Data ONTAP® 7.3 Data Protection Online Backup and Recovery Guide

NetApp, Inc. 495 East Java Drive Sunnyvale, CA 94089 U.S.A. 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: 210-04755_A0 Updated for Data ONTAP 7.3.3 on 15 January 2010

Contents

Copyright information	
Trademark information	19
About this guide	
Audience	
Accessing Data ONTAP man pages	
Terminology	
Where to enter commands	
Keyboard and formatting conventions	
Special messages	
How to send your comments	
Introduction to data protection	
Methods of protecting data	
What online backup and recovery is	
Advantages of online backup and recovery	
Disadvantages of online backup and recovery	31
Methods for online backup and recovery	31
What the NVFAIL feature is	32
What a data loss disaster is	
How to determine a disaster	32
What tools to use to protect against data-loss disasters	33
How to protect data in a SAN environment	35
Policy-based automated data protection using Protection Manager	35
Snapshot management	
What a Snapshot copy is	
How Snapshot copies handle file permissions	38
What you can do with Snapshot copies	
User access to Snapshot copies	39
Access to Snapshot copies over NFS	39
Access to Snapshot copies over CIFS	40
Accessing Snapshot copies from CIFS clients	41
Restricting access to Snapshot copies	41
How Data ONTAP Snapshot copies work in an iSCSI or FC network	42

Using Snapshot copies in the SAN environment	42
Relationship between a LUN and a Snapshot copy	43
Restoring files from Snapshot copies	44
Snapshot restoration using Shadow Copy Client tools	44
Creation of Snapshot copy schedules	44
User-specified Snapshot copy schedules	45
Snapshot copy schedule conflicts	46
Viewing the Snapshot copy schedule using the CLI	46
Displaying the Snapshot schedule using FilerView	46
What the snap sched command arguments mean	47
Strategies for creating a Snapshot copy schedule	47
Changing the Snapshot copy schedule	48
Enabling or disabling automatic Snapshot copies	49
Creating Snapshot copies manually	50
Creating Snapshot copies using FilerView	50
Why you might need to access a particular Snapshot copy	50
Finding the Snapshot copy you need from an NFS client	51
Determining access times from an NFS client	51
Finding the Snapshot copy you need from a CIFS client	52
How to determine access times from a CIFS client	52
What Snapshot disk consumption is	52
How Snapshot copies consume disk space	53
How changing file content consumes disk space	53
Monitoring Snapshot copy disk consumption	54
Displaying Snapshot copy disk consumption statistics	55
What the Snapshot copy reserve is	58
Changing the Snapshot copy reserve	61
What file folding means and how it saves disk space	61
Enabling file folding	62
Displaying the rate of change between Snapshot copies	62
Displaying rates of change between Snapshot copies	63
Deleting Snapshot copies automatically	64
Deleting Snapshot copies automatically without options	65
Viewing settings for the automatic deletion of Snapshot copies	65
Restoring the default settings for the automatic deletion of Snapshot	
copies	66

	Disabling a policy for automatic deletion of Snapshot copies	. 66
	Displaying space reclaimed from deleted Snapshot copies	. 67
	How to determine which Snapshot copies to delete on the basis of size	. 67
	Deleting a Snapshot copy manually	. 68
	Manual deletion of a busy or locked Snapshot copy	. 69
	Renaming Snapshot copies	. 70
Data	recovery using SnapRestore	71
	What SnapRestore is	. 71
	What SnapRestore does	. 71
	When to use SnapRestore	. 72
	Considerations before using SnapRestore	. 72
	Prerequisites for using SnapRestore	. 73
	General cautions for using SnapRestore	. 73
	Caution about reverting the root volume	. 74
	Preserving configuration files	. 74
	Reverting a root volume before using SnapRestore	. 74
	Installing the SnapRestore license	. 74
	Reverting a volume to a selected Snapshot copy	. 75
	Reverting a file to a selected Snapshot copy	. 77
	Obtaining correct incremental backups after reversion	. 80
Data	protection using SnapMirror	81
	How SnapMirror works	. 82
	Applications of SnapMirror	. 83
	What synchronous SnapMirror is	. 84
	Synchronous SnapMirror modes	. 85
	How SnapMirror replicates data synchronously	. 86
	How synchronous SnapMirror handles network issues	. 86
	Guidelines for growing an aggregate with a synchronous SnapMirror	
	destination volume	. 86
	Enabling SnapMirror by entering license codes	. 87
	Turning SnapMirror on	. 87
	Considerations for the use of SnapMirror	. 88
	Prerequisites for SnapMirror	. 88
	Restrictions while using SnapMirror	. 89
	Points of caution while using SnapMirror	. 90
	Symmetrical disk geometry	. 90

Recommended actions while using SnapMirror	91
Deduplication with volume SnapMirror	91
Possible conflicts between SnapMirror operation and Snapshot copy	
schedule	92
Destination accessibility when using CIFS with SnapMirror	92
Considerations before using synchronous SnapMirror	93
Disk types for a V-Series system using synchronous and semi-	
synchronous SnapMirror	95
Estimating aggregate size for synchronous SnapMirror destination	
volumes	95
Deployment of SnapMirror	96
Supported SnapMirror configurations	97
Comparison between volume SnapMirror and qtree SnapMirror	97
SnapMirror deployment variations	98
Migration from traditional volumes to FlexVol volumes	100
SnapMirror commands	100
SnapMirror options	103
SnapMirror files	105
SnapMirror support for IPv6	107
Setting up a basic SnapMirror operation	108
Firewall usage with SnapMirror	110
Data replication from one destination to another in a series (cascading)	110
SnapMirror source to tape to destination scenario	116
Initialization of a SnapMirror destination	121
Quotas for SnapMirror destination qtrees	122
Guidelines for creating a qtree SnapMirror relationship	122
Initialization of a SnapMirror destination from tape	122
Initializing a SnapMirror destination	122
Space guarantee for a volume SnapMirror destination	124
Initializing a destination for non-qtree data	125
How the snapmirror initialize command copies volumes	125
How the snapmirror initialize command copies qtrees	126
What happens after SnapMirror makes the initial copy to the destination	126
How to check the initialization of a volume	126
Checking the initialization of a qtree	127

How the snapmirror initialize command matches source and destination	
volume size	. 127
What you can do if an initial SnapMirror transfer fails	. 127
Maximum number of concurrent replication operations	. 128
Enabling the NearStore personality	. 131
Maximum number of concurrent replication operations in an active/	
active configuration	. 132
Methods for specifying destination systems on the SnapMirror source	. 132
Specifying SnapMirror destinations using the snapmirror.access option	. 132
Specifying SnapMirror destinations using the snapmirror.allow file	. 133
Resolving host names to their IP addresses	. 134
What the snapmirror.conf file does	. 135
Distribution of the snapmirror.conf file	. 135
Limit on entries in the snapmirror.conf file	. 135
Editing the snapmirror.conf file	. 136
Syntax for snapmirror.conf file entries	. 137
Scheduled updates for volumes or qtrees	. 144
Changing scheduled updates for one volume or qtree	. 144
Turning off SnapMirror updates	. 145
Turning off scheduled updates for one volume or qtree	. 146
Manual update of a SnapMirror destination	. 146
Performing a manual SnapMirror update	. 147
Creating extra backup Snapshot copies for SnapMirror qtrees	. 148
What happens after SnapMirror makes incremental updates to the	
destination	. 149
SnapMirror over multiple paths	. 149
Setting up a multipath SnapMirror relationship	. 150
Converting a single-path SnapMirror relationship to multipath	. 150
Compression for SnapMirror transfers	. 151
Enabling compression for a SnapMirror transfer	. 152
Viewing the compression ratio for a SnapMirror transfer	. 153
Checking SnapMirror data transfer status	. 154
What SnapMirror status check shows	. 155
Information messages in the SnapMirror status check	. 158
Adjusting the TCP window size for a SnapMirror relationship	. 162
Setting a maximum transfer rate for all transfers	. 164

