac-unspecified=on  # Allow all access if the RBAC permissions file is
missing
use-https-to-filer=on # Communication with filer done via HTTPS instead of HTTP
```

The following table describes the configuration variables in the snapdrive.conf file.

Variable	Description
`all-access-if-rbac-unspecified=on`	Specifies access control permissions for each host where SnapDrive for UNIX runs by entering the permission string in an access control file. The string you specify controls which SnapDrive for UNIX Snapshot copy and storage operations that host may perform on a given storage system. (These access permissions do not affect the show or list operations.) Set this value to either "on" or "off" where: ◆ on— Specifies SnapDrive for UNIX to enable all access permissions if there is no access control permissions file on the storage system. This is the default value. ◆ off— Specifies the storage system allow the host only the permissions specified in the access control permissions file. If you provide an access control file, this option has no effect. For more information on access control and how to set it up, see "Setting up access control" on page 147.
`audit-log-file="/var/log/sd-audit.log"`	Specifies the location where SnapDrive for UNIX writes the audit log file. The default value depends on your host operating system. The path shown in this example is the default path for a Linux, AIX, or Solaris host. For an HP-UX host, the default path is /var/snapdrive/sd-audit.log. For more information on the audit log file, see "Changing the defaults for audit logs" on page 125.

Variable	Description
`audit-log-max-size=20480`	Specifies the maximum size, in bytes, of the audit log file. When the file reaches this size, SnapDrive for UNIX renames it and starts a new audit log. The default value is 20,480 bytes. Because SnapDrive for UNIX never starts a new log file in the middle of an operation, the actual size of the file could vary slightly from the value specified here. **Note** NetApp recommends that you use the default value. If you decide to change the default value, remember that too many large log files can take up space on your disk and might eventually affect performance.
`audit-log-save=2`	Determines how many old audit log files SnapDrive for UNIX should save. Once this limit is reached, SnapDrive for UNIX discards the oldest file and creates a new one. SnapDrive for UNIX rotates this file based on the value you specify in the audit-log-max-size option. For more information on log rotation, see “Settings affecting log file rotation” on page 124. The default value is 2. **Note** NetApp recommends that you use the default value. If you decide to change the default value, remember that too many log files can take up space on your disk and might eventually affect the performance.

Variable	Description
`autosupport-enabled=off`	Specifies the AutoSupport option for storage systems. Set this value to either “on” to enable AutoSupport or “off” to disable it. The default value is “off” because you need to perform additional setup steps to enable it, including specifying a value for the `autosupport-filer` option. NetApp recommends that you enable this option. See “Setting up AutoSupport” on page 130 for details about how to enable this option.
`autosupport-filer=" "`	Specifies the name or IP address of the storage system that AutoSupport should use to send the message. To disable this option, leave the storage system name blank. This option is disabled by default because it requires information specific to your setup. NetApp recommends that you enable this option. To enable AutoSupport, you must enter a value here and also set `autosupport-enabled` to “on”. See “Setting up AutoSupport” on page 130 for details about how to enable this option.

Variable	Description
`available-lun-reserve=8`	Specifies the number of LUNs that the host must be prepared to create when the current SnapDrive for UNIX operation completes. If very few operating system resources are available to create the number of LUNs specified, SnapDrive for UNIX requests additional resources, based on the value supplied with the `enable-implicit-host-preparation` variable. The default value is 8. **Note** This variable applies only to those systems that require host preparation before you can create LUNs. At present, only Linux and Solaris hosts require this preparation. This variable is only used on configurations that include LUNs.

Variable	Description

`cluster-operation-timeout-secs=600`	Specifies the cluster operation timeout, in seconds. Set this value when working with remote nodes and cluster-wide operations, to determine when the SnapDrive for UNIX operation should time out. The default value is “600” seconds.
	
	Other than the nonmaster node, the cluster master node can also be the remote node, if the SnapDrive for UNIX operation is initiated from a nonmaster node.
	
	If SnapDrive for UNIX operations on any node in the cluster exceed the value you set, or the default of “600” seconds (if you set no value), the operation timesout with the following message:
	
	`Remote Execution of command on slave node sfrac-57 timed out.`
	`Possible reason could be that timeout is too less for that system.`
	`You can increase the cluster connect timeout in snapdrive.conf file.`
	`Please do the necessary cleanup manually.`
	`Also, please check the operation can be restricted to lesser jobs to be done so that time required is reduced.`

`contact-http-port=80`	Specifies the HTTP port to use for communicating with a storage system. If not specified, a default value of “80” is used.
`contact-ssl-port=443`	Specifies the SSL port to use for communicating with a storage system. If not specified, a default value of “443” is used.
`default-noprompt=off`	Specify if you want the `-noprompt` option to be available. The default value is “off” (not available). If you change this option to “on”, SnapDrive for UNIX does not prompt you to confirm an action requested by `-force`.

Variable	Description
`device-retries=3`	Specifies the number of times SnapDrive for UNIX attempts to inquire about the device where the LUN is. The default value is “3”. Specify the number of inquiries that the SnapDrive for UNIX can make about the device where the LUN is located. In normal circumstances, the default value should be adequate. In other circumstances, LUN queries for a snap create operation could fail simply because the storage system is exceptionally busy. If the LUN queries keep failing even though the LUNs are online and correctly configured, you might want to increase the number of retries. This variable is only used on configurations that include LUNs. **Note** NetApp recommends that you configure the same value for the `device-retries` option across all the nodes in the cluster. Otherwise, the device discovery involving multiple cluster nodes can fail on some nodes and succeed on others.

Variable	Description
`device-retry-sleep-secs=1`	Specifies the number of seconds SnapDrive for UNIX waits between inquiries about the device where the LUN resides. The default value is “1” second. In normal circumstances, the default value should be adequate. In other circumstances, LUN queries for a snap create operation could fail simply because the storage system is exceptionally busy. If the LUN queries keep failing even though the LUNs are online and correctly configured, you might want to increase the number of seconds between retries. This variable is only used on configurations that include LUNs. **Note** NetApp recommends that you configure the same value for the `device-retry-sleep-secs` option across all the nodes in the cluster. Otherwise, the device discovery involving multiple cluster nodes can fail on some nodes and succeed on others.

Variable	Description
`default-transport="iscsi" or "fcp"`	Specifies the protocol that SnapDrive for UNIX uses as the transport type, when creating storage, if a decision is required. The acceptable values are “iscsi” or “fcp”. **Note** If a host is configured for only one type of transport and that type is supported by SnapDrive for UNIX, SnapDrive for UNIX uses that transport type, irrespective of the type specified in the snapdrive.conf file. On AIX hosts, ensure the `multipathing-type` option is set correctly. If you specify “fcp”, you must set `multipathing-type` to one of the following values: ◆ NativeMPIO ◆ SANPath ◆ DMP If you specify “iscsi”, you must specify “none” as the value for the `multipathing-type` variable. **Note** If SnapDrive for UNIX operations involve shared disk groups and file systems, you have to specify FCP for the `default-transport` variable across all the nodes in the cluster. Otherwise, the storage creation fails with an error.

Variable	Description
`enable-implicit-host-preparation="on"`	Determines whether SnapDrive for UNIX implicitly requests host preparation for LUNs or simply notifies you that it is required and exits. ◆ "on"—SnapDrive for UNIX implicitly requests the host to make more resources available if there will not be enough resources to create the correct number of LUNs once the current command completes. The number of LUNs being created is specified in the available-lun-reserve variable. This is the default value. ◆ "off"—SnapDrive for UNIX informs you if additional host preparation is necessary for the LUN creation and exits. You can then perform the operations necessary to free up resources needed for the LUN creation. For example, you can execute the `snapdrive config prepare luns` command. Once the preparation is complete, you can re-enter the current SnapDrive for UNIX command. **Note** This variable applies only to systems where host preparation is needed before you can create LUNs. Currently, only Linux and Solaris hosts require that preparation. This variable is only used on configurations that include LUNs.

Variable	Description
`enable-split-clone="off"`	Enables splitting the cloned volumes or LUNs during Snapshot connect and Snapshot disconnect operations, if this variable is set to “on” or “sync”. You can set the following values for this variable: ◆ on—enables an asynchronous split of cloned volumes or LUNs. ◆ sync—enables a synchronous split of cloned volumes or LUNs. ◆ off—disables the split of cloned volumes or LUNs. This the default value. If you set this value to “on” or “sync” during the Snapshot connect operation and “off” during the Snapshot disconnect operation, SnapDrive for UNIX will not delete the original volume or LUN that is present in the Snapshot copy. You can also split the cloned volumes or LUNs by using the `-split` option. For more information on this option, see Chapter 9, “SnapDrive for UNIX options, keywords, and arguments,” on page 348.
`filer-restore-retries=140`	Specifies the number of times SnapDrive for UNIX attempts to restore a Snapshot copy on a storage system if a failure occurs during the restore. The default value is “140”. In normal circumstances, the default value should be adequate. Under other circumstances, this operation could fail simply because the storage system is exceptionally busy. If it keeps failing even though the LUNs are online and correctly configured, you might want to increase the number of retries.

Variable	Description
`filer-restore-retry-sleep-secs=15`	Specifies the number of seconds SnapDrive for UNIX waits between attempts to restore a Snapshot copy. The default value is “15” seconds. In normal circumstances, the default value should be adequate. Under other circumstances, this operation could fail simply because the storage system is exceptionally busy. If it keeps failing even though the LUNs are online and correctly configured, you might want to increase the number of seconds between retries.
`filesystem-freeze-timeout-secs=300`	Specifies the amount of time, in seconds, that SnapDrive for UNIX waits when it cannot access the file system, before trying again. The default value is “300” seconds (5 minutes). This variable is only used on configurations that include LUNs.

Variable	Description
`fstype="vxfs"`	Specifies the type of file system that you want to use in SnapDrive for UNIX operations. The file system must be a type that SnapDrive for UNIX supports for your operating system. Following are the values that you can set for this variable; the default value is different for different host operating systems: ◆ AIX: “jfs”, “jfs2” or “vxfs” The default value is “jfs2”. **Note** The JFS file system type is supported only for Snapshot operations and not for storage operations. ◆ HP-UX: “vxfs” ◆ Linux: “ext3” ◆ Solaris: “vxfs” or “ufs” The default value is “vxfs”. You can also specify the type of file system that you want to use by using the `-fstype` option. For more information on this option, see Chapter 9, “SnapDrive for UNIX options, keywords, and arguments,” on page 348.
`mgmt-retries=3`	Specifies the number of times SnapDrive for UNIX retries an operation on the ManageONTAP control channel after a failure occurs. The default value is “3”.
`mgmt-retry-sleep-secs=2`	Specifies the number of seconds SnapDrive for UNIX waits before retrying an operation on the ManageONTAP control channel. The default value is “2” seconds.
`mgmt-retry-sleep-long-secs=90`	Specifies the number of seconds SnapDrive for UNIX waits before retrying an operation on the ManageONTAP control channel after a failover error occurs. The default value is “90” seconds.

Variable	Description
`multipathing-type="DMP"`	Specifies the multipathing software to use. The default value depends on the host operating system. This option applies only if one of the following is true: ◆ There is more than one multipathing solution available. ◆ The configurations that include LUNs. **Note** If a host is configured for only one type of multipathing, and that type is supported by SnapDrive for UNIX, SnapDrive for UNIX uses that multipathing type, irrespective of the type specified in the snapdrive.conf file. Following are the values that you can set for this variable: **AIX:** The value you set for AIX depends on which protocol you are using. ◆ If you are using FCP, set this to any one of the following values: ❖ NativeMPIO ❖ SANPath ❖ DMP The default value is “SANPath”. ◆ In addition, set the `default-transport` option to “fcp”. ◆ If you are using iSCSI, set this value to “none”. In addition, set the `default-transport` option to “iscsi”. **Note** For the SnapDrive 3.0 for UNIX release, multipathing is not supported on AIX hosts running iSCSI.

Variable	Description
	Solaris: Solaris 10, update 1, you can set the “SOLMPxIO” value to enable multipathing using Solaris MPxIO. To enable multipathing by using SOLMPxIO, you have to add the following lines to the /kernel/drv/scsi_vhci.conf file: `device-type-scsi-options-list = "NETAPP LUN ", ", "symmetric-option"; symmetric-option = 0x1000000;` If Snapdrive for UNIX operations involve shared disk groups and file systems, do either of the following: ◆ Configure this option to "none" value if you do not want multipathing. ◆ Configure this option to "DMP" value if you want VxDMP explicitly on a system where multiple multipathing solutions are available. **Note** Ensure that the `multipathing-type` variable is set to same value across all the nodes in the cluster. **HP-UX:** On this platform, you can set “PVLinks’ and “DMP” values for multipathing. **Linux:** For SnapDrive 3.0 for UNIX release, multipathing is not supported on Linux host, so the default value is “none”.

Variable	Description
`PATH="/sbin:/usr/sbin:/bin:/usr/lib/vxvm/bin:/usr/bin:/opt/NTAP/SANToolkit/bin:/opt/NTAPsanlun/bin:/opt/VRTS/bin:/etc/vx/bin"`	Specifies the search path the system uses to look for tools. NetApp recommends that you verify that this is the correct path for your system. If it is incorrect, change it to the correct path. The default value might vary depending on your operating system. In this case, this is the default path for a Solaris, Linux, or an HP-UX host. AIX hosts do not use this variable because they process the commands differently. **Note** In the snapdrive.conf file of the previous release (SnapDrive 2.1 for UNIX and earlier), the `PATH` variable does not include the new path of the `sanlun` utility.You must copy the new path of `sanlun` specified in the snapdrive.conf.3.0 file to the snapdrive.conf file.
`password-file="/opt/NTAPsnapdrive/.pwfile"`	Specifies the location of the password file for the user login for the storage systems. The default value depends on your operating system. For example, this is the default path for a Solaris host. The default path for an AIX, Linux, or HP-UX host is /opt/NetApp/snapdrive.

Variable	Description
`prefix-filer-lun="  "`	Specifies the prefix that SnapDrive for UNIX applies to all LUN names it generates internally. The default value for this prefix is the empty string. This variable allows the names of all LUNs created from the current host, but not explicitly named on a SnapDrive for UNIX command line, to share an initial string. **Note** This variable is only used on configurations that include LUNs.
`prepare-lun-count=16`	Specifies how many LUNs SnapDrive for UNIX should prepare to create. SnapDrive for UNIX checks this value when there is a request to prepare the host to create additional LUNs. The default value is 16. That means the system will be able to create 16 additional LUNs after the preparation is complete. **Note** This variable applies only to systems where host preparation is needed before you can create LUNs. This variable is only used on configurations that include LUNs. Currently, only Linux and Solaris hosts require that preparation.
`recovery-log-file="/var/log/sd-recovery.log"`	Specifies where SnapDrive for UNIX writes the recovery log file. The default value depends on your host operating system. The path shown in this example is the default path for a Linux, AIX, or Solaris host. For an HP-UX host, the default path is /var/snapdrive/sd-recovery.log. See “Contents of a recovery log” on page 126 for more information.

Variable	Description
`recovery-log-save=20`	Specifies how many old recovery log files SnapDrive for UNIX should save. After this limit is reached, SnapDrive for UNIX discards the oldest file when it creates a new one. SnapDrive for UNIX rotates this log file each time it starts a new operation. For more information on log rotation, see “Settings affecting log file rotation” on page 124. The default value is “20”. **Note** NetApp recommends that you use the default value. If you decide to change the default, remember that having too many large log files can take up space on your disk and might eventually affect performance.

Variable	Description
secure-communication-among-cluster-nodes=on	Specifies a secure communication within the cluster nodes for remote execution of SnapDrive for UNIX commands. You can direct SnapDrive for UNIX to use rsh or ssh by changing this configuration variable. The rsh or ssh methodology adopted by SnapDrive for UNIX for remote execution is decided only by the value set in the /opt/NTAPsnapdrive/snapdrive.conf file of the following two components: ◆ The host in which the SnapDrive for UNIX operation is executed, to get the host WWPN information and device path information of remote nodes. For example, snapdrive storage create executed on master cluster node uses the rsh or ssh configuration variable only in the local snapdrive.conf file to do either of the following: ❖ To determine the remote communication channel ❖ To execute devfsadm command on remote nodes ◆ The nonmaster cluster node, if the SnapDrive for UNIX command is to be executed remotely on the master cluster node. To send the SnapDrive for UNIX command to the cluster master node, the rsh or ssh configuration variable in the local snapdrive.conf file is consulted to determine the rsh or ssh mechanism for remote command execution. The default value of “on” means that ssh will be used for remote command execution. The value “off” means that rsh will be used for execution.

Variable	Description
`snapcreate-cg-timeout=relaxed`	Specifies the interval that the `snapdrive snap create` command allows for a storage system to complete fencing. Values are as follows: ◆ urgent—specifies a short interval. ◆ medium—specifies an interval between urgent and relaxed. ◆ relaxed—specifies the longest interval. This is the default value. If a storage system does not complete fencing within the time allowed, SnapDrive for UNIX creates a Snapshot copy using the methodology for Data ONTAP versions before 7.2. See "Crash-consistent Snapshot copies" on page 238.
`snapcreate-check-nonpersistent-nfs=on`	Enables and disables the Snapshot create operation to work with a nonpersistent NFS file system. The value for this option are as follows: ◆ on—SnapDrive for UNIX checks whether NFS entities specified in the `snapdrive snap create` command are present in the file system mount table. The Snapshot create operation fails if the NFS entities are not persistently mounted through the file system mount table. This is the default value. ◆ off—SnapDrive for UNIX creates a Snapshot copy of NFS entities that do not have a mount entry in the file system mount table. The Snapshot restore operation automatically restores and mounts the NFS file or directory tree that you specify. You can use the `-nopersist` option in the `snapdrive snap connect` command for NFS file systems to prevent adding of mount entries in the file system mount table.

Variable	Description
`snapcreate-consistency-retry-sleep=1`	Specifies the number of seconds between best-effort Snapshot copy consistency retries. The default value is “1” second.
`snapconnect-nfs-removedirectories=off`	Determines whether SnapDrive for UNIX deletes or retains the unwanted NFS directories from the FlexClone volume during the Snapshot connect operation. ◆ on—Deletes the unwanted NFS directories (storage system directories not mentioned in the `snapdrive snap connect` command) from the FlexClone volume during the Snapshot connect operation. The FlexClone volume is destroyed if it is empty during the Snapshot disconnect operation. ◆ off—Retains the unwanted NFS storage system directories during the Snapshot connect operation. This is the default value. During the Snapshot disconnect operation, only the specified storage system directories are unmounted from the host. If nothing is mounted from the FlexClone volume on the host, the FlexClone volume is destroyed during the Snapshot disconnect operation. If this variable is set to “off” during the connect operation or “on” during the disconnect operation, the FlexClone volume will not be destroyed, even if it has unwanted storage system directories: that is, if it is not empty.

Variable	Description
`snapcreate-must-make-snapinfo-on-qtree=off`	Set this value to “on” to enable the Snapshot create operation to create Snapshot copy information on a qtree. The default value is “off” (disabled). SnapDrive for UNIX always attempts to write snapinfo at the root of a qtree if the LUNs being snapped are at the root of that qtree. Setting this option to “on” means that SnapDrive for UNIX fails the Snapshot create operation if it cannot write this data. You should only turn this option on if you are mirroring Snapshot copies using Qtree SnapMirror® software. **Note** Snapshot copies of qtrees work the same way Snapshot copies of volumes do.
`snapcreate-consistency-retries=3`	Specifies the number of times SnapDrive for UNIX attempts a consistency check on a Snapshot copy after it receives a message that a consistency check failed. This option is particularly useful on host platforms that do not include a freeze function. Currently, Linux is the only platform without that function. This variable is only used on configurations that include LUNs. For more information, see “Crash-consistent Snapshot copies” on page 238. The default value is “3”.

Variable	Description
`snapdelete-delete-rollback-with-snap=off`	Set this value to “on” to delete all rollback Snapshot copies related to a Snapshot copy. Set it to “off” to disable this feature. The default value is “off” (disabled). This option only takes effect during a Snapshot delete operation and is used by the recovery log file if you encounter a problem with an operation. NetApp recommends that you accept the default setting.
`snapmirror-dest-multiple-filervolumes-enabled=off`	Set this value to “on” to restore Snapshot copies that span multiple storage systems or volumes on (mirrored) destination storage systems. The default value is “off”.
`snaprestore-delete-rollback-after-restore=on`	Set this value to “on” to delete all rollback Snapshot copies after a successful Snapshot restore operation. Set it to “off” to disable this feature. The default value is “on” (enabled). This option is used by the recovery log file if you encounter a problem with an operation. NetApp recommends that you accept the default setting.

Variable	Description
`snaprestore-make-rollback=on`	Set this value to either “on” to create a rollback Snapshot copy or “off” to disable this feature. The default value is “on” (enabled). A rollback is a copy of the data on the storage system that SnapDrive for UNIX makes before it begins a Snapshot restore operation. That way, if a problem occurs during the Snapshot restore operation, you can use the rollback Snapshot copy to restore the data to the state it was in before the operation began. If you do not want the extra security of a rollback Snapshot copy at restore time, set this option to “off”. If you like the rollback, but not enough to fail your Snapshot restore operation if you cannot make one, set the option `snaprestore-must-make-rollback` to “off”. This option is used by the recovery log file, which you send to NetApp technical support if you encounter a problem. See “Contents of a recovery log” on page 126. NetApp recommends that you accept the default setting.

Variable	Description
`snaprestore-must-make-rollback=on`	Set this value to “on” to fail a Snapshot restore operation if the rollback creation fails. Set the value to “off” to disable this feature. The default value is “on” (enabled). ◆ on—SnapDrive for UNIX attempts to make a rollback copy of the data on the storage system before it begins the Snapshot restore operation. If it cannot make a rollback copy of the data, SnapDrive for UNIX halts the Snapshot restore operation. ◆ off—If you want the extra security of a rollback Snapshot copy at restore time, but not enough to fail the Snapshot restore operation if you cannot make one. This option is used by the recovery log file if you encounter a problem with an operation. NetApp recommends that you accept the default setting.
`snaprestore-snapmirror-check=on`	Set this value to “on” to enable the `snapdrive snap restore` command to check the SnapMirror destination volume. If it is set to “off”, the `snapdrive snap restore` command is unable to check the destination volume. The default value is “on”.

Variable	Description
space-reservations-enabled=on	Enables space reservation when creating LUNs. By default, this option is set to "on"; therefore, the LUNs created by SnapDrive for UNIX have space reservation. This parameter can be used to disable the space reservation for LUNs created by the `snapdrive snap connect` command and `snapdrive storage create` command. NetApp recommends using `-reserve` and `-noreserve` command-line options to enable or disable LUN space reservation in the `snapdrive storage create`, `snapdrive snap connect`, and `snapdrive snap restore` commands. For more information, see "Setting up thin provisioning" on page 140. SnapDrive for UNIX creates LUNs, resizes storage, makes Snapshot copies, and connects or restores the Snapshot copies based on the space reservation permission that is specified in this variable or by the usage of`-reserve` or `-noreseve` command-line options. It does not consider the storage system-side thin provisioning options before performing the preceding tasks.
trace-enabled=on	Set this value to "on" to enable the trace log file, or to "off" to disable it. This file is used by NetApp. The default value is "on". Enabling this file does not affect performance. For more information, see "About the trace log file" on page 128.

Variable	Description
`trace-level=5`	Specifies the types of messages SnapDrive for UNIX writes to the trace log file. This option accepts the following values: ◆ 1—Record fatal errors ◆ 2—Record admin errors ◆ 3—Record command errors ◆ 4—Record warnings ◆ 5—Record information messages ◆ 6—Record in verbose mode The default value is “5”.
`trace-log-file="/var/log/sd-trace.log"`	Specifies where SnapDrive for UNIX writes the trace log file. The default value depends on your host operating system. The path shown in this example is the default path for a Linux, AIX, or Solaris host. For a HP-UX host, the default path is /var/snapdrive/sd-trace.log. For more information, see “About the trace log file” on page 128.

Variable	Description
`trace-log-max-size=0`	Specifies the maximum size of the trace log file in bytes. When the file reaches this size, SnapDrive for UNIX renames it and starts a new trace log. The default value is "0". This value means that for every command, SnapDrive for UNIX creates a separate trace file. SnapDrive for UNIX never starts a new log file in the middle of an operation the actual size of the file could vary slightly from the value specified here. **Note** NetApp recommends that you use the default value. If you decide to change the default, remember that having too many large log files can take up space on your disk and might eventually affect performance.
`trace-log-save=100`	Specifies how many old trace log files SnapDrive for UNIX should save. After this limit is reached, SnapDrive for UNIX discards the oldest file when it creates a new one. This variable works with `trace-log-max-size` variable. By default, `trace-log-max-size=0` saves one command in each file, and `trace-log-save=100` retains the last log 100 files. For more information on log files, see "Settings affecting log file rotation" on page 124.

Variable	Description
`use-https-to-filer=on`	Determines whether you want SnapDrive for UNIX to use SSL encryption (HTTPS) when it communicates with storage system. The default value is “on”. **Note** If you are using a version of Data ONTAP prior to 7.0, you might see slower performance with HTTPS enabled. This is not an issue if you are running Data ONTAP 7.0 or later. On AIX, Solaris, and Linux hosts, the prior versions of SnapDrive 3.0 for UNIX uses HTTP by default to communicate with storage system. So after upgrade or reinstall, the value of the `use-https-to-filer` variable should be set to “off” if you want to continue to use HTTP protocol. On the HP-UX host, SnapDrive 3.0 for UNIX uses HTTP protocol to communicate to storage systems. For more information on the security features available with SnapDrive for UNIX, see the chapter “Setting Up Security Features” on page 145.

Variable	Description
`vmtype="vxvm"`	Specify the type of volume manager you want to use for SnapDrive for UNIX operations. The volume manager must be a type that SnapDrive for UNIX supports for your operating system. Following are the values that you can set for this variable, and the default value is different for different host operating systems: ◆ AIX and HP-UX: “vxvm” or “lvm” The default value is “lvm” ◆ Linux: “lvm” ◆ Solaris: “vxvm” You can also specify the type of volume manager that you want to use by using the `-vmtype` option. For more information on this option, see “SnapDrive for UNIX options, keywords, and arguments” on page 348.

Setting values in snapdrive.conf

To change the values in the snapdrive.conf file or add new name-value pairs, complete the following steps.

Step	Action
1	Log on as root.
2	Make a backup of the snapdrive.conf file.
3	Open the snapdrive.conf file in a text editor.

4	Make the changes to the file.
	To add a name-value pair, use the following format:
	`config-option-name=value # optional comment`
	config-option-name is the name of the option you want to configure; for example, audit-log-file.
	value is the value you want to assign to this option.
	If you want to include a comment with the name-value pair, precede the comment with a pound sign (#).
	You should enter only one name-value pair per line.
	If the name or the value uses a string, enclose the string in either single (`'`) or double (`"`) quotation marks. You can place the quotation marks around either the entire name-value pair or just the value. The following are three examples of how you can use quotes and comments with name-value pairs:
	`"config-option-one=string with white space" # double quotes around the pair`
	`config-option-two="string with white space" # double quotes around the value`
	`config-option-2B='string with white space' # single quotes around the value`

5	To modify an existing name-value pair, replace the current value with the new value. The best way to do this is to perform the following steps: **a.** Comment out the line you want to modify. **b.** Copy the commented-out line. **c.** Un-comment the copy by removing the pound sign. **d.** Modify the value. This way you always have a record of the default value in the file.

Step	Action
	If you want to specify a blank value (for example, you want to disable the audit log file), enter a pair of double quotation marks (`""`).
6	Save the file after you make your changes. SnapDrive for UNIX automatically checks this file each time it starts. Your changes take effect the next time it starts.

Checking your version of SnapDrive for UNIX

To determine which version of SnapDrive for UNIX you are using, complete the following step.

Step	Action

1	Enter the following command:
	
	`snapdrive version`
	
	Example: SnapDrive for UNIX displays its version information when you enter this command:
	
	`# snapdrive version`
	`snapdrive Version 3.0`
	
	Note
	The only argument this command accepts is `-v` (verbose), which displays additional version details. If you include additional arguments, SnapDrive for UNIX displays a warning and then the version number.

Preparing hosts for adding LUNs

Checking host information

If your operating system requires that you prepare it before you create new LUNs, you can use the `snapdrive config` command. This command lets you check information about how many LUNs can be created on a storage system that is mapped to your host.

Not all host platform operating systems require that you prepare the host. Currently, these commands are required only on Linux and Solaris hosts.

Determining how many LUNs can be created

SnapDrive for UNIX lets you determine how many LUNs can be created on the host without exceeding a host-local limit. Use the following command to determine this value:

```
snapdrive config check luns
```

On a Linux host, this command checks the existing /dev/sg files to determine how many are unused.

On a Solaris host, this command scans /kernel/drv/sd.conf to determine how many unused entries it has that would be suitable for LUNs.

Adding host entries for new LUNs

You can also make sure the host is prepared for the creation of a specific number of new LUNs. These LUNs reside on a storage system that is currently mapped to the host. Use the following command:

```
snapdrive config prepare luns -count count [-devicetype shared]
```

`-count` is the number of new LUNs for which you want the host to be prepared.

`-devicetype` is the type of device to be used for SnapDrive for UNIX operations. When specified as `-devicetype shared`, the `snapdrive config prepare luns` command runs on all the nodes in the cluster.

Note

In an SFRAC cluster environment, this command is executed on all nodes in the cluster.

On Linux, this command adds a new /dev/sg device file for each potential LUN for which a device file is not currently available.

On Solaris, this command adds entries to the file /kernel/drv/sd.conf, if necessary, for each potential new LUN that does not have an entry. It also generates an entry for each SCSI target to which the storage system is mapped. On Solaris 8, you must reboot the host after adding sd.conf entries. This command displays a warning whenever a reboot is necessary.

Note

If you have manually edited the /kernel/drv/lpfc.conf file for persistent bindings, ensure that the fcp-bind-WWPN entry is after `# BEGIN: LPUTIL-managed Persistent Bindings.`

Setting up audit, recovery, and trace logging

Supported logs

SnapDrive for UNIX supports the following log files:

- Audit log

  SnapDrive for UNIX logs all commands and their return codes to an audit log. SnapDrive for UNIX makes an entry when you initiate a command and another when the command is complete. That entry includes both the status of the command and the completion time.

- Recovery log

  Some SnapDrive for UNIX operations have the potential to leave the system in an inconsistent or less usable state if interrupted. This could happen if a user terminates the program, or if the host crashes in the middle of an operation. The recovery log contains the steps of a Snapshot restore operation. It documents the steps that were taken and the progress made. This way, NetApp technical support can assist you with the manual recovery process.

- Trace log

  SnapDrive for UNIX reports information useful for diagnosing problems. If you have a problem, NetApp technical support might request this log file.

Enabling and disabling log files

To enable or disable each of these log files in the snapdrive.conf file, complete the following steps.

Step	Action
1	Log in as root.
2	Open the snapdrive.conf file in a text editor.

<table>
<tr><th>Step</th><th colspan="2">Action</th></tr>
<tr><td rowspan="3">3</td><td>If you want to...</td><td>Then...</td></tr>
<tr><td>Enable a log file</td><td>Specify a file name as the value in the name/value pair of the log file you want to enable. SnapDrive for UNIX only writes log files if it has the name of a file to write to. The default names for the log files are as follows:
Audit log: sd-audit.log
Recovery log: sd-recovery.log
Trace log: sd-trace.log
Note
The path to these files may vary depending on your host operating system. For example, on AIX or HP-UX host, the default path for a log file is /var/snapdrive/...; on a Linux or Solaris host the default path is /var/log/... .</td></tr>
<tr><td>Disable a log file</td><td>Do not enter a value for the log file name parameter. If you do not supply a value, there is no file name to which SnapDrive for UNIX can write the log information.
Example: This example disables the audit log file.
<code>audit-log-file=""</code>
Note
NetApp strongly recommends that you leave all log files enabled.</td></tr>
<tr><td>4</td><td colspan="2">Save the file after you make all your changes.
SnapDrive for UNIX automatically checks this file each time it starts, so your changes take effect the next time it starts.</td></tr>
</table>

Settings affecting log file rotation

The values you specify in the snapdrive.conf file enable automatic log file rotations. You can change these values, if necessary, by editing the snapdrive.conf options. The following options affect log file rotation:

- audit-log-max-size
- audit-log-save
- trace-max-size
- trace-log-max-save
- recovery-log-save

Note

For information about the default values for these options, see "Determining options and their default values" on page 86.

With automatic log rotation, SnapDrive for UNIX keeps old log files until it reaches the limit specified in the *audit/trace/recovery*-log-save option. Then it deletes the oldest log file.

SnapDrive for UNIX tracks which file is oldest by assigning the file the number 0 when it creates the file. Each time it creates a new file, it increments by 1 the number assigned to each of the existing log files. When a log file's number reaches the save value, SnapDrive for UNIX deletes that file.

Example: This example uses the `ls` command to produce information about the log files on the system.

Based on those settings, you would see the following information in log files.

```
# ls -l /var/log/sd*
-rw-r--r--   1 root     other      12247 Mar 13 13:09 /var/log/sd-audit.log
-rw-r--r--   1 root     other      20489 Mar 12 16:57 /var/log/sd-audit.log.0
-rw-r--r--   1 root     other      20536 Mar 12 03:13 /var/log/sd-audit.log.1
-rw-r--r--   1 root     other       3250 Mar 12 18:38 /var/log/sd-recovery.log.1
-rw-r--r--   1 root     other       6250 Mar 12 18:36 /var/log/sd-recovery.log.2
-rw-r--r--   1 root     other       6238 Mar 12 18:33 /var/log/sd-recovery.log.3
-rw-r--r--   1 root     other     191704 Mar 13 13:09 /var/log/sd-trace.log
-rw-r--r--   1 root     other     227929 Mar 12 16:57 /var/log/sd-trace.log.0
-rw-r--r--   1 root     other     213970 Mar 12 15:14 /var/log/sd-trace.log.1
-rw-r--r--   1 root     other     261697 Mar 12 14:16 /var/log/sd-trace.log.2
-rw-r--r--   1 root     other     232904 Mar 12 14:15 /var/log/sd-trace.log.3
-rw-r--r--   1 root     other     206905 Mar 12 14:14 /var/log/sd-trace.log.4
```

As shown in this example, the maximum size for each log indicates roughly what size it will be when SnapDrive for UNIX rotates it. SnapDrive for UNIX never rotates a log in the middle of an operation, so the sizes may vary from the default sizes.

Contents of an audit log

The audit log shows information about commands you issued with SnapDrive for UNIX. It maintains a history of the following information:

- The commands issued.
- The return value from those commands.
- The user ID of the user who invoked the command.
- A timestamp indicating when the command started (with no return code) and another timestamp indicating when the command finished (with a return code). The audit log record shows only information about `snapdrive` usage (issued commands).

An audit log file contains the following information.

Field	Description
uid	user ID
gid	group ID
msgText	message text
returnCode	return code from a command

Changing the defaults for audit logs

The snapdrive.conf file enables you to set the following parameters for audit logging.

Note

See "Determining options and their default values" on page 86 for more information about the default values for these options.

- The name of the file containing the audit log.
- The maximum size of the audit log file. The default size is 20K. After the file size reaches the value specified here, SnapDrive for UNIX renames the current audit log file by adding an arbitrary number to the name. Then it starts a new audit file using the name specified by the `audit-log-file` value.

- The maximum number of old audit files that SnapDrive for UNIX saves. The default is 2.

Example of an audit log file

The following example shows an audit log file:

```
2501: Begin uid=0 gid=1 15:35:02 03/12/04 snapdrv snap create -dg
rdg -snapname snap_rdg1
2501: Status=0 15:35:07 03/12/04
2562: Begin uid=0 gid=1 15:35:16 03/12/04 snapdrv snap create -dg
rdg -snapname snap_rdg1
2562: FAILED Status=4 15:35:19 03/12/04
```

The first pair of lines in this example show an operation that succeeded, as indicated by the "Status=0" line.

The second pair of lines indicates an operation that failed. The return code of 4 means “already exists.” If you look at the two command lines, you can see that the first created a Snapshot copy called snap_rdg1. The second line attempted to do the same, but the name already existed, so the operation failed.

Contents of a recovery log

If SnapDrive for UNIX is halted using the Ctrl-C key sequence to kill the program, or if the host or storage system crashes in the middle of an operation, the system might not be able to recover automatically.

So, during any operation that, if interrupted, could leave the system in an inconsistent state, SnapDrive for UNIX writes information to a recovery log file. If a problem occurs, you can send this file to NetApp technical support so they can assist you in recovering the system’s state.

The recovery log utility records the commands that were issued in the process of the operation. Each is marked with an operation_index (a number that uniquely identifies the operation being executed), followed by the date/time stamp and the message text).

Changing the defaults for the recovery logs

The snapdrive.conf file enables you to set the following parameters for recovery logging.

Note

See "Determining options and their default values" on page 86 for more information about the default values for these options.

- The name of the file containing the recovery log, such as recoverylog.
- The maximum number of old recovery files that SnapDrive for UNIX saves. The default is 20. SnapDrive for UNIX keeps this number of recovery logs in case the problem with the process is not immediately discovered. SnapDrive for UNIX starts a new recovery log file each time it completes an operation. It renames the previous one by adding an arbitrary number to the name, such as recoverylog.0, recoverylog.1 and so on.

  Note

  The size of the recovery log file depends on the operation being performed. Each recovery log contains information about a single operation. When that operation is complete, SnapDrive for UNIX starts a new recovery log, regardless of how large the previous file was. As a result, there is no maximum size for a recovery log file.

Example of a recovery log file

The following is an example of entries in a recovery log where SnapDrive for UNIX has restored two Snapshot copies before the operations halted. At this point, you would send this recovery log file to NetApp technical support for assistance in restoring the remaining Snapshot copies.

```
6719: BEGIN 15:52:21 03/09/04 snapdrive snap restore -dg jssdg -
snapname natasha:/vol/vol1:abort_snap_restore
6719: BEGIN 15:52:27 03/09/04 create rollback snapshot:
natasha:/vol/vol1:abort_snap_restore.RESTORE_ROLLBACK_03092004_155
225
6719: END   15:52:29 03/09/04 create rollback snapshot:
natasha:/vol/vol1:abort_snap_restore.RESTORE_ROLLBACK_03092004_155
225 successful
6719: BEGIN 15:52:29 03/09/04 deactivate disk group:  jssdg
6719: BEGIN 15:52:29 03/09/04 stop host volume:
/dev/vx/dsk/jssdg/jvol_1
6719: END   15:52:30 03/09/04 stop host volume:
/dev/vx/dsk/jssdg/jvol_1 successful
6719: BEGIN 15:52:30 03/09/04 unmount file system:  /mnt/demo_fs
6719: END   15:52:30 03/09/04 unmount file system:  /mnt/demo_fs
successful
6719: BEGIN 15:52:30 03/09/04 stop host volume:
/dev/vx/dsk/jssdg/jvol_2
6719: END   15:52:30 03/09/04 stop host volume:
/dev/vx/dsk/jssdg/jvol_2 successful
6719: BEGIN 15:52:30 03/09/04 deport disk group:  jssdg
6719: END   15:52:30 03/09/04 deport disk group:  jssdg successful
6719: END   15:52:30 03/09/04 deactivate disk group:  jssdg
successful
6719: BEGIN 15:52:31 03/09/04 SFSR of LUN: /vol/vol1/lun1 from
snapshot: abort_snap_restore
6719: END   15:52:31 03/09/04 SFSR of LUN: /vol/vol1/lun1 from
snapshot: abort_snap_restore successful
6719: BEGIN 15:52:47 03/09/04 SFSR of LUN: /vol/vol1/lun2 from
snapshot: abort_snap_restore
6719: END   15:52:47 03/09/04 SFSR of LUN: /vol/vol1/lun2 from
snapshot: abort_snap_restore successful
```

About the trace log file

This file is for NetApp technical support's use in cases where there is a problem that needs debugging. Enabling it does not affect system performance. By default, this file is enabled. You can disable it by setting the snapdrive.conf option trace-enabled to off.

Changing the defaults for the trace logs

The snapdrive.conf file enables you to set the following parameters for trace logging.

Note

See "Determining options and their default values" on page 86 for more information about the default values for these options.

- The name of the file containing the trace log.
- The maximum size of the trace log file. The default size is 0 bytes. This value ensures that each trace file will contain only one SnapDrive for UNIX command.

  If you reset the default size to a value other than 0, when the file reaches the size you specified, SnapDrive for UNIX renames the current trace log file by adding an arbitrary number to the name. Then it starts a new trace file using the name specified by the trace-log-file value.
- The maximum number of old trace files that SnapDrive for UNIX saves. The default is 100.
- The types of messages that SnapDrive for UNIX writes to the trace log file. By default, the trace log file contains fatal errors, admin errors, command errors, warnings and information messages.

Setting up AutoSupport

Understanding AutoSupport

NetApp provides AutoSupport with its storage systems as a way to provide better service to you, should you have a problem with your system. With AutoSupport, you can configure your storage system to send an e-mail message to NetApp technical support when an error occurs. Then, if you call in with an issue, NetApp technical support has information about your storage systems and configuration and can more quickly help you to solve the problem.

No secure information is ever sent using AutoSupport.

How SnapDrive for UNIX uses AutoSupport

For this release, SnapDrive for UNIX only sends an AutoSupport message from the storage system to NetApp the first time you execute it after a system reboot. It sends one message for each host reboot from the host that rebooted. At this time, it does not send a message when an error condition occurs.

Note

To use this feature, you must have a user login configured for the storage system and you must enable AutoSupport in the snapdrive.conf file. See "Enabling AutoSupport" on page 132.

The AutoSupport feature in SnapDrive for UNIX logs into the storage system you configured for AutoSupport in the snapdrive.conf file. It uses that storage system to send an AutoSupport message to NetApp. This message specifies the following information:

- SnapDrive for UNIX version
- Message status: 3 for a warning or information message
- Host name
- Host operating system
- Host operating system release number
- Host operating system version
- Protocols used
- Number of disk or volume groups

If the storage system specified for AutoSupport in the snapdrive.conf file cannot send an AutoSupport message to NetApp, SnapDrive for UNIX does not log an error to syslog. This information is only written to the internal SnapDrive for UNIX log files.

Examples of an AutoSupport message

The substance of the AutoSupport message is essentially the same regardless of your operating system. The details of the message, such as information on your operating system, vary according to your system setup.

HP-UX example: The following example is a message sent from a host named DBserver that is running release 5B.11.22 of HP-UX. This is an informational message, as indicated by the number 3 in parentheses: (3).

```
snapdrive: 3.0 (3)
general: host_name = DBserver, host_os=HP-UX,
host_os_release=B.11.22, host_os_version=U,
protos= iscsi, 17 Connect Luns, 13 dgs, 17 hvs
```

Linux example: The following example is a message sent from a host named DBserver that is running Linux. This is an informational message, as indicated by the number 3 in parentheses: (3).

- Red Hat Linux

```
snapdrive: 3.0 (3)
general: host_name = DBserver, host_os=Linux,
host_os_release=2.4.21-9.ELsmp, host_os_version=#1 SMP Thu Jan 8
17:08:56 EST 2004,
protos= iscsi, 17 Connect Luns, 13 dgs, 17 hvs
```

- SUSE Linux

```
snapdrive: 3.0 (3)
general: host_name = DBserver, host_os=Linux,
host_os_release=2.6.5-7.97-bigsmp, host_os_version=#1 SMP Fri Jul 2
14:21:59 UTC 2004
protos= iscsi, 17 Connect Luns, 13 dgs, 17 hvs
```

Solaris example: The following example is a message sent from a host named DBserver that is running release 5.9 of Solaris. This is an informational message, as indicated by the number 3 in parentheses: (3).

```
snapdrive: 3.0 (3)
general: host_name = DBserver, host_os=SunOS, host_os_release=5.9,
host_os_version=Generic,
protos= iscsi, 17 Connect Luns, 13 dgs, 17 hvs
```

IBM-AIX example: The following example is a message sent from a host named DBserver that is running release 5.1 of AIX. This is an informational message, as indicated by the number 3 in parentheses: (3).