Changing the maximum transfer rate for a single SnapMirror transfer	. 165
Use of SnapMirror to copy a volume to a tape	. 165
Considerations before using SnapMirror copy to tape as a backup	
method	. 166
About moving SnapMirror sources	. 167
Moving volume SnapMirror sources	. 167
Moving qtree SnapMirror sources	. 169
Migrating data between volumes by using SnapMirror	. 171
Conversion of a destination to a writable volume or qtree	. 172
Quota restrictions	. 172
Converting a SnapMirror destination to a writable volume or qtree	. 173
After using the snapmirror break command	. 174
Resizing a SnapMirror source and destination volume pair	. 174
Converting asynchronous SnapMirror replication to synchronous	. 178
Stabilizing destinations before a Snapshot copy	. 178
What the quiesce command does	. 179
Resuming transfers after quiescing a destination	. 179
Aborting a SnapMirror transfer	. 180
Releasing partners from a SnapMirror relationship	. 181
SnapMirror data transfer logs	. 183
Checking for SnapMirror logging	. 183
Turning SnapMirror logging on	. 184
Format of SnapMirror log files	. 184
Turning SnapMirror logging off	. 186
Listing SnapMirror Snapshot copies	. 186
Naming conventions for Snapshot copies used by SnapMirror	. 186
Use of the snap list command to display SnapMirror updates on the	
destination volume	. 187
What SnapMirror restarts and retries are	. 189
What the snapmirror resync command does	. 189
Resynchronizing a SnapMirror relationship	. 190
How the snapmirror resync command helps minimize data loss	. 192
Resynchronization of FlexVol volumes	. 193
Testing database applications: A special use of snapmirror resync	. 193
Retrieving data for disaster recovery: A special use of snapmirror resync .	194
Operation of SnapMirror with other features and products	. 195

Comparison between SnapMirror and the vol copy command	196
Comparison between qtree SnapMirror and SnapVault	196
Transfer of LUN clones using qtree SnapMirror	197
Management of SnapMirror operations through Protection Manager	198
Use of SnapMirror with SnapDrive	198
SnapMirror and MultiStore	198
How FlexClone volumes impact SnapMirror	199
Guidelines for creating a clone of a qtree SnapMirror destination volum	ne . 200
SnapVault destination to tape backup	200
How SnapMirror works with the dump command	203
Protection of SnapVault secondaries using volume SnapMirror	204
Use of SnapMirror with S Family storage systems	206
SnapMirror and ACLs	207
SnapMirror over Fibre Channel	207
Hardware requirements for SnapMirror over Fibre Channel	208
Supported Fibre Channel switches	208
SnapMirror over Fibre Channel topology	209
SnapMirror traffic zones	210
Requirements for deploying SnapMirror over Fibre Channel	211
Functionality supported by SnapMirror over Fibre Channel	212
Configuring SnapMirror over Fibre Channel	212
Troubleshooting issues related to SnapMirror over Fibre Channel	219
Troubleshooting of SnapMirror issues	223
What happens if you change a SnapMirror destination volume name	223
Accidental deletion of SnapMirror Snapshot copies	224
Data protection using SnapVault	225
What SnapVault is	226
Advantages of using SnapVault	227
What data gets backed up and restored through SnapVault	227
Types of SnapVault deployment	228
How SnapVault backup works	230
How SnapVault backup works for open systems	231
SnapVault support for IPv6	232
Planning SnapVault backups	233
Planning primary and secondary qtree locations	233
SnapVault primary and secondary on the same system	234

Planning SnapVault backup schedule and Snapshot copy retention	234
Estimating the initial backup time	236
Limit on the number of concurrent SnapVault targets	236
Enabling SnapVault	237
Enabling licenses for SnapVault	238
Setting the snapvault.enable option	238
Setting the ndmpd option	239
Setting the snapvault.access option	239
How to start a SnapVault backup relationship	240
Guidelines for creating a SnapVault relationship	240
Backing up qtree data	241
What non-qtree data is	241
Backing up non-qtree data	242
What volume data backup involves	242
What SnapVault Snapshot copy update schedules are	244
How to avoid Snapshot copy schedule conflicts	245
Scheduling Snapshot copies on the SnapVault primary system	245
Scheduling Snapshot copy backups to the SnapVault secondary system .	247
Scheduling Snapshot copies on the secondary system for archiving	248
Displaying the currently configured Snapshot copy schedule	249
Retaining older SnapVault Snapshot copies on SnapVault secondary	
volumes	249
Unscheduling SnapVault Snapshot copies	251
Disabling Snapshot copies temporarily without unscheduling	252
Enabling Snapshot copies that are temporarily disabled	252
Checking SnapVault transfers	253
Examples for checking the status	254
What the status fields mean	256
Displaying SnapVault Snapshot copies	258
Displaying SnapVault Snapshot copies on a volume	259
Listing Snapshot copies for qtrees	261
Guidelines for creating a clone of a SnapVault destination volume	261
About LUN clones and SnapVault	262
LUN clone transfer in non-optimized mode	263
LUN clones transfer in optimized mode using SnapDrive for Windows	263
How to change SnapVault settings	264

Changing settings for SnapVault backup relationships	265
Why you manually update a qtree on the secondary system	266
Manually updating individual secondary system qtrees	267
Examples of how to update the Snapshot copy on the secondary system	. 267
Why you create a Snapshot copy manually	268
Creating a Snapshot copy manually	268
Specifying a single try for SnapVault Snapshot copy creation	269
Renaming a SnapVault or Open Systems SnapVault secondary volume	270
Restoring SnapVault data to the primary system	271
Examples of restoring SnapVault data	273
Deleting the residual Snapshot copy	274
How to abort SnapVault transfers	275
Aborting primary-to-secondary storage transfers	275
Aborting secondary-to-primary storage transfers	276
Aborting SnapVault Snapshot copy creation	276
Ending SnapVault backups for a qtree	276
Releasing SnapVault relationships	277
Turning SnapVault off	278
Compression feature of Open Systems SnapVault	278
Enabling the compression feature globally for Open Systems SnapVault	
relationships	279
Enabling the compression feature for a new Open Systems SnapVault	
relationship	279
Enabling the compression feature for an existing Open Systems	
SnapVault relationship	280
Disabling the compression feature globally for Open Systems SnapVault	
relationships	280
Disabling the compression feature for a new Open Systems SnapVault	
relationship	. 281
Disabling the compression feature for an existing Open Systems	
SnapVault relationship	281
Setting the default value for compression feature	282
Viewing the compression status for Open Systems SnapVault	
relationships	282
SnapVault secondary system protection	282
How to use SnapMirror to replicate SnapVault data	283

Using backup and standby service for SnapVault	283
How to use SnapVault to protect a volume SnapMirror destination	287
Preserving a Snapshot copy	288
Unpreserving a Snapshot copy	289
SnapVault and MultiStore	290
Error regarding language setting changes on volumes	291
About SnapVault for NetBackup	291
SnapVault for NetBackup benefits	293
SnapVault for NetBackup terminology	293
What the NetBackup catalog is	294
How NetBackup transfers data to the NearStore system	294
How to back up file system data using NetBackup	294
How the NearStore secondary system processes the data	295
Limitations of SnapVault for NetBackup	297
SnapVault for NetBackup usage recommendations	298
How to configure SnapVault for NetBackup transfers	301
Management of data transfers	302
Data restore from the exported file system	313
Removing SnapVault for NetBackup volumes	314
SnapVault for NetBackup with active/active configurations	317
Disaster recovery using volume SnapMirror on NearStore systems	318
How to access backups on the replica of a SnapVault for NetBackup	
volume	327
Troubleshooting for SnapVault for NetBackup	327
Data replication using volume copy	333
Benefits of using volume copy	333
When to copy volumes	334
IPv6 support with volume copy	335
Prerequisites before copying a volume	335
Verifying the size of each volume	336
Verifying the relationship between systems	337
Verifying and changing the status of source and destination volumes	337
Enabling remote access	338
Copying volumes using the vol copy command	339
Number of vol copy operations supported	339
Copying Snapshot copies with the vol copy start command	340

Copying one volume to another volume using the vol copy command	341
Using volume copy to copy LUNs	343
Checking the status of a volume copy operation	343
Displaying the current speed for copying a volume	345
Controlling a volume copy operation speed	345
Aborting a volume copy operation	346
Data mirroring using SyncMirror	349
What SyncMirror is	349
Advantages of using SyncMirror	350
What mirrored aggregates are	350
Requirements for using SyncMirror with disks	351
How SyncMirror works with third-party storage	351
Implications of storage type when mirroring with V-Series systems	352
Requirements for setting up SyncMirror with array LUNs	353
SyncMirror pool assignment planning for third-party storage	357
Example of SyncMirror pool assignments for array LUNs	358
Common errors when setting up SyncMirror pools with array LUNs	360
Troubleshooting errors with SyncMirror pool assignment for array	
LUNs	360
Enabling the SyncMirror license	361
Disabling the SyncMirror license	362
Considerations for using mirrored aggregates	363
How disks are assigned to plexes	363
Viewing plexes and spare pools	364
Creating a mirrored aggregate	366
Converting an aggregate to a mirrored aggregate	368
Addition of disks or array LUNs to a mirrored aggregate	370
Rules for adding disks to a mirrored aggregate	371
Rules for adding array LUNs to a mirrored aggregate	371
Adding disks to a mirrored aggregate, where Data ONTAP selects the	
disks	371
Adding disks or array LUNs to a mirrored aggregate, where the user	
selects the disks	372
Adding disks to a mirrored aggregate, where the user selects the disks	
with assistance from Data ONTAP	373
The states of a plex	374

Viewing the status of plexes	374
Changing the state of a plex	375
Splitting a mirrored aggregate	375
Rejoining split aggregates	376
Removing a plex from a mirrored aggregate	377
Comparing plexes of a mirrored aggregate	378
Stopping plex comparison	379
Suspending plex comparison	379
Resuming plex comparison	380
Viewing the status of a plex comparison	380
Database protection using NVFAIL	
How NVFAIL protects database files	381
Enabling database file protection	383
Where to look for database file verification instructions	
Adding more database file protection	383
Making LUNs accessible to the host after an NVRAM failure	384
Database protection using SnapValidator	
What SnapValidator checks	385
Preparing a database for SnapValidator	386
Implementing SnapValidator checks	387
Licensing SnapValidator	387
Enabling SnapValidator checks on volumes	387
Enabling SnapValidator checks on LUNs	388
Disabling SnapValidator	390
Disabling SnapValidator on a volume	391
Disabling SnapValidator on a LUN	391
Troubleshooting invalid data error messages	391
Virus protection for CIFS	395
How CIFS virus scanning works	395
File types scanned by default	396
Setting up and starting virus scanning	396
Setting up PC clients as virus-scanning clients	397
Enabling virus scanning on the system	397
Setting up secondary scanning clients	398
Setting up McAfee scan detection properties for systems	399
Specifying file types to be scanned	399

Checking the status of virus-scanning ontions	
Controlling vFiler unit usage of host system's virus scanners	410
Allowing file access when the scan cannot be performed	410
Setting and resetting the request timeout for a virus scan	409
Checking vscan information	408
Primary virus scanner not listed	407
Displaying the scanner list	407
Adding shares with virus scanning turned off for read-only access	406
Adding shares with virus scanning turned off	406
Turning scanning on for read-only access	405
Turning scanning off for read-only access	405
Turning scanning on for any access	405
Turning virus scanning off for any access	404
Specifying shares for scanning	404
Using an inclusion list in combination with an exclusion list	403
Resetting the exclude file types list to empty	403
Removing file types to exclude from scanning	403
Adding file types to exclude from scanning	402
Creating a list of file types to exclude from scanning	
Displaying file types to exclude from scanning	402
Excluding file types to be scanned	401
Resetting file types to be scanned	
Repracing file types to be scanned	
Adding file types to be scanned	400
Displaying file types to be scanned	400
Displaying file types to be scanned	400

Copyright information

Copyright [©] 1994–2010 NetApp, 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 NetApp material is subject to the following license and disclaimer:

THIS SOFTWARE IS PROVIDED BY NETAPP "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL NETAPP 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.

NetApp reserves the right to change any products described herein at any time, and without notice. NetApp assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by NetApp. 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 NetApp.

The product described in this manual may be protected by one or more U.S.A. 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

All applicable trademark attribution is listed here.

NetApp, the Network Appliance logo, the bolt design, NetApp-the Network Appliance Company, Cryptainer, Cryptoshred, DataFabric, DataFort, Data ONTAP, Decru, FAServer, FilerView, FlexClone, FlexVol, Manage ONTAP, MultiStore, NearStore, NetCache, NOW NetApp on the Web, SANscreen, SecureShare, SnapDrive, SnapLock, SnapManager, SnapMirror, SnapMover, SnapRestore, SnapValidator, SnapVault, Spinnaker Networks, SpinCluster, SpinFS, SpinHA, SpinMove, SpinServer, StoreVault, SyncMirror, Topio, VFM, and WAFL are registered trademarks of NetApp, Inc. in the U.S.A. and/or other countries. gFiler, Network Appliance, SnapCopy, Snapshot, and The evolution of storage are trademarks of NetApp, Inc. in the U.S.A. and/or other countries and registered trademarks in some other countries. The NetApp arch logo; the StoreVault logo; ApplianceWatch; BareMetal; Camera-to-Viewer; ComplianceClock; ComplianceJournal; ContentDirector; ContentFabric; Data Motion; EdgeFiler; FlexShare; FPolicy; Go Further, Faster; HyperSAN; InfoFabric; Lifetime Key Management, LockVault; NOW; ONTAPI; OpenKey, RAID-DP; ReplicatorX; RoboCache; RoboFiler; SecureAdmin; SecureView; Serving Data by Design; Shadow Tape; SharedStorage; Simplicore; Simulate ONTAP; Smart SAN; SnapCache; SnapDirector; SnapFilter; SnapMigrator; SnapSuite; SohoFiler; SpinMirror; SpinRestore; SpinShot; SpinStor; vFiler; VFM Virtual File Manager; VPolicy; and Web Filer are trademarks of NetApp, Inc. in the U.S.A. and other countries. NetApp Availability Assurance and NetApp ProTech Expert are service marks of NetApp, Inc. in the U.S.A.

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. A complete and current list of other IBM trademarks is available on the Web at http://www.ibm.com/legal/copytrade.shtml.

Apple is a registered trademark and QuickTime is a trademark of Apple, Inc. in the U.S.A. and/or other countries. Microsoft is a registered trademark and Windows Media is a trademark of Microsoft Corporation in the U.S.A. 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 U.S.A. and/or other countries.

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

NetApp, Inc. is a licensee of the CompactFlash and CF Logo trademarks.

NetCache is certified RealSystem compatible.

About this guide

You can use your product more effectively when you understand this document's intended audience and the conventions that this document uses to present information.

This guide describes how to protect, back up, restore, and copy data for systems that run Data ONTAP.

Next topics

Audience on page 21 Accessing Data ONTAP man pages on page 21 Terminology on page 22 Where to enter commands on page 23 Keyboard and formatting conventions on page 24 Special messages on page 25 How to send your comments on page 25

Audience

This document is written with certain assumptions about your technical knowledge and experience.