```
snapdrive: 3.0 (3)
general: host_name = DBserver, host_os=AIX,
host_os_release=1, host_os_version=5
protos= iscsi, 17 Connect Luns, 13 dgs, 17 hvs
```

Enabling AutoSupport

To enable AutoSupport, complete the following steps.

Step	Action
1	Log in as root.
2	Set user logins for the storage systems you want SnapDrive for UNIX to access. See "Specifying the current login information for storage systems" on page 154.

3	Modify the snapdrive.conf file to enable AutoSupport.
	a. Set the `autosupport-enabled` variable to 1. It should read as follows:
	`autosupport-enabled=1`
	b. Specify the name or IP address of a storage system that you want AutoSupport to contact and use for sending the messages. For example, if you want AutoSupport to use the storage system named toaster, the name/value pair would read as follows:
	`autosupport-filer=toaster`
	c. Save the snapdrive.conf file.
	SnapDrive for UNIX automatically checks this file each time it starts, so your changes take effect the next time it starts.

Setting up multipathing

SnapDrive for UNIX supports FCP and iSCSI multipath access to the storage system using the standard multipathing software solution. Following are the multipathing solutions on different host platforms that are supported by SnapDrive for UNIX.

Platform	Multipathing solution
AIX	◆ SANpath ◆ NativeMPIO ◆ DMP
HP-UX	◆ PVLinks ◆ DMP
Linux	Not supported
Solaris	◆ MPxIO ◆ DMP

By using multipathing, you can configure multiple network paths between the host and storage system. If one path fails, the FCP or iSCSI traffic continues on the remaining paths. Multipathing is required if the host has multiple paths to a LUN and it works by making the underlying paths transparent to the user.

Rules

SnapDrive for UNIX uses multipathing solution based on the following rules:

- If the multipathing solution specified in the configuration file for the SnapDrive for UNIX operations is configured and supported, SnapDrive for UNIX uses the specified configured multipathing solution.
- If the multipathing solution specified in the configuration file for the SnapDrive for UNIX operations is not supported or not configured, SnapDrive for UNIX uses the appropriate supported and configured multipathing solution.

Note

SnapDrive for UNIX reports an error if the preceding rules are not met.

Enabling multipathing

To enable multipathing, complete the following steps.

Step	Action
1	Check the SnapDrive for UNIX Compatibility Matrix on the NOW site at (http://now.netapp.com/NOW/knowledge/docs/docs.shtml) for latest information about SnapDrive for UNIX and its requirements.
2	Install the supported HBAs before you install the appropriate Host Utilities software. For more information, see the FCP or iSCSI Host Utilities setup guide at http://now.netapp.com/NOW/knowledge/docs/san/. **Note** To ensure you have the current version of the system components, see the *Compatibility and Configuration Guide for NetApp FCP and ISCSI Products* at http://now.netapp.com/NOW/knowledge/docs/san/fcp_iscsi_config/. Support for new components is added on an on-going basis. This online document contains a complete list of supported HBAs, platforms, applications, and drivers.
3	Load and start the HBA service. For more information, see the FCP or iSCSI Host Utilities setup guide at http://now.netapp.com/NOW/knowledge/docs/san/. If HBA service is not running, you will get the following error on executing the SnapDrive for UNIX commands, such as, `snapdrive storage create`, `snapdrive config prepare luns`: `0001-876 Admin error: HBA assistant not found` Also, for multipathing configuration ensure that the required number of paths are up and running. You can verify using the `sanlun` utility which comes with the Host Utilities software. For example, in FCP multipathing configuration, you can use the `sanlun fcp show adapter -v` command.

Step	Action

	Example 1: In the following example there are two HBA ports (fcd0 and fcd1) connected to the host and are operational (port state). You can also have only one HBA or iSCSI initiator and yet configure multipathing by providing more than one multiple path to the target storage system.
	
	`# sanlun fcp show adapter -v`
	
	`adapter name: fcd0`
	`WWPN: 50060b000038c428`
	`WWNN: 50060b000038c429`
	`driver name: fcd`
	`model: A6826A`
	`model description: Fibre Channel Mass Storage Adapter (PCI/PCI-X)`
	`serial number: Not Available`
	`hardware version: 3`
	`driver version: @(#) libfcd.a HP Fibre Channel ISP 23xx & 24xx Driver B.11.23.04 /ux/core/isu/FCD/kern/src/common/wsio/fcd_init.c:Oct 18 2005,08:19:50`
	`firmware version: 3.3.18`
	`Number of ports: 1 of 2`
	`port type: Fabric`
	`port state: Operational`
	`supported speed: 2 GBit/sec`
	`negotiated speed: 2 GBit/sec`
	`OS device name: /dev/fcd0`
	`adapter name: fcd1`
	`WWPN: 50060b000038c42a`
	`WWNN: 50060b000038c42b`
	`driver name: fcd`
	`model: A6826A`
	`model description: Fibre Channel Mass Storage Adapter (PCI/PCI-X)`
	`serial number: Not Available`
	`hardware version: 3`
	`driver version: @(#) libfcd.a HP Fibre Channel ISP 23xx & 24xx Driver B.11.23.04 /ux/core/isu/FCD/kern/src/common/wsio/fcd_init.c:Oct 18 2005,08:19:50`
	`firmware version: 3.3.18`
	`Number of ports: 2 of 2`

Step	Action

	`port type: Fabric`
	`port state: Operational`
	`supported speed: 2 GBit/sec`
	`negotiated speed: 2 GBit/sec`
	`OS device name: /dev/fcd1`
	
	Example 2: If multipathing is enabled on a host, multiple paths will be visible for the same LUN. You can use `sanlun lun show all` command to verify. In this example you can find multiple paths to the same LUN (fish: /vol/vol1/lun).
	
	`# sanlun lun show all`
	`filer: lun-pathname device filename adapter protocol lun size lun state`
	`fish: /vol/vol1/lun /dev/rdsk/c15t0d0 fcd0 FCP 10m (10485760) GOOD`
	`fish: /vol/vol1/lun /dev/rdsk/c14t0d0 fcd1 FCP 10m (10485760) GOOD`

4	**Conditional:** If a third party multipathing solution is supported by SnapDrive for UNIX or Host Utilities, you have to download the HBA driver software package and applications package from the HBA vendor's Web site. **QLogic:** For QLogic HBAs, go to http://support.qlogic.com/support/drivers_software.asp. Under OEM Models, select NetApp. Locate the driver version listed in the support matrix and download it. **Emulex:** For Emulex HBAs, go to http://www.emulex.com/ts/index.html. Under Storage and System Supplier Qualified and Supported HBAs, select NetApp. Locate the driver version listed in the support matrix and download it. Also download the Emulex applications package from the same location.
5	In FCP configuration, you have to zone the host HBA ports and the target ports using the switch zoning configuration.
6	Install and set up the appropriate FCP or iSCSI Host Utilities software. For installation and setup procedures, see the *Installation and Set Up Guide* at http://now.netapp.com/NOW/knowledge/docs/san/

Step	Action
7	Check the SnapDrive for UNIX stack requirements; see Chapter 1, "SnapDrive for UNIX stack," on page 6.
8	Complete the SnapDrive for UNIX prerequisites; see Chapter 2, "Prerequisites for using SnapDrive for UNIX," on page 16.
9	Install or upgrade SnapDrive for UNIX; see Chapter 3, "Installing and Upgrading SnapDrive for UNIX," on page 29.
10	Verify the SnapDrive for UNIX installation; see Chapter 3, "Installing and Upgrading SnapDrive for UNIX," on page 29.
11	Locate the snadrive.conf file path; see Chapter 3, "Understanding the files installed by SnapDrive for UNIX on the host," on page 74.
12	Set the configuration options; see Chapter 4, "Configuring and Using SnapDrive for UNIX," on page 83.
13	Configure the following configuration variables in the snapdrive.conf file: ◆ `multipathing-type` ◆ `default transport-type` ◆ `fstype` ◆ `vmtype` For detailed information on the above mentioned configuration variables, see Chapter 4, "Determining options and their default values," on page 86. For every host, multipathing type, transport type, file system and volume manger type are dependent on each other. For all the possible combination, see the following table.
14	For the SFRAC cluster environment, execute the `snapdrive config check cluster` command; see Chapter 4, "Some operations available with the snapdrive config command," on page 84.
15	Save the snapdrive.conf file. SnapDrive for UNIX automatically checks this file each time it starts, so your changes take effect the next time it starts.

The following table gives the supported values of the `multipathing-type`, `default transport-type`, `fstype`, and `vmtype` configuration variables. These values are not case-sensitive, so you many not enter the way they are mentioned in the table.

Host platform	default transport-type	multipathing-type	fstype	vmtype
AIX	iscsi	none	jfs2 or jfs	lvm
	fcp	sanpath	jfs2 or jfs	lvm
		nativempio	jfs2 or jfs	lvm
		dmp	vxfs	vxvm
HP-UX	iscsi	pvlinks	vxfs	lvm
	fcp	pvlinks	vxfs	lvm
		dmp	vxfs	vxvm
Solaris	iscsi (hardware)	none	ufs	none
	iscsi (software)	mpxio	ufs	none
		none	ufs	none
	fcp	dmp	vxfs	vxvm
Linux	iscsi	none	ext3	lvm
	fcp	none	ext3	lvm

Refreshing the DMP paths

On hosts with FCP and DMP configurations, the `snapdrive storage delete -lun` command hangs for more than 10 minutes. This happens due to incorrect installation or configuration of the following components:

- NTAPasl
- Veritas stack (storage foundation)
- Multipathing licenses

Refresh the DMP paths information properly after any FCP path is enabled, disabled, or added. To refresh the DMP paths, execute the following commands in the same sequence as they are mentioned in the following table.

Host	Commands
AIX	**1.** `cfgmgr` **2.** `vxdisk scandisk`
HP-UX	**1.** `ioscan -fnC disk` **2.** `ioinit -i` **3.** `vxdisk scandisks`
Solaris	**1.** `devfsadm -Cv` **2.** `vxdisk scandisk`

Setting up thin provisioning

With the thin provisioning feature in SnapDrive for UNIX you can have more storage space for the hosts connecting to the storage system than is actually available on the storage system. For more information on thin provisioning, see the technical report TR-3483 at http://www.netapp.com/library/tr/3483.pdf.

In this release, SnapDrive for UNIX does not support the following Data ONTAP features:

- Fractional reserve
- Space reclamation (Hole punching)
- Snapshot reserve
- Space monitoring command-line interface (CLI)
- LUN fill on demand
- Volume fill on demand
- Volume autogrow/Snapshot autodelete

Enabling thin provisioning

For LUNs: You can enable space reservation either by setting the `space-reservations-enabled` configuration variable value to "on", or by using the `-reserve` and `-noreserve` command-line options in the command. Usage of these command-line options will override the value mentioned in the `space-reservations-enabled` variable.

NetApp recommends using `-reserve` and `-noreserve` command-line options in the `snapdrive storage create`, `snapdrive storage resize`, `snapdrive snap connect`, and `snapdrive snap restore` commands to enable or disable LUN space reservation.

By default, SnapDrive for UNIX enables space reservations for the fresh or new storage create operation. For Snapshot restore and connect operations, it uses the space reservations present in the Snapshot copy if the `-reserve` or `-noreserve` flags are not specified on the CLI or if the value in the configuration file is commented out.

For NFS entities: For Snapshot connect operations, you can enable space reservation for volumes by using the `-reserve` command-line option in the commands involving NFS entities. For NFS entities, SnapDrive for UNIX uses the space reservations available in the Snapshot copy if the `-reserve` or `-noreserve` flags are not specified on the CLI.

Note

You cannot use the `space-reservations-enabled` configuration variable for Snapshot operations involving NFS entities.

For more information about the `space-reservations-enabled` configuration variable, see "Determining options and their default values" on page 86, and about the `-reserve` and `-noreserve` command-line options, see Chapter 9, "SnapDrive for UNIX options, keywords, and arguments," on page 348.

General steps for executing commands

What you can do with the commands

SnapDrive for UNIX supports Snapshot copy management (`snap` commands) and storage provisioning (`storage` and `host` commands). In addition, you can use the `config` command to monitor or control some of SnapDrive for UNIX's configuration details.

Running SnapDrive for UNIX from the command line

You must be logged in as root to run SnapDrive for UNIX. You execute SnapDrive for UNIX from the CLI, either directly from a shell prompt or from scripts. When you execute SnapDrive for UNIX from the command line, remember the following:

- Enter each command on a separate line.
- If you enter a command without arguments, SnapDrive for UNIX gives you a usage line with examples of commands.
- Each SnapDrive for UNIX command must use the following format:

  `snapdrive` *`type_name operation_name`* `[<`*`keyword/options`*`> <`*`arguments`*`>]`

  type_name specifies the type of task you want to perform. You can perform the following types of tasks:

  - Snapshot copy management

    Use the type name `snap` with the `snapdrive` command to indicate you are performing a Snapshot copy management operation.

    `snapdrive snap` *operation* <*arguments*>

  - Storage operations

    Use the type name `storage` with the `snapdrive` command to indicate you are performing an operation dealing with storage on the storage system.

    `snapdrive storage` *operation* <*arguments*>

  - Connect or disconnect the host

    Use the type name `host` with the `snapdrive` command to indicate you are performing an operation dealing only with the host-side entities. The host commands allow you to connect storage on the storage system to the host or disconnect storage from the host. They do not affect the storage on the storage system. They affect only the visibility of the storage on the host.

`snapdrive host` *operation <arguments>*

- Specify or view behavioral options

  Use the type name `config` with the `snapdrive` command to indicate you want to create, modify, or delete a user login for a storage system. You can also use the `config` type name to get SnapDrive for UNIX to view the snapdrive.conf file, view access permissions, or prepare the host for adding LUNs.

  `snapdrive config` *operation <arguments>*

- Determine the version number of SnapDrive for UNIX

  `snapdrive version`

operation_name specifies the operation you want to perform. Each command type has certain operations you can perform. The Snapshot copy management operations are create, rename, restore, show, connect, disconnect, and delete. The storage operations are create, connect, disconnect, resize, show, and delete. The configuration operations are access, show, list, set, delete, prepare luns and check luns. The host operations are connect and disconnect.

<keyword/options> are the keywords that you can use to specify information corresponding to the host and storage system objects with which you are working, as well as options you can specify to get information such as verbose output. (See "SnapDrive for UNIX options, keywords, and arguments" on page 348 for more information.)

<arguments> are the values you supply with the keywords that specify the target. (See "SnapDrive for UNIX options, keywords, and arguments" on page 348 for a list of the acceptable arguments and the value you can supply for them.)

Frequently used command-line options

Following are some frequently used command-line options:

- `-force` (`-f`)

  This option forces SnapDrive for UNIX to attempt an operation that it ordinarily would not undertake. SnapDrive for UNIX prompts you to confirm that you want it to perform this operation before it attempts the operation.

- `-noprompt`

  This option keeps SnapDrive for UNIX from prompting you to confirm the operation. You should include this option if you are running SnapDrive for UNIX in a scripted environment where there is no person to confirm the operation.

- `-verbose` (`-v`)

This option displays detailed or verbose output.

These options appear in the command examples in the following chapters. For more information on the command-line options, see Chapter 9, “Summary of the SnapDrive for UNIX commands,” on page 341.

Setting Up Security Features 5

About this chapter

This chapter contains information about the security features available in SnapDrive for UNIX and how to access them.

Topics in this chapter

This chapter discusses the following topics:

- "Overview of SnapDrive for UNIX security features" on page 146
- "Setting up access control" on page 147
- "Viewing the current access control settings" on page 152
- "Specifying the current login information for storage systems" on page 154
- "Disabling SSL encryption" on page 157

Overview of SnapDrive for UNIX security features

Security features provided by SnapDrive for UNIX

SnapDrive for UNIX provides certain features to allow you to work with it in a more secure fashion. These features include letting you:

- Set up access control permissions
- Specify login information for the storage systems
- Specify that SnapDrive for UNIX use HTTPS (secure socket layer)

The access control feature lets you specify which operations a host running SnapDrive for UNIX can perform on a storage system. You set these permissions individually for each host. For more information, see "Setting up access control" on page 147.

In addition, to allow SnapDrive for UNIX to access a storage system, you must supply the login name and password for that storage system. For more information, see "Specifying the current login information for storage systems" on page 154.

The HTTPS feature lets you specify that you use it for all interactions with the storage system through the ManageONTAP interface, including sending the passwords. This is a default behavior in the SnapDrive 3.0 for UNIX release for AIX, Solaris, and Linux hosts. However, you can disable the SSL encryption by changing the value of the `use-https-to-filer` command to "off". For more information on disabling SSL, see "Disabling SSL encryption" on page 157.

On HP-UX hosts, SnapDrive 3.0 for UNIX uses HTTP protocol to communicate to storage systems.

Together these features give you more control over which users can perform operations on a storage system and from which host.

Setting up access control

Access control in SnapDrive for UNIX

SnapDrive for UNIX allows you control the level of access that each host has to each storage system to which the host is connected. This access level indicates which operations the host is allowed to perform when it targets a given storage system.

With the exception of the show and list operations, the access control permissions can affect all snapshot and storage operations.

Available access control levels

You can set the following access levels:

- NONE –The host has no access to the storage system.
- SNAP CREATE –The host can create Snapshot copies.
- SNAP USE –The host can delete and rename Snapshot copies.
- SNAP ALL –The host can create, restore, delete, and rename Snapshot copies.
- STORAGE CREATE DELETE –The host can create, resize, and delete storage.
- STORAGE USE –The host can connect and disconnect storage.
- STORAGE ALL –The host can create, delete, connect, and disconnect storage.
- ALL ACCESS–The host has access to all of the preceding SnapDrive for UNIX operations.

Each level is distinct. If you specify permission for only certain operations, then SnapDrive for UNIX can execute only those operations. For example, if you specify STORAGE USE, the host can use SnapDrive for UNIX to connect and disconnect to storage, but it cannot perform any other operations governed by access control permissions.

How access control works

SnapDrive for UNIX determines the access control permissions by checking the storage system for a permissions file in the root volume of the storage system. This file is in the directory /vol/vol0/sdprbac (SnapDrive permissions roles-based access control). The file name is sd*host-name*.prbac, where *host-name* is the name of the host to which the permissions apply. You can have a permissions file for each host attached to the storage system.

Setting up access control from a given host to a given vFiler unit is a manual operation. The access from a given host is controlled by a file residing in the root volume of the affected vFiler unit. This file has the name /vol/*root-vfiler-volume*/sdprbac/sd*host-name*.prbac, where *host-name* is the name of the affected host, as returned by `gethostname(3)`. You should ensure that this file is readable, but not writable, from the host that will access it.

Note

To determine the name of host, run the `hostname` command.

If the file is empty, unreadable, or has an invalid format, SnapDrive for UNIX does not grant the host access permission to any of the operations.

If the file is missing, SnapDrive for UNIX checks the configuration variable `all-access-if-rbac-unspecified` in the snapdrive.conf file. If this variable is set to "on" (the default), it allows the hosts complete access to all these operations on that storage system. If it is set to "off", SnapDrive for UNIX denies the host permission to perform any operations governed by access control on that storage system.

Setting access control permissions

To set the access control permissions for a storage system, complete the following steps.

Step	Action
1	Log in as root on the host.
2	On the storage system, create the directory sdprbac in the root volume of the target storage system. **Note** One way to make the root volume accessible is to mount the volume using NFS.

Step	Action
3	In this directory, create the permissions file. Make sure the following is true: ◆ The file must be named sd*host-name*.prbac where *host-name* is the name of the host for which you are specifying access permissions. ◆ The file must be read-only. This ensures that SnapDrive for UNIX can read it, but that it cannot be modified. **Example:** To give a host named dev-sun1 access permission, you would create the following file on the storage system: `/vol/vol1/sdprbac/sddev-sun1.prbac`

Step	Action

4	Set the permissions in the file for that host. You must use the following format for the file:
	◆ You can specify only one level of permissions. To give the host full access to all these operations, enter the string ALL ACCESS.
	◆ The permission string must be the first thing in the file. The file format is invalid if the permission string is not in the first line.
	◆ Permission strings are case insensitive.
	◆ No white space can precede the permission string.
	◆ No comments are allowed.
	The valid permission strings are:
	`NONE`
	`SNAP CREATE`
	`SNAP USE`
	`SNAP ALL`
	`STORAGE CREATE DELETE`
	`STORAGE USE`
	`STORAGE ALL`
	`ALL ACCESS`
	These strings allow the following access:
	◆ NONE –The host has no access to the storage system.
	◆ SNAP CREATE –The host can create Snapshot copies.
	◆ SNAP USE –The host can delete and rename Snapshot copies.
	◆ SNAP ALL–The host can create, restore, delete, and rename Snapshot copies.
	◆ STORAGE CREATE DELETE –The host can create, resize, and delete storage.
	◆ STORAGE USE –The host can connect and disconnect storage.
	◆ STORAGE ALL –The host can create, delete, connect, and disconnect storage.
	◆ ALL ACCESS–The host has access to all of the SnapDrive for UNIX operations.

Step	Action
	Each of these permission strings is discrete. If you specify SNAP USE, the host can delete or rename Snapshot copies, but it cannot create Snapshot copies or restore Snapshot copies or perform any storage provisioning operations. Regardless of the permissions you set, the host can perform show and list operations.
5	Verify the access permissions by entering the following command: `snapdrive config access show` *`filer-name`*

Viewing the current access control settings

About viewing the access control permissions

You use the `snapdrive config access` command to display information about the permissions available for a host on a specific storage system.

Viewing the access control permissions

To view the access control permissions, complete the following step.

Step	Action

1	Execute the `snapdrive config access show` command. This command has the following format:
	`snapdrive config access {show | list} filename`
	You can use the same arguments regardless of whether you enter the `show` or `list` version of the command.
	Note
	For details about using the options and arguments in this command line, see the section "SnapDrive for UNIX options, keywords, and arguments" on page 348.
	Result: This operation displays information about the access permissions available for that host.
	Example 1: This command line checks the storage system toaster to determine which permissions the host has. Based on the permissions shown, the permissions for the host on this storage system are SNAP ALL.
	`# snapdrive config access show toaster`
	`This host has the following access permission to filer, toaster:`
	`SNAP ALL`
	`Commands allowed:`
	`snap create`
	`snap restore`
	`snap delete`
	`snap rename`
	`#`

Step	Action

	Example 2: In this example, the permissions file is not on the storage system, so SnapDrive for UNIX checks the variable `all-access-if-rbac-unspecified` in the snapdrive.conf file to determine which permissions the host has. This variable is set to "on", which is equivalent to creating a permissions file with the access level set to ALL ACCESS.
	
	`# snapdrive config access list toaster`
	`This host has the following access permission to filer, toaster:`
	`ALL ACCESS`
	`Commands allowed:`
	`snap create`
	`snap restore`
	`snap delete`
	`snap rename`
	`storage create`
	`storage resize`
	`snap connect`
	`storage connect`
	`storage delete`
	`snap disconnect`
	`storage disconnect`
	`#`
	
	Example 3: This example shows the kind of message you will receive if there is no permissions file on the storage system toaster and the variable `all-access-if-rbac-unspecified` in the snapdrive.conf file is set to "off".
	
	`# snapdrive config access list toaster`
	`Unable to read the access permission file on filer, toaster. Verify that the file is present.`
	`Granting no permissions to filer, toaster.`

Specifying the current login information for storage systems

About using current logins for storage systems

A user name or password enables SnapDrive for UNIX to access each storage system. It also provides a level of security because, in addition to being logged in as root, the person running SnapDrive for UNIX must supply the correct user name or password when prompted for it. If a login is compromised, you can delete it and set a new user login.

You created the user login for each storage system when you set it up. For SnapDrive for UNIX to work with the storage system, you must supply it with this login information. Depending on what you specified when you set up the storage systems, each storage system could use either the same login or a unique login.

SnapDrive for UNIX stores these logins and passwords in encrypted form on each host. You can specify that it encrypt this information when you send it across the wire by setting the snapdrive.conf variable `use-https-to-filer=on`. For more information, see "Disabling SSL encryption" on page 157.

Specifying login information

To specify the user login information for a storage system, complete the following steps.

Note

Depending on what you specified when you set up the storage system, each storage system could use either the same user name or password or a unique user name or password. If all the storage systems use the same user name or password information, you only need to perform the following steps once. If not, repeat the following steps for each storage system.

Step	Action
1	Log in as root.
2	Enter the following command: `snapdrive config set` *`user_name filername`* `[`*`filername ...`*`]` *user_name* is the user name that was specified for that storage system when you first set it up.

Step	Action
	filername is the name of the storage system. You can enter multiple storage system names on one command line if they all have the same user login/password. You must enter the name of at least one storage system.

3	At the prompt, enter the password, if there is one.
	Note
	If no password was set, press Enter (the null value) when prompted for a password.
	Example 1: This example sets up a user called root for a storage system called toaster:
	`# snapdrive config set root toaster`
	`Password for root:`
	`Retype Password:`
	Example 2: This example sets up one user called root for three storage systems:
	`# snapdrive config set root toaster oven broiler`
	`Password for root:`
	`Retype Password:`

4	If you have another storage system with a different user name or password, repeat these steps.

Verifying storage system user names associated with SnapDrive for UNIX

You can verify which user name SnapDrive for UNIX has associated with a storage system by executing the `snapdrive config list` command.

Note

This command does not query the storage system to determine whether additional user names have been configured for it, nor does it display the password associated with a storage system.

To execute `snapdrive config list`, complete the following step.

Step	Action
1	Log in as root.

2	Enter the following command:
	`snapdrive config list`
	This command displays the user name or storage system pairs for all systems that have users specified within SnapDrive for UNIX. It does not display the passwords for the storage systems.
	Example: This example displays the users associated with the storage systems named rapunzel and mediumfiler:
	`# snapdrive config list`
	`user name filer name`
	`----------`
	`rumplestiltskins rapunzel`
	`longuser mediumfiler`

Deleting a user login for a storage system

To delete a user login for one or more storage systems, complete the following steps.

Step	Action
1	Log in as root.
2	Enter the following command: `snapdrive config delete` *`filername`* `[`*`filername`* `...]` *filername* is the name of the storage system for which you want to delete the user login information. **Result:** SnapDrive for UNIX removes the user name or password login information for the storage systems you specify. **Note** To enable SnapDrive for UNIX to access the storage system, you must specify a new user login.

Disabling SSL encryption

Disabling SnapDrive for UNIX to use HTTPS

The previous versions of SnapDrive for UNIX use HTTP by default, when sending storage system login information across the network. In SnapDrive 3.0 for UNIX release, on AIX, Solaris, and Linux hosts, by default, SnapDrive 3.0 for UNIX uses HTTPS protocol to communicate to storage systems. However, by changing the value of the `use-https-to-filer` configuration variable to "off", you can set SnapDrive 3.0 for UNIX to use HTTP to communicate with the storage system.

On HP-UX hosts, SnapDrive 3.0 for UNIX uses HTTP protocol to communicate to storage systems.

Note

Using HTTPS might result in slower performance with versions of Data ONTAP prior to 7.0. It does not affect performance with Data ONTAP 7.0.

Setting up HTTP

To set up SnapDrive for UNIX to use HTTP for AIX, Solaris, and Linux hosts, complete the following steps.

Note

For more information on the configuration file, see "Setting configuration information" on page 84.

Step	Action
1	Log in as root.
2	Make a backup of the snapdrive.conf file.
3	Open the snapdrive.conf file in a text editor.

Step	Action
4	Change the value for the `use-https-to-filer` variable to “off”. `use-https-to-filer=off` **Tip:** A good practice any time you modify the snapdrive.conf file is to perform the following steps: **a.** Comment out the line you want to modify. **b.** Copy the commented-out line. **c.** Un-comment the copy by removing the pound sign. **d.** Modify the value. This way you always have a record of the default value in the file.
5	Save the file after you make your changes. SnapDrive for UNIX automatically checks this file each time it starts, so your changes take effect the next time it starts.

Provisioning and Managing Storage 6

About this chapter

This chapter provides details about working with storage provisioning. It contains information on using SnapDrive for UNIX to create and connect to storage.

Topics in this chapter

This chapter discusses the following topics:

- "Overview of storage provisioning" on page 160
- "Creating storage" on page 163
- "Displaying information about storage" on page 177
- "Displaying information about devices and LVM entities" on page 185
- "Increasing the size of the storage" on page 192
- "Connecting LUNs and storage entities to the host" on page 198
- "Connecting only the host side of storage" on page 216
- "Disconnecting only the host side of storage" on page 221
- "Deleting storage from the host and storage system" on page 228

Overview of storage provisioning

Using storage provisioning with SnapDrive for UNIX

SnapDrive for UNIX provides end-to-end storage management. With it, you can provision storage from a host to a NetApp storage system and manage that storage with or without using the host LVM.

SnapDrive for UNIX lets you perform the following tasks:

- Create storage by creating LUNs, file systems, logical volumes, and disk groups.
- Display information about storage.
- Connect to storage.
- Resize storage.
- Disconnect from the storage.
- Delete storage.

When you use SnapDrive for UNIX to create storage using the `snapdrive storage create` command, it automatically performs all the tasks needed to set up LUNs, including preparing the host, performing discovery mapping, and connecting to each LUN you create. You can use the `snapdrive storage show` command to display information about the NetApp LUNs, disk groups, host volumes, file systems or NFS directory trees that you create.

SnapDrive for UNIX also provides the `snapdrive storage connect` command. You can use this command to map storage to a new location. This command lets you access existing storage from a different host than the one used to create it. The `snapdrive storage connect` command lets you make existing LUNs, file systems, disk groups and logical volumes accessible on a new host. This operation can be useful if you want to back up a storage entity from the new host to another medium.

The `snapdrive storage resize` command lets you increase the size of your storage in the following ways:

- Specifying a target size that you want the host entity to reach.
- Entering a set number of bytes by which you want to increase the storage.

If at some point, you decide you no longer want your storage mapped to its current location, you can use the `snapdrive storage disconnect` command. This command removes the mappings from one or more host locations to the LUNs making up the storage for that location.

You can also delete the storage. If you execute the `snapdrive storage delete` command, SnapDrive for UNIX removes all the host-side entities you specify as well as all their underlying entities and the LUNs associated with them.

Using storage operations across multiple storage system volumes

SnapDrive for UNIX lets you perform many of the storage operations across multiple storage system volumes as long as they do not manipulate the LVM. This enables you to work with lists of LUNs that exist across multiple storage system volumes.

Considerations for storage operations

For storage operations, consider the following:

- On HP-UX hosts, the default behavior of the LVM is to limit the host to 16 physical volumes per host volume group. This limits the maximum number of LUNs to 16. This limit affects the `snapdrive storage create` and the `snapdrive storage resize` commands. If you attempt to create a host volume with more than 16 LUNs, the `snapdrive storage create` command fails. If you try to resize a volume group so that it contains more than 16 LUNs, the `snapdrive storage resize` command fails.
- To have more than 16 LUNs (physical volumes) on an HP-UX host, you must manually create the host volume group using the `vgcreate` command and set the value for `-p max_pv` to the maximum value you expect to need for LUNs. The only time you may set this limit is when you create the host volume group. See the HP-UX `vgcreate`(1) manual (man) page for more information.If you want to resize your storage, you must use the `-addlun` option on the `snapdrive storage resize` command line. SnapDrive for UNIX does not currently support expanding existing LUNs.
- Support for volume groups spanning multiple storage system volumes or multiple storage systems is limited. Using SnapDrive for UNIX volume groups that span across storage systems cannot be created using the `snapdrive storage create` command.

  Four key commands that SnapDrive for UNIX supports in this cases are:

  - The `snapdrive snap create` command
  - The `snapdrive snap restore` command
  - The `snapdrive snap connect` command
  - The `snapdrive snap disconnect` command
- The `snapdrive storage resize` command does not work with LUNs mapped directly to the host, or with the files systems that they contain. For additional information, see “Increasing the size of the storage” on page 194.

- If you use the `snapdrive storage create` command to create a logical volume or a file system on a LUN, you cannot directly specify the size of the logical volume or file system being created. Instead, you must specify the size of the underlying volume group; any host volume or file system created within it will be marginally smaller than it is.
- In creating host volumes, SnapDrive for UNIX does not provide any options to control their formatting. It creates only concatenated host volumes. SnapDrive for UNIX does operate correctly on host volumes of other formats (such as striped volumes) that were created outside of it.
- You cannot restore a portion of a disk group. SnapDrive for UNIX backs up and restores whole disk groups only.

Creating storage

Creating storage with SnapDrive for UNIX

You can use SnapDrive for UNIX to create the following:

- LUNs
- A file system created directly on a LUN
- Disk groups, host volumes and file systems created on LUNs

SnapDrive for UNIX automatically handles all the tasks needed to set up LUNs associated with these entities, including preparing the host, performing discovery mapping, creating the entity, and connecting to the entity you create. You can also specify which LUNs SnapDrive for UNIX uses to provide storage for the entity you request.

You do not need to create the LUNs and the storage entity at the same time. If you create the LUNs separately, you can create the storage entity later using the existing LUNs.

Creating storage for LVM entities: If you ask SnapDrive for UNIX to create a logical volume or file system using the LVM, SnapDrive for UNIX automatically creates the required disk group. SnapDrive for UNIX creates the file system based on the type supported by the host logical volume manager (LVM). These include the following:

- AIX: JFS2, JFS, and VxFS
- HP-UX: VxFS
- Linux: Ext3
- Solaris: VxFS and UFS

Creating storage for a file system that resides on a LUN: If you ask SnapDrive for UNIX to create a file system that resides directly on a LUN, SnapDrive for UNIX creates and maps the LUN, then creates and mounts the file system without involving the host LVM.

Methods for creating storage

To make it easier to create the storage you want, SnapDrive for UNIX provides some basic formats for the `snapdrive storage create` command. This is because the storage create operation fall into the following general categories:

- Creating LUNs: This command automatically creates the LUNs on the storage system, but does not create any additional storage entities. SnapDrive for UNIX performs all of the tasks associated with host preparation and

discovery for each LUN, as well as mapping and connecting to it. See "Creating LUNs without host entities" on page 170 for the steps required to do this.

- Creating a file system directly on a LUN and setting up the LUN automatically: SnapDrive for UNIX performs all the actions needed to set up the file system. You do not need to specify any LUNs for it to create. See "Creating a file system on a LUN and setting up the LUN automatically" on page 170 for the steps required to do this.
- Creating a file system directly on a LUN and specifying the LUN you want associated with it: In this case, you use the command to specify the file system you want to set up, and the LUN you want to associate with the file system. See "Creating a file system on a LUN and specifying the LUN" on page 171.
- Creating an LVM and setting up the LUN automatically: This command lets you create a file system, a logical volume, or a disk group on the host. SnapDrive for UNIX performs all the actions needed to set up the entity, including automatically creating the required disk group and LUN. You do not need to specify any LUN for it to create. See "Creating an LVM entity and setting up the LUN automatically" on page 172 for the steps required to do this.
- Creating an LVM entity on the host and specifying the LUN you want associated with it: In this case, you use the command to specify both the entity you want to set up (file system, logical volume, or disk group) and the LUN you want associated with that entity. See "Creating an LVM entity and specifying the LUN" on page 174 for the steps required to do this.
- Creating a file system on the shared host, that is, cluster environment: In this case, you create a file system, a logical volume, or a disk group on the shared host. See Chapter 6, "Creating a shared file system," on page 175 for steps to do this.

Guidelines for the storage create operation

Follow these guidelines when using the `snapdrive storage create` command:

- You cannot create a disk group, host volume, or file system using LUNs from more than one storage system volume. If you list LUNs from different storage system volumes with the `-lun` option, you cannot include the `-dg`, `-hostvol`, or `-fs` option on the command line.
- The `-nolvm` option creates a file system directly on a LUN without activating the host LVM. You cannot specify host volumes or disk groups when you use this option.
- You cannot use SnapDrive for UNIX storage provisioning commands for NFS files or directory trees.

- If you use the `snapdrive storage create` command to create a file system directly on a LUN, you cannot specify more than one LUN. SnapDrive for UNIX always creates a new LUN when you use this form of the command.
- Some operating systems, such as Linux and Solaris, have limits on how many LUNs you can create. If your host is running one of these operating systems, you might want to run the `snapdrive storage config` commands. For more information, see "Preparing hosts for adding LUNs" on page 120.
- On Solaris hosts, if both UFS and Veritas stack are installed, use "ufs" value for the `-fstype` command to create an UFS file system directly on a LUN.
- On an HP-UX host with DMP multipathing solution, when you create a raw LUN or a storage entity created with the `-nolvm` option, the DMP multipathing solution is used, whereas the storage entity created without the `-nolvm` option uses the PVLinks multipathing solution.

  Also, creating a file system on a raw LUN with DMP multipathing solution is not supported.

Guidelines for the storage create operation in a cluster environment:

- The `snapdrive storage create` command can be executed from any node in the cluster.
- For storage create operation to be successful neither should be true:
  - The storage entities should not be present on any node in the cluster.
  - The LUNs should not be mapped to any node in the cluster.
- You can create storage entity on a specific node either by using the `-devicetype dedicated` or `shared` option. If you are creating a storage entity that is in a dedicated mode, you can omit the `-devicetype` option in the command line syntax altogether, because the default value is `dedicated`.
- Cluster-wide storage creation of file system is supported on disk groups that uses Veritas volume manager with the Veritas file system (VxFS). This operation is not supported on raw LUNs; that is, the `-nolvm` option is not supported.
- The `-igroup` option is not supported in the storage create operation.
- The storage create operation fails if one of the following happens:
  - If any error occurs during the process of creating a storage entity. SnapDrive for UNIX executes this operation from the master node in a cluster. Before creating the shared storage entities, it creates LUNs, maps the LUNs to the master node, and then maps the LUNs to all the nonmaster nodes. SnapDrive for UNIX will internally creates and manages the igroups for all the nodes.

- If a node in the cluster shuts down and reboots before starting the clustered volume manager (CVM), the shared disk group used by the LUNs should be discovered on the node. By default, the LUNs are visible, if the FCP port address is not changed; otherwise, the LUNs have to be mapped using the `snapdrive storage connect` command.

Information required for snapdrive storage create

The following table lists the information you need to supply when you use the `snapdrive storage create` command to create storage.

Requirement	Argument
Decide the type of storage you want to provision. Based on the command you enter, you can create any of the following: ◆ LUNs If you create one or more LUNs, the first argument must use the long form of the LUN name, which specifies the storage system name, the volume name, and the name of the LUN within the volume. To specify additional LUNs, you can use the LUN name (short name) alone if the new LUN is on the same storage system and volume as the previous LUN. Otherwise, you can specify a new storage system name and volume name (or just a volume name) to replace the previous values. ◆ A file system created directly on a LUN If you create a file system on a LUN, the first argument must be the `-fs` mountpoint. To create the file system on a LUN in a storage system and volume, use the `-filervol` argument and specify the name of the storage system and volume. To create the file system on a specific LUN, use the `-lun` argument and specify the storage system name, volume name, and LUN name. You must also include the `-nolvm` option to create the file system on the LUN without activating the host LVM. By default, SnapDrive for UNIX automatically performs all of the tasks associated with host preparation and discovery for the LUN, as well as mapping and connecting to it. If you create a LUN on a Linux host, SnapDrive for UNIX performs the following actions: ❖ Creates the LUN. ❖ Configures the LUN into one partition.	

Requirement	Argument
◆ LVM disk groups with host volumes and file systems When you specify a disk or volume group, file system, or host or logical volume, SnapDrive for UNIX performs all the actions necessary to create the entity you specify. You can either explicitly specify the LUNs, or just supply the storage system and volume information and let SnapDrive for UNIX create the LUNs automatically. If you are creating an entity such as a file system, you do not need to supply a value for a disk or volume group. SnapDrive for UNIX automatically creates one.	
◆ A LUN (`-lun`)	*long_lun_name*
◆ Additional LUNs	*lun_name* (long or short form)
◆ Disk group (`-dg` *dgname*) or volume group (`-vg` *vgname*)	*disk or volume group name*
SnapDrive for UNIX creates a disk/volume group to hold the LUNs based on the value you enter with the `-dg` option. The name you supply for the group must not exist.	
◆ Host volume (`-hostvol` *file_spec*) or logical volume (`-lvol` *file_spec*)	*Host or logical volume name*
◆ File system (`-fs` *file_spec*)	*filesystem_name*
◆ `-nolvm`	~
Required: If you are creating a file system that resides directly on a LUN, specify the `-nolvm` option.	
◆ Lun size (`-lunsize`)	*size*
◆ Disk group size (`-dgsize`) ◆ Volume group size (`-vgsize`)	*size*
Specify the size in bytes of each entity being created. The size of the LVM entity depends on the aggregated size of the LUNs you request. To control the size of the host entity, use the `-dgsize` option to specify the size in bytes of the underlying disk group.	
◆ Path to storage system volume (`-filervol`)	*long_filer_path*

<table>
<tr><th>Requirement</th><th>Argument</th></tr>
<tr><td>◆ (<code>-lun</code>)</td><td>long_lun_path</td></tr>
<tr><td colspan="2">Specify the storage system and its volume where you want SnapDrive for UNIX to create the LUNs automatically.
◆ Use the <code>-filervol</code> option to specify the storage system and volume where you want the LUNs created.
Do not specify the LUN. SnapDrive for UNIX creates the LUN automatically when you use this form of the <code>snapdrive storage create</code> command. It uses system defaults to determine the LUN IDs, and the size of each LUN. It bases the names of the associated disk/volume groups on the name of the host volume or file system.
◆ Use the <code>-lun</code> option to name the LUNs that you want to use.</td></tr>
<tr><td>◆ File system type (<code>-fstype</code>)</td><td>type</td></tr>
<tr><td colspan="2">If you are creating a file system, supply the string representing the file system type. The following are the values SnapDrive for UNIX accepts for each host platform:
◆ Solaris: VxFS or UFS
◆ HP-UX: VxFS
◆ AIX: JFS2 or VxFS
Note
On an AIX host, the JFS file system type is not supported for storage operations, but supported for Snapshot operations.
◆ Linux: Ext3
Note
By default, SnapDrive for UNIX supplies this value if there is only one file system type for your host platform. In that case, you do not need to enter it.</td></tr>
<tr><td>◆ <code>-vmtype</code></td><td>type</td></tr>
<tr><td colspan="2">Optional: Specifies the type of volume manager to be used for SnapDrive for UNIX operations.</td></tr>
<tr><td>◆ (<code>-fsopts</code>)</td><td>option name and value</td></tr>
<tr><td>◆ <code>-nopersist</code></td><td>~</td></tr>
<tr><td>◆ <code>-reserve | -noreserve</code></td><td>~</td></tr>
</table>

Requirement	Argument
Optional: If you are creating a file system, you can specify the following options: ◆ `-fsopts` to specify options you want to pass to the host command used to create the file systems. For example, you might supply options that the `mkfs` command would use. The value you supply usually needs to be specified as a quoted string and must contain the exact text to be passed to the command. ◆ `-mntopts` option to specify options that you want passed to the host mount command (for example, to specify host system logging behavior). The options you specify are stored in the host file system table file. Allowed options depend on the host file system type. ◆ The `-mntopts` argument is a file system `-type` option that is specified using the `mount` command "-o" flag. Do not include the "-o" flag in the `-mntopts` argument. For example, the sequence `-mntopts tmplog` passes the string `-o tmplog` to the `mount` command, and inserts the text tmplog on a new command line. ◆ `-nopersist` option to create the file system without adding an entry to the file system mount table file on the host (for example, `fstab` on Linux). By default the `snapdrive storage create` command creates persistent mounts. This means that when you create an LVM storage entity on a Solaris, AIX, HP-UX or Linux host, SnapDrive for UNIX automatically creates the storage, mounts the file system, and then places an entry for the file system in the host file system table. On Linux systems, SnapDrive for UNIX adds a UUID in the host file system table. ◆ `-reserve | -noreserve` to create the storage with or without creating a space reservation.	
◆ `-devicetype`	~
Specifies the type of device to be used for SnapDrive for UNIX operations. This can be either "shared" that specifies the scope of LUN, disk group, and file system as cluster-wide or "dedicated" that specifies the scope of LUN, disk group, and file system as local. Although the storage creation process is initiated from the cluster master node, the discovery of LUNS and host preparation of LUNS need to be performed on each cluster node. Therefore, you should ensure that the `rsh` or `ssh` access-without-password-prompt for SnapDrive for UNIX is allowed on all the cluster nodes. You can find the current cluster master node using the SFRAC cluster management commands. The `-devicetype` option specifies the type of device to be used for SnapDrive for UNIX operations. If you do not specify the `-devicetype` option in SnapDrive for UNIX commands that supports this option, then it is equivalent to specifying `-devicetype dedicated`.	
◆ Igroup name (`-igroup`)	*ig_name*
Optional: NetApp recommends that you use the default igroup for your host instead of supplying an igroup name.	

Creating LUNs without host entities

To provision storage by creating LUNs on the storage system, use the following syntax:

```
snapdrive storage create -lun long_lun_name [lun_name ...] -lunsize
size [{-reserve | -noreserve}] [-igroup ig_name [ig_name ...]]
```

Result: SnapDrive for UNIX creates the LUNs you specify.

Example: This example creates three LUNs on the storage system acctfiler. Each LUN is 10 GB.

```
snapdrive storage create -lun acctfiler:/vol/vol1/lunA lunB lunC -
lunsize 10g
```

Creating a file system on a LUN and setting up the LUN automatically

To create a file system directly on a LUN, and have SnapDrive for UNIX automatically create the associated LUN, use the following format:

```
snapdrive storage create -fs file_spec -nolvm [-fstype type] [-
fsopts options] [-mntopts options] [-nopersist] -filervol
long_filer_path -lunsize size [-igroup ig_name [ig_name ...]] [{ -
reserve | -noreserve }]
```

Attention

On HP-UX hosts, creating a file system on a raw LUN with DMP multipathing solution is not supported.

Result: SnapDrive for UNIX creates the file system you specify and creates a LUN for it on the storage system you specify. It performs all of the tasks associated with host preparation and discovery for the LUNs, as well as mapping and connecting the LUNs to the host entity.

Examples

Example 1: This example creates a 100-MB file system that is created directly on the LUN:

```
# snapdrive storage create -fs /mnt/acct1 -filervol
acctfiler:/vol/vol1 -lunsize 100m -nolvm
```

Example 2: This example creates a file system on raw a LUN without any volume manager:

```
# snapdrive storage create -fs /mnt/vxfs2 -fstype vxfs -lun
snoopy:/vol/vol1/lunVxvm2 -lunsize 50m -nolvm
```

```
LUN snoopy:/vol/vol1/lunVxvm2 ... created

mapping new lun(s) ... done
discovering new lun(s) ... done

LUN to device file mappings:
- snoopy:/vol/vol1/lunVxvm2 => /dev/vx/dmp/Disk_1

file system /mnt/vxfs2 created
```

Creating a file system on a LUN and specifying the LUN

To create a file system directly on a LUN, and specify the LUNs that are created as part of it, use the following format:

```
snapdrive storage create -fs file_spec -nolvm [-fstype type]
[-vmtype type] [-fsopts options] [-mntopts options] [-nopersist]
-lun long_lun_name -lunsize size [-igroup ig_name [ig_name ...]]
[{ -reserve | -noreserve }]
```

Result: SnapDrive for UNIX creates the file system on the storage system, volume and LUN you specify. It performs all of the tasks associated with host preparation and discovery for the LUNs, as well as mapping and connecting the LUNs to the host entity.

Examples

Example 1: This example creates a 100-MB file system on luna, in acctfiler:/vol/vol1:

```
# snapdrive storage create -fs /mnt/acct1 -lun
acctfiler:/vol/vol1/luna
-lunsize 100m -nolvm
```

Example 2: This example creates a JFS2 file system on a raw LUN, on an AIX host:

```
# snapdrive storage create -fs /mnt/jfs1 -fstype jfs2 -lun
snoopy:/vol/vol1/lunLvm1 -lunsize 100m -nolvm

        LUN snoopy:/vol/vol1/lunLvm1 ... created

        mapping new lun(s) ... done
        discovering new lun(s) ... done

        LUN to device file mappings:
        - snoopy:/vol/vol1/lunLvm1 => /dev/hdisk2
```

```
        file system /mnt/jfs1 created
```

Example 3: This example creates a JFS2 file system on a raw LUN, on an HP-UX host:

```
# snapdrive storage create -fs /mnt/vxfs1 -fstype vxfs -lun
snoopy:/vol/vol1/lunVxvm1 -lunsize 100m -nolvm

        LUN snoopy:/vol/vol1/lunVxvm1 ... created

        mapping new lun(s) ... done
        discovering new lun(s) ... done

        LUN to device file mappings:
        - snoopy:/vol/vol1/lunVxvm1 => /dev/dsk/c17t0d0

        file system /mnt/vxfs1 created
```

Creating an LVM entity and setting up the LUN automatically

To create an entity on the host, such as a file system, logical volume, or disk group, and have SnapDrive for UNIX automatically create the associated LUN, use the following syntax:

```
snapdrive storage create host_lvm_fspec -filervol long_filer_path
-dgsize size [-igroup ig_name [ig_name ...]]
[{ -reserve | -noreserve }]
```

Remember the following when you execute this command:

- The `host_lvm_fspec` argument lets you specify whether you want to create a file system, logical volume, or disk group. This argument has three general formats. The format you use depends on the entity you want to create.

  To create a file system, use this format:

  ```
  -fs file_spec [-fstype type] [-fsopts options] [-mntopts
  options] [-nopersist] [ -hostvol file_spec] [ -dg dg_name]
  ```

  To create a logical or host volume, use this format:

  ```
  [-hostvol file_spec] [-dg dg_name]
  ```

  To create a disk or volume group, use this format:

  ```
  -dg dg_name
  ```

- If you create a file system, you may also include the host volume specifications, the disk group specifications, or both specifications to indicate the host volume and/or disk group on which the file system will be

based. If you do not include these specifications, SnapDrive for UNIX automatically generates the names for the host volume and/or disk group.

- When you specify a host volume, SnapDrive for UNIX creates a concatenated host volume. While this is the only format SnapDrive for UNIX supports when creating host volumes, it does allow you to manipulate existing striped host volumes.

Result: SnapDrive for UNIX creates the host entity you specify and creates LUNs for it on the storage system you specify. It performs all of the tasks associated with host preparation and discovery for each of the LUNs, as well as mapping and connecting the LUNs to the host entity.

Examples

Example 1: This example creates the file system acctfs with a Solaris file type of VxFS. It sets up LUNs on the storage system acctfiler and creates a disk group that is 1 GB:

```
# snapdrive storage create -fs /mnt/acctfs -fstype vxfs -filervol
acctfiler:/vol/acct -dgsize 1g
```

Example 2: This example on an HP-UX host creates the file system acctfs and specifies only the storage system volume. SnapDrive for UNIX internally creates and names the entities under the file system.