This guide is for system administrators who are familiar with operating systems that run on the storage system's clients, such as UNIX, Windows 95, Windows NT, and Windows 2000. You should be familiar with how to configure the storage system and how the NFS, CIFS, and HTTP protocols are used for file sharing or transfers. This guide does not cover basic system or network administration topics, such as IP addressing, routing, and network topology; it emphasizes the characteristics of the storage system.

Accessing Data ONTAP man pages

You can use the Data ONTAP manual (man) pages to access technical information.

About this task

Data ONTAP manual pages are available for the following types of information. They are grouped into sections according to standard UNIX naming conventions.

Types of information	Man page section
Commands	1

Types of information	Man page section
Special files	4
File formats and conventions	5
System management and services	8

Step

- 1. View man pages in the following ways:
 - Enter the following command at the storage system command line:

man command_or_file_name

- Click the manual pages button on the main Data ONTAP navigational page in the FilerView user interface.
- Use the *Commands: Manual Page Reference*, Volumes 1 and 2 (which can be downloaded or ordered through the NOW site).

Note: All Data ONTAP man pages are stored on the storage system in files whose names are prefixed with the string "na_" to distinguish them from client man pages. The prefixed names are used to distinguish storage system man pages from other man pages and sometimes appear in the NAME field of the man page, but the prefixes are not part of the command, file, or services.

Terminology

To understand the concepts in this document, you might need to know how certain terms are used.

Storage terms

array LUN	Refers to storage that third-party storage arrays provide to storage systems running Data ONTAP software. One array LUN is the equivalent of one disk on a native disk shelf.
LUN (logical unit number)	Refers to a logical unit of storage identified by a number.
native disk	Refers to a disk that is sold as local storage for storage systems that run Data ONTAP software.
native disk shelf	Refers to a disk shelf that is sold as local storage for storage systems that run Data ONTAP software.

storage controller	Refers to the component of a storage system that runs the Data ONTAP operating system and controls its disk subsystem. Storage controllers are also sometimes called <i>controllers, storage appliances, appliances, storage engines, heads, CPU modules,</i> or <i>controller modules.</i>
storage system	Refers to the hardware device running Data ONTAP that receives data from and sends data to native disk shelves, third-party storage, or both. Storage systems that run Data ONTAP are sometimes referred to as <i>filers, appliances, storage appliances, V-Series systems</i> , or <i>systems</i> .
third-party storage	Refers to the back-end storage arrays, such as IBM, Hitachi Data Systems, and HP, that provide storage for storage systems running Data ONTAP.

Cluster and high-availability terms

active/active	In the Data ONTAP 7.2 and 7.3 release families, refers to a pair of storage		
configuration	systems (sometimes called <i>nodes</i>) configured to serve data for each other if on		
	of the two systems stops functioning. Also sometimes referred to as active/active		
	pairs. In the Data ONTAP 7.1 release family and earlier releases, this		
	functionality is referred to as a <i>cluster</i> .		

clusterIn the Data ONTAP 7.1 release family and earlier releases, refers to a pair of
storage systems (sometimes called *nodes*) configured to serve data for each other
if one of the two systems stops functioning. In the Data ONTAP 7.3 and 7.2
release families, this functionality is referred to as an *active/active configuration*.

Where to enter commands

You can use your product more effectively when you understand how this document uses command conventions to present information.

You can perform common administrator tasks in one or more of the following ways:

- You can enter commands either at the system console or from any client computer that can obtain access to the storage system using a Telnet or Secure Shell (SSH) session.
 In examples that illustrate command execution, the command syntax and output shown might differ from what you enter or see displayed, depending on your version of the operating system.
- You can use the FilerView graphical user interface. For information about accessing your system with FilerView, see the *Data ONTAP System Administration Guide*.
- You can enter Windows, ESX, HP-UX, AIX, Linux, and Solaris commands at the applicable client console.

In examples that illustrate command execution, the command syntax and output shown might differ from what you enter or see displayed, depending on your version of the operating system.

• You can use the client graphical user interface.

Your product documentation provides details about how to use the graphical user interface.

• You can enter commands either at the switch console or from any client that can obtain access to the switch using a Telnet session.

In examples that illustrate command execution, the command syntax and output shown might differ from what you enter or see displayed, depending on your version of the operating system.

Keyboard and formatting conventions

You can use your product more effectively when you understand how this document uses keyboard and formatting conventions to present information.

Keyboard conventions

Convention	What it means
The NOW site	Refers to NetApp On the Web at http://now.netapp.com/.
Enter, enter	 Used to refer to the key that generates a carriage return; the key is named Return on some keyboards. Used to mean pressing one or more keys on the keyboard and then pressing the Enter key, or clicking in a field in a graphical interface and then typing information into the field.
hyphen (-)	Used to separate individual keys. For example, Ctrl-D means holding down the Ctrl key while pressing the D key.
type	Used to mean pressing one or more keys on the keyboard.

Formatting conventions

Convention	What it means
<i>Italic</i> font	 Words or characters that require special attention. Placeholders for information that you must supply. For example, if the guide says to enter the arp -d hostname command, you enter the characters "arp -d" followed by the actual name of the host. Book titles in cross-references.

Convention	What it means
Monospaced font	 Command names, option names, keywords, and daemon names. Information displayed on the system console or other computer monitors. Contents of files. File, path, and directory names.
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.

Special messages

This document might contain the following types of messages to alert you to conditions that you need to be aware of.

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

Attention: An attention notice contains instructions that you must follow to avoid a system crash, loss of data, or damage to the equipment.

How to send your comments

You can help us to improve the quality of our documentation by sending us your feedback.

Your feedback is important in helping us to provide the most accurate and high-quality information. If you have suggestions for improving this document, send us your comments by e-mail to *doccomments@netapp.com*. To help us direct your comments to the correct division, include in the subject line the name of your product and the applicable operating system. For example, *FAS6070—Data ONTAP 7.3*, or *Host Utilities—Solaris*, or *Operations Manager 3.8—Windows*.

Introduction to data protection

Data protection means backing up data and being able to recover it. You protect the data by making copies of it so that it is available for restoration even if the original is no longer available.

Businesses need data backup and protection for the following reasons.

- To protect data from accidentally-deleted files, application crashes, data corruption, and viruses
- To archive data for future use
- To recover from a disaster

Next topics

Methods of protecting data on page 27 What online backup and recovery is on page 29 What the NVFAIL feature is on page 32 What a data loss disaster is on page 32 How to protect data in a SAN environment on page 35 Policy-based automated data protection using Protection Manager on page 35

Methods of protecting data

Depending on your data protection and backup needs, Data ONTAP offers a variety of features and methods to ensure against accidental, malicious, or disaster-induced loss of data.

Data protection feature	Description
aggr copy	This is fast block copy of data stored in aggregates; it enables you to quickly copy blocks of stored system data from one aggregate to another.
Snapshot copy	Backup within a volume.
	This feature allows you to manually or automatically create, schedule, and maintain multiple backups (also called Snapshot copies) of data on a volume. Snapshot copies use only a minimal amount of additional volume space, and do not have a performance cost.
	If a user accidentally modifies or deletes crucial data on a volume with Snapshot enabled, that data can be easily and quickly restored from one of the last several Snapshot copies taken.
	You can also create clones of FlexVol volumes and Data ONTAP LUNs using Snapshot copies. For more details, see the <i>Data ONTAP</i> <i>Storage Management Guide</i> .