```
# snapdrive storage create -fs /mnt/acctfs -filervol
acctfiler:/vol/vol1 -vgsize 10g

        LUN acctfs_SdLun ... created

        mapping new lun(s) ... done
        discovering new lun(s) ... done

        LUN to device file mappings:
        - acctfiler:/vol/vol1/acctfs_SdLun => /dev/rdsk/c9t0d6

        disk group acctfs_SdDg created
        host volume acctfs_SdHv created
        file system /mnt/acctfs created
```

Creating an LVM entity and specifying the LUN

If you want create a host entity such as a file system, logical volume, or disk group and specify the LUN that is created as part of it, use the following syntax:

```
snapdrive storage create host_lvm_fspec -lun long_lun_name
[lun_name ...] -lunsize size [-igroup ig_name [ig_name ...]]
[{ -reserve | -noreserve }]
```

Result: SnapDrive for UNIX creates the host entity and the LUNs you specify.

Examples

Example 1: This example creates the file system /mnt/acctfs with an AIX file system type of jfs2. It sets up three LUNs on the storage system acctfiler. Each LUN is 10 GB:

```
# snapdrive storage create -fs /mnt/acctfs -fstype jfs2
-lun acctfiler:/vol/vol1/lunA lunB lunC -lunsize 10g
        LUN acctfiler:/vol/vol1/lunA ... created
        LUN acctfiler:/vol/vol1/lunB ... created
        LUN acctfiler:/vol/vol1/lunC ... created

        mapping new lun(s) ... done
        discovering new lun(s) ... done

        LUN to device file mappings:
        - acctfiler:/vol/vol1/lunA => hdisk2
        - acctfiler:/vol/vol1/lunB => hdisk3
        - acctfiler:/vol/vol1/lunC => hdisk4

        disk group acctfs_SdDg created
        host volume acctfs_SdHv created
        file system /mnt/acctfs created
```

Example 2: This example on a Solaris host creates the file system acctfs on three LUNs and explicitly names the volume group and host volume underneath it. Each LUN is 10 GB:

```
# snapdrive storage create -fs /mnt/acctfs -hostvol
acctfsdg/acctfshv -lun acctfiler:/vol/vol1/lunA lunB lunC -lunsize
10g

        LUN acctfiler:/vol/vol1/lunA ... created
        LUN acctfiler:/vol/vol1/lunB ... created
        LUN acctfiler:/vol/vol1/lunC ... created

        mapping new lun(s) ... done
        discovering new lun(s) ... done
```

```
LUN to device file mappings:
- acctfiler:/vol/vol1/lunA => /dev/vx/rdmp/c4t0d3s2
- acctfiler:/vol/vol1/lunB => /dev/vx/rdmp/c4t0d7s2
- acctfiler:/vol/vol1/lunC => /dev/vx/rdmp/c4t0d8s2

disk group acctfsvg created
host volume acctfshv created
file system /mnt/acctfs created
```

Creating a shared file system

If you want to create a shared file system on a shared Solaris host, using the following syntax:

```
snapdrive storage create -fs file_spec -nolvm [-fstype type]
[-fsopts options] [-mntopts options] [-nopersist]
{ -lun long_lun_name | -filervol long_filer_path }
-lunsize size [-igroup ig_name [ig_name ...]]
[{ -reserve | -noreserve }] -devicetype [{shared | dedicated}]
```

Example: This example creates the file system sfortesting. It sets up LUNs on the storage system f270-197-109 and creates a disk group that is 300m in size in a cluster environment (shared file system). The operation is executed from the non-cluster-master node, but the command is shipped to the master node for execution:

```
# snapdrive storage create -fs /mnt/sfortesting -dgsize 300m -
filervol f270-197-109:/vol/vol2 -devicetype shared
Execution started on cluster master: sfrac-57

LUN sfortesting_SdLun ... created

mapping new lun(s) ... done
discovering new lun(s) ... done


LUN to device file mappings:
- f270-197-109:/vol/vol2/sfortesting_SdLun => /dev/vx/dmp/c5t0d7s2
Connecting cluster node: sfrac-58
mapping lun(s) ... done
discovering lun(s) ... done

LUN f270-197-109:/vol/vol2/sfortesting_SdLun connected
- device filename(s): /dev/vx/dmp/c3t0d1s2
disk group sfortesting_SdDg created
```

```
host volume sfortesting_SdHv created
file system /mnt/sfortesting created
```

Displaying information about storage

Command to use to display available storage

The `snapdrive storage show` or `list` command shows LUNs or NFS directory trees underlying one or more storage entities.You can use the `snapdrive storage show` command to find out what will be in a Snapshot copy of a disk group, host volume, file system, or NFS directory tree.

With this command, you can display the following:

- LUNs available for specific storage systems or storage system volumes
- LUNs associated with file systems, host volumes or disk groups
- NFS mountpoints and directory trees
- LUNs known to a specific host, along with any LVM entities that they include
- Devices known to a specific host
- Resources on the shared and dedicated hosts

Note

You can use either `snapdrive storage show` or `snapdrive storage list` in the command line. These commands are synonyms.

Methods for displaying storage information

To make it easier to display information about storage, SnapDrive for UNIX provides several formats for the `snapdrive storage show` command. This is because storage show operations fall into the following general categories:

- Displaying information about a specific LUN.

  For the steps required to do this, see "Displaying information about LUNs" on page 181.
- Listing information about LUNs available for specific storage systems or storage system volumes.

  For the steps required to do this, see "Displaying LUNs based on the storage system" on page 182.
- Displaying information about LUNs associated with the arguments you specify.

  These arguments can include NFS entities, file systems, host volumes, or disk groups. If you use the -verbose option on the command line, SnapDrive for UNIX provides detailed output, such as showing the storage hierarchy

including the backing LUNs. For the steps required to do this, see "Displaying information about a storage entity" on page 182.

- Displaying information about the devices known to the host.

  For the steps required to do this, see "Displaying information about devices" on page 185.

- Displaying information about all devices and LVM entities known to the host.

  For the steps required to do this, see "Displaying information about devices and LVM entities" on page 185.

- Displaying the status of a resource as shared or dedicated.

  For the steps required to do this, see Chapter 6, "Displaying the status of shared and dedicated resources," on page 189.

Guidelines for the storage show command

The `snapdrive storage show` command has the following guidelines:

- The adapter name is not displayed.
- The `snapdrive storage show -filer` and `snapdrive storage show -filervol` commands do not display output for NFS entities. You must use the `-fs` option to display NFS information.
- The status of non-primary paths is displayed as a hyphen (-) instead of secondary (S).
- If there is no volume manager or host volume underlying a file system, SnapDrive for UNIX displays the following:

```
# snapdrive storage show -fs /pub
 fs: /dev/rdsk/c1t0d1s2         mountpoint: /pub
 device filename adapter path size   state    lun path
 ----------------------------------------------------------
 /dev/rdsk/c1t0d1s2   -   P    10g   online  myfiler:/vol1/lun-1
 /dev/rdsk/c2t0d1s2   -        10g   online  myfiler:/vol1/lun-1
```

The `snapdrive storage show` command requires time to gather information, and may take a few minutes to display the information you request.

Guidelines for the storage show command in a cluster environment:

- The `snapdrive storage show` command can be executed from any node in the cluster.
- The `snapdrive storage show` command with either the `-devicetype shared` or `dedicated` option shows all storage entities that are present.

Information required for snapdrive storage show

The following table lists the information you need to supply when you use the `snapdrive storage show` command to display storage entities.

Requirement	Argument
Based on the command you enter, you can display information about any of the following: ◆ LUNs on a storage system accessible to the current host ◆ LUNs on storage system volumes accessible to the current host ◆ NFS mountpoints and directory trees ◆ LUNs known to a specific host, along with any LVM entities that they include ◆ Devices known to a specific host ◆ Resources on the shared and dedicated hosts ◆ Disk or volume groups ◆ File systems ◆ Host or logical volumes The value you enter for the *file_spec* argument must identify the storage entity about which you want to display information. The command assumes the entities are on the current host. When you display information about a LUN on a storage system volume, you must supply the path to the LUN, including the name of the storage system and the volume, as the value for *long_filer_path* and *long_LUN_name.*	
◆ Storage system (`-filer`)	*filername*
◆ LUNs on a storage system volume on the storage system (`-filervol`)	*long_filer_path*
◆ Specific LUNs (`-lun`)	*long_LUN_name*
◆ Disk group (`-dg` *file_spec*) or volume group (`-vg` *file_spec*)	*name of the disk or volume group*
◆ File system (`-fs` *file_spec*)	*filesystem_name*
◆ Host volume (`-hostvol` *file_spec*) or logical volume (`-lvol` *file_spec*)	*name of the host or logical volume*

Requirement	Argument
To display information about all devices known to the host, use the `-devices` option.	
`-devices`	~
To display information about all devices and LVM entities known to the host, include the `-all` option.	
`-all`	~
To display additional information, include the `-verbose` option.	
`-verbose`	~
To prevent all output, both diagnostic and normal output, include the `-quiet` option. This option overrides the `-verbose`, which can be useful for existence checking within a script.	
`-quiet`	~
To display the type of device to be used for SnapDrive for UNIX operations. This can be either “shared” that specifies the scope of LUN, disk group, and file system as cluster-wide or “dedicated” that specifies the scope of LUN, disk group, and file system as local.	
`-devicetype`	~
Specifies the type of device to be used for SnapDrive for UNIX operations. This can be either “shared” that specifies the scope of LUN, disk group, and file system as cluster-wide or “dedicated” that specifies the scope of LUN, disk group, and file system as local.	
◆ `-fstype`	*type*
◆ `-vmtype`	*type*
Optional: Specifies the type of file system and volume manager to be used for SnapDrive for UNIX operations.	
◆ `-status`	~
Optional: To know if the volume or LUN is cloned.	

Displaying information about LUNs

To display information about LUNs, use the following syntax:

```
snapdrive storage { show | list } -lun long_lun_name [lun_name ...]
[-verbose] [-quiet] [-status]
```

Note

For details about using all the options and arguments available with this command, see "SnapDrive for UNIX options, keywords, and arguments" on page 348.

Result: Depending on the command options you select, SnapDrive for UNIX either displays basic information about the LUNs or lists the LUNs associated with the host entity.

Example: The following are examples of possible command lines:

```
# snapdrive storage show -verbose -lun BigFiler3:/vol/vol1:lunA
lunB lunC

# snapdrive storage show -verbose -filer BigFiler3 BigFiler4
```

Displaying LUNs based on the storage system

To display information about the available LUNs on a certain storage system, use the following syntax:

```
snapdrive storage { show | list } -filer filername [filername...]
[-verbose] [-quiet]
```

```
snapdrive storage { show | list } -filervol long_filer_path
[filer_path ...] [-verbose] [-quiet]
```

Note

For details about using all the options and arguments available with this command, see "SnapDrive for UNIX options, keywords, and arguments" on page 348.

Result: Depending on the command options you select, SnapDrive for UNIX either displays basic information about the LUNs, or lists the LUNs associated with the host entity.

Examples: The following are examples of possible command lines:

```
# snapdrive storage show -verbose -filer BigFiler3 BigFiler4
```

```
# snapdrive storage show -verbose -filervol BigFiler3:/vol/vol1
```

Displaying information about a storage entity

To display information about a storage entity, such as an NFS directory tree, file system, disk group, volume group, or host volume, use the following syntax:

```
snapdrive storage {show | list} {-dg | -fs | -hostvol }
file_spec [file_spec ...]  [{-dg | -fs | -hostvol }
file_spec [file_spec ...]] [-verbose] [-fstype type] [-vmtype type]
[-quiet] [-status]
```

Note

For details about using all the options and arguments available with this command, see "SnapDrive for UNIX options, keywords, and arguments" on page 348.

Result: SnapDrive displays information about the storage entity you requested.

Examples

Example 1: These are examples of command lines that use the `storage show` command:

```
# snapdrive storage list -dg dg1 dg2
# snapdrive storage show -quiet -dg QA_dg_test
```

```
# snapdrive storage show -fs /mnt/fs21 /mnt/fs41
# snapdrive storage show -hostvol dg2/vol1 dg4/myvol3
# snapdrive storage list -dg dg2 -fs /mnt/fs42
# snapdrive storage show -hostvol dg2/vol3 -fs /mnt/fs42
```

Example 2: This example shows output that appears when you display information about the disk group dg1 on a Solaris host (the output varies slightly depending on which host you use):

```
# snapdrive storage show -dg dg1
dg: dg1
hostvol: /dev/vx/dsk/dg1/vol1   state: AVAIL
hostvol: /dev/vx/dsk/dg1/vol3   state: AVAIL
hostvol: /dev/vx/dsk/dg1/vol4   state: AVAIL
fs: /dev/vx/dsk/dg1/vol1        mount point: /pub (persistent)
fs: /dev/vx/dsk/dg1/vol3        mount point: NOT MOUNTED
fs: /dev/vx/dsk/dg1/vol4        mount point: NOT MOUNTED

 device filename        adapter path size proto   state   lun path
----------------------- ------- ---- -----  ----   -----   --------
-------
/dev/vx/rdmp/c1t0d2s2     -      P 10g     fcp online
myfiler:/vol/vol1/lun2
/dev/vx/rdmp/c1t0d7s2     -      P 10g     fcp online
myfiler:/vol/vol1/lun7
```

Example 3: This example uses an HP-UX host. In this example, SnapDrive for UNIX displays information about the volume group vg01.

```
# snapdrive storage show -vg vg01

dg: vg01
hostvol: /dev/vg01/lvol1        state: AVAIL
fs: /dev/vg01/lvol1     mount point: NOT MOUNTED

 device filename        adapter path size proto state   lun path
----------------------- ------- ---- ----- ---- -----   -----------
----
/dev/rdsk/c22t0d0        -    P10g  10g   fcp online
 toaster:/vol/vol1/myhp7-1
/dev/rdsk/c20t0d2         -    P5g  10g   fcp online
toaster:/vol/vol1/myhp7-3
```

Example 4: This example uses a Solaris host. In this example, SnapDrive for UNIX displays information about the disk group dg1:

```
# snapdrive storage show -vg dg1
```

```
saving name dg1

dg: dg1
hostvol: /dev/vx/dsk/dg1/vol1   state: AVAIL
fs: /dev/vx/dsk/dg1/vol1        mount point: NOT MOUNTED

 device filename        adapter path size  proto  state  lun path
----------------------- ------- ---- ----- ------ -----  ----------
-----
/dev/vx/rdmp/c1t0d0s2     -     P    500m   fcp  online
sky:/vol/vol1/mylun_lun0
```

Example 5: This example uses an AIX host. In this example, SnapDrive for UNIX displays information about the disk group dg1:

```
# snapdrive storage show -dg dg1

dg: dg1
hostvol: /dev/dg1_log   state: AVAIL
hostvol: /dev/dg1_vol1  state: AVAIL
fs: /dev/dg1_log        mount point: MOUNTED (persistent)
fs: /dev/dg1_vol1       mount point: MOUNTED (persistent)

 device filename adapter path size   proto state   lun path
----------------  ----   ---- ----   ----- -----   ----------------
-----
/dev/rhdisk2      -       P    2g
   fcp   online flip:/vol/vol1/nate-aix5_lun0

/dev/rhdisk3      -       P    2g    fcp   online
flip:/vol/vol1/nate-aix5_lun1:
```

Example 6: This example shows the output for an NFS directory tree:

```
# snapdrive storage show -fs /mnt/acctfs_nfs
NFS device: myfiler:/vol/vol1    mount point: /mnt/acctfs_nfs (non-
persistent)
```

Displaying information about devices

To display information about the LUNs and devices visible to the host, use the following syntax:

```
snapdrive storage { show | list } -devices
```

Note

For details about using all the options and arguments available with this command, see "SnapDrive for UNIX options, keywords, and arguments" on page 348.

Result: SnapDrive for UNIX displays information about the devices that are visible to the host.

Example: The following example shows the output SnapDrive for UNIX displays when you use the `-devices` option:

```
# snapdrive storage show -devices
Connected LUNs and devices:

device filename      adapter  path  size  proto  state  lun path
---------------      ------   ----  ----  -----  ----   --------
/dev/vx/dmp/c1t0d3s2   -       P    500m   fcp   online
albacore:/vol/
vol1/cs_connect_lun1
/dev/vx/dmp/c1t0d4s2   -       P    500m   fcp   online
albacore:/vol/
vol1/cs_connect_lun2
/dev/vx/dmp/c1t0d5s2   -       P    100m   fcp   online
albacore:/vol/
vol1/luntestvit1
/dev/vx/dmp/c1t0d0s2   -       P    1g     fcp   online
albacore:/vol/
vol1/bosun0
/dev/vx/dmp/c1t0d1s2   -       P    1g     fcp   online
albacore:/vol/
vol1/bosun1
/dev/vx/dmp/c1t0d2s2   -       P    1g     fcp   online
albacore:/vol/
vol1/bosun2
```

Displaying information about devices and LVM entities

To display information about the devices and the LVM entities that are available to the host, use the following syntax:

```
snapdrive storage { show | list } -all
```

Note

For details about using all the options and arguments available with this command, see "SnapDrive for UNIX options, keywords, and arguments" on page 348.

Result: SnapDrive for UNIX displays information about the LVM entities and devices that are visible to the host.

Examples

Example 1: The following example shows a portion of the output SnapDrive for UNIX displays when you use the `-all` option:

```
# snapdrive storage show -all
Connected LUNs and devices:

device filename      adapter  path  size  proto  state  lun path
---------------      -------   ----  ----  -----  ----   --------/dev/vx/dmp/c1t0d3s2
  -       P    500m   fcp   online albacore:/vol/vol1/
cs_connect_lun1
/dev/vx/dmp/c1t0d4s2   -       P    500m   fcp   online albacore:/vol/vol1/
cs_connect_lun2
/dev/vx/dmp/c1t0d5s2   -       P    100m   fcp   online albacore:/vol/vol1/
luntestvit1

Host devices and file systems:

dg: csdg1
hostvol: /dev/vx/dsk/csdg1/hv1_1 state: AVAIL
hostvol: /dev/vx/dsk/csdg1/hv1_2 state: AVAIL
hostvol: /dev/vx/dsk/csdg1/hv1_3 state: AVAIL
fs: /dev/vx/dsk/csdg1/hv1_1 mount point: /mnt/check_submit/csdg1/hv1_1 (persistent)
fs: /dev/vx/dsk/csdg1/hv1_2 mount point: NOT MOUNTED fs: /dev/vx/dsk/csdg1/hv1_3 mount
point: NOT MOUNTED
```

Example 2: The following example shows all the file systems, logical volumes, and disk groups created with Veritas stack and LVM on an AIX host:

```
# snapdrive storage show -all

Connected LUNs and devices:

 device filename      adapter path   size   proto  state  clone  lun path
backing snapshot
----------------      ------- ----   ----   -----  -----  -----  --------
----------------
/dev/hdisk10               -     P       100m    fcp     online   No
tonic:/vol/vol1/lunRaw             -

Host devices and file systems:

dg: vxvm1               dgtype VxVM
hostvol: /dev/vx/dsk/vxvm1/vxfs1_SdHv   state: AVAIL
fs: /dev/vx/dsk/vxvm1/vxfs1_SdHv       mount point: /mnt/vxfs1 (persistent) fstype vxfs

 device filename      adapter path   size   proto  state  clone  lun path
backing snapshot
----------------      ------- ----   ----   -----  -----  -----  --------
----------------
/dev/vx/dmp/F8400_0        -     P       100m    fcp     online   No
tonic:/vol/vol1/lunVxvm1           -

dg: lvm1                dgtype LVM_AIX
hostvol: /dev/jfs1_SdHv state: AVAIL
fs: /dev/jfs1_SdHv      mount point: /mnt/jfs1 (persistent) fstype jfs2

 device filename      adapter path   size   proto  state  clone  lun path
backing snapshot
----------------      ------- ----   ----   -----  -----  -----  --------
----------------
/dev/hdisk6                -     P       100m    fcp     online   No
tonic:/vol/vol1/lunLvm1            -

raw device: /dev/vx/dmp/F8400_1 mount point: /mnt/vxfs2 (persistent) fstype vxfs

 device filename      adapter path   size   proto  state  clone  lun path
backing snapshot
----------------        ------- ----    ----    -----   -----   -----   -------- ------
---------
/dev/vx/dmp/F8400_1        -     P       100m    fcp     online   No
tonic:/vol/vol1/lunVxvm2           -
```

```
raw device: /dev/hdisk8 mount point: /mnt/jfs2 (persistent) fstype jfs2

 device filename      adapter path   size   proto  state  clone  lun path
backing snapshot
----------------      ------- ----   ----   -----  -----  -----  --------
----------------
/dev/hdisk8                -     P       100m    fcp     online   No
tonic:/vol/vol1/lunLvm2            -
```

Displaying the status of shared and dedicated resources

If you want to know whether a resource is shared or dedicated, use the following syntax:

```
snapdrive storage { show | list } {-all | device} -devicetype
{shared|dedicated}
```

Note

For details about using all the options and arguments available with this command, see "SnapDrive for UNIX options, keywords, and arguments" on page 348.

Examples

Example 1: This example displays information of a shared resource:

```
# snapdrive storage show -all -devicetype shared

Connected LUNs and devices:

 device filename      adapter path   size   proto   state   clone   lun path
backing snapshot        shared
----------------      ------- ----   ----   -----   -----   -----   --------
----------------        ------
/dev/vx/dmp/c5t0d0s2       -     P       5g      fcp     online   No     f270-197-
109:/vol/vol1/VXFEN_COORD_LUN1    -            Yes
/dev/vx/dmp/c5t0d1s2       -     P       5g      fcp     online   No     f270-197-
109:/vol/vol1/VXFEN_COORD_LUN2    -            Yes
/dev/vx/dmp/c5t0d2s2       -     P       5g      fcp     online   No     f270-197-
109:/vol/vol1/VXFEN_COORD_LUN3    -            Yes

Host devices and file systems:

dg: shared

 device filename      adapter path   size   proto   state   clone   lun path
backing snapshot        shared
----------------      ------- ----   ----   -----   -----   -----   --------
----------------        ------
/dev/vx/dmp/c5t0d4s2       -     P       72m     fcp     online   No     f270-197-
109:/vol/vol1/shared_SdLun    -            Yes
```

Example 2: This example displays the information about the dedicated resource:

```
# snapdrive storage show -all -devicetype dedicated

Connected LUNs and devices:

 device filename      adapter path   size   proto  state  clone  lun path
backing snapshot        shared
----------------     ------- ----   ----   -----  -----  -----  --------
----------------        ------
/dev/dsk/c5t4d0s2         -     P       10m     fcp     online   No
tonic:/vol/vol1/lunRaw1             -                    No
/dev/dsk/c5t8d0s2         -     P       10m     fcp     online   No
tonic:/vol/vol1/lunRaw1             -                    No
/dev/dsk/c5t4d1s2         -     P       10m     fcp     online   No
tonic:/vol/vol1/lunTest1            -                    No
/dev/dsk/c5t6d0s2         -     P       64m     fcp     online   No
gin:/vol/vol1/lun_one               -                    No

Host devices and file systems:

dg: cmm_SdDg            dgtype VxVM
hostvol: /dev/vx/dsk/cmm_SdDg/cmm_SdHv  state: AVAIL
fs: /dev/vx/dsk/cmm_SdDg/cmm_SdHv       mount point: /mnt/cmm (persistent) fstype VxFS

 device filename      adapter path   size   proto  state  clone  lun path
backing snapshot        shared
----------------     ------- ----   ----   -----  -----  -----  --------
----------------        ------
/dev/vx/dmp/c5t4d2s2      -     P       64m     fcp     online   No
tonic:/vol/vol1/t_lun               -                    No
```

Example 3: This example displays the detailed information about the resources being shared or dedicated using the `-verbose` option in the `snapdrive storage show` command:

```
# snapdrive storage show -all -v

Connected LUNs and devices:

 device filename      adapter path   size   proto  state  clone  lun path
backing snapshot        shared
----------------     ------- ----   ----   -----  -----  -----  --------
----------------        ------
/dev/dsk/c5t4d0s2         -     P       10m     fcp     online   No
tonic:/vol/vol1/lunRaw1             -                    No
/dev/dsk/c5t8d0s2         -     P       10m     fcp     online   No
tonic:/vol/vol1/lunRaw1             -                    No
/dev/dsk/c5t4d1s2         -     P       10m     fcp     online   No
tonic:/vol/vol1/lunTest1            -                    No
/dev/dsk/c5t6d0s2         -     P       64m     fcp     online   No
gin:/vol/vol1/lun_one               -                    No

Host devices and file systems:

dg: cmm_SdDg            dgtype VxVM
hostvol: /dev/vx/dsk/cmm_SdDg/cmm_SdHv  state: AVAIL
fs: /dev/vx/dsk/cmm_SdDg/cmm_SdHv       mount point: /mnt/cmm (persistent) fstype VxFS
Protocol: SCSI

 device filename      adapter path   size   proto  state  clone  lun path
backing snapshot        shared
----------------     ------- ----   ----   -----  -----  -----  --------
----------------        ------
/dev/vx/dmp/c5t4d2s2      -     P       64m     fcp     online   No
tonic:/vol/vol1/t_lun               -                    No
```

Increasing the size of the storage

Resizing the storage to make it larger

SnapDrive for UNIX lets you increase the size of the storage system volume group or disk group. You use the `snapdrive storage resize` command to do this.

Note

This command does not let you resize host volumes or file systems. For example, you can not use the resize command to change the size of a file system on a LUN.

You need to use the LVM commands to resize host volumes and file systems after you have resized the underlying disk group. For information on performing those tasks, see "Resizing host volumes and file systems" on page 196.

You can put the storage resize operations into the following general categories:

- Setting a target size in bytes to which you want to increase the storage
- Specifying a number of bytes by which you want to increase the storage

SnapDrive for UNIX adds a system-generated LUN. If you specify an amount by which you want to increase the storage, such as 50 MB, it makes the LUN 50 MB. If you specify a target size for the storage, it calculates the difference between the current size and the target size. The difference becomes the size of the LUN it then creates.

Guidelines for the storage resize command

Follow these guidelines when you use the SnapDrive storage resize command:

- The storage resize operation can only increase the size of storage. You cannot use it to decrease the size of an entity.
- All LUNs must reside in the same storage system volume.
- The resize operation is not supported directly on logical host volumes, or on file systems that reside on logical host volumes or on LUNs. In those cases, you must use the LVM commands to resize the storage.
- You cannot resize a LUN; you must use the `-addlun` option to add a new LUN.

Guidelines for the storage resize command in a cluster environment:

- The `snapdrive storage resize` command can be executed from any node in the cluster.
- The `snapdrive storage resize` command does not support the `-devicetype` option.

Information required for snapdrive storage resize

The following is a summary of the information you need to supply when you use the `snapdrive storage resize` command.

Requirement	Argument
Decide whether you want to increase the size of a disk or volume group and enter that entity's name with the appropriate argument.	
◆ Disk group (`-dg` *file_spec*) or volume group (`-vg` *file_spec*)	*name of the disk or volume group*
Decide how you want to increase the storage size. Remember the following when you use this command: ◆ Use the `-growby` option to increase the entity size by the bytes specified in the *size* argument. ◆ Use the `-growto` option to increase the entity size so that the new total size is the number of bytes specified in the *size* argument. ◆ Use the `-addlun` option to increase the entity size by adding a new, internally-generated LUN to the underlying disk group. If you do not use this argument, SnapDrive for UNIX increases the size of the last LUN in the disk group to meet the byte size specified in either the `-growby` option or the `-growto` option.	
◆ Specify the number of bytes by which you want to increase the storage (`-growby` *size*)	*number_of_bytes*
◆ Specify the size in bytes that you want the storage to reach (`-growto` *size*)	*number_of_bytes*
◆ Tell SnapDrive for UNIX to increase the size by adding a new LUN to the disk group `(-addlun)`	~

Requirement	Argument
◆ Tell SnapDrive for UNIX to increase the size with or without creating a space reservation `-reserve | -noreserve`	
Optional: NetApp recommends that you use the default igroup for your host instead of supplying an igroup name.	
◆ Igroup name (`-igroup`)	*ig_name*
◆ `-fstype`	*type*
◆ `-vmtype`	*type*
Optional: Specifies the type of file system and volume manager to be used for SnapDrive for UNIX operations.	

Increasing the size of the storage

To increase the size of the storage, use the following syntax:

```
snapdrive storage resize -dg file_spec { -growby | -growto } size
[-addlun [-igroup ig_name [ig_name ...]]] [{ -reserve | -
noreserve}]] [-fstype type] [-vmtype type]
```

Note

You cannot use the `snapdrive storage resize` command to reduce the size of an entity. You can only increase the size using this command.

The `snapdrive storage resize` command is not supported directly on logical volumes or file systems. For example, you can not use the `snapdrive storage resize` command to resize a file system on a LUN.

Result: This command increases the size of the storage entity (logical volume or disk group) by either of the following:

- Adding bytes to storage (`-growby`).
- Increasing the size to the byte size you specify (`-growto`).

Examples

Example 1: This command line is the same one as in example 1, except that it uses the `-addlun` option to increase the size of the disk group my_dg to 155 MB. When you supply the `-addlun` option, SnapDrive for UNIX automatically creates

a new LUN in the disk group. It makes the LUN large enough so that the accumulated disk group size reaches 155 MB. You do not need to specify any values for the `-addlun` option; SnapDrive for UNIX sets it up automatically:

```
# snapdrive storage resize -dg my_dg -addlun -growto 155m
        discovering filer LUNs in disk group my_dg...done
        LUN x-parrot:/vol/vol1/my_dg_2_SdLun ... created
        mapping new lun(s) ... done
        discovering new lun(s) ... done.
        mapping LUN(s)...done.
        initializing LUN(s) and adding to disk group my_dg...done
Disk group my_dg has been resized
```

Example 2: This command line uses the `-growby` option with the `-addlun` option. SnapDrive for UNIX increases the size of the storage by adding a LUN that is 100 MB:

```
# snapdrive storage resize -dg testdg -growby 100m -addlun
        discovering filer LUNs in disk group testdg...done
        LUN toaster:/vol/vol1/testdg_1_SdLun ... created
        mapping new lun(s) ... done
        discovering new lun(s) ... done.
        mapping LUN(s)...done.
        initializing LUN(s) and adding to disk group testdg...done
Disk group testdg has been resized
```

Example 3: This command allows to increase the size of the disk group either by increasing the size of a LUN or by adding a new LUN in a cluster environment:

```
# snapdrive storage resize -dg shared -growby 100m -addlun
discovering filer LUNs in disk group shared...done
        LUN f270-197-109:/vol/vol1/lunShared_SdLun ... created

        mapping new lun(s) ... done
        discovering new lun(s) ... done.

Connecting cluster node: sfrac-58

        mapping lun(s) ... done
        discovering lun(s) ... done

        LUN f270-197-109:/vol/vol1/lunShared_SdLun connected
                - device filename(s): /dev/vx/dmp/c3t0d4s2
        initializing LUN(s) and adding to disk group shared...done
Disk group shared has been resized
Desired resize of host volumes or file systems
```

```
contained in disk group must be done manually
```

Example 4: This example allows to increase the size of the disk group by using the Veritas stack:

```
# snapdrive storage resize -dg vxvm1 -vmtype vxvm -growby 50m -
addlun
        discovering filer LUNs in disk group vxvm1...done
        LUN snoopy:/vol/vol1/vxvm1_SdLun ... created

        mapping new lun(s) ... done
        discovering new lun(s) ... done.
        initializing LUN(s) and adding to disk group vxvm1...done
Disk group lvm1 has been resized
Desired resize of host volumes or file systems
contained in disk group must be done manually
```

Example 5: This example allows to increase the size of the disk group by using LVM:

```
# snapdrive storage resize -dg lvm1 -vmtype lvm -growby 50m -addlun
        discovering filer LUNs in disk group lvm1...done
        LUN snoopy:/vol/vol1/lvm1_SdLun ... created

        mapping new lun(s) ... done
        discovering new lun(s) ... done.
        initializing LUN(s) and adding to disk group lvm1...done
Disk group lvm1 has been resized
Desired resize of host volumes or file systems
contained in disk group must be done manually
```

Resizing host volumes and file systems

The `snapdrive storage resize` command applies only to storage system disk groups and volume groups. If you want to increase the size of your host volume or file system, you must use LVM commands. The following table summarizes the LVM commands you can use on the different platforms. For more information on these commands, see their man pages.

Host	Volume manager	Host volume	File systems
AIX	LVM	`extendlv`	`chfs`
	VxVM	`vxassist`	`fsadm`

Host	Volume manager	Host volume	File systems
HP-UX	LVM	`lvextend`	`extendfs/fsadm`
	VxVM	`vxassist`	`fsadm`
Linux	LVM	`lvextend`	`resize2fs`
Solaris	VxVM	`vxassist`	`fsadm`

Connecting LUNs and storage entities to the host

About the storage connect command

The `snapdrive storage connect` command connects storage entities to the host. Use the `snapdrive storage connect` command to connect to:

- LUNs
- A file system created directly on a LUN
- Disk groups, host volumes and file systems created on LUNs

When you enter the `snapdrive storage connect` command to connect LUNs to the host, SnapDrive for UNIX performs the necessary discovery and mapping. It does not modify LUN contents.

Guidelines for the storage connect command

Follow these guidelines when you use the `snapdrive storage connect` command:

- Storage that includes LVM entities has special requirements. To use the `snapdrive storage connect` command to connect LVM entities, you must create the storage so that each entity in the storage hierarchy has exactly one instance of the next entity. For example, you can use the `snapdrive storage connect` command to connect a storage hierarchy that has one disk group (dg1) with one host volume (hostvol1) and one file system (fs1). However, you cannot use the `snapdrive storage connect` command to connect a hierarchy that has one disk group (dg1) with two host volumes (hostvol1 and hostvol2) and two file systems (fs1 and fs2).
- On Linux hosts, the `snapdrive storage connect` command connects a file system created directly on a LUN only when the underlying LUN is partitioned.

Guidelines for storage connection in a cluster environment:

- If a new node is added to the cluster configuration that uses a shared disk group or file system, use the `snapdrive storage connect -devicetype shared` command. You can execute the `snapdrive storage connect` from any node in the cluster.
- For a storage connect operation to be successful, the following should not happen:
  - The storage entities should not be present on any node in the cluster.
  - The LUNs should not be mapped to any node in the cluster.

- You can connect to a storage entity on a specific node either by using the `-devicetype dedicated` option or by omitting the `-devicetype` option in the command line syntax, because the default value is `dedicated`.
- Cluster-wide storage connection of file system is supported on disk groups that use Veritas volume manager with Veritas file system (VxFS). This operation is not supported on raw LUNs; that is, the `-nolvm` option is not supported.
- The `-igroup` option is not supported in the `snapdrive storage connect` command.
- The storage connect operation fails if either of the following is true:
  - If any error occurs during the process of connecting a storage entity. SnapDrive for UNIX executes this operation from the master node in a cluster. Before creating the shared storage entities, it creates LUNs, maps the LUNs to the master node, and then map the LUNs to all the nonmaster nodes. SnapDrive for UNIX will internally creates and manages the igroups for all the node.
  - If a node in the cluster shuts down and reboots before starting clustered volume manager (CVM), the shared disk group used by the LUNs should be discovered on the node. By default, the LUNs are visible, if the FCP port address is not changed; otherwise, the LUNs have to be mapped using the `snapdrive storage connect` command.
- You can conduct the shared storage connect operation with storage entities on a LUN created with dedicated storage entity data and subsequently disconnected, only if the storage entities do not exist on any cluster node.
- You can conduct the dedicated storage connect operation with storage entities on a LUN with shared storage entity metadata, only if the current node is not part of the cluster or the storage entities do not exist on the cluster.

Information required for snapdrive storage connect

The following table gives the information you need to supply when you use the `snapdrive storage connect` command.

Requirement	Argument
Specify the LUNs, the file system created directly on a LUN, or the LVM entity that you want to connect to the host. ◆ If you connect one or more LUNs, the first argument must use the long form of the LUN name, which specifies the storage system name, the volume name, and the name of the LUN within the volume. To specify additional LUNs, you can use the LUN name alone if the new LUN is on the same storage system and volume as the previous LUN. Otherwise, you can specify a new storage system name and volume name (or just a volume name) to replace the previous values. ◆ If you connect a file system created directly on a LUN, you must include the long form of the LUN name, and also the `-nolvm` option. ◆ If you connect a LUN with a disk group, host volume, and file system, you must use the `-fs` and `-hostvol` options to specify the file system and host volume. The host volume must include the name of the disk group.	
◆ A LUN (`-lun`)	*long_lun_name*
The first value you supply with the `-lun` option must include the storage system name, volume, and LUN name. To connect multiple LUNs on the same volume, you can use relative path names for the `-lun` option after you supply the complete information in the first path name. When SnapDrive for UNIX encounters a relative path name, it looks for the LUN on the same volume as the previous LUN. To connect additional LUNs that are not on the same volume, enter the full path name to each LUN.	
◆ Additional LUNs	*lun_name* (long or short form)
The *file_spec* given to `-fs` is the name of the file system mountpoint when connecting a file system created directly on a LUN.	
◆ A file system (`-fs` *file-spec*)	*filesystem_name*
To connect a file system that is created on a LUN without activating the host LVM:	
◆ `-nolvm`	
To connect a file system on a host volume: The `-fs` *file_spec* and `-hostvol` *file_spec* you supply identify the LVM file system, disk group and host volumes that you want to connect to a new host. The storage hierarchy that you connect must contain a disk group, host volume and file system, and must conform to the requirements in "Guidelines for the storage connect command" on page 198. You must specify a value for `-fs` and `-hostvol`. The `-hostvol` value must include the name of the disk group.	
◆ Host volume (`-hostvol` *file-spec*)	*disk_group_name* and *host_volume_name*

Requirement	Argument
Optional: Use the `-nopersist` option to connect the storage to a new location without creating an entry in the host file system table (for example, `fstab` on Linux). By default the `storage connect` command creates persistent mounts. This means that when you create an LVM storage entity on a Solaris, AIX, HP-UX or Linux host, SnapDrive for UNIX automatically creates the storage, mounts the file system and then places an entry for the file system in the host file system table.	
◆ `-nopersist`	~
Optional: NetApp recommends that you use the default igroup for your host instead of supplying an igroup name.	
◆ Igroup name (`-igroup`)	*ig_name*
◆ `-devicetype`	~
To specify the type of device to be used for SnapDrive for UNIX operations. This can be either “shared” that specifies the scope of LUN, disk group, and file system as cluster-wide or “dedicated” that specifies the scope of LUN, disk group, and file system as local.	
◆ `-fstype`	*type*
◆ `-vmtype`	*type*
Optional: Specifies the type of file system and volume manager to be used for SnapDrive for UNIX operations.	

Connecting to LUNs

To use the `snapdrive storage connect` command to map LUNs to the host, use the following syntax:

```
snapdrive storage connect -lun long_lun_name [lun_name ...]
  [-igroup ig_name [ig_name ...]]
```

Note

For details about using the options and arguments in this command line, see the Chapter 9, "SnapDrive for UNIX options, keywords, and arguments," on page 348.

Examples

Example 1: The following command line connects a single LUN (toaster_lun2) on the storage system (filer1) to the host.

```
# snapdrive storage connect -lun filer1:/vol/vol1/toaster_lun2
```

Example 2: This command line connects multiple LUNs from the same storage system to the host. Because toaster_lun3 and toaster_lun4 do not specify full pathnames, SnapDrive for UNIX looks for them on the same volume of storage system filer1 as toaster_lun2.

```
# snapdrive storage connect -lun filer1:/vol/vol1/toaster_lun2
toaster_lun3 toaster_lun4
```

Example 3: This command line connects LUNs from different storage systems to the host.

```
# snapdrive storage connect -lun filerA:/vol/vol1/lunA
filerB:/vol/vol1/lunB
```

Connecting a file system created directly on a LUN

To use the `snapdrive storage connect` command to connect a file system created directly on a LUN, use the following syntax:

```
snapdrive storage connect -fs file_spec -nolvm -lun long_lun_name
[-igroup ig_name [ig_name ...]] [-nopersist] [-mntopts options] [-
fstype type] [-vmtype type]
```

Note

For details about using the options and arguments in this command line, see Chapter 9, "SnapDrive for UNIX options, keywords, and arguments," on page 348.

Example: The following command connects the file system /acct/acctfs on the storage system toaster to a new host.

```
# snapdrive storage connect -fs /acct/acctfs -lun
toaster:/vol/vol1/acctlun1
-nolvm
```

Connecting LUNs with disk groups, host volumes, and file systems

To use the `snapdrive storage connect` command to connect LUNs that have disk groups, host volumes and file systems, use the following syntax:

```
snapdrive storage connect -fs file_spec -hostvol file_spec
-lun long_lun_name [lun_name ...] [-igroup ig_name [ig_name ...]]
[-nopersist] [-mntopts options] [-fstype type] [-vmtype type]
```

Note

For details about using the options and arguments in this command line, see Chapter 9, "SnapDrive for UNIX options, keywords, and arguments," on page 348.

Examples

Example 1: The following command connects the file system /acct/acctfs in disk group `dg1` to host volume `saleshostvol`. All storage entities are contained on the `acctlun1` LUN on the storage system `toaster`.