Data protection feature	Description
SnapRestore (license required)	Fast, space-efficient restoration of large volumes of data backed up to Snapshot copies.
	The SnapRestore feature performs on-request Snapshot recovery from Snapshot copies on an entire volume.
SnapMirror (license required)	Volume-to-volume and qtree-to-qtree replication.
	This feature enables you to periodically make Snapshot copies of data on one volume or qtree, replicate that data to a partner volume or qtree, usually on another storage system, and archive one or more iterations of that data as Snapshot copies. Replication on the partner volume or qtree ensures quick availability and restoration of data, from the point of the last Snapshot copy, should the storage system containing the original volume or qtree be disabled.
	If you conduct tape backup and archival operations, you can carry them out on the data already backed to the SnapMirror partner, thus freeing the original storage system of this time-consuming, performance-degrading chore.
SnapVault (license required)	Centralized backup of multiple qtrees on multiple storage systems using Snapshot technology.
	This feature enables you to back up qtrees on multiple volumes and storage systems to a single SnapVault secondary storage system specialized for quick backup and restore of its sources.
	You can also install the Open Systems SnapVault agent on Windows, Solaris, Linux, AIX, or HP-UX systems. This agent enables SnapVault to back up and restore data to these systems also.
	If you conduct tape backup and archival operations, you can carry them out on the data already backed up to the SnapVault secondary storage system, thus freeing your storage system of this time- consuming, performance-degrading chore.
Tape backup dump and restore	Tape backup and restore.
commands	The dump and restore commands allow you to back up Snapshot copies to tape. The dump command takes a Snapshot copy of the volume and then copies that data to tape. Because the Snapshot copy, not the active file system, is backed up to tape, Data ONTAP can continue its normal functions while the tape backup takes place.
	For more information, see the <i>Data ONTAP Data Protection Tape</i> <i>Backup and Recovery Guide.</i>
vol copy	Fast block-copy of data from one volume to another.
	The vol copy command enables you to quickly block copy stored data from one volume to another.

Data protection feature	Description
SyncMirror (active/active	Continuous mirroring of data to two separate aggregates.
configuration required)	This feature allows for real-time mirroring of data to matching aggregates physically connected to the same storage system.
	In case of an unrecoverable disk error on one volume, the storage system automatically switches access to the mirrored volume.
	This feature requires an active/active configuration.
nvfail option to the vol options command	Protection against data corruption by failures of nonvolatile RAM (NVRAM).
SnapValidator (license required)	Detects and rejects invalid Oracle data before it is written to the storage system.
Virus scan support	Support for third-party virus-scanning software for files accessed by CIFS clients.
MetroCluster	Stretch MetroCluster provides site protection within a campus, and supports replication up to 500m. Fabric MetroCluster provides site protection within a metro, and supports replication up to 100 km using FC switches. SyncMirror functionality is enhanced to provide continuous volume mirroring.
	For V-Series systems, Fabric MetroCluster support is limited to 30 km.

Related concepts

Data recovery using SnapRestore on page 71 Data protection using SnapVault on page 225 Data replication using volume copy on page 333 Data protection using SnapMirror on page 81 Virus protection for CIFS on page 395 Database protection using SnapValidator on page 385

Related references

Data mirroring using SyncMirror on page 349

What online backup and recovery is

Data ONTAP creates online data backups to enable online data recovery. Online backup data is stored on disks, or on array LUNs in the case of third-party storage, rather than on tape. Data stored on disk is available for quick restoring in the event that disaster recovery operations are necessary.

Online backup and recovery solutions include: Snapshot, SnapMirror, SnapRestore, SnapVault, SyncMirror, MetroCluster, the vol copy command, and the ndmpcopy command.

- The Snapshot feature enables you to schedule weekly, daily, or hourly online backups. Snapshot technology makes online point-in-time copies in the same volume as the original data. It enables users to recover their own deleted or modified files without assistance from a system administrator.
- The SnapMirror feature allows you to schedule regular automatic copies of file system Snapshot copies of a volume or qtree onto another volume or qtree (on the same or a different storage system).
- The SnapRestore feature restores an entire volume to the state recorded in a previously created Snapshot copy with maximum speed and disk space efficiency. The SnapVault feature protects the data in one or more qtrees in a series of Snapshot copies stored on a separate storage system.
- SnapVault maintains an online, asynchronous, permanently read-only replica of the qtree data. SnapVault backup and Snapshot copy creation runs on an automated schedule.

Note: SnapVault, in addition to providing storage system backup, also provides direct backup to servers running Windows NT, Windows 2000, Solaris, or HP-UX.

- SyncMirror provides continuous real-time mirroring of data between two partner volumes on a shared or partner storage system.
- The MetroCluster feature provides SyncMirror continuous mirroring over extended distances (500 meters to 100 kilometers). For V-Series, Fabric MetroCluster support is limited to 30 km.
- The vol copy command uses a Snapshot copy and copies a volume, manually or by means of a script.
- The ndmpcopy command copies any subtree to any location on any storage system. For more information, see the *Data ONTAP Data Protection Tape Backup and Recovery Guide*.

You can use these online data backup and recovery systems to supplement tape backup and recovery.

Next topics

Advantages of online backup and recovery on page 30 Disadvantages of online backup and recovery on page 31 Methods for online backup and recovery on page 31

Related concepts

Data protection using SnapMirror on page 81

Advantages of online backup and recovery

Online backup and recovery protection offers better speed and ease of use, compared to tape archives.

The main advantages of online backup and recovery are as follows.

- Speedy backups and restores greatly reduce backup time requirements.
- Backups can be made more frequently because they are faster.
- It is easy to recover a particular file, directory, or volume from an online backup.
- Disaster recovery is quicker with online mirroring and restores.

- Data availability is higher because of the high speed of data recovery.
- More data can be backed up in less time.

Disadvantages of online backup and recovery

Online data protection has some disadvantages over tape archives.

The important disadvantages of online backup and recovery compared to tape archives are the following.

- Online data protection is often more expensive to procure.
- Online data protection consumes resources, such as disk space, that can otherwise be used for everyday activities.

Methods for online backup and recovery

You can perform online backup and recovery using several methods.

Data ONTAP provides the following features for online backup and recovery:

- Snapshot copies to make a read-only image of a file system on the same disk
- SnapRestore to restore data to a corrupted volume from a previous Snapshot copy
- SnapMirror to maintain a replica of one volume in another volume, and one qtree in another qtree
- SnapVault to keep copies of volumes on the server, from which individual qtrees are available at any time to the client
- The vol copy command to copy data from one volume to another
- SyncMirror to maintain two identical copies of a volume at all times

Related concepts

Snapshot management on page 37 Data recovery using SnapRestore on page 71 Data protection using SnapVault on page 225 Data replication using volume copy on page 333 Data protection using SnapMirror on page 81

Related references

Data mirroring using SyncMirror on page 349

What the NVFAIL feature is

If NVRAM problems occur that compromise database validity, the NVFAIL feature can warn you and automatically rename the database so that it does not restart automatically. You can then ensure that the database is valid before restarting it.

Data ONTAP provides database protection using the nvfail option of the vol options command.

Note: You can use this feature only when there are databases on the storage system.

Related concepts

Database protection using NVFAIL on page 381

What a data loss disaster is

A data loss disaster is a situation in which service from one physical site (for example, a building or a corporate campus) on the network is lost for an extended period of time.

The following are examples of disasters:

- Fire
- Earthquake
- Prolonged power outages at a site
- · Prolonged loss of connectivity from clients to the storage system at a site

When a disaster occurs, it can affect all the computing infrastructure including storage systems, application servers, networking connectivity, and client connectivity. When you create a disaster plan, you should take into consideration all of your computing infrastructure.

Next topics

How to determine a disaster on page 32 *What tools to use to protect against data-loss disasters* on page 33

How to determine a disaster

It is critical that you follow some predefined procedure to confirm whether a disaster really has occurred.

You can use any of the following procedures to determine the status of the supposed disaster site.

- Use the following external interfaces:
 - Ping
 - Remote shell

- FilerView
- Protection Manager
- Use network management tools to test connectivity to the site.
- Physically inspect the site, if possible.