```
# snapdrive storage connect -fs /acct/acctfs -hostvol
dg1/saleshostvol/ -lun toaster:/vol/vol1/acctlun1
```

Example 2: The following example connects to the file system /mnt/vxfs1 create on Veritas stack:

```
# snapdrive storage connect -fs /mnt/vxfs1 -fstype vxfs -hostvol
vxvm1/vxfs1_SdHv -lun snoopy:/vol/vol1/lunVxvm1

        mapping lun(s) ... done
        discovering lun(s) ... done

        LUN snoopy:/vol/vol1/lunVxvm1 connected
                - device filename(s): /dev/vx/dmp/Disk_0
        Importing vxvm1
        Connected fs /mnt/vxfs1
```

Example 3: The following example connects to the file system /mnt/jfs1 create on LVM stack:

```
# snapdrive storage connect -fs /mnt/jfs1 -fstype jfs2 -hostvol
lvm1/jfs1_SdHv -lun snoopy:/vol/vol1/lunLvm1

        mapping lun(s) ... done
        discovering lun(s) ... done

        LUN snoopy:/vol/vol1/lunLvm1 connected
                - device filename(s): /dev/hdisk7
        Importing lvm1
```

```
Connected fs /mnt/jfs1
```

Connecting existing LUNs with shared resources

If a new node is added to the cluster configuration that uses a shared disk group or file system, use the following syntax:

```
# snapdrive storage connect -fs file_spec -lun long_lun_name
[lun_name...] [-devicetype shared] [-mntopts options]
```

Note

For details about using the options and arguments in these command lines, see Chapter 9, "SnapDrive for UNIX options, keywords, and arguments," on page 348.

Example: This example enables access to existing LUNs, disk groups, and file systems that are not in Snapshot copies:

```
# snapdrive storage connect -fs /mnt/sharedA -hostvol
sharedA/sharedA_SdHv -lun f270-197-109:/vol/vol2/lunSharedA -
devicetype shared

Execution started on cluster master: sfrac-57

        mapping lun(s) ... done
        discovering lun(s) ... done

        LUN f270-197-109:/vol/vol2/lunSharedA connected
                - device filename(s): /dev/vx/dmp/c5t0d7s2

Connecting cluster node: sfrac-58

        mapping lun(s) ... done
        discovering lun(s) ... done

        LUN f270-197-109:/vol/vol2/lunSharedA connected
                - device filename(s): /dev/vx/dmp/c3t0d7s2
        Importing sharedA
        Activating hostvol sharedA_SdHv
        Connected fs /mnt/sharedA
```

Disconnecting LUN mappings from the host

Using storage disconnect

The storage disconnect operation removes the LUNs, or the LUNs and storage entities that were mapped to the host using the `snapdrive storage create` or `snapdrive storage connect` command. Use the `snapdrive storage disconnect` command to disconnect:

- LUNs
- A file system created directly on a LUN
- Disk groups, host volumes and file systems created on LUNs

When SnapDrive for UNIX removes the LUN mappings, it exports the disk groups or file systems that the LUNs contain. This action, which marks the disk and file system as exported, is the only change that disconnecting the mappings has on the contents of the LUNs.

Methods for disconnecting storage

To make it easier to disconnect the storage, SnapDrive for UNIX provides several formats for the `snapdrive storage disconnect` command. This is because the disconnect operations fall into the following general categories:

- Specifying the LUNs that you want to disconnect from the host.

  For the steps required to do this, see "Disconnecting LUNs from the host" on page 212.
- Specifying a file system that is created directly on a LUN that you want to disconnect from the host.

  SnapDrive for UNIX disconnects both the file system and LUN. For the steps required to do this, see "Disconnecting a file system created on a LUN from the host" on page 213.
- Specifying a disk group, host volume or file system that resides on LUNs you want to disconnect from the host.

  SnapDrive for UNIX disconnects all the LUNs associated with that entity, and also removes mappings for the file system, host volume, and disk group that comprise the entity you disconnected. For the steps required to do this, see "Disconnecting LUNs and storage entities from the host" on page 213.
- Disabling a node from using a shared disk group or file system in a cluster environment.

  For the steps required to do this, see Chapter 6, "Disabling a node or a cluster from using shared resources," on page 215.

Guidelines for the snapdrive storage disconnect command

Follow these guidelines when using the `snapdrive storage disconnect` command:

- When you disconnect a file system, SnapDrive for UNIX always removes the mountpoint.

  Linux hosts allow you to attach multiple file systems to a single mountpoint. However, SnapDrive for UNIX requires a unique mountpoint for each file system. The `snapdrive storage disconnect` command fails if you use it to disconnect file systems that are attached to a single mountpoint.
- If you use the `-lun` option to specify the name of a LUN that is a member of either a host disk group or a file system, the `snapdrive host disconnect` command fails.
- If you use `-lun` option to specify the name of the LUN that is not discovered by multipathing software on the host, the `snapdrive storage disconnect` command fails.

Guidelines for using the disconnect command in a cluster environment:

- The `snapdrive storage disconnect` command can be executed from any node in the cluster.
- For the storage disconnect operation to be successful, either of the following should be true:
  - The storage entities should be shared across all the nodes in the cluster.
  - The LUNs should be mapped to all the nodes in the cluster.
- You can disconnect a storage entity from a specific node using the `-devicetype dedicated` option or by omitting the `-devicetype` option in the command altogether, because the default value is `dedicated`.
- The `snapdrive storage disconnect` command gives an error if a shared storage entity or LUN is disconnected with `dedicated` option, or if a dedicated storage entity or LUN is disconnected with `shared` option.
- SnapDrive for UNIX executes the `snapdrive storage disconnect` command on the master node. It destroys the storage entities, disconnects the LUNs on all the nonmaster nodes, and then disconnects the LUNs from the master node in the cluster. If any error occurs during this sequence, the storage disconnect operation fails.

Tips for using storage disconnect

When you use the `snapdrive storage disconnect` command on some operating systems, you lose information such as the host volume names, the file system mountpoint, the storage system volume names, and the names of the LUNs. Without this information, reconnecting the storage at a later point in time is difficult.

To avoid this potential problems, NetApp recommends that you first create a Snapshot copy of the storage using the `snapdrive snap create` command **before** you execute the `snapdrive storage disconnect` command.

That way, if you want to reconnect the storage later, you can use the following workaround.

Step	Action
1	Execute the command the following command: `snapdrive snap restore` *`filespec`* `-snapname` *`long_snap_name`* Include the full path to the Snapshot copy in this command.
2	Now you can remove the Snapshot copy, if you choose, by executing the `snapdrive snap delete` command.

Information required for snapdrive storage disconnect

The following table gives the information you need to supply when you use the `snapdrive storage disconnect` command.

Requirement	Argument
Based on the command you enter, you can remove mappings from any of the following: ◆ LUNs If you disconnect one or more LUNs, the first argument must use the long form of the LUN name, which specifies the storage system name, the volume name, and the name of the LUN within the volume. To specify additional LUNs, you can use the LUN name alone if the new LUN is on the same storage system and volume as the previous LUN. Otherwise, you can specify a new storage system name and volume name (or just a volume name) to replace the previous values. ◆ File systems on LUNs The *file_spec* given to `-fs` is the name of the file system mountpoint. SnapDrive for UNIX automatically locates and disconnects the LUN that is associated with the file system you specify. ◆ Disk or volume groups ◆ File systems on disk or volume groups ◆ Host or logical volumes The value you enter for the *file_spec* argument must identify the storage entity you are disconnecting.	
◆ A LUN (`-lun`)	*long_lun_name*
The first value you supply with the `-lun` option must include the storage system name, volume, and LUN name. To disconnect multiple LUNs on the same volume, you can use relative path names for the `-lun` option after you supply the complete information in the first path name. When SnapDrive for UNIX encounters a relative path name, it looks for the LUN on the same volume as the previous LUN. To disconnect additional LUNs that are not on the same volume, enter the full path name to each LUN.	
◆ Additional LUNs	*lun_name* (long or short form)
◆ Disk group (`-dg` *file_spec*) or volume group (`-vg` *file_spec*)	*name of the disk or volume group*
◆ File system (`-fs` *file_spec*)	*filesystem_name*
◆ Host volume (`-hostvol` *file_spec*) or logical volume (`-lvol` *file_spec*)	*name of the host or logical volume*

Requirement	Argument
If you want SnapDrive for UNIX to disconnect the storage you specify even if you include on the command line a host-side entity that has other entities (such as a disk group that has one or more host volumes), include the `-full` option on the command line. If you do not include this option, you must specify only empty host-side entities.	
`-full`	~
If you want to disable a node or a cluster from sharing a file system	
`-devicetype`	
◆ `-fstype`	*type*
◆ `-vmtype`	*type*
Optional: Specifies the type of file system and volume manager to be used for SnapDrive for UNIX operations.	

Disconnecting LUNs from the host

To use the `snapdrive storage disconnect` command to remove the mappings for the LUNs you specify, use the following syntax:

```
snapdrive storage disconnect -lun long_lun_name [lun_name...]
```

Note

For details about using the options and arguments in this command line, see "SnapDrive for UNIX options, keywords, and arguments" on page 348.

Examples

Example 1: The following command line disconnects a single LUN (toaster_lun2) on filer1.

```
# snapdrive storage disconnect -lun filer1:/vol/vol1/toaster_lun2
```

Example 2: This command line disconnects LUNs from different storage systems.

```
# snapdrive storage disconnect -lun filerA:/vol/vol1/lunA
filerB:/vol/vol1/lunB
```

Disconnecting a file system created on a LUN from the host

To use the `snapdrive storage disconnect` command to remove a file system created directly on a LUN, use the following syntax:

```
snapdrive storage disconnect -fs file_spec [-fstype type]
[-vmtype type]
```

Note

For details about using the options and arguments in this command line, see "SnapDrive for UNIX options, keywords, and arguments" on page 348.

Example: The following command line disconnects the file system /mnt/`acc1`.

```
# snapdrive storage disconnect -fs /mnt/acc1
```

Disconnecting LUNs and storage entities from the host

To use the `snapdrive storage disconnect` command to remove the mappings for the LUNs associated with storage entities, use the following syntax:

```
snapdrive storage disconnect
{ -dg | -fs | -hostvol } file_spec [file_spec ...]
[{ -dg | -fs | -hostvol } file_spec [file_spec ...] ...] [-full]
[-fstype type] [-vmtype type]
```

Example 1: The following command line disconnects the disk group dg1:

```
# snapdrive storage disconnect -dg dg1
```

Example 2: The following example disconnects the file system /mnt/vxfs1 create on Veritas stack:

```
# snapdrive storage disconnect -fs /mnt/vxfs1 -fstype vxfs -full

 disconnecting file system /mnt/vxfs1
         - fs /mnt/vxfs1 ...      - fs /mnt/vxfs1 ... disconnected
         - hostvol vxvm1/vxfs1_SdHv ... disconnected
         - dg vxvm1 ... disconnected
         - LUN snoopy:/vol/vol1/lunVxvm1 ... disconnected

0001-669 Warning:
      Please save information provided by this command.
      You will need it to re-connect disconnected filespecs.
```

Example 3: The following example disconnects file system /mnt/jfs1 create on LVM stack:

```
# snapdrive storage disconnect -fs /mnt/jfs1 -fstype jfs2 -full
```

```
 disconnecting file system /mnt/jfs1
         - fs /mnt/jfs1 ...      - fs /mnt/jfs1 ... disconnected
         - hostvol lvm1/jfs1_SdHv ... disconnected
         - dg lvm1 ... disconnected
         - LUN snoopy:/vol/vol1/lunLvm1 ... disconnected

0001-669 Warning:
      Please save information provided by this command.
      You will need it to re-connect disconnected filespecs.
```

Disabling a node or a cluster from using shared resources

You have to modify the /etc/VRTSvcs/conf/config/main.cf file to disable a node from using a shared resource. For more information about the main.cf file, refer *Veritas Cluster Server 4.1 Installation Guide for Solaris*.

To disable a node from using a shared resource, use the following syntax:

```
snapdrive storage disconnect -fs file_spec -lun long_lun_name
[lun_name...] [-devicetype shared]
```

Note

For details about using the options and arguments in this command lines, see Chapter 9, "SnapDrive for UNIX options, keywords, and arguments," on page 348.

Examples

Example 1: This example disables a node from sharing a file system:

```
# snapdrive storage disconnect -fs /mnt/fs

 disconnect file system /mnt/fs
         - fs /mnt/fs ... disconnected
         - hostvol fs_SdDg/fs_SdHv ... disconnected
         - dg fs_SdDg ... disconnected
         - LUN f270-197-109:/vol/vol2/fs_SdLun ... disconnected
```

Example 2: This example disables a cluster from sharing a file system:

```
# snapdrive storage disconnect -fs /mnt/shared -devicetype shared
Execution started on cluster master: sfrac-57

 disconnect file system /mnt/shared
         - fs /mnt/shared ... disconnected
         - hostvol shared/shared_SdHv ... disconnected
         - dg shared ... disconnected

Disconnecting cluster node: sfrac-58
         - LUN f270-197-109:/vol/vol1/lunShared ... disconnected
         - LUN f270-197-109:/vol/vol1/lunShared ... disconnected

0001-669 Warning:
      Please save information provided by this command.
      You will need it to re-connect disconnected filespecs.
```

Connecting only the host side of storage

How the host connect command works

The `snapdrive host connect` command performs the host side operations that connect a storage entity. Use the `snapdrive host connect` command to connect the following:

- LUNs
- A file system created directly on a LUN
- Disk groups, host volumes and file systems created on LUNs

When you enter the `snapdrive storage connect` command to connect LUNs to the host, SnapDrive for UNIX performs the necessary discovery and mapping. It does not modify LUN contents.

Connecting LUNs: When you use the `snapdrive host connect` command to connect LUNs, SnapDrive for UNIX assumes that the LUNs have been fully provisioned and mapped to the host. The command performs only the host operations (discovery) required to connect the LUN.

Connecting LUNs that contain storage entities: When you use the `snapdrive host connect` command to connect a LUN that has a file system, host volume, or disk group, SnapDrive for UNIX assumes that the LUN has been created on a storage system and mapped to an igroup. The command performs the host operations (discovery, file system mounting, and so on) required to connect the storage.

Note

The storage entity you connect must conform to specific configuration requirements. For additional information, see "Guidelines for the host connect command" section.

Guidelines for the host connect command

Follow these guidelines when connecting a Snapshot copy to the host:

- Storage that has LVM entities has special requirements. To use the `snapdrive host connect` command to connect LVM entities, you must create the storage so that each entity in the storage hierarchy has exactly one instance of the next entity. For example, you can use the `snapdrive host connect` command to connect a storage hierarchy that has one disk group (dg1) with one host volume (hostvol1) and one file system (fs1). However, you cannot use the `snapdrive host connect` command to connect a

hierarchy that has one disk group (dg1) with two host volumes (hostvol1 and hostvol2) and two file systems (fs1 and fs2).

- On Linux hosts, the `snapdrive host connect` command connects a file system created directly on a LUN only when the underlying LUN is partitioned.

Information required for snapdrive host connect

The following table gives the information you need to supply when you use the `snapdrive host connect` command.

Requirement	Argument
Specify the LUNs, a file system created directly on a LUN, or a LUN with a disk group, host volume and file system that you want to connect to the host. ◆ If you connect one or more LUNs, the first argument must use the long form of the LUN name, which specifies the storage system name, the volume name, and the name of the LUN within the volume. To specify additional LUNs, you can use the LUN name alone if the new LUN is on the same storage system and volume as the previous LUN. Otherwise, you can specify a new storage system name and volume name (or just a volume name) to replace the previous values. ◆ If you connect a a file system that is created directly on a LUN, you must include the long form of the LUN name, and also the `-nolvm` option. ◆ If you connect a LUN with a disk group, host volume, and file system, you must use the `-fs` and `-hostvol` options to specify the file system and host volume. The host volume must include the name of the disk group.	
◆ A LUN (`-lun`)	*long_lun_name*
◆ Additional LUNs	*lun_name* (long or short form)
A file system (`-fs` *file_spec*)	*filesystem_name*
To add a file system that is created on a LUN without activating the host LVM:	
◆ `-nolvm`	
To add a file system on a host volume:	
◆ Host volume (`-hostvol` *file_spec*)	*disk_group_name* and *host_volume_name*

Requirement	Argument
The `-fs` *file_spec* and `-hostvol` *file_spec* you supply identifies the LVM file system, disk group and host volumes that you want to connect to a new host. The storage hierarchy that you connect must contain a disk group, host volume and file system, and must conform to the requirements in "Guidelines for the storage connect command" on page 198. You must specify a value for the `-fs` and `-hostvol`. The `-hostvol` must include the name of the disk group.	
Optional: Use the -nopersist option to connect the storage to a new location without creating an entry in the host file system table (for example, `fstab` on Linux). By default the `storage connect` command creates persistent mounts. This means that when you create an LVM storage entity on a Solaris, AIX, HP-UX or Linux host, SnapDrive for UNIX automatically creates the storage, mounts the file system and then places an entry for the file system in the host file system table.	
◆ `-nopersist`	~
◆ `-fstype`	*type*
◆ `-vmtype`	*type*
Optional: Specifies the type of file system and volume manager to be used for SnapDrive for UNIX operations.	

Connecting to LUNs

To use the `snapdrive host connect` command to map LUNs to the host, use the following syntax:

```
snapdrive host connect -lun long_lun_name [lun_name ...]
```

Note

For details about using the options and arguments in this command line, see "SnapDrive for UNIX options, keywords, and arguments" on page 348.

Example: The following command line connects the LUNs lunA and lunB.

```
# snapdrive host connect -lun acctfiler:/vol/vol1/lunA lunB
```

Connecting a file system created directly on a LUN

To use the `snapdrive host connect` command to connect a LUN with a file system created directly on it, use the following syntax:

```
snapdrive host connect -fs file_spec -nolvm -lun long_lun_name [-
nopersist][-mntopts options][-fstype type] [-vmtype type]
```

Note

For details about using the options and arguments in this command line, see Chapter 9, "SnapDrive for UNIX options, keywords, and arguments," on page 348.

Example: The following command connects the file system `my_fs that resides on acctlun1` on the storage system `toaster`.

```
# snapdrive host connect -fs /my_fs -lun toaster:/vol/vol1/acctlun1
-nolvm
```

Connecting LUNs with disk groups, host volumes, and file systems

To use the `snapdrive host connect` command to connect LUNs that have disk groups, host volumes and file systems, use the following syntax:

```
snapdrive host connect -fs file_spec -hostvol file_spec
-lun long_lun_name [lun_name][-nopersist] [-mntopts options]
[-fstype type] [-vmtype type]
```

Note

For details about using the options and arguments in this command line, see Chapter 9, "SnapDrive for UNIX options, keywords, and arguments," on page 348.

Example: The following command connects the file system /acct/acctfs in diskgroup dg1 and host volume saleshostvol. All storage entities are contained on the acctlun1 on the storage system toaster.

```
# snapdrive host connect -fs /acct/acctfs -hostvol
dg1/saleshostvol/ -lun toaster:/vol/vol1/acctlun1
```

Disconnecting only the host side of storage

Disconnecting host entities from storage

The `snapdrive host disconnect` command disconnects the host side entities you specify without changing the state on the storage system where the entities reside. Use the `snapdrive host disconnect` command to disconnect the following:

- LUNs
- A file system created directly on a LUN
- Disk groups, host volumes and file systems created using the LVM

This command performs the host portion of the `snapdrive storage disconnect` actions. Because it does not unmap or delete the LUNs, you do not risk disconnecting other hosts from the storage entity if all the hosts share an igroup on the storage system. It only affects the host.

Methods for disconnecting host side storage

To make it easier to disconnect storage from the host, SnapDrive for UNIX provides several formats for the `snapdrive storage disconnect` command. This is because storage disconnect operations fall into the following general categories:

- Specifying the LUNS you want to disconnect from the host.

  For the steps required to do this, see "Disconnecting LUNs from the host" on page 225.
- Specifying the file system created directly on the host that you want to disconnect.

  For the steps required to do this, see "Disconnecting a file system created on a LUN from the host" on page 226.
- Specifying the LVM disk groups, file systems or host volumes that you want to disconnect.

  For the steps required to do this, see "Disconnecting LUNs and storage entities from the host" on page 227.

Guidelines for the host disconnect command

Remember these guidelines when you use the `snapdrive host disconnect` command:

- When you disconnect a file system, SnapDrive for UNIX always removes the mountpoint.

Linux hosts allow you to attach multiple file systems to a single mountpoint. However, SnapDrive for UNIX requires a unique mountpoint for each file system. The `snapdrive host disconnect` command fails if you use it to disconnect file systems that are attached to a single mountpoint.

- If you use the `-lun` option to specify the name of a LUN that is a member of either a host disk group or a file system, the `snapdrive host disconnect` command fails.
- If you use the `-lun` option to specify the name of the LUN that is not discovered by multipathing software on the host, the `snapdrive storage delete` command fails. For example, on Solaris hosts, the LUN has to be under DMP control. In other words, the LUN has to have a corresponding /dev/vx/dmp device.

Tips for using host disconnect

When you use the `snapdrive host disconnect` command on some operating systems, you lose information, such as the host volume names, the file system mountpoint, the storage system volume names, and the names of the LUNs. Without this information, reconnecting the storage at a later point in time is difficult.

To avoid this potential problem, NetApp strongly recommends that you first create a Snapshot copy of the storage using the `snapdrive snap create` command **before** you execute the `snapdrive host disconnect` command.

That way, if you want to reconnect the storage later, you can use the following workaround.

Step	Action
1	Execute the following command: `snapdrive snap restore` *`filespec`* `-snapname` *`long_snap_name`* Include the full path to the Snapshot copy in this command.
2	Now you can remove the Snapshot copy, if you choose, by executing the `snapdrive snap delete` command.

Information required for snapdrive host disconnect

The following table gives the information you need to supply when you use the `snapdrive host disconnect` command.

Requirement	Argument
Based on the command you enter, you can disconnect any of the following storage entities without changing the state of the storage system: ◆ LUNs If you disconnect one or more LUNs, the first argument must use the long form of the LUN name, which specifies the storage system name, the volume name, and the name of the LUN within the volume. To specify additional LUNs, you can use the LUN name alone if the new LUN is on the same storage system and volume as the previous LUN. Otherwise, you can specify a new storage system name and volume name (or just a volume name) to replace the previous values for file systems created directly on a LUN. ◆ A file system on a LUN ◆ Disk or volume groups ◆ File systems on disk or volume groups ◆ Host or logical volumes ◆ Disk groups, host volumes and file systems created using the LVM You can disconnect an LVM disk group, file system, or host volume. Enter the name of the entity as the value for the *file_spec* argument. The command disables the entity you specify and disconnects the LUNs associated with it. The value you enter for the *file_spec* argument must identify the storage entity you are disconnecting.	
◆ Disk group (`-dg` *file_spec*) or volume group (`-vg` *file_spec*)	*name of the disk or volume group*
◆ File system (`-fs` *file_spec*)	*filesystem_name*
The *file_spec* given to `-fs` is the name of the file system mountpoint. SnapDrive for UNIX automatically locates and disconnects any host volume, disk group, or LUN that is associated with the file system you specify.	
◆ Host volume (`-hostvol` *file_spec*) or logical volume (`-lvol` *file_spec*)	*name of the host or logical volume*
◆ LUN (`-lun` l*ong_lun_name*)	*long LUN name*

Requirement	Argument
The first value you supply with the `-lun` option must include the storage system name, volume, and LUN name. To delete multiple LUNs on the same volume, you can use relative path names for the `-lun` option after you supply the complete information in the previous path name. When SnapDrive for UNIX encounters a relative pathname, it looks for the LUN on the same volume as the previous LUN. To delete additional LUNs that are not on the same volume, enter the full path name to each LUN.	
If you want SnapDrive for UNIX to disconnect the storage you specify even if you include on the command line a storage entity that has other entities (such as a disk group that has one or more host volumes), include the `-full` option on the command line. If you do not include this option, the command fails. If you do not include this option, you must specify only empty host-side entities.	
`-full`	~
◆ `-fstype`	*type*
◆ `-vmtype`	*type*
Optional: Specifies the type of file system and volume manager to be used for SnapDrive for UNIX operations.	

Disconnecting LUNs from the host

To use the `snapdrive host disconnect` command to disconnect LUNs from the host, use the following syntax:

```
snapdrive host disconnect -lun long_lun_name [lun_name...]
```

Note

For details about using the options and arguments in this command line, see "SnapDrive for UNIX options, keywords, and arguments" on page 348.

Example 1: The following command line disconnects a single LUN (toaster_lun2) on the storage system filer1.

```
# snapdrive host disconnect -lun filer1:/vol/vol1/toaster_lun2
```

Example 2: This command line disconnects LUNs from different storage systems to the host.

```
# snapdrive host disconnect -lun filerA:/vol/vol1/lunA
filerB:/vol/vol1/lunB
```

Disconnecting a file system created on a LUN from the host

To use the `snapdrive host disconnect` command to remove a file system created directly on a LUN, use the following syntax:

```
snapdrive host disconnect -fs file_spec [-fstype type]
[-vmtype type]
```

Note

For details about using the options and arguments in this command line, see Chapter 9, "SnapDrive for UNIX options, keywords, and arguments," on page 348.

Example: The following command line disconnects a file system acc1.

```
# snapdrive host disconnect -fs /mnt/acc1
```

Disconnecting LUNs and storage entities from the host

To use the `snapdrive host disconnect` command to disconnect a LUN and storage entities from the host with an LVM entity, use the following syntax:

```
snapdrive host disconnect
{-dg | -fs | -hostvol} file_spec [file_spec ...]
[{ -dg | -fs | -hostvol } file_spec [file_spec ...] ...] [-full]
[-fstype type] [-vmtype type]
```

Note

For details about using the options and arguments in this command line, see "SnapDrive for UNIX options, keywords, and arguments" on page 348.

Example 1: The following command line disconnects the disk groups dg1, dg2, and dg3.

```
# snapdrive host disconnect -dg dg1 dg2 dg3
```

Example 2: This command line disconnects the logical volume dg1/mylvol1.

```
# snapdrive host disconnect -lvol dg1/mylvol1
```

Deleting storage from the host and storage system

Using storage delete command

The `snapdrive storage delete` command removes the storage entities on the host as well as all underlying host side entities and storage system LUNs backing them.

CAUTION

This command deletes data. Use caution in running it.

Methods for deleting storage

To make it easier to delete the storage, SnapDrive for UNIX provides several formats for the `snapdrive storage delete` command. This is because the delete operations fall into the following general categories:

- Specifying the LUNs that you want to delete from the host.

  For the steps required to do this, see "Deleting specific LUNs" on page 232.
- Specifying the host entity for which you want to delete the storage, including all associated objects and all storage system LUNs.

  For the steps required to do this, see "Deleting specific host entities" on page 233.
- Specifying the shared host entity for which you want to delete the storage. For the steps required to do this, see Chapter 6, "Deleting shared host entities," on page 234.

Guidelines for using the storage delete command

The `snapdrive storage delete` command has the following restrictions in SnapDrive for UNIX:

- When you delete a file system, SnapDrive for UNIX always removes the file system's mountpoint.

  Linux hosts allow you to attach multiple file systems to a single mountpoint. However, SnapDrive for UNIX requires a unique mountpoint for each file system. The `snapdrive storage delete` command fails if you use it to delete file systems that are attached to a single mountpoint.
- If you use the `-lun` option to specify the name of a LUN that is a member of either a host disk group or a file system, the `snapdrive storage delete` command fails.
- If you use `-lun` option to specify the name of the LUN that is not discovered by multipathing software on the host, the `snapdrive storage delete`

command fails. For example, on Solaris hosts, the LUN has to be under DMP control. In other words, the LUN has to have a corresponding /dev/vx/dmp device.

Guidelines for storage deletion in a cluster environment:

- If you initiate the `snapdrive storage delete` command with `-devicetype shared` option from any nonmaster node in the cluster, the command is sent to the master node and executed. For this to happen, you have to ensure that the `rsh` or `ssh` access-without-password-prompt is allowed on all the cluster nodes.
- The `snapdrive storage delete` command can be executed from any node in the cluster.
- For the storage delete operation to be successful, neither should be false:
  - The storage entities have to be shared.
  - The LUNs should be mapped to all the nodes in the cluster.
- You can delete a storage entity on a specific node either by using the `-devicetype dedicated` option or by omitting the `-devicetype` option in the command line syntax, altogether because the default value is `dedicated`.
- The `snapdrive storage delete` command gives an error if a shared storage entity or LUN is deleted with `-devicetype dedicated` option, or if a dedicated storage entity or LUN is deleted with shared option.
- The storage delete operations fails, if one of the following happens:
  - If any error occurs during the process of deleting a storage entity

    SnapDrive for UNIX deletes the storage entities, disconnect the LUNs from all the nonmaster nodes and then disconnect and delete the LUNs from the master node in the cluster.
  - If a node in the cluster shuts down and reboots before the `snapdrive storage delete` command is invoked

    This happens because the LUNs are still be mapped to the non-existing node. To avoid this, use the `-force` option. For more information on this option, see Chapter 4, “Determining options and their default values,” on page 86.

Information required for snapdrive storage delete

The following is a summary of the information you need to supply when you use the `snapdrive storage delete` command.

Requirement	Argument
Based on the command you enter, you can delete any of the following types of storage: ◆ LUNs Specify one or more LUNs that you want to delete from the storage system. If you delete one or more LUNs, the first argument must use the long form of the LUN name, which specifies the storage system name, the volume name, and the name of the LUN within the volume. To specify additional LUNs, you can use the LUN name alone if the new LUN is on the same storage system and volume as the previous LUN. Otherwise, you can specify a new storage system name and volume name (or just a volume name) to replace the previous values. ◆ A file system created directly on a LUNs ◆ Disk or volume groups ◆ File systems on disk or volume groups ◆ Host or logical volumes The value you enter for the *file_spec* argument must identify the storage entity you are deleting.	
◆ A LUN (`-lun`)	*long_lun_name*
◆ Additional LUNs	*lun_name* (long or short form)
◆ Disk group (`-dg` *file_spec*) or volume group (`-vg` *file_spec*)	*name of the disk group or volume group*
◆ File system (`-fs` *file_spec*)	*filesystem_name*
◆ Host volume (`-hostvol` *file_spec*) or logical volume (`-lvol` *file_spec*)	*name of the host volume or logical volume* **Note** You must supply both the requested volume and the disk group containing it; for example, `-hostvol dg3/acct_volume`.
If you want SnapDrive for UNIX to delete the storage you specify even if you include on the command line a host-side entity that has other entities (such as a disk group that has one or more host volumes), include the `-full` option on the command line. If you do not include this option, you must specify only empty host-side entities.	

Requirement	Argument
`-full`	~
To specify the shared host entity for which you want to delete the storage.	
`-devicetype`	
◆ `-fstype`	*type*
◆ `-vmtype`	*type*
Optional: Specifies the type of file system and volume manager to be used for SnapDrive for UNIX operations.	

Deleting specific LUNs

To delete specific LUNs, use the following syntax:

```
snapdrive storage delete -lun long_lun_name [lun_name...]
```

Note

For details about using the options and arguments in this command line, see "SnapDrive for UNIX options, keywords, and arguments" on page 348.

Example: The following command line tells SnapDrive for UNIX to delete LUNs from storage systems.

```
# snapdrive storage delete -lun filer1:/vol/vol1/lun1 lun2
```

Deleting specific host entities

To delete a storage entity such as a disk group and its associated LUNs, or a file system that resides directly on a LUN, use the following syntax:

```
snapdrive storage delete
{-dg | -fs | -hostvol } file_spec [file_spec ...]
[{-dg | -fs | -hostvol } file_spec [file_spec ...]] [-full]
```

Note

For details about using the options and arguments in this command line, see "SnapDrive for UNIX options, keywords, and arguments" on page 348.

Example: The following command lines tell SnapDrive for UNIX to delete the objects from specific host entities, such as disk groups, logical volumes and file systems.

```
# snapdrive storage delete -full -dg dg1 dg2 dg3
# snapdrive storage delete -lvol mydg/vol3 mydg/vol5
# snapdrive storage delete -fs /mnt/acct2
```

Deleting shared host entities

To delete a shared storage entity such as a disk group, file system, and its associated LUNs, use the following syntax:

```
snapdrive storage delete {-vg | -dg | -fs} file_spec [file_spec...]
[-devicetype shared]
```

Note

For details about using the options and arguments in these command line, see Chapter 9, "SnapDrive for UNIX options, keywords, and arguments," on page 348.

Example: This example deletes the shared host entity and its associated LUNs. The operation is executed from the non-cluster-master node, but the command is shipped to the master node for execution.

```
#snapdrive storage delete -dg testdg -devicetype shared -full
Execution started on cluster master: sfrac-57

deleting disk group testdg
- dg testdg ... deleted

Disconnecting cluster node: sfrac-58
- LUN f270-197-109:/vol/vol2/testdg_SdLun ... disconnected
- LUN f270-197-109:/vol/vol2/testdg_SdLun ... deleted
```

Creating and Using Snapshot Copies 7

About this chapter

This chapter provides details about creating and using SnapDrive for UNIX Snapshot copies.

Topics in this chapter

This chapter discusses the following topics:

- "Overview of Snapshot operations" on page 236
- "Creating Snapshot copies" on page 238
- "Displaying information about Snapshot copies" on page 249
- "Renaming a Snapshot copy" on page 257
- "Restoring a Snapshot copy" on page 259
- "Connecting to a Snapshot copy" on page 272
- "Disconnecting a Snapshot copy" on page 288
- "Deleting a Snapshot Copy" on page 294

Overview of Snapshot operations

SnapDrive for UNIX lets you use Data ONTAP Snapshot technology to make an image (Snapshot copy) of host data that is stored on a NetApp storage system. This Snapshot copy provides you with a copy of that data, which you can restore later. The data in the Snapshot copy can exist on one storage system or span multiple storage systems and their volumes. These storage systems can be in cluster-wide shared or node-local file systems or disk groups or LUNs in a cluster environment.

On a nonclustered UNIX host with SnapDrive for UNIX installed, you can create a Snapshot copy of one or more volume groups on a storage system.The Snapshot copy can contain file systems, logical volumes, disk groups, LUNs, and NFS directory trees. After you create a Snapshot copy, you can rename it, restore it, or delete it. You can also connect it to a different location on the same host or to a different host. After you connect it, you can view and modify the content of the Snapshot copy, or you can disconnect the Snapshot copy. In addition, SnapDrive for UNIX lets you display information about Snapshot copies that you created.

On a clustered UNIX host with SnapDrive for UNIX installed, you can Conduct Snapshot operations on a cluster-wide shared storage system that includes disk groups and file systems. The Snapshot operations include create, rename, restore, connect, disconnect, display, and delete.

Considerations when working with Snapshot copies

When working with Snapshot operations, consider the following:

- SnapDrive for UNIX works only with Snapshot copies that it creates. It cannot restore Snapshot copies that it did not create.
- When you create a Snapshot copy on a destination storage system, the Snapshot copy is automatically replicated, from the source storage system on which it is created to the destination storage system. SnapDrive for UNIX allows you to restore the Snapshot copy on the destination storage system as well. For additional information, see “Restoring Snapshot copies on destination storage system” on page 260.
- Connecting to the originating host occurs when you use the `snapdrive snap connect` command to connect to a Snapshot copy at a new location on the same host where it was last connected (or is still connected).
- On Linux hosts, SnapDrive 3.0 for UNIX supports Snapshot connect operation on the originating host, unless the LUN or a LUN with a file

system is part of the Linux LVM1 volume manager. For additional information, see “Connecting to a Snapshot copy” on page 272.

- Snapshot support for storage entities spanning multiple storage system volumes or multiple storage systems is limited on configurations that do not permit a freeze operation in the software stack. For additional information, see “Crash-consistent Snapshot copies” on page 238.
- When you export the volume through the NFS protocol, set the Anonymous User ID option to “0” for the SnapDrive for UNIX commands to work.

Creating Snapshot copies

About creating Snapshot copies

You can use the `snapdrive snap create` command to create Snapshot copies, which are point-in-time, read-only images of data on storage system volumes. The Snapshot create operation ensures that you have backed up your LUNs or NFS files and directory trees. You can use the Snapshot copy you create to restore your data if you encounter corruption or other problems.

Crash-consistent Snapshot copies

When you create a Snapshot copy of a storage entity, like a file system or disk group, SnapDrive for UNIX creates a Snapshot copy that contains the image of all the storage system volumes that comprise the entity you specified using a *file_spec* argument. The *file_spec* argument specifies the storage entity, like the file system, LUN, or NFS directory tree, that SnapDrive for UNIX uses to create the Snapshot copy.

SnapDrive for UNIX makes consistent storage components that comprise the entity you requested in the Snapshot copy. This means that LUNs or directories being used outside those specified by the `snapdrive snap create` command's *file_spec* argument may not have consistent images in the Snapshot copy. SnapDrive for UNIX enables you to restore only the entities specified by the *file_spec* argument that are made consistent in the Snapshot copy.

Snapshot copies of entities contained on a single storage system volume are always crash-consistent. SnapDrive for UNIX takes special steps to ensure that Snapshot copies that span multiple storage systems or storage system volumes are also crash-consistent. The method that SnapDrive for UNIX uses to ensure crash consistency depends on the Data ONTAP version where the storage entities in your Snapshot copy reside.

Crash consistency before Data ONTAP 7.2: When you create a Snapshot copy that spans multiple storage system volumes on storage systems prior to 7.2, SnapDrive for UNIX ensures consistency by freezing I/O to the requested LUNs. If a freeze is not provided by the host, as for example on a Linux host, SnapDrive for UNIX makes a best effort to create a consistent Snapshot copy by taking the Snapshot copy without freezing the target storage, and then checking for read-write I/Os that occurred to the storage entities when the Snapshot copy was taken. If SnapDrive for UNIX can create a crash-consistent Snapshot copy, the `snap create` command succeeds. If it cannot, SnapDrive for UNIX discards the Snapshot copy and informs the user of the failure. SnapDrive for UNIX will never create a Snapshot copy unless the data is crash-consistent.

The following table shows the host systems and Snapshot copy entities where SnapDrive for UNIX can guarantee a crash-consistent Snapshot copy. You can sometimes create crash-consistent Snapshot copies in configurations that are not guaranteed, but this requires additional steps and may also require multiple attempts, especially if the target storage is under load. You must perform whatever steps are necessary to quiesce the application before taking a Snapshot copy on a configuration where Snapshot copies are not guaranteed. Note that database hot backup facilities depend on the methods used by the Database Management System (DBMS), and do not always quiesce I/O to database files.

Host	Snapshot entities that span multiple volumes				
	LVM file systems	**LVM host volume or disk group**	**File system on LUN (two or more)**	**LUN (two or more)**	**NFS file or directory tree (two or more)**
Solaris	Guaranteed	Best Effort	Guaranteed	Best Effort	Best Effort
HP-UX	Guaranteed	Best Effort	Guaranteed	Best Effort	Best Effort
AIX	Guaranteed	Guaranteed	Best Effort	Best Effort	Best Effort
Linux	Best Effort	Best Effort	Best Effort	Best Effort	Best Effort

Crash consistency with Data ONTAP 7.2 and later: Data ONTAP versions 7.2 and greater provides support for consistency groups and storage system fencing. SnapDrive for UNIX uses these features to ensure that all Snapshot copies that span multiple volumes are crash consistent.

To create a crash consistent Snapshot copy across multiple volumes, SnapDrive for UNIX

- fences (freezes) I/O to every volume that contains a storage entity.
- takes a Snapshot copy of each volume.

The time it takes to fence the volume and create the Snapshot copy is limited, and is controlled by Data ONTAP.

The `snapcreate-cg-timeout` parameter in the snapdrive.conf file specifies the amount of time, within Data ONTAP limitations, that you wish to allow for storage system fencing. You can specify an interval that is urgent, medium, or relaxed. For information about the `snapcreate-cg-timeout` parameter, see "Determining options and their default values" on page 86. If the storage system requires more time than allowed to complete the fencing operation, SnapDrive

creates the Snapshot copy using the consistency methodology for previous Data ONTAP 7.2 versions. You can also specify this methodology by using the `-nofilerfence` option when you create the Snapshot copy.

If you request a Snapshot copy for a storage entity that spans storage systems with both Data ONTAP 7.2 and previous Data ONTAP versions, SnapDrive for UNIX also creates the Snapshot copy using the consistency method for Data ONTAP versions before 7.2. For additional information, see "Crash consistency before Data ONTAP 7.2" on page 238.

Application-consistent Snapshot copies

To ensure that a Snapshot copy is application-consistent, you may need to stop or do whatever steps are required to quiesce the application before taking the Snapshot copy. Note that database hot backup facilities depend on the methods used by the DBMS, and do not always quiesce I/O to database files.

If the application has not completed its transactions and written data to the storage system, the resulting Snapshot copy might not be application-consistent.

Note

If your application can recover from a crash-consistent Snapshot copy, you do not need to stop it. Consult the documentation for your application. For more information on taking application-consistent Snapshot copies, see Chapter 1, "Considerations when using SnapDrive for UNIX," on page 7.

You should take a new Snapshot copy whenever you add or remove a host volume, LUN, or NFS directory tree, or resize host volumes or file systems. This ensures that you have a consistent copy of the newly configured disk group that you can use if you need to restore the disk group.

Snapshot copies that span storage systems or volumes

SnapDrive for UNIX allows you to take Snapshot copies that span multiple storage system volumes or multiple storage systems. These volumes can reside on the same storage system or different storage systems. Although the `snapdrive snap create` command creates a Snapshot copy of all the volumes that comprise the entity you request, SnapDrive for UNIX will restore only the entities that you specify in the `snapdrive snap create` command.

When you use the `snapdrive snap create` command to make a Snapshot copy that spans multiple volumes, you do not need to name the volumes on the command line. SnapDrive for UNIX gets this information from the *file_spec* argument that you specify.

- If the *file_spec* you enter requests a disk group, or a file system or host volume that resides on a disk group, SnapDrive for UNIX automatically creates a Snapshot copy that includes all the storage system volumes for the disk group, volume, or file system you specified.
- If the *file_spec* you enter requests a LUN, SnapDrive for UNIX takes a Snapshot copy of the storage system volume that contains the LUN.
- If the *file_spec* you enter requests a file system that resides directly on a LUN, SnapDrive for UNIX takes a Snapshot copy of the storage system volume that contains the LUN and file system that you specified.
- If the *file_spec* you enter requests an NFS directory, SnapDrive for UNIX creates a Snapshot copy of the volume that contains the NFS directory tree.

In addition to using a *file_spec* argument that is built on entities from multiple storage systems and storage system volumes, you can also use a combination of *file_spec* arguments where each value is based on single storage system or storage system volume. For example, suppose you have a setup where the disk group dg1 spans the storage systems storage system2 and storage system3, dg2 is on storage system2, and dg3 is on storage system3. In this case, any of the following command lines would be correct:

```
snapdrive snap create -dg dg1 -snapname snapdg1
snapdrive snap create -dg dg2 dg3 -snapname snapdg23
snapdrive snap create -dg dg1 dg2 dg3 -snapname snapdg123
```

Something to keep in mind when creating Snapshot copies that span storage systems and volumes is that SnapDrive for UNIX creates the Snapshot copy on each storage systems volume using a short name. It does not include the full path name in the name, even if the volumes are on different storage system. This means that if you later rename the Snapshot copy, you must go to each storage system and volume and rename it there as well.

Creating Snapshot copies of unrelated entities

Unless you specify otherwise, SnapDrive for UNIX assumes that all entities that you specify on a given `snapdrive snap create` command line are related; in other words the validity of updates to one entity may depend on updates to the other entities specified. When storage entities have dependent writes in this way, SnapDrive for UNIX takes steps to create a Snapshot copy that is crash-consistent for all storage entities as a group.

The following example shows how SnapDrive for UNIX creates a Snapshot copy of storage entities that may have dependent writes. In the example below, the `snapdrive snap create` command specifies a file system on a LUN and also a disk group. The disk group consists of LUNs residing on a single storage system

volume. The file system on a LUN resides on a different storage system and storage system volume. As a group, the file system and the disk group span multiple storage system volumes; individually they do not.

The following command specifies a Snapshot copy that contains both the file system /mnt/fs1 and the disk group dg1:

```
snapdrive snap create -fs /mnt/fs1 -dg dg1 -snapname fs1_dg1
```

Because these storage entities can have dependent writes, SnapDrive for UNIX attempts to create a crash-consistent Snapshot copy, and treats the file system /mnt/fs1 and the disk group dg1 as a group. This means SnapDrive for UNIX is required to freeze I/O operations to the storage system volumes before creating the Snapshot copy.

Creating crash-consistent Snapshot copies for multiple storage entities across volumes takes extra time, and is not always possible if SnapDrive for UNIX cannot freeze I/O operations. Because this is so, SnapDrive for UNIX allows you to create Snapshot copies of unrelated storage entities. Unrelated storage entities are entities that you can specify that have no dependent writes when the Snapshot copy is taken. Because the entities have no dependent writes, SnapDrive for UNIX does not take steps to make the entities consistent as a group. Instead, SnapDrive for UNIX creates a Snapshot copy in which each of the individual storage entities is crash-consistent.

The following command specifies a Snapshot copy of the file system on a LUN and the disk group described previously. Because the `-unrelated` option is specified, SnapDrive for UNIX creates a Snapshot copy in which the file system /mnt/fs1 and the disk group dg1 are crash-consistent as individual storage entities, but are not treated as a group. The following command does not require SnapDrive for UNIX to freeze I/O operations on the storage system volumes:

```
snapdrive snap create -fs /mnt/fs1 -dg dg1 -unrelated -snapname
fs1_dg1
```

For additional information about how SnapDrive for UNIX ensures crash consistency, see “Crash-consistent Snapshot copies” on page 238.

Guidelines for Snapshot copy creation

Follow these guidelines when you enter commands that create Snapshot copies:

- You can keep a maximum of 255 Snapshot copies per storage system volume. This limit is set by the storage system. The total number can vary depending on whether other tools use these Snapshot copies.

When the number of Snapshot copies has reached the maximum limit, the Snapshot create operation fails. You must delete some of the old Snapshot copies before you can use SnapDrive for UNIX to take any more.

- SnapDrive for UNIX does not support Snapshot copies that it does not create. For example, it does not support Snapshot copies that are created from the storage system console, because such a practice can lead to inconsistencies within the file system.
- You cannot use SnapDrive for UNIX to create Snapshot copies of the following:
  - Root disk groups

    The Snapshot create operation fails when you try to take a Snapshot copy of a root disk group for an LVM.
  - Boot device or swap device

    SnapDrive for UNIX does not take a Snapshot copy of a system boot device or a system swap device.
- When a Snapshot copy spans multiple storage systems or storage system volumes, SnapDrive for UNIX requires a freeze operation to guarantee crash-consistency. For information about creating Snapshot copies on configurations for which a freeze operation is not provided, see "Crash-consistent Snapshot copies" on page 238.

Guidelines for Snapshot copy creation in a cluster environment:

- SnapDrive for UNIX can create Snapshot copies of disk groups and file systems that are shared with a host cluster partner in the Veritas SFRAC 4.1 environment. For instructions on creating a shared storage entity, see Chapter 7, "Creating a Snapshot copy," on page 246.
- The Snapshot create operation can be invoked from any node in the cluster.
- The multiple file systems and disk groups that are specified in this operation should have the same scope: that is, either all should be shared or all should be dedicated.
- An NFS file system in cluster-wide shared mode is not supported, but an NFS file system in dedicated mode in clustered nodes is supported.
- File systems are not supported on raw LUNs.

Information required for snapdrive snap create

The following table gives the information you need to supply when you use the `snapdrive snap create` command.

Requirement	Argument
Determine the type of storage entity you want to capture in the Snapshot copy. You can specify NFS entities, LUNs, file systems created directly on LUNs, and LVM entities on a single command line. Supply that entity's name with the appropriate argument. This is the value for the *file_spec* argument. ◆ If you specify a disk group that has a host volume or file specification, the argument translates into a set of disk groups on the storage system. SnapDrive for UNIX creates the entire disk group containing the entity, even if the entity is a host volume or file system. ◆ If you specify a file specification that is an NFS mountpoint, the argument translates to the directory tree on the storage system volume. ◆ If you specify a LUN, or a LUN that has a file system, the argument translates to the LUN, or to the LUN that has the file system. For information about how SnapDrive for UNIX freezes data for the host and storage system you specify, see "Guidelines for Snapshot copy creation" on page 242.	
◆ LUN (`-lun` *file_spec*)	*name of the LUN* *You must include the name of the storage system, volume, and LUN*
◆ Disk group (`-dg` *file_spec*) or volume group (`-vg` *file_spec*)	*name of the disk or volume group*
◆ File system (`-fs` *file_spec*)	*filesystem_name*
◆ Host volume (`-hostvol` *file_spec*) or logical volume (`-lvol` *file_spec*)	*name of the host or logical volume* **Note** You must supply both the requested volume and the disk group containing it; for example, `-hostvol dg3/acct_volume`.
◆ Snapshot copy name (`-snapname` *snap_name*)	*Snapshot copy_name*
Specify the name for the Snapshot copy. This can be either the long version of the name that includes the storage system and volume with the Snapshot copy name or the short version that is just the Snapshot copy name.	

Requirement	Argument
◆ `-unrelated`	~
Optional: Decide if you want to create a Snapshot copy of storage entities that have no dependent writes when the Snapshot copy is taken. Because the entities have no dependent writes, SnapDrive for UNIX creates a crash-consistent Snapshot copy of the individual storage entities, but does not take steps to make the entities consistent with each other.	
◆ `-force`	~
◆ `-noprompt`	~
Optional: Decide if you want to overwrite an existing Snapshot copy. Without this option, this operation halts if you supply the name of an existing Snapshot copy. When you supply this option and specify the name of an existing Snapshot copy, the command prompts you to confirm that you want to overwrite the Snapshot copy. To prevent SnapDrive for UNIX from displaying the prompt, include the `-noprompt` option also. (You must always include the `-force` option if you want to use the `-noprompt` option.)	
◆ `-devicetype`	~
Optional: Specify the type of device to be used for SnapDrive for UNIX operations. This can be either "shared" that specifies the scope of LUN, disk group, and file system as cluster-wide or "dedicated" that specifies the scope of LUN, disk group, and file system as local. If you specify the `-devicetype dedicated` option, all the options of `snapdrive snap create` command currently supported in SnapDrive 2.1 for UNIX function as they always have. If you initiate the `snapdrive snap create` command with the `-devicetype shared` option from any nonmaster node in the cluster, the command is shipped to the master node and executed. For this to happen, you must ensure that the `rsh` or `ssh` access-without-password-prompt for the root user should be configured for all nodes in the cluster.	
◆ `-fstype`	*type*
◆ `-vmtype`	*type*
Optional: Specify the type of file system and volume manager to be used for SnapDrive for UNIX operations.	

Creating a Snapshot copy

To create a Snapshot copy, use the following syntax:

```
snapdrive snap create
{-lun | -dg | -fs | -hostvol } file_spec  [file_spec ...]
[ {-lun |-dg | -fs | -hostvol } file_spec [file_spec...]]
-snapname snap_name [ -force [-noprompt]][-unrelated]
[-nofilerfence][-fstype type] [-vmtype type]
```

Note

Before you execute this syntax, you must understand the options, keywords, and arguments mentioned in this command. For details, see Chapter 9, "SnapDrive for UNIX options, keywords, and arguments," on page 348.

Result: The *file_spec* arguments represent a set of storage entities on one or more storage systems. The Snapshot create operation takes a Snapshot copy of the storage system volume containing those entities and gives it the name specified in the *snap_name* argument.

Examples

Example 1: This example creates a multivolume Snapshot copy for a Linux host. The Snapshot copy contains the disk group vgmultivol, which include the host volumes lvol1 and lvol2:

```
# snapdrive snap create -vg vgmultivol -snapname snapmultivol
Successfully created snapshot snapmultivol on 2 filer volumes:
        toaster:/vol/vol1
        toaster:/vol/vol2
snapshot snapmultivol contains:
        disk group vgmultivol containing host volumes
                lvol1
                lvol2
```

Example 2: These examples create Snapshot copies of file systems that are created directly on LUNs.

The following creates a Snapshot copy of the file system mounted at /mnt/fs1. The Snapshot copy contains the LUN on which the file system is created:

```
# snapdrive snap create -fs /mnt/fs1 -snapname snapfs1
```

The next example creates a Snapshot copy of multiple file systems. When the file system spans storage systems or storage system volumes, SnapDrive for UNIX automatically locates the storage system volumes that contain the LUNs with the file systems:

```
# snapdrive snap create -fs /mnt/fs1 /mnt/fs2 -snapname snapfs12
```

Example 3: This example creates a Snapshot copy of LUNs that are mapped directly to the host. The `-lun` specification must include the storage system, volume and LUN name:

```
# snapdrive snap create -lun ham:/vol/vol1/luna lunb
-snapname twoluns
```

Example 4: This example creates a Snapshot copy of LUNs that reside on multiple storage systems:

```
# snapdrive snap create -lun filer1:/vol/vol1/luna
filer2:/vol/vol1/lunb
-snapname lunsab
```

Example 5: This example creates a Snapshot copy of storage system entities that do not have dependent writes during Snapshot copy creation. SnapDrive for UNIX creates a Snapshot copy in which the file system /mnt/fs1 and the disk group dg1 are crash-consistent as individual storage entities, but are not treated as a group:

```
# snapdrive snap create -fs /mnt/fs1 -dg dg1 -unrelated
-snapname fs1_dg1
```

Example 6: This example creates a Snapshot copy of a shared file system on storage system f270-197-109:/vol/vol2:

```
# snapdrive snap create -fs /mnt/sfortesting -snapname
testsfarcsnap

Successfully created snapshot testsfarcsnap on f270-197-
109:/vol/vol2

snapshot testsfarcsnap contains:
disk group sfortesting_SdDg containing host volumes
sfortesting_SdHv (filesystem: /mnt/sfortesting)
```

Example 7: This example creates a Snapshot copy using the Veritas stack:

```
# snapdrive snap create -fs /mnt/vxfs1 -fstype vxfs -snapname
snapVxvm
Successfully created snapshot snapVxvm on snoopy:/vol/vol1

snapshot snapVxvm contains:
        disk group vxvm1 containing host volumes
                vxfs1_SdHv (filesystem: /mnt/vxfs1)
```

Example 8: This example creates a Snapshot copy using the JFS file system:

```
# snapdrive snap create -fs /mnt/jfs1 -fstype jfs -snapname snapLvm
Successfully created snapshot snapLvm on snoopy:/vol/vol1

        snapshot snapLvm contains:
        disk group lvm1 containing host volumes

                jfs1_SdHv (filesystem: /mnt/jfs1)
```

Example 9: These examples use a host called DBserver. The disk group dg1 has host volumes myvol1 and myvol2. The host volume dg1/myvol2 has a file system mounted on /fs2. The disk group dg1 has three LUNs in it: toaster:/vol/vol1/lun0, toaster:/vol/vol1/lun1, and toaster: /vol/vol1/lun2. The following command lines are all valid, and back up the same data. These command lines fall into three general categories. The first command line uses a file system; the next to refer to disk/volume groups, and the last two refer to host volumes/logical volumes. All the command lines create a Snapshot copy called toaster:/vol/vol1:snap1.

```
# snapdrive snap create -fs /fs2 -snapname snap1

# snapdrive snap create -dg dg1 -snapname snap1

# snapdrive snap create -vg dg1 -snapname snap1

# snapdrive snap create -hostvol dg1/myvol1 -snapname snap1

# snapdrive snap create -lvol dg1/myvol2 -fs /fs2 -snapname snap1
```

Displaying information about Snapshot copies

Command to use to display Snapshot copy information

You can use the `snapdrive snap show` (or `list`) command to display information about each Snapshot copy taken by SnapDrive for UNIX. You can use this command to display information on the following:

- Storage systems
- Volumes on storage systems
- Storage entities such as NFS files and directory trees, volume groups, disk groups, file systems, logical volumes, and host volumes
- Snapshot copies

Note

The `show` and `list` forms of this command are synonymous.

For SnapDrive 2.0 for UNIX and later, you must use the long form of the Snapshot copy name when you display information about Snapshot copies.

Guidelines for displaying Snapshot copies

Follow these guidelines when displaying Snapshot copies:

- You can use the wildcard (*) character in Snapshot copy names. The Snapshot show operation lets you use the wildcard character to show all Snapshot copy names that match a certain pattern or all Snapshot copy names on a particular volume. The following rules apply to using wildcard in Snapshot copy names:
  - You can use a wildcard at the end of the name only. You cannot use the wildcard at the beginning or the middle of a Snapshot copy name.
  - You cannot use the wildcard in the storage system or storage system volume fields of a Snapshot copy name.
- You can also use this command to list all of the Snapshot copies on specific objects, including storage systems and their volumes, disk groups, host volume groups, file systems, host volumes, and logical volumes.
- If you enter a `snapdrive snap show` command and SnapDrive for UNIX does not locate any Snapshot copies, it displays the message “no matching Snapshot copies.” If you specify arguments on the command line, and some portions of them do not exist, SnapDrive for UNIX returns a partial listing of those for which Snapshot copies are found. It also lists the arguments that were invalid.

- If the `snapdrive snap create` command is abruptly aborted, an incomplete .stoc.xml file is stored in the volume on the storage system. Due to this, all scheduled Snapshot copies made by the storage system will have a copy of the incomplete .stoc.xml file. For the `snapdrive snap list` command to work successfully, complete the following steps:

Step	Action
1	Delete the incomplete .stoc.xml file in the volume.
2	Delete the scheduled Snapshot copies made by the storage system containing the incomplete .stoc.xml file.

Information required for snapdrive snap show or list

The following table gives the information you need to supply when you use the `snapdrive snap show | list` command.

Note

You can use the same arguments regardless of whether you enter `snapdrive snap show` or `snapdrive snap list` as the command. These commands are synonyms. For details about using all the options, keywords, and arguments available with this command, see Chapter 9, "SnapDrive for UNIX options, keywords, and arguments," on page 348.

Requirement	Argument
Based on the command you enter, you can display information about any of the following: ◆ Storage systems ◆ Storage system volumes ◆ Disk or volume groups ◆ File systems ◆ Host or logical volumes ◆ Snapshot copies The value you enter for the *file_spec* argument must identify the storage entity about which you want to display information. The command assumes the entities are on the current host.	
◆ Storage system (`-filer`)	*filername*
◆ A volume on the storage system (`-filervol`)	*filervol*

Requirement	Argument
◆ Disk group (`-dg` *file_spec*) or volume group (`-vg` *file_spec*)	*name of the disk or volume group*
◆ File system (`-fs` *file_spec*)	*filesystem_name*
◆ Host volume (`-hostvol` *file_spec*) or logical volume (`-lvol` *file_spec*)	*name of the host or logical volume*
◆ Snapshot copy name (`-snapname` *long_snap_name*)	*long_snap_name*
◆ Additional Snapshot copy names	*snap_name* (long or short version)
If you want to display information about a Snapshot copy, specify the name for the Snapshot copy. For the first Snapshot copy, *long_snap_name*, enter the long version of the name, which includes the storage system name, volume, and Snapshot copy name. You can use the short version of the Snapshot copy name if it is on the same storage system and volume.	
`-verbose`	~
To display additional information, include the `-verbose` option.	

Displaying Snapshot copies residing on a storage system

To display information about Snapshot copies residing on a storage system, use the following syntax:

```
snapdrive snap show -filer filername [filername...]
  [-verbose]
```

Displaying Snapshot copies of a storage system volume

To display information about Snapshot copies of a storage system volume, use the following syntax:

```
snapdrive snap show -filervol filervol [filervol...]
[-verbose]
```

Displaying Snapshot copies of a LUN, and storage entities

To display information about Snapshot copies of a LUN, disk or volume group, file system, or host or logical volume, use the following syntax:

```
snapdrive snap { show | list } {-lun |-dg | -fs | -hostvol }
file_spec [file_spec ...]   [-verbose]
```

Displaying a Snapshot copy

To display information about a Snapshot copy, use the following syntax:

```
snapdrive snap show [-snapname] long_snap_name   [snap_name ...]
[-verbose]
```

Result: This operation displays, at a minimum, the following information about the Snapshot copy:

- The name of the storage system where the Snapshot copy was taken
- The name of the host that took the Snapshot copy
- The path to the LUNs on the storage system
- The date and time the Snapshot copy was taken
- The name of the Snapshot copy
- The names of the disk groups included in the Snapshot copy

Examples

Example 1: The following are examples of `snapdrive snap show` commands:

```
# snapdrive snap show -snapname toaster:/vol/vol2:snapA snapX snapY

# snapdrive snap show -verbose toaster:/vol/vol2:snapA
/vol/vol3:snapB snapC

# snapdrive snap show toaster:/vol/vol2:snapA

# snapdrive snap list -dg dg1 dg2
```

Example 2: This example uses a wildcard to display information about the available Snapshot copies on a particular volume:

```
# snapdrive snap show toaster:/vol/vol1:*
snap name                          host                date
snapped
-----------------------------------------------------------------
-------------
toaster:/vol/vol1:sss1               DBserver                Mar 12
16:19 dg1
toaster:/vol/vol1:testdg             DBserver                Mar 12
15:35 dg1
toaster:/vol/vol1:t1                 DBserver                Mar 10
18:06 dg1
toaster:/vol/vol1:hp_1               HPserver                Mar  8
19:01 vg01
toaster:/vol/vol1:r3                 DBserver                Mar  8
13:39 rdg1
toaster:/vol/vol1:r1                 DBserver                Mar  8
13:20 rdg1
```

```
toaster:/vol/vol1:snap2             DBserver                Mar  8
11:51 rdg1toaster:/vol/vol1:snap_p1       DBserver
               Mar  8 10:18 rdg1
```

Example 3: In this example, the verbose option is included in the command line in an AIX host:

```
# snapdrive snap list betty:/vol/vol1:testsnap -v
snap name                          host                date
snapped

-------------------------------------------------------------------
-------------
betty:/vol/vol1:testsnap             aix198-75             Jul 31
10:43 test1_SdDg
host OS: AIX 3 5

snapshot name: testsnap

Volume Manager: aixlvm 5.3
disk group:     test1_SdDg
host volume:    test1_SdHv
file system:    test1_SdHv      file system type:  jfs2
mountpoint:
/mnt/test1

lun path                         dev paths
-----------------------------------------------------

betty:/vol/vol1/aix198-75_luntest1_SdLun        /dev/hdisk32
```

Example 4: This example includes messages about Snapshot copies on an AIX host that were not created with SnapDrive for UNIX:

```
# snapdrive snap show -filer toaster
snap name                   host                    date
snapped
-------------------------------------------------------------------
-------------
toaster:/vol/vol1:hourly.0  non-snapdrive snapshot
toaster:/vol/vol1:hourly.0  non-snapdrive snapshot
toaster:/vol/vol1:snap1    DBserver1             Oct 01 13:42  dg1
dg2
toaster:/vol/vol1:snap2     DBserver2               Oct 10 13:40
DBdg/hvol1
toaster:/vol/vol1:snap3     DBserver3               Oct 31 13:45
DBdg
```

Example 5: This example displays a Snapshot copy of an LVM-based file system on an AIX host using the `snapdrive snap show` command and the `-verbose` option:

```
# snapdrive snap show -v -fs /mnt/check_submit/csdg2/hv3_0

snapname                      host          date          snapped
-------------------------------------------------------------------
-------------
toaster:/vol/vol1:mysnapshot sales-aix     Aug 24 10:55   csdg2

host OS: AIX 1 5

snapshot name: mysnapshot

Volume Manager: aixlvm 5.1

disk group: csdg2
host volume: csdg2_log
host volume: csdg2_hv3_0
file system: csdg2_hv3_0 file system type: jfs2 mountpoint:
/mnt/check_submit/csdg2/hv3_0

lun path dev paths
-----------------------------------------------------
spinel:/vol/vol1/check_submit_aix-4 /dev/hdisk4
```

Example 6: This example shows a Snapshot copy of an NFS-mounted directory tree on a Linux host using the `snapdrive snap list` command with the `-verbose` option:

```
# snapdrive snap list -fs /mnt/acctfs1 -v

snap name                              host         date
          snapped
-------------------------------------------------------------------
----------
besser:/vol/vol1:acctfs-s1          childs       Aug  8 18:58
/mnt/acctfs1

host OS: Linux 2.4.21-9.ELsmp #1 SMP Thu Jan 8 17:08:56 EST 2004

snapshot name: acctfs-s1 file system: type: nfs   mountpoint:
/mnt/acctfs1
filer dir: besser:/vol/vol1
```

Example 7: This example executes the `snapdrive snap show` command on a Linux host:

```
# snapdrive snap show -snapname surf:/vol/vol1:swzldg5snapped

snap name                          host                date
snapped
-------------------------------------------------------------------
-------------
surf:/vol/vol1:bagel5snapped        pons                   Aug 18
20:06 dg5
#
# ./linux/ix86/snapdrive snap show -v -snapname
surf:/vol/vol1:bagel5snapped >

snap name                          host                date
snapped
-------------------------------------------------------------------
-------------
surf:/vol/vol1:bagel5snapped        pons                   Aug 18
20:06 dg5

host OS: Linux 2.4.21-9.ELsmp #1 SMP Thu Jan 8 17:08:56 EST 2004

snapshot name: bagel5snapped

Volume Manager: linuxlvm 1.0.3

disk group:     dg5
host volume:    vol1
host volume:    vol2
host volume:    vol3

lun path                        dev paths
-----------------------------------------------------
surf:/vol/vol1/glk19    /dev/sdu
```

Example 8: The following examples use wildcard:

```
# snapdrive snap show toaster:/vol/volX:*
# snapdrive snap show -v toaster:/vol/volX:DB1*
filer1:/vol/volY:DB2*
# snapdrive snap show toaster:/vol/vol2:mysnap* /vol/vol2:yoursnap*
hersnap*
# snapdrive snap show toaster:/vol/volX:*
```

Example 9: In this example use of a wildcard is invalid because the wildcard is in the middle of the name instead of at the end:

```
# snap show toaster:/vol/vol1:my*snap
```

Other way to get Snapshot copy names

Another way to get a Snapshot copy name is to log on to the storage system and use the `snapdrive snap list` command there. This command displays the names of the Snapshot copies.

Note

The `snapdrive snap show` command is equivalent to the storage system `snapdrive snap list` command.

Renaming a Snapshot copy

Command to use to rename a Snapshot copy

You can use the `snapshot snap rename` command to change the name of an existing Snapshot copy.

Renaming a Snapshot copy that spans systems, volumes

You can also use this command to rename a Snapshot copy that is across multiple storage systems or multiple storage system volumes.

If you rename one of these Snapshot copies, you must also rename all the related Snapshot copies using the same name. This is because SnapDrive for UNIX uses a short name when it creates the Snapshot copy, even though it spans multiple storage systems or volumes. The rename command changes the name of the current Snapshot copy but it does not change the name of the related Snapshot copies in the other locations.

Guidelines for renaming Snapshot copies

Follow these guidelines when you use the `snapdrive snap rename` command:

- An error occurs if you try to rename a Snapshot copy to a different storage system volume.
- An error occurs if the new name for the Snapshot copy already exists. You can use the `-force` option to force SnapDrive for UNIX to change the name without generating an error.

Information required for snapdrive snap rename

The following table gives the information you need to supply when you use the `snapdrive snap rename` command.

Requirement	Argument
◆ `-snapname`	
Current name of the Snapshot copy; use the long form of this name	*old_long_snap_name*
New name of the Snapshot copy	*new_snap_name*

Requirement	Argument
◆ -force	~
◆ -noprompt	~
Optional: Decide if you want to overwrite an existing Snapshot copy. Without this option, this operation halts if you supply the name of an existing Snapshot copy. When you supply this option and specify the name of an existing Snapshot copy, it prompts you to confirm that you want to overwrite the Snapshot copy. To prevent SnapDrive for UNIX from displaying the prompt, include the -noprompt option also. (You must always include the -force option if you want to use the -noprompt option.)	

Changing a Snapshot copy name

To change the name of a Snapshot copy, use the following syntax:

```
snapdrive snap rename [-snapname] old_long_snap_name new_snap_name
[-force [-noprompt]]
```

Note

Before you execute this syntax, you must understand the options, keywords, and arguments mentioned in this command. For details, see Chapter 9, "SnapDrive for UNIX options, keywords, and arguments," on page 348.

Result: The Snapshot rename operation changes the name of the source Snapshot copy to the name specified by the target argument.

Example: The following are examples of the snapdrive snap rename command. The first command line includes the -force option because a Snapshot copy named newsnapshot already exists. In the second example, both Snapshot copy names use the long form of the name, but they both resolve to the same storage system volume:

```
snapdrive snap rename -force filer1:/vol/vol1:oldsnap newsnapshot
```

```
snapdrive snap rename filer1:/vol/vol1:FridaySnap
filer1:/vol/vol1:Snap040130
```

Restoring a Snapshot copy

Command to use to restore Snapshot copies

The `snapdrive snap restore` command restores data from the Snapshot copy you specify on the command line to the storage system. This operation replaces the contents of the *file_spec* arguments (for example disk groups, NFS files, NFS directory trees, file systems created directly on LUNs) that you specified on the `snapdrive snap restore` command with the contents of the *file_spec* arguments found in the specified Snapshot copy.

You can also restore Snapshot copies for non-existent *file_spec* arguments. This happens when the value you specify no longer exists on the host, but existed when you took the Snapshot copy. For example, it might be a file system that you have now unmounted or a disk group that you have removed.

Normally, you restore Snapshot copies from the host where you took the Snapshot copies (in other words, the originating host). You can also restore Snapshot copies using a different, or non-originating, host (see "Restoring a Snapshot copy from a different host" on page 271).

Note

SnapDrive for UNIX can only restore Snapshot copies it takes.

How SnapDrive restores Snapshot copies

SnapDrive for UNIX performs the following operations when you restore Snapshot copies:

- When you restore Snapshot copies for disk groups, or for host volumes and file systems that are created on them, SnapDrive for UNIX restores the whole disk group. If you specify part of a disk group, SnapDrive for UNIX still restores the entire disk group. An error occurs if you enter only a subset of the host volumes and/or file systems in each disk group on the command line. You can include the `-force` option to override this error; however, SnapDrive for UNIX then restores the entire disk group.
- When you restore a Snapshot copy for a file system created directly on a LUN, SnapDrive for UNIX restores the LUN where the file system resides and mounts the file system.
- When you restore Snapshot copies of LUNs (`-lun`), SnapDrive for UNIX restores the LUNs you specify.

  If you use the `-lun` option and specify a Snapshot copy that contains file system, disk group, or host volume entities, SnapDrive for UNIX restores the

LUN that you specify without restoring the storage entity. You must enter the long name for the LUN, and you must use the `-force` option.

- When you restore an NFS directory tree, SnapDrive for UNIX restores all the directories and files in the directory tree. You can also restore individual NFS files. Within the directory tree, SnapDrive for UNIX will delete any new NFS files or directories that you create after you created the Snapshot copy.
- If the configuration of the disk group you are trying to restore has changed since the Snapshot copy was taken, the restore operation fails. If you have added or removed a host volume, a file system, or LUNs, changed the way your data is striped, or resized any volume manager entity above the disk group level, you can override and restore an older Snapshot copy by including the `-force` option.

Note

You should always take a new Snapshot copy whenever a LUN or NFS directory tree has been added to or removed from a disk group.

- You cannot restore Snapshot copies of
  - Root disk groups
  - Boot device
  - Swap device

Restoring Snapshot copies on destination storage system

When you create a Snapshot copy on a destination storage system, the Snapshot copy is automatically replicated, from the source system, where it is created to the destination storage system. SnapDrive for UNIX allows you to restore the Snapshot copy on the source storage system. You can also restore the Snapshot copy on the destination storage system, provided you meet the following guidelines.

Restoring a single storage entity on a storage system or storage system cluster: You can restore a Snapshot copy that contains a single storage entity that resides on a storage system or on a clustered storage system. The name of the volume on the destination storage system must match the name of the volume on the source storage system.

Restoring multiple storage entities: To restore a Snapshot copy that contains storage entities that reside on multiple destination storage systems, you must meet the following requirements:

- The storage entities you specify on the command line must reside on a single storage system, or on a clustered storage system pair.

- The name of the volume of the source storage system must match the name of the volume of the destination storage system.
- You must set the `snapmirror-dest-multiple-filervolumes-enabled` argument in the snapdrive.conf file to `on`.

You can use one command to restore storage entities that reside on a single storage system or on a clustered storage system pair. See "Example 1" on page 246.

Guidelines for restoring a Snapshot copy in a cluster environment:

- The `snapdrive snapshot restore` command can be executed from any node in the cluster.
- The file system or disk groups have to be shared across all the nodes in the cluster, if they are live.
- The Snapshot restore operation on a shared file system or disk group fails, if any of the LUNs are mapped to a node outside the cluster. Ensure that none of the shared LUNs are mapped to a node outside the cluster.
- The Snapshot create operation can be conducted on a dedicated file system or disk group, but to restore the Snapshot copy in a shared mode, you have to ensure that the file system or disk group does not exist in dedicated mode on any node in the cluster. Otherwise, SnapDrive for UNIX gives you an error.
- If a file system or disk group does not exist in the cluster, SnapDrive for UNIX creates the LUNs from the Snapshot copy, maps them to all nodes in the cluster, and activates the disk group and file system. After mapping the LUNs to all nodes in the cluster, Veritas cluster volume manager refreshes the LUN information between all the nodes in the cluster. If the disk group activation is attempted before the LUNs information is refreshed among the CVM instances in cluster nodes, the Snapshot restore operation might fail. For CVM refresh, you have to reissue the `snapdrive snap restore` command.
- A Snapshot copy created on a node outside of a cluster can be restored and shared in the cluster only if the following is true:
  - The file system or disk group does not exist in the dedicated mode on any node in the cluster.
  - The LUNs are invisible to the node outside of the cluster.
- You cannot restore Snapshot copies on shared and dedicated systems in one Snapshot restore operation.
- If the `snapdrive snap restore` command is issued with the `-devicetype dedicated` option or without a `-devicetype` option specified on a shared disk group or file system, SnapDrive for UNIX alerts you that the LUNs connected to multiple nodes will be restored.

- If the disk group configuration is changed between Snapshot copy creation and Snapshot copy restore, SnapDrive for UNIX alerts you that the configuration is changed.

Considerations for restoring a Snapshot copy

Before restoring a Snapshot copy, consider the following important information:

- Make sure you are not in *any* directory on a file system that you want to restore. You can perform the `snapdrive snap restore` command from any directory except the one on a file system to which you want to restore the information.
- Do not interrupt the restore operation by entering Ctrl-C. Doing so could leave your system in an unusable configuration. If that happens, you will need to consult with NetApp technical support to recover.
- When exporting the NFS entities to a volume, set the Anonymous User ID option to "0" for the `snapdrive snap restore` command to work successfully.

Information required for snapdrive snap restore

The following table gives the information you need to supply when you use the `snapdrive snap restore` command.

Requirement	Argument
Decide the type of storage entity that you wish to restore and enter that entity's name with the appropriate argument. ◆ If you specify a host volume or file system to be restored, the argument you give is translated to the disk group containing it. SnapDrive for UNIX then restores the entire disk group. SnapDrive for UNIX freezes any file systems in host volumes in those disk groups and takes a Snapshot copy of all storage system volumes containing LUNs in those disk groups. ◆ If you specify a file specification that is an NFS mountpoint, the argument translates to a directory tree. SnapDrive for UNIX restores only the NFS directory tree or file. Within the directory tree, SnapDrive for UNIX will delete any new NFS files or directories that you created after you created the Snapshot copy. This ensures that the state of the restored directory tree will be the same as when the Snapshot copy of the tree was made. ◆ If you restore a LUN, SnapDrive for UNIX restores the LUN you specify.	

Requirement	Argument
◆ If you restore a file system that is created directly on a LUN, SnapDrive for UNIX restores the LUN and the file system. ◆ If the Snapshot copy contains storage entities that span multiple storage system volumes, you can restore any of the entities in that Snapshot copy.	
◆ LUN (`-lun` *file_spec*)	*name of the LUN. You must include the name of the storage system, volume, and LUN.*
◆ Disk group (`-dg` *file_spec*) or volume group (`-vg` *file_spec*)	*name of the disk or volume group*
◆ File system (`-fs` *file_spec*)	*filesystem_name*
◆ File (`-file` *file_spec*)	*name of the NFS file*
◆ Host volume (`-hostvol` *file_spec*) or logical volume (`-lvol` *file_spec*)	*name of the host or logical volume* **Note** You must supply both the requested volume and the disk group containing it; for example, `-hostvol dg3/acct_volume`.
Specify the name for the Snapshot copy. If any of the *file_spec* arguments you supply on the command line currently exist on the local host, you can use a short form of the Snapshot copy name. If none of the *file_spec* arguments exist on the host, you must use a long form of the Snapshot copy name where you enter the storage system name, volume, and Snapshot copy name. If you use a long name for the Snapshot copy and the path name does not match the storage system and/or storage volume information on the command line, SnapDrive for UNIX fails. The following is an example of a long Snapshot copy name: `big_filer:/vol/account_vol:snap_20031115` In some cases the value supplied with the *file_spec* argument may not exist on the host. For example, if you had unmounted a file system or removed a disk group by exporting it, deporting it, or destroying it, that file system or disk group could still be a value for the *file_spec* argument. It would, however, be considered a non-existent value. SnapDrive for UNIX can restore Snapshot copies for such a non-existent *file_spec*, but you must use the long Snapshot copy name.	

Requirement	Argument
When you restore Snapshot copies that span multiple storage systems and volumes, and contain an non-existent *file_spec* argument, SnapDrive for UNIX permits an inconsistency in the command line. It does not allow for existing *file_spec* arguments. If you want to restore only one storage entity from a multiple storage system Snapshot copy, the Snapshot copy you specify does not need to be on the same storage system as the storage system containing the storage entity. The short form of the same Snapshot copy name would omit the storage system and storage system volume name, so it would appear as: `snap_20031115`	
◆ Snapshot copy name (`-snapname`)	*snap_name*
It can be either a short name, such as mysnap1, or a long name that includes the storage system name, volume, and Snapshot copy name. Generally, NetApp recommends that you use the short name. If the *file_spec* argument is non-existent: that is, it no longer exists on the host; see the explanation of the *file_spec* argument. Then you must use the long name for the Snapshot copy.	
`-reserve | -noreserve`	
Optional: If you want SnapDrive for UNIX to create a space reservation when you restore the Snapshot copy.	
◆ `-force`	~
◆ `-noprompt`	~
Optional: Decide if you want to overwrite an existing Snapshot copy. Without this option, this operation halts if you supply the name of an existing Snapshot copy. When you supply this option and specify the name of an existing Snapshot copy, it prompts you to confirm that you want to overwrite the Snapshot copy. To prevent SnapDrive for UNIX from displaying the prompt, include the `-noprompt` option also. (You must always include the `-force` option if you want to use the `-noprompt` option.) You must include the `-force` option on the command line if you attempt to restore a disk group where the configuration has changed since the last Snapshot copy. For example, if you changed the way data is striped on the disks since you took a Snapshot copy, you would need to include the `-force` option. Without the `-force` option, this operation fails. This option asks you to confirm that you want to continue the operation unless you include the `-noprompt` option with it. **Note** If you added or deleted a LUN, the restore operation fails, even if you include the `-force` option.	

Requirement	Argument
◆ `-devicetype`	~
Optional: Specify the type of device to be used for SnapDrive for UNIX operations. This can be either "shared" that specifies the scope of LUN, disk group, and file system as cluster-wide or "dedicated" that specifies the scope of LUN, disk group, and file system as local. If you specify the `-devicetype dedicated` option, all the options of `snapdrive restore connect` command currently supported in SnapDrive 2.1 for UNIX function as they always have. If you initiate the `snapdrive restore connect` command with the `-devicetype shared` option from any nonmaster node in the cluster, the command is shipped to the master node and executed. For this to happen, you must ensure that the `rsh` or `ssh` access-without-password-prompt for the root user should be configured for all nodes in the cluster.	

Restoring a Snapshot copy

The restore operation can take several minutes, depending on the type and amount of data being restored.

To restore a Snapshot copy, use the following syntax:

```
snapdrive snap restore -snapname snap_name { -lun | -dg | -fs | -
hostvol | -file } file_spec [file_spec ...]
[{ -lun | -dg | -fs | -hostvol | -file } file_spec [file_spec ...]
...] [-force [-noprompt]] [{-reserve | -noreserve}]
[-devicetype {shared | dedicated}]
```

Note

For details about using these options and arguments, see the section "SnapDrive for UNIX options, keywords, and arguments" on page 348.

Result: SnapDrive for UNIX replaces the existing contents of the LUNs you specify in the `snapdrive snap restore` command line with the contents of the LUNs in the Snapshot copy you specify. This operation can take several minutes. When the operation is complete, SnapDrive for UNIX displays a message similar to the following:

```
Snap restore <filespec list> succeeded
```

Examples

Example1: In the following example, file system 1 (fs1) resides on storage system1, and file system 2 (fs2) resides on storage system1 and also on storage system 2, which is the partner storage system. File system 3 (fs3) resides on storage system1, partner storage system 2, and storage system3, which is not part of the cluster. An additional file system, fs4, resides entirely on storage system 4.

The following command creates a Snapshot copy of fs1, fs2, fs3 and fs4:

```
snapdrive snap create -fs /mnt/fs1 /mnt/fs2 /mnt/fs3 /mnt/fs4
-snapname fs_all_snap
```

The next command restores fs1 and fs2 on the destination storage system. Both fs1 and fs2 reside on a clustered pair, so you can restore them with one command:

```
snapdrive snap restore -fs /mnt/fs1 /mt/fs2 -snapname fs_all_snap
```

The following command restores fs4:

```
snapdrive snap restore -fs /mnt/fs4 -snapname fs_all_snap
```

SnapDrive for UNIX cannot restore fs3 on the destination storage system, because this file system resides on storage system1, storage system 2, and storage system 3.

Example 2: If you supply a non-existent *file_spec* argument that, at the time the Snapshot copy was created, spanned multiple storage system volumes, you can specify any of the Snapshot copies in which the values for the *file_spec* argument are stored.

For example, suppose disk group dg1 spans filer1:/vol/vol1 and filer2:/vol/vol1. When you performed the Snapshot create operation `snapdrive snap create -dg dg1 -snapname snapdg1`, SnapDrive for UNIX would have created two Snapshot copies: filer1:/vol/vol1:snapdg1 and filer2:/vol/vol1:snapdg1.

If you want to restore dg1 and it currently exists on the host (that is, it is *live*), you only need to supply the short name of the Snapshot copy on the command line and SnapDrive for UNIX finds both Snapshot copies and restores them:

```
snapdrive snap restore -dg dg1 -snapname snapdg1
```

If you exported disk group dg1, it becomes a non-existent entity on the host. You can still restore the Snapshot copy, but now you must specify the long Snapshot copy name that contains the storage system and volume information. The `snapdrive snap restore` command only accepts one Snapshot copy name, so you can supply the name of either Snapshot copy that was created by the `snap create` command. Enter one of the following command lines:

```
snapdrive snap restore -dg dg1 -snapname filer1:/vol/vol1:snapdg1
```

```
snapdrive snap restore -dg dg1 -snapname filer2:/vol/vol1:snapdg1
```

Example 3: This example restores a Snapshot copy on a Linux host:

```
# snapdrive snap restore -dg dg5 -snapname bagel5snapped
Starting to restore LUNs in disk group dg5
  WARNING:  This can take several minutes.
            DO NOT CONTROL-C!
            If snap restore is interrupted, the disk group
            being restored may have inconsistent or
            corrupted data.
  For detailed progress information, see the log file /var/log/sd-
recovery.log
        Importing dg5
snap restore: snapshot bagel5snapped contains:
        disk group dg5 containing host volumes
                vol1
                vol2
                vol3
snap restore: restored snapshot surf:/vol/vol1:bagel5snapped
```

Example 4: This example takes you through the steps of creating a Snapshot copy across multiple storage systems and volumes and then restoring it, even though the disk group it contains no longer exists on the host. The example uses disk group dg2, which is on filer2:/vol/vol2, and disk group dg3, which is on filer3:/vol/vol3. The following `snap create` command was used:

```
# snapdrive snap create -dg dg2 dg3 -snapname snapdg23
```

This command created the Snapshot copies filer2:/vol/vol2:snapdg23 and filer3:/vol/vol3:snapdg23.

If dg2 and dg3 still exist on the host, you can use the short name to restore the Snapshot copy:

```
# snapdrive snap restore -dg dg2 dg3 -snapname snapdg23
```

Suppose, however, that you had exported both dg2 and dg3. In that case, they no longer exist on the host. You can restore them by specifying either of the Snapshot copies created by the `snap create` command line, but you must include the long Snapshot copy name on the `snap restore` command line.You can enter either of the following command lines:

```
# snapdrive snap restore -dg dg2 dg3 -snapname
  filer3:/vol/vol3:snapdg23
```

```
# snapdrive snap restore -dg dg2 dg3 -snapname
  filer2:/vol/vol2:snapdg23
```

If you only want to restore dg2, enter one of the following command lines:

```
# snapdrive snap restore -dg dg2 -snapname
filer2:/vol/vol2:snapdg23

# snapdrive snap restore -dg dg2 -snapname
filer3:/vol/vol3:snapdg23
```

Because dg2 no longer exists on the host, you can specify the Snapshot copy on storage system3 even though dg2 was created on storage system2. You can only use this inconsistency if the Snapshot copy spans multiple storage systems or volumes and the host entity no longer exists. If dg2 still existed on the host, the second command line would fail.

Example 5: The following example shows how to use the `-lun` option to restore a specific LUN without restoring the storage entities that it contains. This syntax is useful in cases where a LUN that contains a storage entity is accidently removed.

In this example, disk group 1 (dg1) resides on luna and lunb. The command restores luna, which was removed accidently, but does not restore lunb or dg1. The long lun name and `-force` option are required:

```
# snapdrive snap restore -lun toaster:/vol/vol1/luna
-snapname dg1_snapshot -force
```

Example 6: The following example uses the full LUN name to restore a LUN on a qtree:

```
# snapdrive snap restore -lun fatboy:/vol/vol1/qt/sr2
fatboy:/vol/vol1/qt/sr fatboy:/vol/vol1/sr     -snapname
fatboy:/vol/vol1:sr
Starting to restore qt/sr2, qt/sr, sr
  WARNING:  This can take several minutes.
            DO NOT CONTROL-C!
            If snap restore is interrupted, the filespecs
            being restored may have inconsistent or corrupted
            data.
  For detailed progress information, see the log file /var/log/sd-
recovery.log
Successfully restored snapshot sr on fatboy:/vol/vol1

        raw LUN: fatboy:/vol/vol1/qt/sr2
        raw LUN: fatboy:/vol/vol1/qt/sr
        raw LUN: fatboy:/vol/vol1/sr
```

Example 7: The following example restores three LUNs that are at different locations with a single `snapdrive snap restore` command: one in a storage system volume, and two in two different qtrees:

```
# snapdrive snap restore -lun
   mylun qt1/mylyn1 qt2/mylyn2 filer:/vol/vol1/mylun3
filer:/vol/vol1/qt3/mylun4
   -snapname filer:/vol/vol:snapall
```

Example 8: This example restores a Snapshot copy using the Veritas stack:

```
# snapdrive snap restore -fs /mnt/vxfs1 -snapname snapVxvm
Starting to restore /mnt/vxfs1
  WARNING:  This can take several minutes.
            DO NOT CONTROL-C!
            If snap restore is interrupted, the filespecs
            being restored may have inconsistent or corrupted
            data.
  For detailed progress information, see the log file /var/log/sd-
recovery.log
        Importing vxvm1
        Activating hostvol vxfs1_SdHv
Successfully restored snapshot snapVxvm on snoopy:/vol/vol1

        disk group vxvm1 containing host volumes
                vxfs1_SdHv (filesystem: /mnt/vxfs1)
```

Example 9: This example restores a Snapshot copy using LVM:

```
# snapdrive snap restore -fs /mnt/jfs1 -snapname snapLvm
Starting to restore /mnt/jfs1
  WARNING:  This can take several minutes.
            DO NOT CONTROL-C!
            If snap restore is interrupted, the filespecs
            being restored may have inconsistent or corrupted
            data.
  For detailed progress information, see the log file /var/log/sd-
recovery.log
        Importing lvm1
        Activating hostvol jfs1_SdHv
Successfully restored snapshot snapLvm on snoopy:/vol/vol1
disk group lvm1 containing host volumes
                jfs1_SdHv (filesystem: /mnt/jfs1)
```

Example 10: The following is the standard warning message that appears during a Snapshot restore operation:

```
Starting to restore LUNs in disk groups: <dg1>
WARNING:  This can take several minutes.
DO NOT CONTROL-C!
If snap restore is interrupted, the disk groups being restored may
have inconsistent or corrupted data.
```

SnapDrive for UNIX also displays a warning message during a restore operation if any logical volumes have been added or removed since the Snapshot copy was taken. The message tells you which logical volumes have been added or removed. For example, if you added a volume called /dev/tester2/vol2 and removed the volume /dev/tester2/vol3 after you took the Snapshot copy snap1, you would see the following message when you executed the `snap restore` command:

```
Warning (Warning 0001-178): diskgroup configuration has changed
since snapshot lobster:/vol/vol1:snap1 was taken:

added hostvol /dev/tester2/vol2 removed hostvol /dev/tester2/vol3
```

Example 11: The following example illustrates how a *file_spec* is restored in a shared storage system:

```
# snapdrive snap restore -fs /mnt/sfortesting -snapname f270-197-
109:/vol/vol2:testsfarcsnap -devicetype shared -force

Execution started on cluster master: sfrac-57
Starting to restore /mnt/sfortesting
WARNING:  This can take several minutes.
DO NOT CONTROL-C!
If snap restore is interrupted, the filespecs
being restored may have inconsistent or corrupted
data.
For detailed progress information, see the log file
/var/log/sd-recovery.log

Connecting cluster node: sfrac-58

mapping lun(s) ... done
discovering lun(s) ... done

LUN f270-197-109:/vol/vol2/sfortesting_SdLun connected
- device filename(s): /dev/vx/dmp/c3t0d1s2
Importing sfortesting_SdDg
Activating hostvol sfortesting_SdHv
Successfully restored snapshot testsfarcsnap on f270-197
109:/vol/vol2disk group sfortesting_SdDg containing host volumes
sfortesting_SdHv (filesystem: /mnt/sfortesting)
```

Note

The preceding command can be used to restore both failed and takeover storage entities in a clustered environment.

Examples 12: The following are some additional examples of the `snapdrive snap restore` command:

```
# snapdrive snap restore -fs /mnt/dir -snapname
filer1:/vol/vol1:NewSnap33

# snapdrive snap restore -lun filer1:/vol/vol1/luna lunb
-snapname Sunday

# snapdrive snap restore -lun filer1:/vol/vol1/luna
filer2:/vol/vol1/lunb -snapname Thursday

# snapdrive snap restore -vg OracleVg1 Datavg -snapname Monday

# snapdrive snap restore -dg dg1 dg2 -snapname Tuesday
```

This example uses a host called DBserver and a disk group dg1 that has host volumes myvol1 and myvol2. The volume dg1/myvol2 has a file system mounted on /fs2. The disk group dg1 has three LUNs in it: toaster:/vol/vol1/lun0, toaster:/vol/vol1/lun1, and toaster: /vol/vol1/lun2. You can use any of the following commands to restore the same data:

```
# snapdrive snap restore -dg dg1 -snapname snap1

# snapdrive snap restore -v dg1 -snapname snap1

# snapdrive snap restore -hostvol dg1/myvol2 dg1/myvol1
-snapname toaster:/vol/vol1:snap1

# snapdrive snap restore -lvol dg1/myvol1 -fs /fs2
```

Restoring a Snapshot copy from a different host

In most cases, you restore a Snapshot copy from the host where you took the Snapshot copy. On occasion, you might need to restore a Snapshot copy using a different, or non-originating host. To restore a Snapshot copy using a non-originating host, use the same `snapdrive snap restore` command that you would normally use. If the Snapshot copy you restore contains NFS entities, the non-originating host must have permission to access the NFS directory. For additional information, see Chapter 2, “NFS considerations,” on page 20.

Connecting to a Snapshot copy

Connecting to a Snapshot copy

SnapDrive for UNIX lets you connect a host to a Snapshot copy from a different location on a host. This new location can be on the host where you took the Snapshot copy (the originating host) or on a different host (the non-originating host).

Being able to set up the Snapshot copies in a new location means you can back up a Snapshot copy to another medium, perform maintenance on a disk group, or simply access the Snapshot copy data without disrupting the original copy of the data.

With this command, you can connect a host to Snapshot copy that contains any of the following:

- LUNs
- A file system created directly on a LUN
- Disk groups, host volumes and file systems created on LUNs
- NFS directory trees
- Disk groups, host volumes, and file systems on shared storage system

How snapdrive snap connect works

When you use the `snapdrive snap connect` command, it clones the storage for the entity you specify and imports it to the host:

- If you specify a Snapshot copy that contains a LUN (`-lun`), SnapDrive for UNIX maps a new copy of the LUN to the host. You cannot use the `snapdrive snap connect` command to specify a LUN on the same command line with other storage entities (`-vg`, `-dg`, `-fs`, `-lvol`, or `-hostvol`).
- If you specify a file system that resides directly on a LUN, SnapDrive for UNIX maps the LUN to the host and mounts the file system.
- If you specify a Snapshot copy that contains a disk group, or a host volume or file system that is part of a disk group, the `snapdrive snap connect` command connects the entire target disk group. To make the connection, SnapDrive for UNIX re-activates all of the logical volumes for the target disk group and mounts all the file systems on the logical volumes.
- If you specify a Snapshot copy that contains an NFS directory tree, SnapDrive for UNIX creates a clone of the FlexVol volume that contains the NFS directory tree. SnapDrive for UNIX then connects the volume to the

host and mounts the NFS file system. Within the directory tree, SnapDrive for UNIX deletes any new NFS files or directories that you create after you created the Snapshot copy. SnapDrive for UNIX deletes from the FlexVol volume any files or directories that are outside of the NFS directories that you connect, if the `snapconnect-nfs-removedirectories` configuration option is set to "on".

If you connect a Snapshot copy that contains NFS directory trees using the `-readonly` option, SnapDrive for UNIX mounts the Snapshot copy of the directory directly without creating a clone.

You cannot use the `snapdrive snap connect` command to specify NFS mountpoints on the same command line as non-NFS entities; that is, using the options `-vg`, `-dg`, `-fs`, `-lvol`, or `-hostvol`.

Connecting Snapshot copies on mirrored storage systems

When you create a Snapshot copy on a mirrored storage system, the Snapshot copy is automatically replicated, from the source system where it is created, to the destination (mirrored) storage system. SnapDrive for UNIX allows you to connect the Snapshot copy on the source storage system. You can also connect the Snapshot copy on the destination storage system, provided you meet the following guidelines.

Connecting a single storage entity on a storage system or storage system cluster: You can connect a Snapshot copy that contains a single storage entity that resides on a storage system or on a clustered storage system. The name of the volume on the destination storage system must match the name of the volume on the source storage system.

Connecting multiple storage entities: To connect a Snapshot copy that contains storage entities that reside on multiple destination storage systems you must meet the following requirements:

- The storage entities you specify on the command line must reside on a single storage system, or on a clustered storage system.
- The name of the volume of the source storage system must match the name of the volume of the destination storage system.
- You must set the `snapmirror-dest-multiple-filervolumes-enabled` variable in the snapdrive.conf file to "on".

You can use one command to connect storage entities that reside on a single storage system or on a clustered storage system.

Snapshot connect and Snapshot restore operations

Unlike the Snapshot restore operation, the Snapshot connect operation does not replace the existing contents of the LUNs that make up the host entity with the Snapshot copy contents. It clones the information.

Once the connection is made, both Snapshot connect and Snapshot restore operations perform similar activities:

- The Snapshot connect operation activates logical volumes for the storage entity, mounts file systems, and optionally adds an entry to the host file system table.
- The Snapshot restore operation activates the logical volumes for the storage entity, mounts the file systems, and applies the host file system mount entries that were preserved in the Snapshot copy.

Guidelines for connecting Snapshot copies

Follow these guidelines when connecting to Snapshot copies:

- The `snapdrive snap connect` command only works with Snapshot copies created with version 2.x of SnapDrive for UNIX. It does not work with Snapshot copies created using a version 1.x of SnapDrive for UNIX.
- On a non-originating host, SnapDrive 3.0 for UNIX supports the Snapshot connect operation using Linux LVM1 or LVM2. However, it does not support the Snapshot connect operation on the originating host, if the LUN is part of the Linux LVM1 volume manager.
- On an originating host, SnapDrive 3.0 for UNIX supports connecting and restoring Snapshot copies that are created by SnapDrive 2.x for UNIX.

  Note

  On a Linux originating host, the Snapshot connect operation works only with Linux LVM2, and Snapshot copies created by SnapDrive for UNIX.

- On Linux hosts, the `snapdrive snap connect` command is supported if the Snapshot copy you connect contains a LUN, or a LUN with a file system, that was created without activating the Linux LVM1. SnapDrive for UNIX does not support the `snapdrive snap connect` command for Linux entities that are created using the Linux LVM1. For additional information, see "Connecting to a Snapshot copy" on page 272.

  The `snapdrive snap connect` command does not permit you to rename the disk group on a Linux host. For example, the following command is not supported:

  ```
  snapdrive snap connect -dg dg1 dg1copy -snapname
  toaster:/vol/vol1:dg1snapshot
  ```

- For read and write access to NFS directory trees, the `snapdrive snap connect` command uses the Data ONTAP FlexVol volume feature, and therefore requires Data ONTAP 7.0 or later. Configurations with Data ONTAP 6.5 can connect NFS files or directory trees, but are provided with read-only access.
- On an originating host, the Snapshot connect operation is not supported with NativeMPIO multipathing type.
- If you set the `enable-split-clone` configuration variable value to "on" or "sync" during the Snapshot connect operation and "off" during the Snapshot disconnect operation, SnapDrive for UNIX will not delete the original volume or LUN that is present in the Snapshot copy.
- You have to set the value of Data ONTAP 7.2.2 configuration option `vfiler.vol_clone_zapi_allow` to "on" to connect to a Snapshot copy of a volume or LUN in a vFiler unit.
- The Snapshot connect operation is not supported on the hosts having different host configurations.
- The `snapdrive snapshot connect` command used to connect to a root volume of a physical storage system or a vFiler unit will fail because Data ONTAP does not allow cloning of a root volume.

Guidelines for connecting Snapshot copies in a cluster environment:

- The `snapdrive snapshot connect` command can be executed from any node in the cluster. If you initiate the `snapdrive snap connect` command with the `-devicetype shared` option from any nonmaster node in the cluster, the command is sent to the master node and executed. For this to happen, ensure that the `rsh` or `ssh` access-without-password-prompt is allowed on all the cluster nodes.
- The multiple file systems and disk groups that are specified in this operation should have the same device type scope; that is, either all should be shared or all should be dedicated.
- The `snapdrive snap connect` command with NFS or storage entities on raw LUNs is not supported.
- The `-igroup` option is supported with the `-devicetype dedicated` option and not with the `-devicetype shared` option in the `snapdrive snap connect` command.
- SnapDrive for UNIX executes the `snapdrive snap connect` command on the master node. Before creating the shared storage entities, it creates and maps the LUN on the master node and then maps the LUNs on all the nonmaster nodes. Is also creates and manages the igroups for all the nodes in

the cluster. If any error occurs during this sequence, the Snapshot connect operation fails.

- The `snapdrive snap connect` command can be used to connect the following storage entities:
  - A shared file system or disk group that is already present in a shared or dedicated mode in the cluster.
  - A dedicated file system or disk group to a single node in the cluster even if the file system or disk group is already present in a shared mode in the cluster.
  - A Snapshot copy of a file system or disk group that is created on a node outside the cluster.
- A dedicated file system or disk group that is already present in a nonmaster node cannot be connected again in a shared mode in the cluster without the `-destdg` option for a disk group and the `-autorename` option for a file system.

  That is, if a file system is already present in dedicated mode in one of the nonmaster nodes in the cluster, you have to specify the `snapdrive snap connect` command with the `-destdg` and `-autorename` options, or explicitly specify the destination file system in the command.

Information required for snapdrive snap connect

The following table gives the information you need to supply when you use the `snapdrive snap connect` command.

Requirement	Argument
Decide the type of storage entity that you want to use to attach the Snapshot copy and supply that entity's name with the appropriate argument. This is the value for the *src_fspec* argument. ◆ If you connect a Snapshot copy of a LUN, SnapDrive for UNIX connects the LUN you specify. You cannot use the `-lun` option on the same command line with the `-vg -dg -fs -lvol` or `-hostvol` options. The short name of the LUN in the lun_name or qtree_name/lun_name format. ◆ If you connect a Snapshot copy of a file system that is created directly on a LUN, SnapDrive for UNIX connects the LUN that has the file system. ◆ If you connect a Snapshot copy of a disk group that has a host volume or file specification, the argument translates into a set of disk groups on the storage system. SnapDrive for UNIX connects the entire disk group containing the entity, even if the entity is a host volume or file system. ◆ If you connect a Snapshot copy of an NFS file system, the argument translates to the NFS directory tree. SnapDrive for UNIX creates a FlexClone of the volume, removes directory trees that are not specified in the Snapshot copy, and then connects and mounts the NFS directory tree. If you specify an NFS mountpoint, you cannot specify non-NFS entities (`-vg`,`-dg`, `-fs`, `-lvol` or `-hostvol`) on the same command line.	
◆ LUN (`-lun` *file_spec*)	*short name of the LUN.*
The *s_lun_name* specifies a LUN that exists in the `-snapname` *long_snap_name*. The short *lun_name* is required. You cannot include a storage system or storage system volume name. The *d_lun_name* specifies the name at which the LUN will be connected. The short *lun_name* is required. You cannot include a storage system or storage system volume name. You must specify a *d_lun_name.*	
◆ Disk group (`-dg` *file_spec*) or volume group (`-vg` *file_spec*)	*name of the disk or volume group*
◆ File system (`-fs` *file_spec*)	*filesystem_name*
◆ Host volume (`-hostvol` *file_spec*) or logical volume (`-lvol` *file_spec*)	*name of the host or logical volume*

Requirement	Argument
Connect a Snapshot copy with an NFS directory tree to Data ONTAP 6.5 or 7.0 configurations. ❖ If your configuration uses Data ONTAP 6.5 or a later version of Data ONTAP with traditional (not FlexVol) volumes, you must specify this option to connect the Snapshot copy with read-only access (required). ❖ If your configuration uses Data ONTAP 7.0 and later and FlexVol volumes, SnapDrive for UNIX automatically provides read-write access. Specify this option only if you want to restrict access to read-only (optional).	
◆ `-readonly`	
Optional: Supply a name by which the target entity will be accessible after the storage entity is connected. SnapDrive for UNIX uses this name to connect the destination entity. This is the *dest_file_spec* argument. If you omit this name, the `snap connect` command uses the value you supplied for *src_fspec*.	
◆ Name of target entity	*dest_file_spec*
Optional: Specify the names for the destination storage entities. If you included this information as part of the *dest_fspec*/*src_fspec* pair, you do not need to enter it here. You can use the *-destxx* options to specify names for destination storage entities if this information is not part of the *dest_fspec*/*src_fspec* pair. For example, the `-fs` option names only a destination mountpoint so you can use the `-destdg` option to specify the destination disk group. If you do not specify the name needed to connect an entity in the destination disk group, the `snapdrive snap connect` command takes the name from the source disk group. If you do not specify the name needed to connect an entity in the destination disk group, the `snap connect` command takes the name from the source disk group. If it cannot use that name, the operation fails, unless you included `-autorename` on the command line.	
◆ Destination disk group (`-destdg`) or destination volume group (`-destvg`)	*dgname*
◆ Destination logical volume (`-destlv`) or destination host volume (`-desthv`)	*lvname*

Requirement	Argument
Specify the name for the Snapshot copy. Use the long form of the name where you enter the storage system name, volume, and Snapshot copy name.	
◆ Snapshot copy name (`-snapname`)	*long_snap_name*
◆ `-nopersist`	~
Optional: Connect the Snapshot copy to a new location without creating an entry in the host file system table. ❖ The `-nopersist` option allows you to connect a Snapshot copy to a new location without creating an entry in the host file system table (for example, `fstab` on Linux). By default SnapDrive for UNIX creates persistent mounts. This means that: ❖ When you connect a Snapshot copy on a Solaris, AIX or HP-UX host, SnapDrive for UNIX mounts the file system and then places an entry for the LUNs that comprise the file system in the host file system table file. ❖ When you connect a Snapshot copy on a Linux host, SnapDrive for UNIX mounts the file system, resets the file system universal unique identifier (UUID) and label, and places the UUID and mountpoint in the host file system table file. ❖ You cannot use `-nopersist` to connect a Snapshot copy that contains an NFS directory tree.	
◆ `-reserve | -noreserve`	~
Optional: Connect the Snapshot copy to a new location with or without creating a space reservation.	
◆ Igroup name (`-igroup`)	*ig_name*
Optional: NetApp recommends that you use the default igroup for your host instead of supplying an igroup name.	
◆ `-autoexpand`	~

Requirement	Argument
To shorten the amount of information you must supply when connecting to a volume group, include the `-autoexpand` option on the command line. This option lets you name only a subset of the logical volumes or file systems in the volume group. It then expands the connection to the rest of the logical volumes or file systems in the disk group. This way you do not need to specify each logical volume or file system. SnapDrive for UNIX uses this information to generate the name of the destination entity. This option applies to each disk group specified on the command line and all host LVM entities within the group. Without the `-autoexpand` option (default), you must specify *all* affected host volumes and file systems contained in that disk group in order to connect the entire disk group. **Note** If the value you enter is a disk group, you do not need to enter all the host volumes or file systems because SnapDrive for UNIX knows what the disk group is connecting to. NetApp recommends that, if you include this option, you should also include the `-autorename` option. If the `-autoexpand` option needs to connect the destination copy of an LVM entity, but the name is already in use, the command fails unless the `-autorename` option is on the command line.	
The command fails if you do not include `-autoexpand` and you do not specify all the LVM host volumes in all the disk groups referred to on the command line (either by specifying the host volume itself or the file system).	
◆ `-autorename`	~
When you use the `-autoexpand` option without the `-autorename` option, the `snap connect command` fails if the default name for the destination copy of an LVM entity is in use. If you include the `-autorename` option, SnapDrive for UNIX renames the entity when the default name is in use. This means that with the `-autorename` option on the command line, the Snapshot connect operation continues regardless of whether all the necessary names are available. This option applies to all host-side entities specified on the command line. If you include the `-autorename` option on the command line, it implies the `-autoexpand` option, even if you do not include that option.	
◆ `-devicetype`	~

Requirement	Argument
Optional: Specify the type of device to be used for SnapDrive for UNIX operations. This can be either "shared" that specifies the scope of LUN, disk group, and file system as cluster-wide or "dedicated" that specifies the scope of LUN, disk group, and file system as local. If you specify the `-devicetype dedicated` option, all the options of `snapdrive snap connect` command currently supported in SnapDrive 2.1 for UNIX function as they always have. If you initiate the `snapdrive snap connect` command with the `-devicetype shared` option from any nonmaster node in the cluster, the command is shipped to the master node and executed. For this to happen, you must ensure that the `rsh` or `ssh` access-without-password-prompt for the root user should be configured for all nodes in the cluster.	
◆ `-split`	~
Enables to split the cloned volumes or LUNs during Snapshot connect and Snapshot disconnect operations.	

Connect to a Snapshot copy that contains LUNs

To connect to a Snapshot copy that contains LUNs, use the following syntax:

```
snapdrive snap connect -lun s_lun_name d_lun_name [[-lun]
s_lun_name d_lun_name ...] -snapname long_snap_name [-igroup
ig_name [ig_name ...]] [-split]
```

Note

The *s_lun_name* and *d_lun_name* should be in the format lun_name or qtree_name/lun_name.

Result: SnapDrive for UNIX clones the LUNs you specify and connects them to a new location.

Examples

Example 1: The following example connects the LUN mylun1, in hornet/vol/vol/tuesdaysnapshot to mylun1copy:

```
# ./snapdrive snap connect -lun mylun1 mylun1copy -snapname
hornet:/vol/vol1:somesnapshot

connecting hornet:/vol/vol1/mylun1:
    LUN copy mylun1copy ... created
      (original: hornet:/vol/vol1/mylun1)
```

```
    mapping new lun(s) ... done
    discovering new lun(s) ... done
```

Example 2: The following example connects two LUNs, mylun1 and mylun2, to mylun1copy and mylun2copy, respectively:

```
# ./snapdrive snap connect -lun mylun1 mylun1copy -lun mylun2
mylun2copy -snapname hornet:/vol/vol1:tuesdaysnapshot

connecting hornet:/vol/vol1/mylun1:
    LUN copy mylun1copy ... created
       (original: hornet:/vol/vol1/mylun1)

    mapping new lun(s) ... done

connecting hornet:/vol/vol1/mylun2:
    LUN copy mylun2copy ... created
       (original: hornet:/vol/vol1/mylun2)

    mapping new lun(s) ... done
    discovering new lun(s) ... done
```

Connecting to a Snapshot copy of storage entities other than LUNs

To connect to a Snapshot copy that contains storage entities other than LUNs, use the following syntax:

```
snapdrive snap connect fspec_set [fspec_set...]
-snapname long_snap_name [-igroup ig_name [ig_name ...]]
[-autoexpand] [-autorename] [-nopersist] [{-reserve | -noreserve}]
[-readonly] [-split]
```

In the preceding usage, *fspec_set* has the following format:

```
  {-dg | -fs | -hostvol} src_file_spec   [dest_file_spec]
[{-destdg | -destvg} dgname]    [{-destlv | -desthv} lvname]
```

Note

This command must always start with the name of the storage entity you want to connect (for example, `-dg`, `-hostvol`, or `-fs`). If you specify an NFS mountpoint, you cannot specify non-NFS entities (`-vg`, `-dg`, `-fs`, `-lvol` or `-hostvol`) on the same command line.

The command fails, if either of the following is true:

- Any destination names you supply must not currently be in use.
- A file system name that is being used as a mountpoint.

Note

On Linux hosts, SnapDrive 3.0 for UNIX supports the Snapshot connect operation on the originating host, unless the LUN is part of the Linux LVM1 volume manager.

Attention

When you connect from a non-originating host to a Snapshot copy containing the VxFS file system mounted with the default mount `qio` option, you should have the Veritas license for Veritas File Device Driver (VxFDD) installed.

Result: SnapDrive for UNIX clones the LUNs you specify and connects them to a new location.

Examples

Example 1: The following command line connects a disk group and uses the default names as the destination names (that is, it creates them from the source names):

```
# snapdrive snap connect -vg vg1 -snapname
filer1:/vol/vol1:vg1snapshot

 connecting vg1:
        LUN copy vg1_lun1_0 ... created
                 (original: filer1:/vol/vol1/vg1_lun1)

        mapping new lun(s) ... done
        discovering new lun(s) ... done
        Importing vg1
```

Example 2: The following command line connects a disk group with a single host volume. It also specifies a name for the destination host volume and disk group:

```
# snapdrive snap connect -lvol vg1/vol1 vg1copy/vol1copy -snapname
filer1:/vol/vol1:vg1snapshot

 connecting vg1:
        LUN copy vg1_lun1_0 ... created
                 (original: filer1:/vol/vol1/vg1_lun1)
mapping new lun(s) ... done
        discovering new lun(s) ... done
        Importing vg1copy
```

Example 3: The following command line connects a disk group with two LUNs and two file systems. It specifies a destination name for each of the file systems, the host volume for one of the file systems, and the disk groups for both file systems:

```
# snapdrive snap connect -fs mnt/fs1 /mnt/fs1copy -destvg vg1copy \
-fs /mnt/fs2 /mnt/fs2copy -destlv vg1copy/vol2copy -destvg vg1copy
\ -snapname filer1:/vol/vol1:vg1snapshot

 connecting vg1:
        LUN copy vg1_lun1_0 ... created
                 (original: filer1:/vol/vol1/vg1_lun1)
        LUN copy vg1_lun2_0 ... created
                 (original: filer1:/vol/vol1/vg1_lun2)

        mapping new lun(s) ... done
        discovering new lun(s) ... done
        Importing vg1copy
```

Example 4: The following command line includes the `-autoexpand` option as it connects a disk group with two file systems. It uses the default names as the destination names (that is, it creates them from the source names):

```
# snapdrive snap connect -lvol mnt/fs1 -snapname
filer1:/vol/vol1:vg1snapshot \
-autoexpand

 connecting vg1:
        LUN copy vg1_lun1_0 ... created
                 (original: filer1:/vol/vol1/vg1_lun1)
        LUN copy vg1_lun2_0 ... created
                 (original: filer1:/vol/vol1/vg1_lun2)

        mapping new lun(s) ... done
        discovering new lun(s) ... done
        Importing vg1
```

Example 5: The following command line includes the `-autorename` option as it connects a disk group with two file systems and two LUNs:

```
# snapdrive snap connect -fs mnt/fs1 -snapname
filer1:/vol/vol1:vg1snapshot \
-autorename

 connecting vg1:
        LUN copy vg1_lun1_0 ... created
```

```
                (original: filer1:/vol/vol1/vg1_lun1)
        LUN copy vg1_lun2_0 ... created
                (original: filer1:/vol/vol1/vg1_lun2)

        mapping new lun(s) ... done
        discovering new lun(s) ... done
        Importing vg1_0
```

Example 6: The following example connects to a Snapshot copy Snapshot copy with file system, disk group created on Veritas stack:

```
# snapdrive snap connect -fs /mnt/vxfs1 /mnt/vxfs1_clone -snapname
snoopy:/vol/vol1:snapVxvm -autorename

 connecting vxvm1:
        LUN copy lunVxvm1_0 ... created
                (original: snoopy:/vol/vol1/lunVxvm1)

        mapping new lun(s) ... done
        discovering new lun(s) ... done
 Importing vxvm1_0
Successfully connected to snapshot snoopy:/vol/vol1:snapVxvm
        disk group vxvm1_0 containing host volumes
                vxfs1_SdHv_0 (filesystem: /mnt/vxfs1_clone)
```

Example 7: The following example connects to a Snapshot copy Snapshot copy with file system, disk group created on LVM stack:

```
# snapdrive snap connect -fs /mnt/jfs1 /mnt/jfs1_clone -snapname
snoopy:/vol/vol1:snapLvm -autorename

 connecting lvm1:
        LUN copy lunLvm1_0 ... created
                (original: snoopy:/vol/vol1/lunLvm1)

        mapping new lun(s) ... done
        discovering new lun(s) ... done
        Importing lvm1_0
Successfully connected to snapshot snoopy:/vol/vol1:snapLvm
        disk group lvm1_0 containing host volumes
                jfs1_SdHv_0 (filesystem: /mnt/jfs1_clone)
```

Example 8: In the following example, file system 1 (fs1) resides on storage system1, and file system 2 (fs2) resides on storage system1 and also on storage system2, which is the partner storage system. File system 3 (fs3) resides on storage system1, partner storage system 2, and storage system 3, which is not part of the cluster. An additional file system, fs4, resides entirely on storage system 4.

The following command creates a Snapshot copy of fs1, fs2, fs3 and fs4:

```
snapdrive snap create -fs /mnt/fs1 /mnt/fs2 /mnt/fs3 /mnt/fs4
-snapname fs_all_snap
```

The next command connect fs1 and fs2 on the destination storage system. Both fs1 and fs2 reside on a clustered pair, so you can restore them with one command:

```
snapdrive snap connect -fs /mnt/fs1 /mt/fs2 -snapname fs_all_snap
```

The following command restores fs4:

```
snapdrive snap connect -fs /mnt/fs4 -snapname fs_all_snap
```

SnapDrive for UNIX cannot connect fs3 on the destination storage system, because this file system resides on storage system1, storage system 2, and storage system 3.

Connecting to Snapshot copies of shared storage entities other than LUNs

To connect to Snapshot copies that contain shared storage entities other than LUNs, use the following syntax:

```
snapdrive snap connect fspec_set [fspec_set...] -snapname
long_snap_name [-devicetype shared] [-split]
```

In this syntax, *fspec_set* is:

```
{-dg | -fs} src_file_spec [dest_file_spec] [-destdg dgname]
```

Examples

Example 1: The following example connects to a Snapshot copy that contains shared storage entities on an originating cluster. The operation is executed from the non-cluster-master node, but the command is shipped to the master node and executed:

```
# snapdrive snap connect -fs /mnt/sfortesting /mnt/sfortesting2 -
snapname f270-
197-109:/vol/vol2:testsfarcsnap -devicetype shared -autorename
Execution started on cluster master: sfrac-57
connecting sfortesting_SdDg:
LUN copy sfortesting_SdLun_0 ... created
(original: f270-197-109:/vol/vol2/sfortesting_SdLun)

mapping new lun(s) ... done
discovering new lun(s) ... done

Connecting cluster node: sfrac-58
```

```
mapping lun(s) ... done
discovering lun(s) ... done

LUN f270-197-109:/vol/vol2/sfortesting_SdLun_0 connected
- device filename(s): /dev/vx/dmp/c3t0d22s2
Importing sfortesting_SdDg_0
Activating hostvol sfracvxfstestfs_SdHv_0
Successfully connected to snapshot f270-197-
109:/vol/vol2:testsfarcsnap
disk group sfortesting_SdDg_0 containing host volumes
sfortesting_SdHv_0 (filesystem: /mnt/sfortesting2)
```

Example 2: The following example connects to a Snapshot copy that contains shared storage entities on a non-originating cluster. The operation is executed from the non-cluster-master node, but the command is shipped to the master node and executed:

```
# snapdrive snap connect -fs /mnt/sfortesting -snapname f270-197-
109:/vol/vol2:testsfarcsnap -devicetype shared
Execution started on cluster master: sfrac-57
connecting sfortesting_SdDg:
LUN copy sfortesting_SdLun_0 ... created
(original: f270-197-109:/vol/vol2/sfortesting_SdLun)
Step Action
184 Connecting to a Snapshot copy

mapping new lun(s) ... done
discovering new lun(s) ... done

Connecting cluster node: sfrac-58

mapping lun(s) ... done
discovering lun(s) ... done
LUN f270-197-109:/vol/vol2/sfortesting_SdLun_0 connected
- device filename(s): /dev/vx/dmp/c3t0d1s2
Importing sfortesting_SdDg
Activating hostvol sfortesting_SdHv
Successfully connected to snapshot f270-197-
109:/vol/vol2:testsfarcsnap
disk group sfortesting_SdDg containing host volumes
sfortesting_SdHv (filesystem: /mnt/sfortesting)
```

Disconnecting a Snapshot copy

Using the Snapshot disconnect operation

You use the `snapdrive snap disconnect` command to remove the mappings for LUNs, or for storage entities and the underlying LUNs, or for NFS directories in the Snapshot copy.

You can use this command to disconnect Snapshot copies that span multiple storage system volumes or multiple storage systems. The storage entities and volumes can reside on the same storage system or different storage systems.

Use this command to disconnect the any of the following:

- LUNs
- A file system created directly on a LUN
- Disk groups, host volumes and file systems created on LUNs
- NFS directory trees
- Shared disk groups, host volumes, and file systems created on LUNs

The disconnect operation does not modify the connected Snapshot copy. However, by default, the operation does delete any temporary LUNs or clones created by the corresponding connect operation.

Note

For LUNs, file systems on LUNs, and LVM entities, this command is equivalent to `snapdrive storage delete`.

Guidelines for disconnecting Snapshot copies

Follow these guidelines when disconnecting Snapshot copies:

- When you disconnect a file system, SnapDrive for UNIX always removes the mountpoint.
- Linux hosts allow you to attach multiple file systems to a single mountpoint. However, SnapDrive for UNIX requires a unique mountpoint for each file system. The `snapdrive snap disconnect` command fails if you use it to disconnect file systems that are attached to a single mountpoint.
- To undo the effects of the Snapshot connect operation, use the Snapshot disconnect command.
- If you set the `enable-split-clone` configuration variable value to “on” or “sync” during the Snapshot connect operation and “off” during the Snapshot

disconnect operation, SnapDrive for UNIX will not delete the original volume or LUN that is present in the Snapshot copy.

- On an HP-UX host with DMP multipathing solution, SnapDrive for UNIX does not support file system on a raw LUN.

For NFS entities: Follow these guidelines when disconnecting Snapshot copies that contain NFS entities:

- If you disconnect an NFS directory tree that you connected with read-only permission, SnapDrive for UNIX performs the following actions:
  - Unmounts the file system
  - Removes the mount entry in the file system table file
  - Removes the mountpoint
- If you disconnect an NFS directory tree that you connected with read-write permission, SnapDrive for UNIX performs the following actions:
  - Unmounts the file system
  - Removes the mount entry in the file system table file
  - Deletes the NFS directory tree that corresponds to the file system in the FlexVol volume clone
  - Destroys the underlying FlexVol volume clone (if it is empty)
  - Removes the mountpoint

Guidelines for disconnecting Snapshot copies in a cluster environment:

- The `snapdrive snap disconnect` command can be executed from any node in the cluster.
- For the Snapshot disconnect operation to be successful, either of the following should be true:
  - The storage entities should be shared across all the nodes in the cluster.
  - The LUNs should be mapped to all the nodes in the cluster.
- You can disconnect a storage entity from a specific node by using the `-devicetype dedicated` or `shared` option. If you are disconnecting a storage entity that is in a dedicated mode, you can omit `-devicetype` option from the command line syntax, because the default value is `dedicated`.
- The `snapdrive snap disconnect` command gives an error if a shared storage entity or LUN is disconnected with the `dedicated` option, or if a dedicated storage entity or LUN is disconnected with the `shared` option.
- SnapDrive for UNIX executes the `snapdrive snap disconnect` command on the master node. It destroys the storage entities, disconnects the LUNs on all the nonmaster nodes, and then disconnects the LUNs from the master

node in the cluster. If any error occurs during this sequence, the Snapshot disconnect operation fails.

Information required for snapdrive snap disconnect

The following table gives the information you need to supply when you use the `snapdrive snap disconnect` command.

Requirement	Argument
◆ LUN (`-lun` *file_spec*)	*name of the LUN. Include the name of the filer, volume and LUN.*
◆ Disk group (`-dg` *file_spec*) or volume group (`-vg` *file_spec*)	*name of the disk or volume group*
◆ File system (`-fs` *file_spec*)	*filesystem_name*
◆ Host volume (`-hostvol` *file_spec*) or logical volume (`-lvol` *file_spec*)	*name of the host or logical volume*
Specify the type of storage entity that you want to use to disconnect the Snapshot copy and supply that entity's name with the appropriate argument. This is the value for the *file_spec* argument.	
`-devicetype`	~
Optional: Specifies the type of device to be used for SnapDrive for UNIX operations. This can be either "shared" that specifies the scope of LUN, disk group, and file system as cluster-wide or "dedicated" that specifies the scope of LUN, disk group, and file system as local. If you specify the `-devicetype dedicated` option, all the options of the `snapdrive snap disconnect` command currently supported in SnapDrive 2.1 for UNIX function as they always have. If you initiate the `snapdrive snap disconnect` command with the `-devicetype shared` option from any nonmaster node in the cluster, the command is shipped to the master node and executed. For this to happen, you must ensure that the `rsh` or `ssh` access-withoutpassword-prompt for the root user should be configured for all nodes in the cluster.	
`-full`	~

Requirement	Argument
Include the `-full` option on the command line if you want SnapDrive for UNIX to disconnect the objects from the Snapshot copy even if a host-side entity on the command line has other entities (such as a disk group that has one or more host volumes). If you do not include this option, you must specify only empty host-side entities.	
◆ `-fstype`	*type*
◆ `-vmtype`	*type*
Optional: Specify the type of file system and volume manager to be used.	
`-split`	~
Enables to split the cloned volumes or LUNs during Snapshot connect and Snapshot disconnect operations.	

Disconnecting Snapshot copy with LUNs and no storage entities

To disconnect a Snapshot copy that contains LUNs having no storage entities, use the following syntax:

```
snapdrive snap disconnect -lun long_lun_name [lun_name ...]
[-devicetype {shared | dedicated}] [-split]
```

Result: SnapDrive for UNIX removes the mappings for the storage entities specified in the command line.

Example: This command removes the mappings to luna and lunb on the storage system toaster:

```
# snapdrive snap disconnect -lun toaster:/vol/vol1/luna lunb
```

Disconnecting Snapshot copy with storage entities

To disconnect a Snapshot copy that contains a disk group, host volume, file system, or NFS directory tree, use the following syntax:

```
snapdrive snap disconnect
{-dg | -fs | -hostvol } file_spec [file_spec ...]
[-dg | -fs | -hostvol } file_spec [file_spec ...]] [-full]
[-devicetype {shared | dedicated}] [-fstype type] [-vmtype type]
[-split]
```

Note

This command must always start with the storage entity (for example, `-lun -dg`, `-hostvol`, or `-fs`):

- If you specify a LUN (`-lun`), you must enter the long LUN name. You cannot specify a LUN with the `-lun` option on the same command line as other storage entities (`-vg`,`-dg`, `-fs`, `-lvol` or `-hostvol` options).
- If you specify an NFS mountpoint, you cannot specify non-NFS entities (`-vg`, `-dg`, `-fs`, `-lvol` or `-hostvol`) on the same command line. You must use a separate command to specify the NFS mountpoint.

An error occurs if the host entity is using LUNs that are not part of the Snapshot copy.

An error also occurs if you specify a subset of the host volumes and/or file systems contained in each target disk group.

Result: SnapDrive for UNIX removes the mappings for the storage entities specified in the command line.

Examples

Example 1: This command line removes the mappings to all the LUNs underlying the host volume dg5/myvolume. It removes any temporary LUNs that were created with a Snapshot connect operation:

```
# snapdrive snap disconnect -hostvol dg5/myvolume
```

Example 2: This command disconnects the mapping to disk group 1 (dg1) and to the underlying LUN. It also removes any temporary LUNs that were created with the Snapshot connect operation:

```
# snapdrive snap disconnect -lun toaster:/vol/vol1/luna -dg dg1
```

Example 3: This command line removes the mapping to the file system fs1, and to the LUN that underlies it. It also removes any temporary LUNs that were created with the Snapshot connect operation:

```
# snapdrive snap disconnect -fs mnt/fs1
```

Example 4: This command line removes the mappings for disk groups dg1, dg2, and dg3. It removes any temporary LUNs that might have been created with the Snapshot connect operation:

```
# snapdrive snap disconnect -dg dg1 dg2 dg3
```

Example 5: This example disconnects a Snapshot copy with file system, disk group on Veritas stack:

```
# snapdrive snap disconnect -fs /mnt/vxfs1_clone -fstype vxfs

 delete file system /mnt/vxfs1_clone
        - fs /mnt/vxfs1_clone ... deleted
        - hostvol vxvm1_0/vxfs1_SdHv_0 ... deleted
        - dg vxvm1_0 ... deleted
        - LUN snoopy:/vol/vol1/lunVxvm1_0 ... deleted
```

Example 6: This example disconnects a Snapshot copy with file system, disk group on LVM stack:

```
# snapdrive snap disconnect -fs /mnt/jfs1_clone -fstype jfs2

 delete file system /mnt/jfs1_clone
        - fs /mnt/jfs1_clone ... deleted
        - hostvol lvm1_0/jfs1_SdHv_0 ... deleted
        - dg lvm1_0 ... deleted
        - LUN snoopy:/vol/vol1/lunLvm1_0 ... deleted
```

Disconnecting Snapshot copies with shared storage entities

To disconnect Snapshot copies with shared storage entities, use the following syntax:

```
snapdrive snap disconnect
{-dg | -fs} file_spec [file_spec ...]
[{-dg | -fs} file_spec [file_spec ...] ...] long_snap_name [-full]
[-devicetype shared] [-fstype type] [-vmtype type] [-split]
```

Note

For details about using the options and arguments in this command line, see Chapter 9, "Command-line options," on page 348.

Example: This example disconnects shared file system:

```
# snapdrive snap disconnect -fs /mnt/oracle -devicetype shared
```

Deleting a Snapshot Copy

Command to use to delete Snapshot copies

The `snapdrive snap delete` command removes the Snapshot copies you specify from a storage system. This command does not perform any operations on the host. It only removes the Snapshot copy from a storage system, if you have permission to do so. (If you want to keep the LUNs and mappings, see the information on the `snapdrive storage delete` command on "Deleting storage from the host and storage system" on page 228.)

Reasons to delete Snapshot copies

You might delete older Snapshot copies for the following reasons:

- To keep fewer stored Snapshot copies than the hard limit of 255 on a storage system volume.

  After this limit is reached, attempts to create new Snapshot copies fail.
- To free space on the storage system volume.

  Even before the Snapshot copy limit is reached, a Snapshot copy fails if the disk does not have enough reserved space for it.
- You can also use the wildcard (*) character in Snapshot copy names. The Snapshot show operation enables you to use the wildcard character to show all Snapshot copy names that match a certain pattern. The following rules apply to using wildcard in Snapshot copy names:
  - You can use a wildcard at the end of the name only. You cannot use the wildcard at the beginning or the middle of a Snapshot copy name.
  - You cannot use the wildcard in the storage system or storage system volume fields of a Snapshot copy name.

Guidelines for deleting Snapshot copies

Follow these guidelines when you use the `snapdrive snap delete` command:

- The Snapshot delete operation fails if any of the Snapshot copies you want to delete are in use or were not created by SnapDrive for UNIX. You can override this behavior by including the `-force` option with the `snapdrive snap delete` command.
- If you have a Snapshot copy that spans multiple storage system volumes, you must manually delete the Snapshot copy on each volume.

Information required for snapdrive snap delete

The following table gives the information you need to supply when you use the `snapdrive snap delete` command.

Requirement	Argument
Specify the name for the Snapshot copy. Use the long form of the Snapshot copy name where you enter the storage system name, volume, and Snapshot copy name. The following is an example of a long Snapshot copy name: `big_filer:/vol/account_vol:snap_20031115` If you want to specify additional Snapshot copies, you can use the short form of the name if they are on the same storage system and volume as the first Snapshot copy. Otherwise, use the long form of the name again.	
◆ Snapshot copy name (`-snapname`)	*long_Snapshot copy_name*
◆ Additional Snapshot copies	*Snapshot copy_name* (either long or short form)
◆ `-verbose`	~
To display a list of the Snapshot copies being deleted, include the `-verbose` option. This option fills in the missing storage system and volume information in cases where you used the short form of the Snapshot copy name.	
◆ `-force`	~
◆ `-noprompt`	~
Optional: Decide if you want to overwrite an existing Snapshot copy. Without this option, this operation halts if you supply the name of an existing Snapshot copy. When you supply this option and specify the name of an existing Snapshot copy, it prompts you to confirm that you want to overwrite the Snapshot copy. To prevent SnapDrive for UNIX from displaying the prompt, include the `-noprompt` option also. (You must always include the `-force` option if you want to use the `-noprompt` option.)	

Deleting a Snapshot copy

To delete a Snapshot copy, use the following syntax:

```
snapdrive snap delete [-snapname] long_snap_name [snap_name ...]
[-verbose] [-force [-noprompt]]
```

Note

The section "SnapDrive for UNIX options, keywords, and arguments" on page 348 provides details about using the options and arguments in this command line.

Note

If the Snapshot copy you specify is in use, this operation fails. SnapDrive for UNIX only reports that this operation completed successfully if all the Snapshot copies are removed.

Result: SnapDrive for UNIX deletes the existing contents of the LUNs you specify in the `snap delete` command line and replaces them with the contents of the LUNs in the Snapshot copy you specify.

Examples

Example 1: This example displays a list of what is being deleted:

```
# snapdrive snap delete -v filer1:/vol/vol1/snap1 snap2 snap3
snapdrive: deleting
               filer1:/vol/vol1/snap1
               filer1:/vol/vol1/snap2
               filer1:/vol/vol1/snap3
```

Example 2: You can also use the wildcard character to specify the Snapshot copy name:

```
# snap delete myfiler:/vol/vol1:mysnap* /vol/vol2:hissnap*
yoursnap* hersnap
```

Troubleshooting 8

About this chapter

This chapter provides information about the troubleshooting tool available with SnapDrive for UNIX. This tool is for gathering information as well as solving problems.

At the time of this release, there were some known issues and limitations for SnapDrive for UNIX. While some issues affect all SnapDrive for UNIX host platforms, others affect only a specific host platform. To locate information on known issues and troubleshooting tips, see the SnapDrive for UNIX *Release Notes* at http://now.netapp.com.

Topics in this chapter

This chapter discusses the following topics:

- "Data collection utility" on page 298
- "Understanding error messages" on page 301
- "Common error messages" on page 303
- "Standard exit status values" on page 326

Data collection utility

About the data collection utility

SnapDrive for UNIX provides a data collection utility (snapdrive.dc) that collects diagnostic information about SnapDrive for UNIX and your system setup. It does this by running NetApp diagnostic utilities and copying SnapDrive for UNIX log files to a special directory. Then it creates a compressed file containing this information that you can send to NetApp technical support for analysis.

Note

This utility only gathers basic information about the system and the configuration of SnapDrive for UNIX. It does not copy the file containing login information for the storage systems. It also does not make any configuration changes.

Tasks performed by snapdrive.dc

The snapdrive.dc utility performs the following tasks:

- Runs the *host*_info and filer_info utilities to collect information about the host and the storage systems connected to the host, and saves this information to a compressed file. The *host*_info utility and filer_info utility comes along with SnapDrive for UNIX installation package.

  For example, the Solaris kit includes the solaris_info utility, while the HP-UX kit includes the hpux_info utility. On AIX, it is the aix_info utility and on Linux, it is the linux_info utility.
- Creates a directory called /tmp/netapp/ntap_snapdrive_*name*. (the directory path name can vary depending on the host; see the FCP or iSCSI Host Utilities documentation for more information on this path name). The tool places copies of the following files in the directory:
  - SnapDrive for UNIX version, as indicated by running the `snapdrive version` command
  - The snapdrive.conf file
  - The audit log files
  - The trace log files
  - The recovery log files
  - The files created by the *host*_info utility
- Creates a compressed file of the directory contents and displays a message stating you send this file to NetApp technical support.

Executing the data collection utility

To execute the data collection utility, complete the following steps.

Step	Action
1	Log in as root.
2	Change to the SnapDrive for UNIX diagnostic directory. The path is: *install_directory*/diag *install_directory* is the SnapDrive for UNIX installation directory for your host operating system. This directory can vary depending on your host operating system. See the installation steps to determine where this directory is on your host.
3	At the command prompt, enter the following command: **snapdrive.dc [-d *directory*] [-n *file_name*] [-f]** -d *directory* specifies the location for the compressed file that this utility creates. The default location is /tmp/netapp. -n *file_name* specifies a string to be included in the name for the directory and compressed output file. If you supply a value for this argument, the snapdrive.dc utility creates a directory called ntap_snapdrive_*name* and a file name called ntap_snapdrive_*name*.tar.Z. The default path name is /tmp/netapp/ntap_snapdrive_info.tar.Z. -f forces SnapDrive for UNIX to overwrite the files if they currently exist. Without the -f option, the snapdrive.dc utility returns an error message if the output directory or the file name exists. It does not overwrite any directories or file names. **Result:** This utility runs the *host*_info script for your host operating system, as well as copying log files and the configuration file for SnapDrive for UNIX. It places this information in a directory, and then creates a compressed file containing the contents of the directory.
4	Send the *directory*/ntap_snapdrive_*name*.tar.Z file to NetApp technical support for analysis.

Examples of using snapdrive.dc

The following are examples of using the snapdrive.dc utility.

Example 1: This example runs the snapdrive.dc utility without supplying any command-line arguments:

```
# snapdrive.dc
SnapDrive configuration info and logs are in directory
/tmp/netapp/ntap_snapdrive_info.
Compressed file is /tmp/netapp/ntap_snapdrive_info.tar.Z.
Please send this file to technical support for analysis.
```

Example 2: This example uses the command-line options to specify a directory and a name for the resulting file:

```
# snapdrive.dc -d . -n mysystem
SnapDrive configuration info and logs are in directory
./ntap_snapdrive_mysystem.
Compressed file is ./ntap_snapdrive_mysystem.tar.Z.
Please send this file to technical support for analysis.
```

Understanding error messages

Error message locations

SnapDrive for UNIX provides information about error messages in the following places:

- The command output

  It displays all messages to the standard error output of the SnapDrive for UNIX command.

- The system log

  SnapDrive for UNIX logs all errors that have a severity level of Fatal and Admin error to the system log using the syslog(3) mechanism.

In addition, SnapDrive for UNIX records information about the commands and their execution in the log files, specifically the following:

- The audit log file

  The audit log records the following information for each SnapDrive for UNIX command:

  - Who issued it
  - When it was issued
  - What its exit status was. This is very useful in determining what actually happened on a system.

- The trace log file

  The trace log records more detailed information about any errors that occur. NetApp technical support uses this log when diagnosing problems.

Error message format

SnapDrive for UNIX returns the standard error code information, which provides a more specific description of what caused the initial error condition. SnapDrive for UNIX error messages conform to the following format:

```
return code message-ID error type: message text
```

- *return code*—SnapDrive for UNIX error message ID that is linked to an exit status value which indicates the basic cause of the error condition. For more information, see "Standard exit status values" on page 326.
- *message-ID*—A unique identifier used by NetApp technical support to locate the specific code that produced the error. If you need to call NetApp technical support, NetApp recommends that you record the message ID that accompanied the error message.

- *error type*—Specifies the type of error that SnapDrive for UNIX encountered. Return values include the following:
  - *Warning*—SnapDrive for UNIX executed the command but issued a warning about conditions that might require your attention.
  - *Command*—SnapDrive for UNIX failed to execute the command due to an error in the command line. Check the command line format and variables to ensure they are correct.
  - *Admin*—SnapDrive for UNIX failed to execute the command due to incompatibilities in the system configuration. Contact your System Administrator to review your configuration parameters.
  - *Fatal*—SnapDrive for UNIX failed to execute the command due to an unexpected condition. Fatal errors are rare. If a fatal error occurs and you have problems resolving it, contact NetApp technical support for assistance.
- *message text*—Information that explains the error. This text might include information from another component to provide more detail about the error. For example, if a command-line argument such as a disk group is missing, the error message tells you what is missing. Or the ManageONTAP APIs that SnapDrive for UNIX uses to control the storage system might supply additional text to help explain the error. In this case, the text follows the basic SnapDrive for UNIX error message.

Sample error message: The following message indicates a problem on the command line. The message-ID is 0001-377.