What tools to use to protect against data-loss disasters

Data ONTAP provides features and products that enable the administrator to back up or replicate data stored on a local storage system to an off-site network location. This ensures that data can be restored if data loss is caused by disaster at a primary data storage site.

The following table describes the Data ONTAP tools that enable data protection.

Feature	Description
SnapVault : Inter-site Snapshot copy backup and restorability	A SnapVault secondary system can be located off-site, any distance from the primary storage systems that it is backing up.
	Data recoverability : In the event of a data-loss disaster at a primary storage site, data that is backed up to the SnapVault secondary storage can be restored to primary storage systems that have not suffered physical damage or that have been replaced either at the disaster site or at an alternate location.
	Currency of restore : Data can be restored from the time that the last Snapshot copy was created on the secondary system.
	Connection requirements : DSL connections or faster are recommended between the primary and secondary systems. Even connections with very low bandwidth, such as 56 Kbps, are possible.
	Routers, switches, and DNS servers should be preconfigured to direct users to alternate storage sites if the primary system that they first attempt to access becomes unavailable.
	Advantage: Centralized, inexpensive off-site backup

Feature	Description
SnapMirror : Inter-site Snapshot copy backup, availability, and restorability	A SnapMirror destination system can be located off-site, any distance from the source system whose volumes it is replicating.
	Data availability : In the event of a data-loss disaster at a source site, SnapMirror data at the destination site can be made quickly available.
	Data recoverability : SnapMirror data can be restored to source storage systems that have not suffered physical damage or that have been replaced either at the disaster site or at an alternate location.
	Currency of restore : Data can be restored from the time of the last SnapMirror Snapshot copy transfer from source to destination.
	Connection requirements : DSL connections or faster are recommended between the source and destination. Even connections with very low bandwidth, such as 56 Kbps, are possible. Routers, switches, and DNS servers should be preconfigured to direct users to the destination storage site if the source they are attempting to access becomes unavailable.
	Advantage: Combined off-site protection and availability
MetroCluster : Inter-site realtime backup, availability, and restorability	Synchronously mirrored MetroCluster storage systems can be located at different sites, up to 100 km from one another. For V-Series systems, Fabric MetroCluster support is limited to 30 km.
	Data availability : In the event of a data-loss disaster at one site, data that has been mirrored to the partner site can be made quickly available.
	Data recoverability : This data can also be mirrored to source storage units that have not suffered physical damage or that have been replaced either at the disaster site or at an alternate location.
	Currency of restore : Data can be restored from the time of the last NVRAM checkpoint.
	Connection requirements : Data ONTAP cluster connections supplemented with switches and DSL or faster connections are required. Routers, switches, and DNS servers should be preconfigured to direct users to the MetroCluster partner if the clustered system that they first attempt to access becomes unavailable.
	Advantage: Combined real-time off-site protection and availability.

For more information about MetroCluster, see the Data ONTAP Active/Active Configuration Guide.

Related concepts

Data protection using SnapVault on page 225 Data protection using SnapMirror on page 81 What a Snapshot copy is on page 37

How to protect data in a SAN environment

If the volumes contain logical units of storage (LUNs) created to enable integration into a storage area network (SAN) environment, the procedures to implement data protection might have to be modified.

For descriptions of data backup and restore on volumes containing Data ONTAP LUNs, see the *Data* ONTAP Block Access Management Guide for iSCSI and FC.

Policy-based automated data protection using Protection Manager

Typically, data and resource management is time consuming because it involves manual analysis and management of storage capacity, network bandwidth, schedules, retention policies, and other infrastructure variables. Protection Manager simplifies this work by employing configuration policies (that you can assign to multiple storage systems, volumes, or qtrees), convenient wizards, and automated verification of certain aspects of the data protection configuration.

The protection application can perform the following actions:

- Use protection policies to manage primary data, storage, and backup and mirror relationships.
- Manage local and remote backups and mirror copies.
- Provision the secondary storage for backups and mirrored copies based on policies you assign.
- Enable disaster recovery capability if you install the licensed disaster recovery option.
- Automatically validate your backup and disaster recovery configuration with a conformance checker.

For details, see the Provisioning Manager and Protection Manager Administration Guide.
Snapshot management

Data ONTAP maintains a configurable Snapshot schedule that creates and deletes Snapshot copies automatically for each volume. You can also create and delete Snapshot copies, and manage Snapshot schedules based on your requirements.

Next topics

What a Snapshot copy is on page 37 User access to Snapshot copies on page 39 How Data ONTAP Snapshot copies work in an iSCSI or FC network on page 42 Restoring files from Snapshot copies on page 44 Creation of Snapshot copy schedules on page 44 Creating Snapshot copies manually on page 50 Why you might need to access a particular Snapshot copy on page 50 What Snapshot disk consumption is on page 52 What file folding means and how it saves disk space on page 61 Displaying the rate of change between Snapshot copies on page 62 Deleting Snapshot copies automatically on page 64 Displaying space reclaimed from deleted Snapshot copies on page 67 How to determine which Snapshot copies to delete on the basis of size on page 67 Renaming Snapshot copies on page 70

What a Snapshot copy is

A Snapshot copy is a read-only image of a traditional volume, a FlexVol volume, or an aggregate that captures the state of the file system at a point in time. Snapshot copies are your first line of defense to back up and restore data.

For information about traditional volumes, FlexVol volumes, or aggregates, see the *Data ONTAP Storage Management Guide*.

Data ONTAP maintains a configurable Snapshot copy schedule that creates and deletes Snapshot copies automatically for each volume. Snapshot copies can also be created and deleted manually.

You can store up to 255 Snapshot copies at one time on each volume.

You can specify the percentage of disk space that Snapshot copies can occupy. The default setting is 20 percent of the total (both used and unused) space on the disk.

Next topics

How Snapshot copies handle file permissions on page 38

What you can do with Snapshot copies on page 38

Related concepts

Creation of Snapshot copy schedules on page 44 *What Snapshot disk consumption is* on page 52

How Snapshot copies handle file permissions

Snapshot files carry the same permissions and inode numbers as the original files, keeping the integrity of the security system intact.

Inodes are data structures that hold information (including permissions information) about files on the storage system. Every file in the file system is uniquely identified by its inode in the same file system.

Note: The inode number for a file in a Snapshot copy is the same as the inode number for the corresponding file in the active file system. As a result, some programs on UNIX clients consider the two files to be the same. For example, if you use an older version of the GNU diff program to compare the two files, it might not find any differences between them. However, newer versions of GNU diff should work. In a few cases, if you try to restore a file from a Snapshot copy, you might see the following error message:

cp:.snapshot/xxx and xxx are identical.

To ensure that the two files have different inode numbers before the copying or comparison, copy one of the files to a different name.

What you can do with Snapshot copies

Snapshot copies enable system administrators and end users to perform important tasks in backup and recovery.

Snapshot copies enable system administrators to perform the following tasks:

- Create instantaneous backups
- Create a clone of a FlexVol volume
- Create a clone of a Data ONTAP LUN

For information about cloning a Data ONTAP LUN, see the *Data ONTAP Block Access Management Guide for iSCSI and FC*.

For information about cloning a FlexVol volume, see the Data ONTAP Storage Management Guide.

Snapshot copies enable end users to do the following:

- · Recover older versions or sets of files that were accidentally changed or deleted
- · Restore their own files without needing a system administrator to restore files from tape

User access to Snapshot copies

By default, every volume contains a directory named .snapshot through which users can access old versions of files in that directory. Users can gain access to Snapshot copies depending on the file-sharing protocol used—NFS or CIFS. Access to Snapshot copies can be turned off.

Snapshot files carry the same read permissions as the original file. A user who has permission to read a file in the volume can read that file in a Snapshot copy. A user without read permission to the volume cannot read that file in a Snapshot copy. Snapshot copies do not have write permissions.

Next topics

Access to Snapshot copies over NFS on page 39 Access to Snapshot copies over CIFS on page 40 Accessing Snapshot copies from CIFS clients on page 41 Restricting access to Snapshot copies on page 41

Access to Snapshot copies over NFS

On an NFS client, the user can obtain access to Snapshot copies.

The following illustration shows the directory structure on an NFS client with the volo volume named toaster mounted on the /n/toaster directory.



Figure 1: Directory structure of an NFS client

In this example, the user can obtain access to Snapshot copies in the /n/toaster/.snapshot directory. Notice that the .snapshot directory is shown only at the mountpoint, although it actually

exists in every directory in the tree. The user, however, can only see the .snapshot directory at the mountpoint. That is, the .snapshot directory is accessible by name in each directory, but is only seen in the output of the ls command at the mountpoint.

For example, at the mountpoint of a file system, a directory listing looks like this:

```
systemA> ls -a
. .. .snapshot dir1 dir2
```

The same command entered in a directory below the mountpoint does not show the .snapshot directory; for example:

```
systemA> cd dir1
systemA> ls -a
. .. file1 file2
```

If you enter the 1s command with the directory name .snapshot, you can see a directory for each of the Snapshot copies for the dir1 directory:

```
systemA> ls .snapshot
hourly.0 hourly.4 nightly.0 nightly.4
hourly.1 hourly.5 nightly.1 nightly.5
hourly.2 hourly.6 nightly.2 weekly.0
hourly.3 hourly.7 nightly.3 weekly.1
```

If the .snapshot directory entry appeared in every directory, it would cause many commands to work improperly. For instance, all recursive commands for deleting files would fail because everything below the .snapshot directory is read-only. The recursive commands would copy everything in the Snapshot copies as well as files in the active file system. A find command would generate a list much longer than expected.

Access to Snapshot copies over CIFS

By default, CIFS users cannot see the .snapshot directory. To allow CIFS users to see the .snapshot directory, you can set the cifs.show_snapshot option to on.

To CIFS users, the .snapshot directory appears only at the root of a share. For example, if a user's home directory is a share named bill that corresponds to the /vol/vol0/home/bill directory, only the /vol/vol0/home/bill/.snapshot directory is visible. When this user displays the contents of the home directory, the .snapshot directory is displayed as ~snapshot if the operating system supports long file names and as ~SNAPSHT if the operating system supports only short file names.

Note: The .snapshot directory can be viewed in a directory listing or Windows Explorer display if the client operating system is configured to show hidden files.

In each directory within the share, a snapshot directory exists but is not visible to clients. For example, if the client operating system supports long file names, the applications on that operating system can use the Snapshot copy at each level of the share by using .snapshot, ~snapshot, or

 \sim SNAPSHT as the directory name. The user cannot, however, display the directory name in any listing.

Accessing Snapshot copies from CIFS clients

You can access the data within Snapshot copies from CIFS clients.

Before you begin

Ensure that the cifs.show_snapshot option is set to on.

Step

1. To access Snapshot copies on Windows NT 4 or other Windows clients (Windows 95 or later), click **Start > Run menu**, then enter the following command:

\\systemname\share\.snapshot (or ~snapshot or ~SNAPSHT)

systemname is the name of the storage system you are using.

share is the name of the Windows share that you want to access.

Example

\\systemA\home\.snapshot

Snapshot copies can also be accessed lower in the share by providing a path to a lower directory. Snapshot copies can be accessed through DOS on any system by changing to the ~SNAPSHT directory.

Restricting access to Snapshot copies

You can restrict client access to Snapshot copies for a particular volume. This restriction could be due to security issues, or to prevent access to virus-infected files. You can enable or disable client access by using the vol options command.

Step

1. To specify client access to Snapshot copies within a volume, enter the following command.

vol options vol_name nosnapdir {on|off}

vol_name is the name of the volume for which you want to set client access to Snapshot copies.

You can set the nosnapdir option to either on or off.

- on Disables client access to Snapshot copies, and hides the .snapshot directory from clients.
- off Enables client access to Snapshot copies, and makes the .snapshot directory visible to clients.

Disabling client access to Snapshot copies

To disable client access to Snapshot copies within the volA volume, enter the following command.

vol options volA nosnapdir on

How Data ONTAP Snapshot copies work in an iSCSI or FC network

If you take a Snapshot copy of a file system when an application is running, the Snapshot copy might contain inconsistent data. You can take measures (such as quiescing the application) to ensure that the data is consistent before you take the Snapshot copy.

To take a Snapshot copy of these types of applications, you should ensure that the files are closed and cannot be modified. When you quiesce an application or take it offline, the file system caches are committed before the Snapshot copy is taken. The Snapshot copy takes less than one second to complete, after which the application can resume normal operation.

Some applications take a lot of time to quiesce. To avoid a scenario in which the application is unavailable for a long time, some applications have a built-in hot backup mode. This allows a Snapshot copy or a backup to occur while the application operates in a degraded mode, with limited performance.

Data ONTAP cannot take Snapshot copies of applications that have the ability to work with raw device partitions. It is recommended that you use specialized modules from a backup software vendor tailored for such applications.

To back up raw partitions, it is best to use hot backup mode for the duration of the backup operation. For more information about backup and recovery of databases using SAN configurations, see the appropriate technical report for the database.

Next topics

Using Snapshot copies in the SAN environment on page 42 Relationship between a LUN and a Snapshot copy on page 43

Related information

www.netapp.com/tech_library/

Using Snapshot copies in the SAN environment

You can use Snapshot copies in the SAN environment when the data within a Data ONTAP LUN is in a consistent state.

About this task

Data ONTAP cannot ensure that the data within a LUN is in a consistent state. That is, Data ONTAP does not know whether an application is accessing the data inside the LUN. Therefore, before

creating a Snapshot copy, you need to quiesce the application or file system using the LUN. This action flushes the host file system buffers to disk. Quiescing ensures that the Snapshot copy is consistent.

One way to accomplish this is to use batch files and scripts on a host that has administrative access to the system.

Steps

- **1.** Make the data within the LUN consistent with the application by quiescing a database, placing the application in hot backup mode, or taking the application offline.
- 2. Use the rsh or ssh command in the script to access the system.
- **3.** Use the snap command to create the Snapshot copy on the system (this takes only a few seconds, regardless of volume size or use).
- **4.** Return the application to normal operation.

Snapshot copy scripts can be scheduled to run at specified intervals. On Windows hosts, you can use the Windows Task Scheduler. On UNIX hosts, you can use cron or other utilities. Also, you can use SnapDrive to save the contents of the host file system buffers to disk and to create Snapshot copies. For more information, see the *SnapDrive Installation and Administration Guide*.

Relationship between a LUN and a Snapshot copy

When you take a Snapshot copy of a Data ONTAP LUN, the Snapshot copy is initially backed by data in the LUN. After the Snapshot copy is taken, data written to the LUN is available in the active file system.

After you have a Snapshot copy, you can use it to create a LUN clone for temporary use as a prototype for testing data or scripts in applications or databases. Because the LUN clone is backed by the Snapshot copy, you cannot delete the Snapshot copy until you split the clone from it.

To restore the LUN from a Snapshot copy, you can use SnapRestore. However, the restored LUN does not have any updates to the data since the Snapshot copy was taken.

When you create the LUN, space reservation is enabled by default. This means that enough space is reserved so that write operations to the LUNs are guaranteed. The more space that is reserved, the less free space is available. If free space within the volume is below a certain threshold, Snapshot copies cannot be taken.

Restoring files from Snapshot copies

You might need to restore a file from a Snapshot copy if the file was accidentally erased or corrupted. If you have purchased the SnapRestore option, you can automatically restore files or volumes from Snapshot copies with one command.