```
Return Code: 43

0001-377 Command error: Disk group name dg2 is already in use or
conflicts with another entity.
```

Common error messages

Introduction

This section provides information about most common error messages that you may encounter. This chapter does not include information for every component or error condition.

Operating system limits on open files

SnapDrive for UNIX checks for operating system limitations on the number of files opened by a process.

Note

The default limit for the number of file handles opened simultaneously by one process varies based on your operating system. Check your operating system documentation to determine the limit.

If the number of open LUNs for one operation exceeds the operating system limit on the number of file handles opened simultaneously by one process, SnapDrive for UNIX exits with the following error message:

```
0001-001 Admin error: Unable to open device path-to-device
```

Example: You see an error message similar to the following one if this limit is exceeded on a Solaris host:

```
0001-001 Admin error: Unable to open device /dev/rdsk/c1t1d26s2
```

Error message values

The following table gives you detailed information about the most common errors that you can encounter when using SnapDrive for UNIX.

Error code	Return code	Type	Description	Solution
0001-019	3	Command	invalid command line -- duplicate filespecs: <dg1/vol2 and dg1/vol2>	This happens when the command executed has multiple host entities on the same host volume. For example, the command explicitly specified the host volume and the file system on the same host volume. **What to do:** Complete the following steps: 1. Remove all the duplicate instances of the host entities. 2. Execute the command again.
0001-023	11	Admin	Unable to discover all LUNs in disk group <dg1>. Devices not responding: </dev/dsk/c27t0d6> Please check the LUN status on the filer and bring the LUN online if necessary	This happens when a SCSI inquiry on the device fails. A SCSI inquiry on the device can fail for multiple reasons. **What to do:** Execute the following steps in the same order if the preceding step does not solve the issue: 1. Set the device-retries configuration variable to a higher value. For example, set it to "10" (the default value is "3") and execute the command again. 2. Use snapdrive storage show command with the -all option to get information about the device. 3. Check if the FCP or iSCSI service is up and running on the storage system. If not, contact the storage administrator to bring the storage system online.

Error code	Return code	Type	Description	Solution
				4. Check if the FCP or iSCSI service is up on the host. For more information, see the FCP or iSCSI *Host Utilities Setup Guide* at http://now.netapp.com/NOW/knowledge/docs/san/. If the preceding solutions do not solve the issue, contact NetApp technical support to identify the issue in your environment.
9000-023	1	Command	`No arguments for keyword -lun`	This error occurs when the command with the `-lun` keyword does not have the *lun_name* argument. **What to do:** Do either of the following; ◆ Specify the *lun_name* argument for the command with the `-lun` keyword. ◆ Check the SnapDrive for UNIX help message.
0001-028	1	Command	`File system </mnt/qa/dg4/vol1> is of a type (hfs) not managed by snapdrive Please resubmit your request, leaving out the file system <mnt/qa/dg4/vol1>`	This error occurs when a non-supported file system type is part of a command. **What to do:** Exclude or update the file system type and then use the command again. For the list of file system types that SnapDrive for UNIX supports, See the SnapDrive for UNIX *Compatibility Matrix* on the NOW site.

Error code	Return code	Type	Description	Solution
9000-030	1	Command	-lun may not be combined with other keywords	This error occurs when you combine the -lun keyword with the -fs or -dg keyword. This is a syntax error and indicates invalid usage of command. **What to do:** Execute the command again only with the -lun keyword.
0001-034	1	Command	mount failed: mount: <device name> is not a valid block device"	This error occurs only when the cloned LUN is already connected to the same filespec present in Snapshot copy and then you try to execute the snapdrive snap restore command. The command fails because the iSCSI daemon remaps the device entry for the restored LUN when you delete the cloned LUN. **What to do:** Do either of the following: ◆ Execute the snapdrive snap restore command again. ◆ Delete the connected LUN (if it is mounted on the same filespec as in Snapshot copy) before trying to restore a Snapshot copy of a original LUN.

Error code	Return code	Type	Description	Solution
0001-046 and 0001-047	1	Command	Invalid snapshot name: </vol/vol1/NO_FILER_PREFIX> or Invalid snapshot name: NO_LONG_FILERNAME - filer volume name is missing	This is a syntax error which indicates invalid use of command, where a Snapshot operation is attempted with an invalid Snapshot name. **What to do:** Complete the following steps: 1. Use the `snapdrive snap list -filer <filer-volume-name>` command to get a list of Snapshot copies. 2. Execute the command with the *long_snap_name* argument.
9000-047	1	Command	More than one -snapname argument given	SnapDrive for UNIX cannot accept more than one Snapshot name in the command line for performing any Snapshot operations. **What to do:** Execute the command again, with only one Snapshot name.
9000-049	1	Command	-dg and -vg may not be combined	This error occurs when you combine the `-dg` and `-vg` keywords. This is a syntax error and indicates invalid usage of commands. **What to do :** Execute the command either with the `-dg` or `-vg` keyword.

Error code	Return code	Type	Description	Solution
9000-050	1	Command	-lvol and -hostvol may not be combined	This error occurs when you combine the -lvol and -hostvol keywords. This is a syntax error and indicates invalid usage of commands. **What to do:** Complete the following steps: 1. Change the -lvol option to -hostvol option or vice-versa in the command line. 2. Execute the command.
9000-057	1	Command	Missing required -snapname argument	This is a syntax error that indicates an invalid usage of command, where a Snapshot operation is attempted without providing the *snap_name* argument. **What to do:** Execute the command with an appropriate Snapshot name.
0001-067	6	Command	Snapshot hourly.0 was not created by snapdrive.	These are the automatic hourly Snapshot copies created by Data ONTAP.
0001-092	6	Command	snapshot <non_existant_24965> doesn't exist on a filervol exocet: </vol/vol1>	The specified Snapshot copy was not found on the storage system. **What to do:** Use the snapdrive snap list command to find the Snapshot copies that exist in the storage system.

Error code	Return code	Type	Description	Solution
0001-099	10	Admin	Invalid snapshot name: <exocet:/vol2/dbvol:New SnapName> doesn't match filer volume name <exocet:/vol/vol1>	This is a syntax error that indicates invalid use of commands, where a Snapshot operation is attempted with an invalid Snapshot name. **What to do:** Complete the following steps: **1.** Use the `snapdrive snap list -filer <filer-volume-name>` command to get a list of Snapshot copies. **2.** Execute the command with the correct format of the Snapshot name that is qualified by SnapDrive for UNIX. The qualified formats are: *long_snap_name* and *short_snap_name*.
0001-122	6	Admin	Failed to get snapshot list on filer <exocet>: The specified volume does not exist.	This error occurs when the specified storage system (filer) volume does not exist. **What to do:** Complete the following steps: **1.** Contact the storage administrator to get the list of valid storage system volumes. **2.** Execute the command with a valid storage system volume name.

Error code	Return code	Type	Description	Solution
0001-124	111	Admin	Failed to remove snapshot <snap_delete_multi_inuse_24374> on filer <exocet>: LUN clone	The Snapshot delete operation failed for the specified Snapshot copy because the LUN clone was present. **What to do:** Complete the following steps: 1. Use the `snapdrive storage show` command with the `-all` option to find the LUN clone for the Snapshot copy (part of the backing Snapshot copy output). 2. Contact the storage administrator to split the LUN from the clone. 3. Execute the command again.
0001-155	4	Command	Snapshot <dup_snapname23980> already exists on <exocet: /vol/vol1>. Please use -f (force) flag to overwrite existing snapshot	This error occurs if the Snapshot name used in the command already exists. **What to do:** Do either of the following: ◆ Execute the command again with a different Snapshot name. ◆ Execute the command again with the `-f` (force) flag to overwrite the existing Snapshot copy.
0001-158	84	Command	diskgroup configuration has changed since <snapshotexocet:/vol/vol1:overwrite_noforce_25078> was taken. removed hostvol </dev/dg3/vol4> Please use '-f' (force) flag to override warning and complete restore	The disk group can contain multiple LUNs and when the disk group configuration changes, you encounter this error. For example, when creating a Snapshot copy, the disk group consisted of X number of LUNs and after making the copy, the disk group can have X+Y number of LUNs. **What to do:** Use the command again with the `-f` (force) flag.

Error code	Return code	Type	Description	Solution
0001-185	NA	Command	storage show failed: no NETAPP devices to show or enable SSL on the filers or retry after changing snapdrive.conf to use http for filer communication.	This problem can occur for the following reasons: ◆ If the iSCSI daemon or the FCP service on the host has stopped or is malfunction, the snapdrive storage show -all command fails, even if there are configured LUNs on the host. **What to do:** See the Host Utilities *Setup Guide* to resolve the malfunctioning iSCSI or FCP service. ◆ The storage system on which the LUNs are configured is down or is undergoing a reboot. **What to do:** Wait until the LUNs are up. ◆ The value set for the use-https-to-filer configuration variable might not be a supported configuration. **What to do:** Complete the following steps: **1.** Use the sanlun lun show all command to check if there are any LUNs mapped to the host. **2.** If there are any LUNs mapped to the host, follow the instructions mentioned in the error message. Change the value of the use-https-to-filer configuration variable (to "on" if the value is "off"; to "off' if the value is "on").

Error code	Return code	Type	Description	Solution
0001-226	3	Command	'snap create' requires all filespecs to be accessible Please verify the following inaccessible filespec(s): File System: </mnt/qa/dg1/vol3>	This error occurs when the specified host entity does not exist. **What to do:** Use the `snapdrive storage show` command again with the `-all` option to find the host entities which exist on the host.

Error code	Return code	Type	Description	Solution
0001-242	18	Admin	Unable to connect to filer: <filername>	SnapDrive for UNIX attempts to connect to a storage system through the secure HTTP protocol. The error can occur when the host is unable to connect to the storage system. **What to do:** Complete the following steps: **1.** Network problems: **a.** Use the `nslookup` command to check the DNS name resolution for the storage system that works through the host. **b.** Add the storage system to the DNS server if it does not exist. You can also use an IP address instead of a host name to connect to the storage system. **2.** Storage system Configuration: **a.** For SnapDrive for UNIX to work, you must have the license key for the secure HTTP access. **b.** After the license key is set up, check if you can access the storage system through a Web browser. **3.** Execute the command after performing either Step 1 or Step 2 or both.

Error code	Return code	Type	Description	Solution
0001-243	10	Command	Invalid dg name: < >	This error occurs when the disk group is not present in the host and subsequently the command fails. **What to do:** Complete the following steps: 1. Use the `snapdrive storage show -all` command to get all the disk group names. 2. Execute the command again, with the correct disk group name.
0001-246	10	Command	Invalid hostvolume name: </mnt/qa/dg2/BADFS>, the valid format is <vgname/hostvolname>, i.e. <mygroup/vol2>	**What to do:** Execute the command again, with the following appropriate format for the host volume name: `vgname/hostvolname`
0001-360	34	Admin	Failed to create LUN </vol/badvol1/nane-hp13_lunnewDg_fve_SdLun > on filer <exocet>: No such volume	This error occurs when the specified path includes a storage system volume which does not exist. **What to do:** Contact your storage administrator to get the list of storage system volumes which are available for use.

Error code	Return code	Type	Description	Solution
0001-372	58	Command	Bad lun name:: </vol/vol1/sce_lun2a> - format not recognized	This error occurs if the LUN names that are specified in the command do not adhere to the pre-defined format that SnapDrive for UNIX supports. SnapDrive for UNIX requires LUN names to be specified in the following pre-defined format: <filer-name: /vol/<volname>/<lun-name> **What to do:** Complete the following steps: **1.** Use the `snapdrive help` command to know the pre-defined format for LUN names that SnapDrive for UNIX supports. **2.** Execute the command again.
0001-373	6	Command	The following required 1 LUN(s) not found: exocet:</vol/vol1/NotARealLun>	This error occurs when the specified LUN is not found on the storage system. **What to do:** Do either of the following: ◆ To see the LUNs connected to the host, use the `snapdrive storage show -dev` command or `snapdrive storage show -all` command. ◆ To see the entire list of LUNs on the storage system, contact the storage administrator to get the output of the `lun show` command from the storage system.

Error code	Return code	Type	Description	Solution
0001-377	43	Command	Disk group name <name> is already in use or conflicts with another entity.	This error occurs when the disk group name is already in use or conflicts with another entity. **What to do:** Do either of the following: ◆ Execute the command with the -autorename option. ◆ Use the snapdrive storage show command with the -all option to find the names that the host is using. Execute the command specifying another name that the host is not using.
0001-380	43	Command	Host volume name <dg3/vol1> is already in use or conflicts with another entity.	This error occurs when the host volume name is already in use or conflicts with another entity **What to do:** Do either of the following: ◆ Execute the command with the -autorename option. ◆ Use the snapdrive storage show command with the -all option to find the names that the host is using. Execute the command specifying another name that the host is not using.

Error code	Return code	Type	Description	Solution
0001-417	51	Command	The following names are already in use: <mydg1>. Please specify other names.	**What to do:** Do either of the following: ◆ Execute the command again with the -autorename option. ◆ Use snapdrive storage show -all command to find the names that exists on the host. Execute the command again to explicitly specify another name that the host is not using.
0001-430	51	Command	You cannot specify both -dg/vg dg and -lvol/hostvol dg/vol	This is a syntax error which indicates an invalid usage of commands. The command line can accept either -dg/vg keyword or the -lvol/hostvol keyword, but not both. **What to do:** Execute the command with only the -dg/vg or -lvol/hostvol keyword.
0001-434	6	Command	snapshot exocet:/vol/vol1:NOT_EXIST doesn't exist on a filervol exocet:/vol/vol1	This error occurs when the specified Snapshot copy is not found on the storage system. **What to do:** Use the snapdrive snap list command to find the Snapshot copies that exist in the storage system.

Error code	Return code	Type	Description	Solution
0001-435	3	Command	You must specify all host volumes and/or all file systems on the command line or give the -autoexpand option. The following names were missing on the command line but were found in snapshot <hpux11i2_5VG_SINGLELUN_REMOTE>: Host Volumes: <dg3/vol2> File Systems: </mnt/qa/dg3/vol2>	The specified disk group has multiple host volumes or file system, but the complete set is not mentioned in the command. **What to do:** Do either of the following: ◆ Re-issue the command with the -autoexpand option. ◆ Use the snapdrive snap show command to find the entire list of host volumes and file systems. Execute the command specifying all the host volumes or file systems.
0001-440	6	Command	snapshot hpux11i2_5VG_SINGLELUN_REMOTE does not contain disk group 'dgBAD'	This error occurs when the specified disk group is not part of the specified Snapshot copy. **What to do:** To find if there is any Snapshot copy for the specified disk group, do either of the following: ◆ Use the snapdrive snap list command to find the Snapshot copies in the storage system. ◆ Use the snapdrive snap show command to find the disk groups, host volumes, file systems, or LUNs that are present in the Snapshot copy. ◆ If a Snapshot copy exists for the disk group, execute the command with the Snapshot name.

Error code	Return code	Type	Description	Solution
0001-442	1	Command	More than one destination <dis> and <dis1> specified for a single snap connect source <src>. Please retry using separate commands.	**What to do:** Execute a separate `snapdrive snap connect` command, so that the new destination disk group name (which is part of the `snap connect` command) is not the same as what is already part of the other disk group units of the same `snapdrive snap connect` command.
0001-465	6	Command	The following filespecs do not exist and cannot be deleted: Disk Group: <nane-hp13_dg1>	The specified disk group does not exist on the host, therefore the deletion operation for the specified disk group failed. **What to do:** See the list of entities on the host by using the `snapdrive storage show` command with the `-all` option.

Error code	Return code	Type	Description	Solution
0001-476	NA	Admin	Unable to discover the device associated with <long lun name>. If multipathing in use, possible multipathing configuration error.	There can be many reasons for this failure. ◆ Invalid host configuration: The iSCSI, FCP, or the multipathing solution is not properly setup. ◆ Invalid network or switch configuration: The IP network is not setup with the proper forwarding rules or filters for iSCSI traffic, or the FCP switches are not configured with the recommended zoning configuration. The preceding issues are very difficult to diagnose in an algorithmic or sequential manner. **What to do:** NetApp recommends that before you use SnapDrive for UNIX, you follow the steps recommended in the Host Utilities *Setup Guide* (for the specific operating system) for discovering LUNs manually. After you discover LUNs, use the SnapDrive for UNIX commands.

Error code	Return code	Type	Description	Solution
0001-486	12	Admin	LUN(s) in use, unable to delete. Please note it is dangerous to remove LUNs that are under Volume Manager control without properly removing them from Volume Manager control first.	SnapDrive for UNIX cannot delete a LUN that is part of a volume group. **What to do:** Complete the following steps: **1.** Delete the disk group using the command `snapdrive storage delete -dg <dgname>`. **2.** Delete the LUN.
0001-494	12	Command	Snapdrive cannot delete <mydg1>, because 1 host volumes still remain on it. Use -full flag to delete all file systems and host volumes associated with <mydg1>	SnapDrive for UNIX cannot delete a disk group until all the host volumes on the disk group are explicitly requested to be deleted. **What to do:** Do either of the following: ◆ Specify the `-full` flag in the command. ◆ Complete the following steps: **a.** Use the `snapdrive storage show -all` command to get the list of host volumes that are on the disk group. **b.** Mention each of them explicitly in the SnapDrive for UNIX command.

Error code	Return code	Type	Description	Solution
0001-541	65	Command	Insufficient access permission to create a LUN on filer, <exocet>.	SnapDrive for UNIX uses the sd-*hostname*.prbac file on the root storage system (filer) volume for its pseudo access control mechanism. **What to do:** Do either of the following: ◆ Modify the sd-*hostname*.prbac file in the storage system to include the following requisite permissions (can be one or many): ❖ NONE ❖ SNAP CREATE ❖ SNAP USE ❖ SNAP ALL ❖ STORAGE CREATE DELETE ❖ STORAGE USE ❖ STORAGE ALL ❖ ALL ACCESS ◆ In the snapdrive.conf file, ensure that the all-access-if-rbac-unspecified configuration variable is set to "on".
0001-570	6	Command	Disk group <dg1> does not exist and hence cannot be resized	This error occurs when the disk group is not present in the host and subsequently the command fails. **What to do:** Complete the following steps: **1.** Use the snapdrive storage show -all command to get all the disk group names. **2.** Execute the command with the correct disk group name.

Error code	Return code	Type	Description	Solution
0001-574	1	Command	<VmAssistant> lvm does not support resizing LUNs in disk groups	This error occurs when the volume manager that is used to perform this task does not support LUN resizing. SnapDrive for UNIX depends on the volume manager solution to support the LUN resizing, if the LUN is part of a disk group. **What to do:** Check if the volume manager that you are using supports LUN resizing.
0001-616	6	Command	1 snapshot(s) NOT found on filer: <exocet:/vol/vol1:MySnapName>	SnapDrive for UNIX cannot accept more than one Snapshot name in the command line for performing any Snapshot operations. To rectify this error, re-issue the command with one Snapshot name. This is a syntax error which indicates invalid use of command, where a Snapshot operation is attempted with an invalid Snapshot name. To rectify this error, complete the following steps: ◆ Use the `snapdrive snap list -filer <filer-volume-name>` command to get a list of Snapshot copies. ◆ Execute the command with the *long_snap_name* argument.
0001-640	1	Command	Root file system / is not managed by snapdrive	This error occurs when the root file system on the host is not supported by SnapDrive for UNIX. This is an invalid request to SnapDrive for UNIX.

Error code	Return code	Type	Description	Solution
0001-684	45	Admin	Mount point <fs_spec> already exists in mount table	**What to do:** Do either of the following: ◆ Execute the SnapDrive for UNIX command with a different mountpoint. ◆ Check that the mountpoint is not in use and then manually (using any editor) delete the entry from the following files: ❖ Linux: /etc/fstab ❖ Solaris: /etc/vfstab ❖ AIX: /etc/filesystems ❖ HPUX: /etc/fstab
0001-796 and 0001-767	3	Command	More than one lun name cannot be specified with -nolvm	SnapDrive for UNIX does not support more than one LUN in the same command with the -nolvm option. **What to do:** Do either of the following: ◆ Use the command again to specify only one LUN with the -nolvm option. ◆ Use the command without the -nolvm option. This will use the supported volume manager present in the host, if any.

Error code	Return code	Type	Description	Solution
0001-876	NA	Admin	HBA assistant not found	If the HBA service is not running, you will get this error on executing the SnapDrive for UNIX commands, such as, `snapdrive storage create`, `snapdrive config prepare luns`. **What to do:** Check the status of the FCP or iSCSI service. If it is not running, start the service and execute the SnapDrive for UNIX command.

Standard exit status values

Understanding exit status values of error messages

Each SnapDrive error message ID is linked to an exit status value. Exit status values contain the following information:

- *Exit status value*—indicates the basic cause of the error condition.
- *Type*—indicates the type of error. The level of seriousness depends on the message, not the value. The following are the possible values:
  - *Warning*—SnapDrive for UNIX executed the command but issued a warning about conditions that might require your attention.
  - *Command*—SnapDrive for UNIX failed to execute the command due to an error in the command line. Check the command line format to ensure they are correct.
  - *Admin*—SnapDrive for UNIX failed to execute the command due to incompatibilities in the system configuration. Contact your System Administrator to review your configuration parameters.
  - *Fatal*—SnapDrive for UNIX failed to execute the command due to an unexpected condition. Fatal errors are rare. If a fatal error occurs and you have problems resolving it, contact NetApp technical support for assistance in determining the steps you need to take to recover correctly and fix any error condition.

Using exit status values

Exit status values are used in scripts to determine the success or failure of a SnapDrive for UNIX command.

- A value of zero indicates that the command completed successfully.
- A value other than zero indicates that the command did not complete, and provides information about the cause and severity of the error condition.

Script example

The following script uses SnapDrive for UNIX exit status values:

```
#!/bin/sh
# This script demonstrates a SnapDrive
# script that uses exit codes.

RET=0;
#The above statement initializes RET and sets it to 0

snapdrive snap create -dg vg22 -snapname vg22_snap1;
```

```
# The above statement executes the snapdrive command

RET=$?;
#The above statement captures the return code.
#If the operation worked, print
#success message. If the operation failed, print
#failure message and exit.

if [ $RET -eq 0 ]; then
echo "snapshot created successfully"
else
echo "snapshot creation failed, snapdrive exit code was $RET"
exit 1
fi

exit 0;
```

If RET=0, the command executed successfully and the script outputs the following:

```
# ./tst_script
snap create: snapshot vg22_snap1 contains:
disk group vg22 containing host volumes
lvol1
snap create: created snapshot betty:/vol/vol2:vg22_snap1
snapshot created successfully
```

If RET= a value other than zero, the command did not execute successfully. The following example shows typical output:

```
# ./tst_script
0001-185 Command error: snapshot betty:/vol/vol2:vg22_snap1 already
exists on betty:/vol/vol2.
Please use -f (force) flag to overwrite existing snapshot
snapshot creation failed, snapdrive exit code was 4
```

Exit status values

The following table contains information about exit status values. The exit status values are numbered sequentially. If SnapDrive for UNIX does not currently implement an error, that exit status value is not included in the table. As a result, there can be some gaps in the numbers.

Exit value	Error name	Type	Description
1	Not supported	Command error	A function was invoked that is not supported in this version of SnapDrive for UNIX.
2	No memory	Fatal	The system has run out of memory. SnapDrive for UNIX cannot proceed until you free enough memory for it to work. Check other applications running to verify that they are not consuming excessive memory.
3	Invalid command	Command error	You issued an invalid command; this is likely to be a syntax error in the text of the command you entered.
4	Already exists	Command error	You requested that something be created that already exists. Usually, this error refers to a Snapshot copy name, which must not exist on the storage system volume where you are taking the Snapshot copy.
5	Create thread failed	Admin error	SnapDrive for UNIX could not create a process thread. Check the other processes running on the system to make sure that enough thread resources are available.
6	Not found	Command error	You included a file, data group, host volume, file system, or other argument on the SnapDrive for UNIX command line that does not exist.
7	Not a mounted file system	Command error	The file system you want to access either is not a valid file system or is not mounted.
9	Volume manager error	Command error	An error was returned when accessing the volume manager. See the specific error message to get details of which error, and why.

Exit value	Error name	Type	Description
10	Invalid name	Command error	You supplied a name on the command line that was not correctly formatted. For example, a storage system volume was not specified as *filer*:/vol/*volname*. This message also occurs when an invalid character is given in either a storage system or a volume manager-based name.
11	Device not found	Admin error	SnapDrive for UNIX cannot access a LUN in the disk group that you want to take a Snapshot copy of. Check the status of all LUNs, both on the host and on the storage system. Also check that the storage system volume is online, and that the storage system is up and connected to the host.
12	Busy	Command error	The LUN device, file, directory, disk group, host volume, or other entity is busy. This is generally a nonfatal error that goes away when you retry the command. It sometimes indicates that a resource or process is hung, causing the object to be busy and unavailable for SnapDrive for UNIX to use. It could also indicate you are trying to make a Snapshot copy during a period when the I/O traffic is too heavy for the Snapshot copy to be made successfully.
13	Unable to initialize	Fatal	SnapDrive for UNIX could not initialize third-party material that it needs. This can refer to file systems, volume managers, cluster software, multipathing software, and so on.

Exit value	Error name	Type	Description
14	SnapDrive busy	Command error	Another user or process is performing an operation on the same hosts or storage systems at the same time that you asked SnapDrive for UNIX to perform an operation. Retry your operation. Occasionally this message means that the other process is hung and you must kill it. **Note** The Snapshot restore operation can take a long time under some circumstances. Be sure that the process you think is hung is not just waiting for a Snapshot restore operation to be completed.
15	Config file error	Fatal	The snapdrive.conf file has invalid, inadequate, or inconsistent entries. See the specific error message for details. You must correct this file before SnapDrive for UNIX can continue. For information about modifying the snapdrive.conf file, see “Setting values in snapdrive.conf” on page 118.
17	Bad permissions	Command error	You do not have permission to execute this command. You must be logged in as root to run SnapDrive for UNIX.
18	No filer	Admin error	SnapDrive for UNIX cannot contact the storage system needed for this command. Check the connectivity to the storage system indicated in the error message.
19	Bad filer login	Admin error	SnapDrive for UNIX cannot log in to the storage system using the login information you supplied.
20	Bad license	Admin error	A service SnapDrive for UNIX requires is not licensed to run on this storage system.

Exit value	Error name	Type	Description
22	Cannot freeze fs	Admin error	A Snapshot create operation failed because SnapDrive for UNIX could not freeze the file systems specified in order to make the Snapshot copy. Confirm that the system I/O traffic is light enough to freeze the file system and then retry the command.
27	Inconsistent Snapshot copy	Admin error	The Snapshot restore operation failed because you requested a restore from a Snapshot copy with inconsistent images of the disk group. Inconsistent images can occur in the following cases: ◆ You did not make the Snapshot copy using SnapDrive for UNIX. ◆ The Snapshot create operation was interrupted before it set consistent bits, and thus, could not clean up (as in the case of a catastrophic system failure). ◆ Some type of data problem occurred with the Snapshot copy after it was made.
28	HBA failure	Admin error	SnapDrive for UNIX encountered an error while trying to retrieve information from the HBA.
29	Bad metadata	Admin error	SnapDrive for UNIX encountered an error in the Snapshot copy metadata that it wrote when it created the Snapshot copy.
30	No Snapshot copy metadata	Admin error	SnapDrive for UNIX cannot perform a Snapshot restore operation because the metadata does not contain all requested disk groups.

Exit value	Error name	Type	Description
31	Bad password file	Admin error	The password file has a bad entry. Use the `snapdrive config delete` command to delete the login entry for this storage system. Then re-enter the login information using the `snapdrive config set` *user_name* command. For information on specifying user logins to storage systems, see "Specifying the current login information for storage systems" on page 154.
33	No password file entry	Admin error	The password file has no entry for this storage system. Run the `snapdrive config set` *username filername* command for every storage system on which you need to run SnapDrive for UNIX. Then try this operation again.
34	Not aNetApp LUN	Admin error	A SnapDrive for UNIX command encountered a LUN that is not on a NetApp storage system.
35	User aborted	Admin error	The system displayed a prompt asking you to confirm an operation and you indicated that you did not want the operation performed.
36	I/O stream error	Admin error	The system input or system output routines returned an error that SnapDrive for UNIX did not understand. Run snapdrive.dc and send that information to NetApp technical support so that they can help you determine which steps to perform to complete the recovery.
37	File system full	Admin error	An attempt to write a file failed because there was insufficient space on the file system. SnapDrive for UNIX can proceed when you free enough space on the appropriate file system.

Exit value	Error name	Type	Description
38	File error	Admin error	An I/O error occurred when SnapDrive for UNIX was reading or writing a system configuration file or a temporary file.
39	Duplicate diskgroup	Command error	SnapDrive for UNIX got a duplicate minor node number when trying to activate a disk group.
40	File system thaw failed.	Admin error	A snap create command failed due to system activity on the file system. This usually occurs when the SnapDrive for UNIX file system freeze, required for the Snapshot copy, times out before the Snapshot copy is complete.
43	Name already in use	Command error	SnapDrive for UNIX attempted to create a disk group, host volume, file system or LUN but the name was already in use. To correct, select a name that is not in use, and re-enter the SnapDrive for UNIX command.
44	File system manager error	Fatal	SnapDrive for UNIX encountered an unexpected error from the file system when: ◆ attempting to create the file system ◆ making an entry in the file system mount table to automatically mount the file system at boot. The text of the error message displayed with this code describes the error that the file system encountered. Record the message, and send it to NetApp technical support so that they can help you determine which steps to perform to complete the recovery.
45	Mountpoint error	Admin error	The file system mountpoint appeared in the system mount table file. To correct, select a mountpoint that is not in use or listed in the mount table, and re-enter the SnapDrive for UNIX command.

Exit value	Error name	Type	Description
46	LUN not found	Command error	A SnapDrive for UNIX command attempted to access a LUN that did not exist on the storage system. To correct, check that the LUN exists and that the name of the LUN is entered correctly.
47	Initiator group not found	Admin error	A storage system initiator group could not be accessed as expected. As a result, SnapDrive for UNIX cannot complete the current operation. The specific error message describes the problem and the steps you need to perform to resolve it. Fix the problem and then repeat the command.
48	Object offline	Admin error	SnapDrive for UNIX attempted to access an object (such as a volume) but failed because the object was offline.
49	Conflicting entity	Command error	SnapDrive for UNIX attempted to create an igroup, but encountered an igroup of the same name.
50	Cleanup error	Fatal	SnapDrive for UNIX encountered an item that should be removed but is still there.
51	Disk group ID conflict	Command error	A `snapdrive snap connect` command requested a disk group ID that conflicts with an existing disk group. This usually means that a `snapdrive snap connect` command on an originating host is being attempted on a system that does not support it. To fix this problem, attempt the operation from a different host.
52	LUN not mapped to any host	Admin error	A LUN is not mapped to any host. In other words, it does not belong to a storage system initiator group. To be accessible, the LUN must be mapped to the current host outside SnapDrive for UNIX.

Exit value	Error name	Type	Description
53	LUN not mapped to local host	Admin error	A LUN is not mapped to the current host. In other words, it does not belong to a storage system initiator group that includes initiators from the current host. To be accessible, the LUN must be mapped to the current host outside SnapDrive for UNIX.
54	LUN is mapped using foreign igroup	Admin error	A LUN is mapped using a foreign storage system initiator group. In other words, it belongs to a storage system igroup containing only initiators not found on the local host. As a result, SnapDrive for UNIX cannot delete the LUN. To use SnapDrive for UNIX to delete a LUN, the LUN must belong only to local igroups; that is, igroups containing only initiators found on the local host.
55	LUN is mapped using mixed igroup	Admin error	A LUN is mapped using a mixed storage system initiator group. In other words, it belongs to a storage system igroup containing both initiators found on the local host and initiators not found there. As a result, SnapDrive for UNIX cannot disconnect the LUN. To use SnapDrive for UNIX to disconnect a LUN, the LUN must belong only to local igroups or foreign igroups; not mixed igroups. (Local igroups contain only initiators found on the local host; foreign igroups contain initiators not found on the local host.)

Exit value	Error name	Type	Description
56	Snapshot copy restore failed	Admin error	SnapDrive for UNIX attempted a Snapshot restore operation, but it failed without restoring any LUNs in the Snapshot copy. The specific error message describes the problem and the steps you need to perform to resolve it. Fix the problem and then repeat the command.
58	Host reboot needed	Admin error	The host operating system requires a reboot in order to update internal data. SnapDrive for UNIX has prepared the host for this update, but cannot complete the current operation. Reboot the host and then re-enter the SnapDrive for UNIX command line that caused this message to appear. After the reboot, the operation will be able to complete.
59	Host, LUN preparation needed	Admin error	The host operating system requires an update to internal data in order to complete the current operation. This update is required to allow a new LUN to be created. SnapDrive for UNIX cannot perform the update, because automatic host preparation for provisioning has been disabled because the snapdrive.conf variable `enable-implicit-host-preparation` is set to "off". (For more information, see "Setting configuration information" on page 84.) With automatic host preparation disabled, you should use either the `snapdrive config prepare luns` command to prepare the host to provision LUNs or perform the preparation steps manually.

Exit value	Error name	Type	Description
			To avoid this error message, set the `enable-implicit-host-preparation` value to "on" in the snapdrive.conf file. **Note** Currently, only the Linux and Solaris platforms require host preparation.
61	Cannot support persistent mount	Command error	For Linux hosts only, an error occurred because a `snapdrive snap connect` command, `snapdrive storage connect` command, or `snapdrive host connect` command requested a persistent mount for a file system that is not on a partitioned LUN.
62	Not empty	Command error	An error occurred because SnapDrive for UNIX could not remove a storage system volume or directory. This may happen when another user or another process creates a file at exactly the same time and in the same directory that SnapDrive tries to delete. To avoid this error, make sure that only one user works with the storage system volume at the time.
63	Timeout expired	Command error	An error occurred because SnapDrive for UNIX could not restore a LUN within the time-out period of 50 minutes. Record the message, and send it to NetApp technical support so that they can help you determine which steps to perform to complete the recovery.
64	Service not running	Admin error	An error occurred because a SnapDrive for UNIX command specified an NFS entity and the storage system was not running the NFS service.

Exit value	Error name	Type	Description
126	Unknown error	Admin error	An unknown error occurred that might be serious. Run the snapdrive.dc utility and send its results to NetApp technical support for analysis.
127	Internal error	Fatal	A SnapDrive for UNIX internal error occurred. Run snapdrive.dc and send its results to NetApp technical support for analysis.

Command Reference 9

About this chapter

This chapter contains a list of the commands supported in SnapDrive for UNIX and information about the options, keywords, and arguments that work with them.

Topics in this chapter

This chapter discusses the following topics:

- "Collecting information needed by SnapDrive for UNIX commands" on page 340
- "Summary of the SnapDrive for UNIX commands" on page 341
- "SnapDrive for UNIX options, keywords, and arguments" on page 348

Collecting information needed by SnapDrive for UNIX commands

Collecting information needed by commands

This chapter provides checklists you can use as you execute SnapDrive for UNIX. For each command, it supplies the following:

- Recommended usage formats
- Information on the keywords, options, and arguments available with the commands and the values you should supply
- Examples of the commands

These checklists provide a quick overview of using these commands. For more information on the commands, see the Chapter 7, "Creating and Using Snapshot Copies," on page 235 and the Chapter 6, "Provisioning and Managing Storage," on page 159.

General notes about the commands

The following are general notes about the commands:

- The `-dg` and `-vg` options in the command lines are synonyms that reflect the fact that some operating systems refer to disk groups and others refer to volume groups. This guide uses `-dg` to refer to both disk groups and volume groups.
- The `-lvol` and `-hostvol` options in the command lines are synonyms that reflect the fact that some operating systems refer to logical volumes and others refer to host volumes. This guide uses `-hostvol` to refer to both logical volumes and host volumes.
- NetApp strongly recommends that you use the default igroup and not specify an igroup explicitly by including the `-igroup` option on your command line. For information on how SnapDrive for UNIX handles igroup names, see "Command-line keywords" on page 357. If you need to specify an igroup, see the SnapDrive for UNIX man page for information on doing that.

Summary of the SnapDrive for UNIX commands

Command summary

SnapDrive for UNIX supports the following command lines:

Configuration command lines: The following table gives various command-line options for configuration operations.

`snapdrive config access {show | list} filername`
`snapdrive config check luns`
`snapdrive config delete filername [filername...]`
`snapdrive config list`
`snapdrive config prepare luns -count count [-devicetype {shared | dedicated}]`
`snapdrive config set user_name filername [filername...]`
`snapdrive config show [host_file_name]`
`snapdrive config check cluster`

Storage provisioning command lines: The following table gives various command-line options for storage provisioning operations.

Operation	Command-line option
Create	`snapdrive storage create -lun` *`long_lun_name`* `[`*`lun_name`* `...] -lunsize size [{ -dg | -vg } `*`dg_name`*`] [-igroup `*`ig_name`* `[`*`ig_name`* `...]] [{ -reserve | -noreserve }][-fstype `*`type`*`] [-vmtype `*`type`*`]`
	`snapdrive storage create {-lvol | -hostvol}` *`file_spec`* `[{-dg | -vg} dg_name] {-dgsize | -vgsize} size -filervol long_filer_path [-devicetype {shared | dedicated}] [{-noreserve | -reserve}] [-fstype `*`type`*`] [-vmtype `*`type`*`]`
	`snapdrive storage create -fs` *`file_spec`* `-nolvm [-fsopts options] [-mntopts options] [-nopersist] { -lun` *`long_lun_name`* `| -filervol` *`long_filer_path`* `} -lunsize size [-igroup` *`ig_name`* `[`*`ig_name`* `...]] [{ -reserve | -noreserve }] [-devicetype {shared | dedicated}] [-fstype `*`type`*`] [-vmtype `*`type`*`]`
	`snapdrive storage create` *`host_lvm_fspec`* `-filervol` *`long_filer_path`* `-dgsize size [-igroup` *`ig_name`* `[`*`ig_name`* `...]] [{ -reserve | -noreserve }] [-devicetype {shared | dedicated}]`

	`snapdrive storage create` *`host_lvm_fspec`* `-lun` *`long_lun_name`* `[`*`lun_name`* `...] -lunsize size [-igroup` *`ig_name`* `[`*`ig_name`* `...]] [{ -reserve | -noreserve }]`
	Note
	You can use one of the following three formats for the *-file_spec* argument, depending on the type of storage you want to create. (Remember that `-dg` is a synonym for `-vg`, and `-hostvol` is a synonym for `-lvol`.)
	To create a file system directly on a LUN, use this format:
	`-fs` *`file_spec`* `[-nolvm -fs` *`type`*`] [-fsops` *`options`*`] [-mntopts` *`options`*`] [-vmtype` *`type`*`]`

	To create a file system that uses a disk group or host volume, use this format:
	`-fs file_spec [-fstype type] [-fsopts options] [-mntops options] [-hostvol file_spec] [-dg dg_name] [-vmtype type]`
	To create a logical or host volume, use this format:
	`[-hostvol file_spec] [-dg dg_name] [-fstype type] [-vmtype type]`
	To create a disk group, use this format:
	`-dg dg_name [-fstype type] [-vmtype type]`

Connect	`snapdrive storage connect -fs file_spec -nolvm -lun long_lun_name [-igroup ig_name [ig_name ...]] [-nopersist] [-mntopts options] [-devicetype {shared | dedicated}] [-fstype type] [-vmtype type]`
	`snapdrive storage connect -fs file_spec -hostvol file_spec -lun long_lun_name [lun_name ...] [-igroup ig_name [ig_name ...]][-nopersist] [-mntopts options] [-devicetype {shared | dedicated}] [-fstype type] [-vmtype type]`
	`snapdrive storage connect -lun long_lun_name [lun_name ...][-igroup ig_name [ig_name ...]]`
	`snapdrive storage connect -lun long_lun_name [lun_name...][-devicetype {shared | dedicated}]`
	`snapdrive storage connect -fs file_spec {-hostvol | -lvol} file_spec -lun long_lun_name [lun_name...] [-devicetype {shared | dedicated}] [-nopersist] [-mntopts options] [-fstype type] [-vmtype type]`

Disconnect	snapdrive storage disconnect -lun *long_lun_name* [*lun_name*...] [-devicetype {shared | dedicated}]
	snapdrive storage disconnect {-vg | -dg | -fs | -lvol | -hostvol} *file_spec* [*file_spec* ...] [{-vg | -dg | -fs | -lvol | -hostvol} *file_spec* ...] ...] [-full] [-devicetype {shared | dedicated}] [-fstype *type*] [-vmtype *type*]
Resize	snapdrive storage resize {-dg | -vg} *file_spec* [*file_spec* ...]{-growby | -growto} size [-addlun [-igroup *ig_name* [*ig_name* ...]]] [{ -reseserve | -noreserve }]] [-fstype *type*] [-vmtype *type*]
Show	snapdrive storage { show | list } -filer *filername* [*filername* ...] [-verbose] [-quiet]
	snapdrive storage { show | list } -filervol *long_filer_path* [*filer_path*...][-verbose] [-quiet]
	snapdrive storage { show | list } {-all | device} [-devicetype {shared | dedicated}]
	snapdrive storage show [-verbose] {-filer *filername* [*filername*...] | -filervol volname [volname...]} [-devicetype {shared | dedicated}]
	snapdrive storage { show | list } -lun *long_lun_name* [*lun_name* ...] [-verbose] [-quiet] [-status]
	snapdrive storage { show | list } { -vg | -dg | -fs | -lvol |-hostvol } *file_spec* [*file_spec* ...] [{ -vg | -dg | -fs | -lvol | -hostvol } *file_spec* [*file_spec* ...]] [-verbose] [-quiet] [-fstype *type*] [-vmtype *type*] [-status]

Delete	snapdrive storage delete -lun *long_lun_name* [*lun_name*...] [-devicetype {shared | dedicated}]
	snapdrive storage delete { -vg | -dg | -fs | -lvol | -hostvol } *file_spec* [*file_spec* ...] [{ -vg | -dg | -fs | -lvol | -hostvol } *file_spec* [*file_spec* ...]] [-full] [-devicetype {shared | dedicated}] [-fstype *type*] [-vmtype *type*]

Host-side command lines: The following table gives various command-line options for host-side operations.

Operation	Command-line options
Host connect	snapdrive host connect -lun *long_lun_name* [*lun_name* ...]
	snapdrive host connect -fs *file_spec* -nolvm -lun *long_lun_name* [-nopersist][-mntopts options] [-fstype *type*] [-vmtype *type*]
	snapdrive host connect -fs *file_spec* -hostvol file_spec -lun *long_lun_name* [*lun_name*][-nopersist] [-mntopts options]
Host disconnect	snapdrive host disconnect -lun *long_lun_name* [*lun_name*...]
	snapdrive host disconnect -fs *file_spec* [-fstype *type*] [-vmtype *type*]
	snapdrive host disconnect {-vg | -dg | -fs | -lvol | -hostvol} *file_spec* [*file_spec* ...] [{-vg | -dg | -fs | -lvol | -hostvol} *file_spec* *[file_spec* ...]...] [-full] [-fstype *type*] [-vmtype *type*]

Snapshot operation command lines: The following table gives various command-line options for Snapshot operations.

Create	snapdrive snap create { -lun | -vg | -dg | -fs | -lvol | -hostvol } *file_spec* [*file_spec ...*] [{ -lun | -vg | -dg | -fs | -lvol | -hostvol } *file_spec* [*file_spec...*]] -snapname *snap_name* [-force [-noprompt]] [-unrelated] [-devicetype {shared | dedicated}] [-fstype *type*] [-vmtype *type*]
Show	snapdrive snap { show | list } -filer *filername* [*filername...*][-verbose]
	snapdrive snap { show | list } -filervol filervol [filervol...][-verbose]
	snapdrive snap { show | list } {-lun |-vg | -dg | -fs | -lvol | -hostvol } *file_spec* [*file_spec ...*] [-verbose]
	snapdrive snap { show | list } -snapname *long_snap_name* [snap_name ...] [-verbose]
Connect	snapdrive snap connect -lun *s_lun_name d_lun_name* [[-lun] *s_lun_name d_lun_name ...*] -snapname *long_snap_name* [-igroup *ig_name* [*ig_name ...*]] [-split] **Note** In a `snapdrive snap connect` command, the LUN name should be in the format lun_name or qtree_name/lun_name.
	snapdrive snap connect *fspec_set* [*fspec_set...*] -snapname *long_snap_name* [-igroup *ig_name* [*ig_name ...*]] [-autoexpand] [-autorename] [-nopersist] [{-reserve | -noreserve}] [-readonly] [-split] **Note** The *fspec_set* argument has the following format: {-lun | -vg | -dg | -fs | -lvol | -hostvol} *src_fspec* [*dest_fspec*] {-destdg | -destvg } dg_name] [{-destlv | -desthv } *lv_name*]
Rename	snapdrive snap rename -snapname *old_long_snap_name new_snap_name* [-force [-noprompt]]

Restore	snapdrive snap restore -snapname *snap_name* { -lun | -vg | -dg | -fs | -lvol | -hostvol | -file } *file_spec* [*file_spec ...*] [{ -lun | -vg | -dg | -fs | -lvol | -hostvol | -file } *file_spec* [*file_spec...*]...] [-force [-noprompt]] [{-reserve |-noreserve}] [-devicetype {shared | dedicated}]
Disconnect	snapdrive snap disconnect -lun *long_lun_name* [*lun_name...*] [-devicetype {shared | dedicated}] [-split]
	snapdrive snap disconnect {-vg | -dg | -fs | -lvol | -hostvol} *file_spec* [*file_spec ...*] [-vg | -dg | -fs | -lvol | -hostvol} *file_spec* [*file_spec ...*]] [-full] [-devicetype {shared | dedicated}] [-fstype *type*] [-vmtype *type*] [-split]
Delete	snapdrive snap delete [-snapname] *long_snap_name* [*snap_name...*][-verbose] [-force [-noprompt]]

SnapDrive for UNIX options, keywords, and arguments

Command-line options

SnapDrive for UNIX enables you to include the following options as appropriate with its commands. In certain cases, you can abbreviate these options. For example, you can use `-h` instead of `-help`.

Option	Description
`-addlun`	Tells SnapDrive for UNIX to add a new, internally-generated LUN to a storage entity in order to increase its size.
`-all`	Used with the `snapdrive storage {show | list}` command to display all devices and LVM entities known to the host.
`-autoexpand`	Used with the `snapdrive snap connect` command to enable you to request that a disk group be connected when you supply a subset of the logical volumes or file systems in the disk group.
`-autorename`	Used with the `snapdrive snap connect` command to enable the command to rename any newly-connected LVM entities for which the default name is already in use.
`-device`	Used with the `storage {show | list}` command to display all devices known to the host.

Option	Description
-devicetype	Specifies the type of device to be used for SnapDrive for UNIX operations. Following are the values for this option: ◆ shared— specifies the scope of LUN, disk group, and file system as cluster-wide. ◆ dedicated—specifies the scope of LUN, disk group, and file system as local. This is the default value. If you do not specify the -devicetype option in SnapDrive for UNIX commands that supports this option, it is equivalent to specifying -devicetype dedicated.
-dgsize (synonymous with -vgsize)	Used with the snapdrive storage create command to specify the size in bytes of the disk group you want to create.
-force (or -f)	Causes operations to be attempted that SnapDrive for UNIX would not undertake ordinarily. SnapDrive for UNIX prompts you to ask for confirmation before it executes the operation.
-fsopts	The options you want passed to the host operation that creates the new file system. Depending on your host operating system, this host operation might be a command such as the mkfs command. The argument you supply with this option usually needs to be specified as a quoted string and must contain the exact text to be passed to the command. For example, you might enter -o largefiles as the option you want passed to the host operation.

Option	Description
`-fstype`	The type of file system you want to use for the SnapDrive for UNIX operations. The file system must be a type that SnapDrive for UNIX supports for your operating system. Current values that you can set for this variable are as follows: ◆ AIX: “jfs”, “jfs2” or “vxfs” The default value is “jfs2”. ◆ HP-UX: “vxfs” ◆ Linux: “ext3” ◆ Solaris: “vxfs” or “ufs” The default value is “vxfs”. You can also specify the type of file system that you want to use by using the `-fstype` configuration variable. For more information on this variable, see Chapter 4, “Determining options and their default values,” on page 86.
`-full`	Allows operations on a specified host-side entity to be performed even if the entity is not empty (for example, the entity might be a volume group containing one or more logical volumes).
`-growby`	The number of bytes you want to add to a LUN or disk group in order to increase its size.
`-growto`	The target size in bytes for a LUN, disk group, or volume group. SnapDrive for UNIX automatically calculates the number of bytes necessary to reach the target size and increases the size of the object by that number of bytes.

Option	Description

-help	Prints out the usage message for the command and operation. Enter this option by itself without other options. Following are the examples of possible command lines.
	
	Example 1: In this example, the help option is executed with the SnapDrive for UNIX command:
	
	`# snapdrive help`
	`snapdrive: For detailed syntax of individual commands, type`
	`snapdrive: 'snapdrive command operation help'`
	`snapdrive: Supported commands and operations are:`
	`snapdrive snap show`
	`snapdrive snap list`
	`snapdrive snap create`
	`snapdrive snap delete`
	`snapdrive snap rename`
	`snapdrive snap connect`
	`snapdrive snap disconnect`
	`snapdrive snap restore`
	`snapdrive storage show`
	`snapdrive storage list`
	`snapdrive storage create`
	`snapdrive storage delete`
	`snapdrive storage resize`
	`snapdrive storage connect`
	`snapdrive storage disconnect`
	`snapdrive host connect`
	`snapdrive host disconnect`
	`snapdrive version`
	`snapdrive config access`
	`snapdrive config prepare`
	`snapdrive config check`
	`snapdrive config show`
	`snapdrive config set`
	`snapdrive config delete`
	`snapdrive config list`

Option	Description
-help... continued	**Example 2:** In this example, the help option is executed in a SnapDrive operation: `# snapdrive  snap create help` `Usage: snapdrive snap create {-lun	-dg	-vg	-hostvol	-lvol	-fs} file_spec [file_spec ...][{-lun	-dg	-vg	-hostvol	-lvol	-fs} file_spec [file_spec ...] ...] -snapname snap_name [-force [-noprompt]] [-unrelated]` `Examples:` `snapdrive snap create -dg dg_name1 dg_name2 -snapname snap_name` `snapdrive snap create -fs /fs_mount_dir -lvol logical_volname -snapname snap1`  **Example 3:** In this example, an incomplete SnapDrive for UNIX operation is being executed without any parameters. In such a scenario, SnapDrive for UNIX provides a usage message for that operation:  `# snapdrive snap create` `Usage: snapdrive snap create {-lun	-dg	-vg	-hostvol	-lvol	-fs} file_spec [file_spec ...][{-lun	-dg	-vg	-hostvol	-lvol	-fs} file_spec [file_spec ...] ...] -snapname snap_name [-force [-noprompt]] [-unrelated]` `Examples:` `snapdrive snap create -dg dg_name1 dg_name2 -snapname snap_name` `snapdrive snap create -fs /fs_mount_dir -lvol logical_volname -snapname snap1`
-lunsize	The size of the LUN in bytes to be created by a given command.

Option	Description
-mntopts	Specifies options that you want passed to the host mount command (for example, to specify file system logging behavior). Options are also stored in the host file system table file. The options allowed depend on the host file system type. The -mntopts argument that you supply is a file system-type option that is specified using the mount command "- o" flag. Do not include the "- o" flag in the -mntopts argument. For example, the sequence -mntopts tmplog passes the string -o tmplog to the mount command line, and inserts the text "tmplog" on a new command line.
-nofilerfence	Suppresses the use of the Data ONTAP consistency group feature in creating Snapshot copies that span multiple filer volumes. In Data ONTAP 7.2, you can suspend access to an entire filer volume. By using the -nofilerfence option, you can freeze access to an individual LUN.
-nolvm	Connects or creates a file system directly on a LUN without involving the host LVM. All commands that take this option for connecting or creating a file system directly on a LUN will not accept it for cluster-wide or shared resources. This option is allowed only for local resources. If you have enabled the -devicetype shared option, then this option cannot be used, because -nolvm option is valid only for local resources and not for shared resources.

Option	Description
`-nopersist`	Connects or creates a file system, or a Snapshot copy that has a file system, without adding an entry in the host's persistent mount entry file.
`-reserve` `-noreserve`	Used with the `snapdrive storage create` or `snapdrive snap connect` commands to specify whether or not SnapDrive for UNIX creates a space reservation. By default, SnapDrive for UNIX creates reservation for storage create, resize, and Snapshot create operations, and does not create reservation for Snapshot connect operation.
`-noprompt`	Suppresses prompting during command execution. By default, any operation that might have dangerous or non-intuitive side effects prompts you to confirm that SnapDrive for UNIX should be attempted. This option overrides that prompt; when combined with the `-force` option, SnapDrive for UNIX performs the operation without asking for confirmation.
`-quiet (or -q)`	Suppresses the reporting of errors and warnings, regardless of whether they are normal or diagnostic. It returns zero (success) or non-zero status. The `-quiet` option overrides the `-verbose` option. This option will be ignored for `snapdrive storage show`, `snapdrive snap show`, and `snapdrive config show` commands.

Option	Description
`-readonly`	Required for configurations with Data ONTAP 6.5 or any configuration that uses traditional volumes. Connects the NFS file or directory with read-only access. Optional for configurations with Data ONTAP 7.0 that use FlexVol volumes. Connects the NFS file or directory tree with read-only access. (Default is read/write).
`-split`	Enables to split the cloned volumes or LUNs during Snapshot connect and Snapshot disconnect operations. You can also split the cloned volumes or LUNs by using the `enable-split-clone` configuration variable. For more information on this variable, see Chapter 4, "Determining options and their default values," on page 86.
`-status`	Used with the `snapdrive storage show` command to know if the volume or LUN is cloned.
`-unrelated`	Creates a Snapshot copy of *file_spec* entities that have no dependent writes when the Snapshot copy is taken. Because the entities have no dependent writes, SnapDrive for UNIX creates a crash-consistent Snapshot copy of the individual storage entities, but does not take steps to make the entities consistent with each other.
`-verbose` (or `-v`)	Displays detailed output, where appropriate. All commands and operations accept this option, although some might ignore it.
`-vgsize` (synonymous with `-dgsize`)	Used with the `storage create` command to specify the size in bytes of the volume group you want to create.

Option	Description
`-vmtype`	The type of volume manager you want to use for the SnapDrive for UNIX operations. If the user specifies the `-vmtype` option in the command line explicitly, SnapDrive for UNIX uses the value specified in the option irrespective of the value specified in the `vmtype` configuration variable. If the`-vmtype` option is not specified in the command-line option, SnapDrive for UNIX uses the volume manager that is in the configuration file. The volume manager must be a type that SnapDrive for UNIX supports for your operating system. Current values that you can set for this variable are as follows: ◆ AIX and HP-UX: “vxvm” or “lvm” The default value is “lvm”. ◆ Linux: “lvm” ◆ Solaris: “vxvm” You can also specify the type of volume manager that you want to use by using the `vmtype` configuration variable. For more details, see Chapter 4, “Determining options and their default values,” on page 86.

Rules for keywords

SnapDrive for UNIX uses keywords to specify sequences of strings corresponding to the host and storage system objects with which you are working. The following rules apply to SnapDrive for UNIX keywords:

- Precede each keyword with a hyphen (-).
- Do not concatenate keywords.
- Enter the entire keyword and hyphen, not an abbreviation.

Command-line keywords

Here are the keywords you can use with the SnapDrive for UNIX commands. You use them to specify the targets of the SnapDrive for UNIX operations. These keywords can take one or more arguments.

Note

Some LVMs refer to *disk groups* and some refer to *volume groups*. In SnapDrive for UNIX, these terms are treated as synonyms. Moreover, some LVMs refer to *logical volumes* and some refer to *volumes*. SnapDrive for UNIX treats the term *host volume* (which was created to avoid confusing *host logical volumes* with *storage system volumes*) and the term *logical volume* as synonymous.

Keyword	Argument used with this keyword
`-dg` (synonymous with `-vg`)	The name of the host disk group. You can enter the name of either a disk group or a volume group with this option.

`-destdg`
`-desthv`
`-destlv`
`-destvg`
	The destination group or volume.

`-destfv`	The name of the FlexClone volume specified on the command line for volume clones created by SnapDrive for UNIX during the NFS Snapshot connect operation. **Note** This argument supports NFS volumes only and not NFS directories. **Example:** If a `snapdrive snap connect` command contains three FlexClone volumes, this argument can be used to rename only one or more FlexClone volumes. In this example, the argument is used to rename only one FlexClone volume: `snapdrive snap connect -fs/mnt1/pt1/ -fs/mnt2/pt2 -fs/mnt3/pt3 -destfv filer:/vol/<vol-name> <dest-vol-name> -snapname`

Keyword	Argument used with this keyword
`-file`	The name of a NFS file.
`-filer`	The name of a storage system.
`-filervol`	The name of the storage system and a volume on it.
`-fs`	The name of a file system on the host. The name used is the directory where the file system is currently mounted or is to be mounted (the mountpoint).
`-hostvol` (synonymous with `-lvol`)	The host volume name, including the disk group that contains it. For example, you might enter large_vg/accounting_lvol. **Note** In Solaris Volume Manager, the host volume name format should be in dx—dxxxx format, where x denotes the maximum number of volumes that is supported. On Solaris 10, x varies from 0 to 8191, and on Solaris 9, x varies from 0 to 127. For more information on host volume name format, see the *Solaris Volume Manager Administration Guide*.

Keyword	Argument used with this keyword
`-igroup`	The name of an initiator group (igroup). NetApp strongly recommends that you use the default igroup that SnapDrive for UNIX creates instead of specifying an igroup on the target storage system. The default igroup is hostname_protocol_SdLg. ◆ *hostname* is the local (non-domain qualified) name of the current host. ◆ *protocol* is either `fcp` or `iscsi`, depending which protocol the host is using. If the igroup hostname_protocol_SbIg does not exist, SnapDrive for UNIX creates it and places all the initiators for the host in it. If it exists and has the correct initiators, SnapDrive for UNIX uses the existing igroup. If the igroup exists, but does not contain the initiators for this host, SnapDrive for UNIX creates a new igroup with a different name and uses that igroup in the current operation. To avoid using the same name, SnapDrive for UNIX includes a unique number when it creates the new name. In this case, the name format is hostname-number_protocol_SdIg. If you supply your own igroup name, SnapDrive for UNIX does not validate the contents of the igroup. This is because it cannot always determine which igroups are on the hosts. All commands that take this option for specifying initiator groups will not accept it with shared disk groups and file systems. This option is allowed only for local resources.

Keyword	Argument used with this keyword
	If you have enabled `-devicetype shared` option, then this option cannot be used, because `-igroup` option is valid only for local resources and not for shared resources. For details on specifying igroups, see the SnapDrive for UNIX man page. The SnapDrive for UNIX command fails if any foreign igroups are involved in the command line. Ensure that all the igroups specified in the command line contain initiators from the local host.
`-lun`	The name of a LUN on a storage system. For the first LUN name you supply with this keyword, you must supply the full path name (storage system name, volume name, and LUN name). For additional LUN names, you can specify either only the names within their volume (if the volume stays unchanged) or a path to indicate a new storage system name or a new volume name (if you just want to switch volumes). **Note** In a `snapdrive snap connect` command, the lun_name should be in the lun_name or tree_name/lun_name format.
`-lvol` (synonymous with `-hostvol`)	The logical volume name, including the volume group that contains it. For example, you might enter large_vg/accounting_lvol as the logical volume name.
`-snapname`	The name of a Snapshot copy.
`-vg` (synonymous with `-dg`)	The name of the volume group. You can enter the name of either a disk group or a volume group with this option.

Command-line arguments

The following table describes the arguments you can specify with the keywords.

Use the format `snapdrive type_name operation_name [<keyword/option> <arguments>]`; for example, if you wanted to create a Snapshot copy called snap_hr from the host file system /mnt/dir, you would enter the following command line:

```
snapdrive snap create -fs /mnt/dir -snapname snap_hr
```

Argument	Description
dest_fspec	The name by which the target entity will be accessible after its disk groups or LUNs are connected.
dgname	The name of a disk group or volume group.
d_lun_name	Allows you to specify a destination name that SnapDrive for UNIX uses to make the LUN available in the newly-connected copy of the Snapshot copy.
filername	The name of a storage system.
filer_path	A path name to a storage system object. This name can contain the storage system name and volume, but it does not have to if SnapDrive for UNIX can use default values for the missing components based on values supplied in the previous arguments. The following are examples of path names: test_filer:/vol/vol3/qtree_2 /vol/vol3/qtree_2 qtree_2

Argument	Description

file_spec	The name of a storage entity, such as a host volume, LUN, disk or volume group, file system, or NFS directory tree.
	
	In general, you use the *file_spec* argument as one of the following:
	◆ An object you want SnapDrive for UNIX to make a Snapshot copy of or to restore from a Snapshot copy
	◆ An object that you want to either create or use when provisioning storage
	
	The objects do not have to be all of the same type. If you supply multiple host volumes, they must all belong to the same volume manager.
	
	If you supply values for this argument that resolve to redundant disk groups or host volumes, the command fails.
	
	Example of incorrect usage: This example assumes dg1 has host volumes hv1 and hv2, with file systems fs1 and fs2. As a result, the following arguments would fail because they involve redundant disk groups or host volumes.
	
	`-dg dg1 -hostvol dg1/hv1`
	
	`-dg dg1 -fs /fs1`
	
	`-hostvol dg1/hv1 -fs /fs1`
	
	Example of correct usage: This example shows the correct usage for this argument.
	
	`-hostvol dg1/hv1 dg1/hv2`
	
	`-fs /fs1 /fs2`
	
	`-hostvol dg1/hv1 -fs /fs2`

Argument	Description
fspec_set	Used with the `snap connect` command to identify a ◆ A host LVM entity ◆ A file system contained on a LUN The argument also lets you specify a set of destination names that SnapDrive for UNIX uses when it makes the entity available in the newly connected copy of the Snapshot copy. The format for *fspec_set* is: `{ -vg | -dg | -fs | -lvol | -hostvol } src_fspec [dest_fspec] [{ -destdg | -destvg } dg_name] [{ -destlv | -desthv } lv_name]`

host_lvm_fspec	Lets you specify whether you want to create a file system, logical volume, or disk group when you are executing the `storage create` command. This argument may have any of the three formats shown below. The format you use depends on the entity you want to create.
	Note
	The `-dg` and `-vg` options are synonyms that reflect the fact that some operating systems refer to disk groups and others refer to volume groups. In addition, `-lvol` and `-hostvol` are also synonyms. This guide uses `-dg` to refer to both disk groups and volume groups and `-hostvol` to refer to both logical volumes and host volumes.
	To create a file system, use this format:
	`-fs file_spec [-fstype type] [-fsopts options] [-hostvol file_spec] [-dg dg_name]`
	To create a logical or host volume, use this format:
	`[-hostvol file_spec] [-dg dg_name] | -hostvol`
	To create a disk or volume group, use this format:
	`file_spec [-dg dg_name] | -dg dg_name`

Argument	Description
	You must name the top-level entity that you are creating. You do not need to supply names for any underlying entities. If you do not supply names for the underlying entities, SnapDrive for UNIX creates them with internally generated names. If you specify that SnapDrive for UNIX create a file system, you must specify a type that SnapDrive for UNIX supports with the host LVM. These types include the following: ◆ AIX: JFS2 or VxFS The default file system type is JFS2. **Note** The JFS file system type is supported only for Snapshot operations and not for storage operations. ◆ HP-UX: VxFS ◆ Linux: Ext3 ◆ Solaris: VxFS or UFS The option `-fsopts` is used to specify options to be passed to the host operation that creates the new file system; for example, *mkfs*.
ig_name	The name of an initiator group.

long_filer_path	A path name that includes the storage system name, volume name, and possibly other directory and file elements within that volume. The following are examples of long path names:
	`test_filer:/vol/vol3/qtree_2`
	`10.10.10.1:/vol/vol4/lun_21`

long_lun_name	A name that includes the storage system name, volume, and LUN name. The following is an example of a long LUN name: `test_filer:/vol/vol1/lunA`

Argument	Description

long_snap_name	A name that includes the storage system name, volume, and Snapshot copy name. The following is an example of a long Snapshot copy name:
	`test_filer:/vol/account_vol:snap_20040202`
	With the `snapdrive snap show` and `snapdrive snap delete` commands, you can use the asterisk (*) character as a wildcard to match any part of a Snapshot copy name. If you use a wildcard character, you must place it at the end of the Snapshot copy name. SnapDrive for UNIX displays an error message if you use a wildcard at any other point in a name.
	Example: This example uses wildcards with both the `snap show` command and the `snap delete` command:
	`snap show myfiler:/vol/vol2:mysnap* myfiler:/vol/vol2:/yoursnap*`
	`snap show myfiler:/vol/vol1/qtree1:qtree_snap*`
	`snap delete 10.10.10.10:/vol/vol2:mysnap* 10.10.10.11:/vol/vol3:yoursnap* hersnap`
	Limitation for wildcards: You cannot enter a wildcard in the middle of a Snapshot copy name. For example, the following command line produces an error message because the wildcard is in the middle of the Snapshot copy name:
	`banana:/vol/vol1:my*snap`

lun_name	The name of a LUN. This name does not include the storage system and volume where the LUN is located. The following is an example of a LUN name: `lunA`
path	Any path name.
s_lun_name	Indicates a LUN entity that is captured in the Snapshot copy specified by *long_snap_name*.

Argument	Description

short_snap_name	A name that consists of only the Snapshot copy name. You can use the short form of the Snapshot copy name when you do not need to include the storage system and storage system volume as part of the name. For example, when you create a Snapshot copy, the storage system and volume must be the same as those of the original data.
	
	You can use wildcard with the `snapdrive snap show` and `snapdrive snap delete` operations. If you use a wildcard, you must place it at the end of the Snapshot copy name. SnapDrive for UNIX displays an error message if you enter a wildcard in the middle of a Snapshot copy name.
	
	Example: This example uses wildcard with the `snapdrive snap show` command:
	
	`snap show toaster:/vol/vol1:mysnap*`
	
	Limitations for wildcards: You cannot enter a wildcard in the middle of a Snapshot copy name. For example, the following Snapshot copy name is incorrect:
	
	`toaster:/vol/vol1:test*snap`

size	The size in bytes of an object such as a LUN or a disk group. You can abbreviate sizes; for example, you might use either “2 g” or “500 m”.
snap_name	A Snapshot copy name in either the long or short form.
src_fspec	Identifies an LVM entity within the Snapshot copies you want to connect to a different location using `snapdrive snap connect` command. This name is used to locate the disk group within the Snapshot copy that needs to be connected in order to make the entity available.
type	Specifies the type of supported value you have to use for SnapDrive for UNIX operations. For example, you might use “jfs2” or “jfs” for the `-fstype` option on an AIX host.
volname	The name of the storage system volume.

Glossary

cluster

You can have a storage system cluster or a host cluster.

The standard storage system cluster involves having a pair of storage systems attached via fabric to a switch in such a way that one can serve its partner's data if the partner fails.

A host cluster refers to a host with multiple nodes. See the Network Appliance FCP Host Utilities (Attach Kit) or iSCSI Host Utilities (Support Kit) documentation for your operating system for information on the number of supported nodes on a host.

clustered failover (CFO)

This is a method that ensures data availability by transferring the data service of a failed storage system to another storage system in the cluster. Transfer of data service is often transparent to users and applications

disk group

A unit of storage that can contain one or more volumes and one or more file systems.

A set of LUNs managed by an LVM on the host. LUNs in a disk group are combined in stripes, mirrors, and other RAID combinations by the host LVM. These combinations can then be divided into host volumes, which can be used to store data directly (as if they were raw disks), or can be used to host a file system. The data on a disk group is generally spread over all LUNs in the disk group, making it difficult to restore only a single LUN's data. SnapDrive for UNIX uses the term *disk group* interchangeably with the term *volume group*.

failover

Failover refers to situations where a system component fails and another component takes over its functions while the system continues to operate. Failover can occur between storage system heads configured in a CFO cluster, among redundant FCP or iSCSI paths from the host to storage system LUNs, or from one clustered host to another.

In a clustered failover, an alternate system takes over and emulates the primary system if the primary system becomes unusable.

FCP initiator

See initiator.

file_spec

Any supported object, like a host volume, disk group, file system, NFS file, or directory tree, that SnapDrive for UNIX uses to create a Snapshot copy.

file system

A file system in the SnapDrive for UNIX context refers to the file organization system supported by NetApp on the host operating system.

host

A host is the machine running a UNIX operating system that accesses storage on a storage system.

host bus adapter (HBA)

A host bus adapter (HBA) refers to the adapters used to connect hosts and storage systems in a NetApp SAN so that hosts can access LUNs on the storage systems using FCP. *See also* LUN (logical unit number).

host utilities

Host Utilities is a kit that enables the host to connect to the NetApp storage systems. The kit consists of utilities, scripts, support files and document set.

host volume

A host volume exists within a logical volume manager. It is a subset of a disk group (volume group) that acts as a disk to applications accessing it. The data in a host volume can be distributed over all or a subset of the disks in the disk group to which it belongs. Use of a host volume allows the administrator to easily resize the amount of storage assigned to a file system or application, without having to change the *disk* the application is using. The term *host volume* is used interchangeably with *logical volume*. The term *volume*, which is often used in conjunction with host logical volume managers, is avoided in this document, to avoid confusion with storage system volumes.

initiator

The hardware that initiates data exchange.

Initiator group (igroup)

Initiators that are grouped as authorized to access particular targets.

There are three categories of igroups that apply to SnapDrive for UNIX:

- Foreign: This igroup contains only initiators not found on the local host.
- Local: This igroup contains only initiators found on the local host.
- Mixed. This igroup contains initiators that are found as well as not found on the local host.

logical volume

See host volume.

LUN (logical unit number)

LUN refers to a logical unit of storage identified by a number.

LUN ID

This is the numerical identifier for a LUN.

LVM storage entities

Disk groups with or without host volumes and file systems that are created using the LVM on the host.

network interface card (NIC)

A network interface card (NIC) refers to a Gigabit Ethernet (commonly known as GbE) or a Fast Ethernet card that is compliant with the IEEE 802.3 standards.

These cards can provide the following connectivity functions:

- Connect hosts and storage systems to a local area network (LAN)
- Connect hosts and storage systems to data-center switching fabrics, specifically, enable hosts to connect to LUNs on storage systems using iSCSI

raw data

Data that is passed to an I/O device without being interpreted or processed. Raw access to a disk device implies that it is being used directly by an application to store data, rather than having a file system on it and letting the application store data in files in the file system.

raw storage entities

LUNs or LUNs that contain file systems that are mapped directly to the host. Raw entities are created without using LVM.

rollback

This is a Snapshot copy of the data on the storage system that SnapDrive for UNIX makes before it begins a Snapshot restore operation. Having a rollback Snapshot copy means that, in the event of a problem with a Snapshot restore operation, you can restore the data on the storage system to the state it was in before the operation began. This option is enabled in the snapdrive.conf file by default (see "Setting values in snapdrive.conf" on page 118).

SAN (storage area network)

A SAN (storage area network) is a storage setup composed of one or more storage systems and connected to one or more hosts in an FCP or an iSCSI environment.

To a host running SnapDrive for UNIX, a connected SAN is just another target storage device within which SnapDrive for UNIX can create and manage LUNs.

SFRAC

Veritas Storage Foundation for Oracle Real Application Clusters (SFRAC). This software enables administrators of Oracle Real Application Clusters (RAC) to operate a database in an environment of systems in local or global clusters running Veritas Cluster Server™ (VCS) and the cluster features of Veritas Volume Manager™ and Veritas File System™, also known as CVM and CFS, respectively.

Snapshot copy

A Snapshot copy refers to the Data ONTAP Snapshot copy™ technology. This technology enables the recovery after accidental deletion or modification of the data stored on a NetApp storage system by referencing a point-in-time image of that data.

storage entity

The storage object, like a host volume, disk group, file system or NFS file or directory tree, that Snapshot copy uses to create a Snapshot copy.

storage system

A storage system (sometimes called a filer) is a NetApp storage system that supports the FCP (Fibre Channel Protocol), iSCSI, and/or GbE (Gigabit Ethernet) protocols.

storage system volume

A storage system volume is a functional unit of storage. It comprises a collection of physical disks. A volume can be composed of one or more RAID groups to ensure data integrity and availability if multiple disks fail simultaneously within the same volume. For more information about storage system volumes, see the Data ONTAP *Storage Management Guide*.

support kit

The support kit is the standard software and documentation NetApp supplies to allow you to connect your host to a NetApp storage system using an iSCSI protocol.

target port

For NetApp SANs, a target is the HBA port on the storage system used to receive the SCSI I/O commands that the host initiator port sends. *See also* Initiator.

vFiler unit

A virtual storage system you create using MultiStore, which enables you to partition the storage and network resources of a single storage system so that it appears as multiple storage systems on the network.

volume

See either host volume or storage system volume.

Index

Symbols

93, 106, 107, 108, 110, 113
<host>_info utility 298

A

access permissions 6
    control levels 147
    examples 152
    permissions file 147
    setting 147
    setting configuration option 89
    steps for setting 148
    steps for viewing 152
-addlun command option 348
AIX host
    example of successful installation 37
    example of successful uninstallation 44
    example of uninstall warning 45
    files installed 74
    installation steps for SnapDrive for UNIX 31
    requirements 30
    uncompressing downloaded file 31
    uninstalling SnapDrive for UNIX 40
    using SMIT 32
    verify installation 38
arguments
    d_lun_name 361
    dest_fspec 361
    destfv 357
    dgname 361
    file_spec 362
    filer_path 361
    filername 361
    fspec_set 363
    host_lvm_fspec 363
    ig_name 364
    long_filer_path 364
    long_lun_name 364
    long_snap_name 365
    lun_name 365
    path 365
    rules for using 361
    s_lun_name 365
    short_snap_name 366
    size 366
    snap_name 366
    src_fspec 366
    type 366
    volname 366
attach kit
    install 23
audit log file 89, 122
    disabling 123
    example 126
    maximum size 90
    number of old files 90
    specifying pathname 123
    using 125
audit-log-file option 89
audit-log-save option 90
-autoexpand command option 348
-autorename command option 348
AutoSupport
    configuration options 91
    example of a message 131
    setting up 132
    snapdrive.conf option 91
    using 130
autosupport-enabled option 91
autosupport-filer 91
available-lun-reserve option 92

B

bytes
    adding to storage 194

C

CD-ROM
    getting software 26
checklist
    general information 340
cluster

definition of 367
host 367
storage system 367
support in SnapDrive for UNIX 16
clustered failover
definition 367
command line mode
using 142
commands
-addlun option 348
arguments 361
-autoexpand option 348
-autorename option 348
config check luns 85
config delete 85
config list 84, 155
config prepare luns 85
config set 85
config show 84
-dg keyword 357
file_spec arguments 362
filer_path argument 361
-force option 349
-fsopts option 349
-fstype option 350
-full option 350
-growby option 350
-growto option 350
-help option 351
host connect 218
host disconnect 227
-hostvol keyword 358
-igroup keyword 359
keyword 357
keyword rules 356
long_filer_path argument 364
-lunsize option 352
-noprompt option 354
options 348
path argument 365
-quiet option 354
running from the command line 142
short_snap_name argument 366
snap create 246
snap delete 295
snap list 256
snap rename 258
snap show 251
snap_name argument 366
snapdrive config delete 156
snapdrive config set 154, 156
snapdrive version 119
-snapname keyword 360
storage show 181, 182, 185
summary of all commands 341
swverify 58
-verbose option 355
Compatibility and Configuration Guide for NetApp's FCP and iSCSI Products 12
config command
check luns 85
delete 85
list 84
options 84
prepare luns 85
set 85
show 84
config delete command
SnapDrive for UNIX 156
config list command 155
config list operation 84
config set command 154
SnapDrive for UNIX 156
config show operation 84
configuration options
all-access-if-rbac-unspecified 89
audit-log-file 89
audit-log-max-size option 90
audit-log-save 90
autosupport-enabled 91
autosupport-filer 91
available-lun-reserve 92
cluster-operation-timeout-secs=600 93
contact-http-port 93
contact-ssl-port 93
default-noprompt 93
default-transport 96
device-retries option 94
device-retry-sleep-secs 95
enable-implicit-host-preparation 97

filer-restore-retries 98
filer-restore-retry-sleep 99
filesystem-freeze-timeout 99
mgmt-retries 100
mgmt-retry-sleep 100
mgmt-sleep-long-secs 100
multipathing-type 101
password-file 103
path 103
prefix-filer-lun 104
prepare-lun-count 104
recovery-log-file 104
recovery-log-save 105
secure-communication-among-cluster-nodes 106
snapconnect-nfs-removedirectories=on 108
snapcreate-cg-timeout 107
snapcreate-check-nonpersistent-nfs 107
snapcreate-consistency-retries 109
snapcreate-consistency-retry-sleep 108
snapdelete-delete-rollback-with-snap 110
snapmirror-dest-multiple-filervolumes-enabled 110
snaprestore-delete-rollback-after-restore 110
snaprestore-make-rollback 111
snaprestore-must-make-rollback 112
snaprestore-must-make-snapinfo-on-qtree 109
snaprestore-snapmirror-check 112
space-reservations-enabled 113
trace-enabled 113
trace-level 114
trace-log-file 114, 115
trace-log-save 115
use-https-to-filer 116
viewing with config show 84
configuring
storage system volumes 21
connecting Snapshot copies
restrictions 274
contact-http-port-option 93
contact-ssl-port 93
conventions
formatting x
keyboard x
creating Snapshot copies
examples 248
steps for creating 246
using SnapDrive for UNIX 246
creating storage 163

D

data
raw 369
data collection utility
examples 300
executing 299
information collected 298
Data ONTAP
version 18
Data ONTAP Block Access Management Guide 13
default-noprompt option 93
default-transport option 96
deleting Snapshot copies
examples 296
reasons 294
restrictions 294
steps 295
depot file
HP-UX installation 48
device-retries option 94
device-retry-sleep-secs option 95
-dg 357
disconnecting Snapshot copies
examples 292
disconnecting storage 207, 213
disk group
definition 367
increasing size 192
displaying Snapshot copies
examples 252
steps for displaying 251
documentation
additional 12
Compatibility and Configuration Guide for NetApp's FCP and iSCSI Products 12
Data ONTAP Block Access Management Guide 13
FCP host attach kit 13
iSCSI host support kit 13

SnapDrive for UNIX Compatibility Matrix 12
SnapDrive for UNIX man page 12
SnapDrive for UNIX Quick Start 12
SnapDrive for UNIX Release Notes 12
storage system 18
System Configuration Guide 13

E

enable-implicit-host-preparation option 97
error messages 301
limit on open files 303
return codes 326
examples 296

F

failover
cluster 16
definition 367
FCP
license required 19
FCP host attach kit documentation 13
Fibre Channel Protocol *See* FCP
file system
creating 164
definition 368
file sytems
resizing 196
file_spec 362
filer
cluster 16
defined x
filer *See* storage system
filer volume *See* storage system volume
filer_path
example 361
filer-restore-retries option 98
filer-restore-retry-sleep option 99
FilerView
setting snap reserve 22
files
.rmp (Linux) 56, 63
<host>_info utlity 298
access permissions 147
audit log 89, 90, 122
audit log file 125, 126
audit-log-save 90
depot (HP-UX) 48
installed on hosts 74
limit open files 303
RBAC permissions 89
recovery log 105, 122
recovery log file 126, 128
sdhost-name.prbac 147
snapdrive.conf 84
snapdrive.dc 298
trace log 114, 115, 122
filesystem-freeze-timeout option 99
-force
command option 349
-fsopts
command option 349
-fstype
command option 350
-full command option 350

G

-growby
command option 350
growby
command option 194
-growto
command option 350
growto
command option 194
guidelines
creating storage system volumes 21

H

HAK (Host Attach Kit)
install 23
-help
command option 351
host
<host>_info utility 298
cluster 367
clustered 16
displaying storage information 182
limit on open files 303

non-originating 271
volume 368, 371
Host Attach Kit *See* HAK
Host Bus Adapter (HBA)
definition 368
host connect command
steps for using 218
host disconnect command
examples 227
steps for using 227
host entities
creating
storage create command 164
deleting 233
disconnecting LUNs 213
host volume
definition 368
hosts
AIX requirements 30
comparing platforms 9
differences in commands 341
differences in platforms 10
HP-UX requirements 46
installation steps on AIX host 31
installation steps on HP-UX host 48
installation steps on Linux host 56
installation steps on Solaris host 63
Linux requirements 54
preparing for LUNs 120
resizing 196
Solaris requirements 61
terminology differences 10
-hostvol 358
HP-UX host
depot file 48
example of successful installation 51
example of successful uninstall 53
files installed 74
http 146
installation steps for SnapDrive for UNIX 48
moving file to host 48
requirements 46
uninstalling SnapDrive for UNIX 52
HTTPS 5
disabling 157
HP-UX host 157

I

-igroup keyword 359
increasing storage size 192
installing
completing the installation 73
steps for AIX host 31
steps for HP-UX host 48
steps for Linux host 56
steps for Solaris host 63
upgrading SnapDrive for UNIX 79
verify 58
IP address
partner 19
iSCSI
license required 19
iSCSI host suppor kit documentation 13

K

keywords 357
-destdg 357
-desthv 357
-destlv 357
-destvg 357
-dg 357
-file 358
-filer 358
-filervol 358
-fs 358
-hostvol 358
-igroup 359
-lun 360
-lvol 360
rules for using 356
-snapname 360
-vg 360

L

license
FCP 19
iSCSI 19
storage system 19

Linux
Red Hat and SUSE linux 6
Linux host
.rpm file 56, 63
adding host entries for LUNs 120
example of successful installation 57, 63
files installed 74
installation steps for SnapDrive for UNIX 56
moving downloaded file 55
preparing for LUNs 120
requirements 54
uninstalling SnapDrive for UNIX 60
verify installation 58
log files
audit 90
audit log 89, 122, 125, 126
disabling 123
enabling 122, 123
recovery 105
recovery log 122, 126, 128
specifying pathnames 123
trace 114, 115
trace log 122
Logical Unit Numbers *See* LUN
logical volume
creating
disk group 164
Logical volume manager *See*LVM ix
logical volumes
definition 369
logins
storage system 154
long_filer_path
example 364
LUN ID
defined x
definition 369
LUNs
adding host entries for LUNs 120
connecting to host 218
connecting to storage 198
defined x
definition 369
deleting 228
disconnecting from host entity 213
disconnecting from storage 207
disconnecting to host 227
displaying storage information 181
preparing host 120
preparing the host 85
-lunsize
command option 352
LVM
using to resize 196

M

man page
SnapDrive for UNIX 12
mgmt-retries option 100
mgmt-retry-sleep option 100
mgmt-sleep-long-secs option 100
Multipathing
refreshing the DMP paths 138
setting 133
multipathing-type option 101

N

name/value pair
snapdrive.conf file 118, 158
Network Interface Card (NIC), definition 369
non-originating host 271
-noprompt
command option 354
NOW description page
most up-to-date requirements 7
NOW site
downloading software 25
NTAPasl
install 24

O

open files
limit 303
operating system
limit on open files 303
storage system 18
options
-addlun 348

-all 348
-autoexpand 348
-autorename 348
-device 348
-dgsize 349
-force 349
-fsopts 349
-fstype 350
-full 350
-growby 350
-growto 350
-help 351
--lunsize 352
-mntopts 353
-nolvm 353
-nopersist 354
-noprompt 354
-noreserve 354
-quiet 354
-readonly 355
-reserve 354
-split 355
-status 355
-unrelated 355
using on command lines 348
-verbose 355
-vgsize 355

P

partner
specify IP address 19
password-file option 103
passwords
storage system 154
path 365
path configuration option 103
prefix-filer-lun option 104
prepare-lun-count option 104

Q

Quick Start Guide 12
-quiet
command option 354

R

raw data
definition 369
RBAC *See* access permissions
recommendations
snap reserve 22
storage system volumes 21
recovery log file 105, 122
disabling 123
enabling 123
example 128
specifying pathname 123
using 126
recovery-log-fileoption 104
recovery-log-save option 105
Red Hat Linux host
see Linux host 54
renaming Snapshot copies
examples 258
steps for renaming 258
requirements
AIX host 30
HP-UX host 46
Linux host 54
most current 7
Solaris host 61
storage system 18
restoring Snapshot copies
examples 271
restoring from a different host 271
restrictions 259
steps for restoring 265
time required 265
restrictions
on connecting Snapshot copies 274
on restoring Snapshot copies 259
Snapshot copies 242
return codes
error messags 326
rollback 110, 111, 112
definition 370
rpm file
Linux installation 56, 63
rsh
check for rsh usage 86

configuration variable 106
snapdrive snap connect 275
snapdrive storage create 169
snapdrive storage delete 229
Solaris cluster environment 5

S

SAN (Storage Area Network), definition 370
secure-communication-among-cluster-nodes 106
security
considerations 5
features 146
short_snap_name 366
SMIT
using on AIX 32
using on AIX host 31
snap create command 246
examples 248
snap delete command 295, 296
commands
snap delete 294
snap disconnect command
examples 292
snap rename command 258
examples 258
snap reserve 21
recommended value 20
resetting 22
snap restore command
examples 271
snap show command
examples 252
snap_name 366
snapcreate-cg-timeout 107
snapcreate-check-nonpersistent-nfs 107
snapcreate-consistency-retries option 109
snapcreate-consistency-retry-sleep 108
snapdelete-delete-rollback-with-snap option 110
SnapDrive for UNIX
access permissions 6, 147
additional documentation 12
AIX requirements 30
audit log 125
AutoSupport 130
capabilities 2
cautions 20
checking version 119
command arguments 361
command keywords 356
command options 348
comparing platforms 9
config check luns 85
config delete 85
config delete command 156
config list 84
config prepare luns 85
config set 85
config set command 154, 156
config show 84
configurations 16
connecting Snapshot copies 272
creating Snapshot copies 246
data collection utlity 298
deleting Snapshot copies 294, 295
deleting user logins 156
disabling log files 123
disconnecting Snapshot copies 288
displaying information on storage 177
displaying Snapshot copies 249
downloading software 25
enabling log files 122
error messages 301
files installed on host 74
getting software 25
getting software from CD-ROM 26
how it manages 4
how it works 4
how it works on a shared storage system 3
HP-UX requirements 46
HTTPS 5
installation steps on AIX 31
installation steps on HP-UX 48
installation steps on Linux 56
installation steps on Solaris 63
Linux requirements 54
log files 122
multiple platforms 9
prerequisites 16
recommendations 20

recovery log 126
renaming Snapshot copies 257
requires attach kit 23
restoring Snapshot copies 259
running from the command line 142
security 5
security features 146
setting snap reserve 22
setting up AutoSupport 132
setting values in snapdrive.conf 118
Snapshot copy
created with SnapDrive for UNIX 2
snapshot restrictions 242
Solaris requirements 61
specifying log file pathnames 123
specifying logins for storage system 154
stack 6
storage 2
storage system volumes 21
summary of all commands 341
supported Snapshot copies 238
time to restore Snapshot copies 265
uninstalling from AIX host 40
uninstalling from HP-UX host 52
uninstalling from Linux host 60
uninstalling from Solaris host 72
upgrading versions 79
version command 119
volume manager terminology 10
SnapDrive for UNIX Compatibility Matrix 12
SnapDrive for UNIX Quick Start 12
SnapDrive for UNIX Release Notes 12
snapdrive.conf
all-access-if-rbac-unspecified option 89
audit-log-file option 89
audit-log-max-size option 90
audit-log-save option 90
autosupport-enabled option 91
autosupport-filer option 91
available-lun-reserve option 92
contact-http-port 93
contact-ssl-port 93
default values 86
default-noprompt option 93
default-transport option 96
device-retries option 94
device-retry-sleep-secs option 95
enable-implicit-host-preparation option 97
filer-restore-retries option 98
filer-restore-retry-sleep option 99
filesystem-freeze-timeout option 99
mgmt-retries option 100
mgmt-retry-sleep option 100
multipathing-type option 101
name/value pair 118, 158
password-file option 103
path option 103
prefix-filer-lun option 104
prepare-lun-count option 104
recovery-log-file option 104
recovery-log-save option 105
secure-communication-among-cluster-nodes
106
setting values 118
snapcreate-cg-timeout 107
snapcreate-check-nonpersistent-nfs 107
snapcreate-consistency-retries option 109
snapcreate-consistency-retry-sleep 108
snapdelete-delete-rollback-with-snap option
110
snapmirror-dest-multiple-filervolumes-
enabled 110
snaprestore-delete-rollback-after-restore
option 110
snaprestore-make-rollback option 111
snaprestore-must-make-rollback option 112
snaprestore-must-make-snapinfo-on-qtree
option 109
snaprestore-snapmirror-check option 112
space-reservations-enabled 113
trace-enabled option 113
trace-level option 114
trace-log-file option 114, 115
trace-logs-ave option 115
use-https-to-fileroption 116
viewing 84
snapdrive.dc 298
examples 300
executing 299
tasks performed 298

SnapDriver for UNIX
Snapshot copies 2
snapmirror-dest-multiple-filervolumes-enabled 110
-snapname 360
SnapRestore
license required 19
snaprestore-delete-rollback-after-restore option 110
snaprestore-make-rollback option 111
snaprestore-must-make-rollback 112
snaprestore-must-make-snapinfo-on-qtree option 109
snaprestore-snapmirror-check option 112
Snapshot copies
about 238
connecting from different locations 272
Snapshot copy
creating 246
definition 370
deleting 294, 295
disconnecting 288
displaying information 249
examples of snap create command 248
examples of snap delete command 296
examples of snap disconnect command 292
examples of snap rename command 258
examples of snap restore command 271
examples of snap show command 252
listing on storage system 256
listing with wildcards 256
reasons to delete 294
renaming a Snapshot copy 257
restoring 259
restoring from a different host 271
restrictions 242
restrictions on connecting 274
restrictions on deleting 294
restrictions on restoring 259
rollback 110, 111, 370
rollback option 112
spanning storage systems 240
spanning volumes 240
steps for displaying 251
steps for renaming 258
steps for restoring 265
time to restore 265
using wildcards 366
software
downloading 25
getting 25
getting from CD-ROM 26
uncompressing 31
uncompressing on Solaris 62
Solaris host
adding host entries for LUNs 120
files installed 74
installation steps for SnapDrive for UNIX 63
preparing for LUNs 120
requirements 61
uncompressing SnapDrive for UNIX software 62
uninstalling SnapDrive for UNIX 72
space reservations 20
space-reservations-enabled 113
special messages xi
stack
required 6
storage
connecting 198
connecting LUNs to host 218
created with SnapDrive for UNIX 2
creating 163
deleting 228, 232, 233
disconnecting 207, 213
disconnecting LUNs to host 227
disconnecting specific LUNs 212
displaying information 177
examples of storage show command 183
growing 194
resizing 192
steps for deleting host entities 233
steps for deleting LUNs 232
steps for displaying information 181
steps for displaying information on host entities 182
steps for displaying information on storage systems 182
Storage Area Network *See* SAN
storage commands

using across storage systems 161
storage connect command
examples 202, 203, 204, 219, 220
steps for using 202
storage create command
examples 174
173
storage delete command 232, 233
limitations 228
storage disconnect command 213
examples 212, 213, 225, 226
limitations 208, 221
steps for using 212
tips 209
storage provisioning 160
storage resize command 194
limitations 192
steps for using 194
storage show command 181, 182, 185
examples 182, 183
restrictiosns 178
storage system
cluster 367
clustered 16
configuring volumes 21
definition 370
deleting SnapDrive for UNIX logins 156
displaying storage information 182
documentation 18
failover 16
guides for creating volumes 21
login required 5
logins 154
operating system 18
preparing 18
required license 19
requirements 18
resetting snap reserve 22
setup program 19
specify login information 154
specify partner IP address 19
target 371
volume 371
volume preparation 21
storage system volume
definition 371
storage systems
spanning 161, 240
verifying user names 155
support kit
definition 371
swverify command
verify installation 58
System Configuration Guide 13

T

target
definition 371
Terminology x
trace log file 114, 115, 122
disabling 123
enabling 123
specifing pathname 123
trace-enabled option 113
trace-level option 114
trace-log-file option 114, 115
trace-log-save option 115
troubleshooting
data collection utility 298
error message resturn codes 326
error messages 301
limits on open files 303

U

uninstalling
SnapDrive for UNIX from AIX host 40
SnapDrive for UNIX from HP-UX host 52
SnapDrive for UNIX from Linux host 60
SnapDrive for UNIX from Solaris host 72
upgrading
SnapDrive for UNIX 79
use-https-to-filer option 116
user logins
deleting from SnapDrive for UNIX 156
displaying names with config list 155
utilities
<host>_info 298
data collection 298

V

-verbose
command option 355
VERITAS
prerequisite 24
verifying configuration 81
version command
SnapDrive for UNIX 119
volume
logical 369
volume group
increasing size 192
volumes
definition 371
host 368
preparing for storage system 21
recommendations for storage system 21
spanning 240
storage system 21, 371

W

wildcards
example 365
using with Snapshot copies 366