Steps

- 1. If the original file still exists and you do not want it overwritten by the file in a Snapshot copy, then use your UNIX or Windows client to rename the original file or move it to a different directory.
- 2. Locate the Snapshot copy that contains the version of the file you want to restore.
- 3. Copy the file from the .snapshot directory to the directory in which the file originally existed.

Related concepts

When to use SnapRestore on page 72

Related tasks

Reverting a file to a selected Snapshot copy on page 77

Snapshot restoration using Shadow Copy Client tools

You can access and restore Data ONTAP Snapshot files using the Windows Shadow Copy Client. The Shadow Copy Client provides a Previous Versions tab in the Properties menu from which you can view and restore Data ONTAP Snapshot images.

The Shadow Copy Client software for Windows 2003 is called the Previous Versions Client. Downloads available from Microsoft allow you to use Shadow Copy client tools on most older versions of Windows. For more information about Shadow Copy Client or Previous Versions Client software, consult the Microsoft documentation.

Creation of Snapshot copy schedules

Data ONTAP provides a default Snapshot copy schedule for each volume. You can configure the schedule to fit your needs. The schedule creates Snapshot copies automatically, and deletes older Snapshot copies after a specified period.

The default Snapshot copy schedule automatically creates one nightly Snapshot copy Monday through Saturday at midnight, and four hourly Snapshot copies at 8 a.m., noon, 4 p.m., and 8 p.m. Data ONTAP retains the two most recent nightly Snapshot copies and the six most recent hourly Snapshot copies, and deletes the oldest nightly and hourly Snapshot copies when new Snapshot copies are created.

Next topics

User-specified Snapshot copy schedules on page 45 Snapshot copy schedule conflicts on page 46 Viewing the Snapshot copy schedule using the CLI on page 46 Displaying the Snapshot schedule using FilerView on page 46 What the snap sched command arguments mean on page 47 Strategies for creating a Snapshot copy schedule on page 47 Changing the Snapshot copy schedule on page 48 Enabling or disabling automatic Snapshot copies on page 49

Related tasks

Changing the Snapshot copy schedule on page 48

User-specified Snapshot copy schedules

You can configure weekly, nightly, or hourly Snapshot copy schedules using the snap sched command.

Туре	Description
Weekly	Data ONTAP creates these Snapshot copies every Sunday at midnight.
	Weekly Snapshot copies are named weekly. <i>n</i> , where <i>n</i> is an integer. The most recent weekly Snapshot copy is weekly. 0, weekly. 1 is the next most recent weekly Snapshot copy, and so on.
Nightly	Data ONTAP creates these Snapshot copies every night at midnight, except when a weekly Snapshot copy is scheduled to occur at the same time.
	Nightly Snapshot copies are named nightly. <i>n</i> , where <i>n</i> is an integer. The most recent nightly Snapshot copy is nightly. 0, nightly.1 is the next most recent nightly Snapshot copy, and so on.
Hourly	Data ONTAP creates these Snapshot copies on the hour or at specified hours, except if a weekly or nightly Snapshot copy is scheduled to occur at the same time.
	Hourly Snapshot copies are named hourly. <i>n</i> , where <i>n</i> is an integer. The most recent hourly Snapshot copy is hourly. 0, hourly. 0 is the next most recent hourly Snapshot copy, and so on.

The following table describes the available types of Snapshot copy schedules.

When Data ONTAP creates a weekly, nightly, or hourly Snapshot copy, the value of n is adjusted for all the weekly, nightly, or hourly Snapshot copies; the earlier Snapshot copies in the series are renamed. The higher the value of n, the older the Snapshot copy.

Snapshot copy schedule conflicts

If SnapMirror or SnapVault is scheduled to perform Snapshot copy management at the same time as a snap sched command operation, then the Snapshot copy management operations scheduled using the snap sched command might fail with syslog messages.

The syslog messages are:

Skipping creation of hourly snapshot

and

```
Snapshot already exists
```

To avoid this condition, you should stagger the Snapshot copy update schedules so that SnapMirror activity does not begin or end at the exact minute a snap sched operation attempts to create a Snapshot copy. Additionally, if snap sched Snapshot copies conflict with SnapVault activity, you should use the snapvault snap sched command to configure equivalent schedules.

Viewing the Snapshot copy schedule using the CLI

You can view the Snapshot copy schedule for a volume by using the snap sched command.

Step

1. To view the Snapshot copy schedule for a volume, enter the following command:

snap sched [vol_name]

Note: If you do not specify a volume name, *snap sched* displays the Snapshot copy schedule for each volume on the system.

Example

```
systemA> snap sched vol1
Volume vol1: 2 6 8@8,12,16,20
```

Displaying the Snapshot schedule using FilerView

You can use FilerView to display the Snapshot schedule for a volume.

Steps

- 1. In FilerView, click Volumes in the list on the left.
- 2. In the list under Volumes, click **Snapshots**.
- 3. In the list under Snapshots, click Configure.

What the snap sched command arguments mean

The snap sched command arguments allow you to specify the schedule for creating Snapshot copies. You can also specify the number of Snapshot copies to be retained.

The following illustration explains the arguments in a sample snap sched command output.



Figure 2: Arguments in a sample snap sched command output

Snapshot copy schedule results: The command shown in the illustration keeps the two most recent weekly Snapshot copies, the six most recent nightly Snapshot copies, and the eight most recent hourly Snapshot copies, created at 8 a.m., 12 p.m., 4 p.m., and 8 p.m. every day. Whenever the Snapshot copy schedule creates a new Snapshot copy of a particular type, it deletes the oldest Snapshot copy and renames the existing Snapshot copies. On the hour, for example, the system deletes hourly.7, renames hourly.0 to hourly.1, and so on.

Note: If you omit the @ argument specifying the hours for the hourly Snapshot copies, Data ONTAP creates a Snapshot copy every hour. Nightly and weekly Snapshot copies are always created at midnight.

Strategies for creating a Snapshot copy schedule

You should create a Snapshot copy schedule that meets the needs of your organization and users.

Following are some strategies for scheduling and retaining Snapshot copies:

• If users rarely lose files or typically notice lost files right away, you can use the default Snapshot copy schedule. This schedule creates no weekly Snapshot copy; it creates a Snapshot copy every night and keeps two; and it creates hourly Snapshot copies at 8 a.m., 12 p.m., 4 p.m., and 8 p.m, and keeps six. Following is the default Snapshot copy schedule command:

```
snap sched vol_name 0 2 6@8,12,16,20
```

• If users commonly lose files or do not typically notice lost files right away, you should delete the Snapshot copies less often than you would if you used the default schedule.

Following is the recommended Snapshot copy schedule for this situation. It keeps two weekly Snapshot copies, six nightly Snapshot copies, and eight hourly Snapshot copies:

snap sched vol1 2 6 8@8,12,16,20

On many systems, only 5 to 10 percent of the data changes each week, so the Snapshot copy schedule of six nightly and two weekly Snapshot copies consumes 10 to 20 percent of disk space. Considering the benefits of Snapshot copies, it is worthwhile to reserve this amount of disk space for Snapshot copies.

• You can create different Snapshot copy schedules for different volumes on a storage system. On a very active volume, you should schedule Snapshot copies every hour and keep them for just a few hours, or turn off Snapshot copies. For example, the following schedule creates a Snapshot copy every hour and keeps the last three:

snap sched vol2 0 0 3

This schedule does not consume much disk space, and it lets users recover files in recent Snapshot copies as long as they notice their mistake within a couple of hours.

• When you create a new volume, the new volume inherits the Snapshot copy schedule from the root volume. After you use the volume for a while, you should check how much disk space the Snapshot copies consume and how often users need to recover lost files, and then adjust the schedule as necessary.

Related concepts

What Snapshot disk consumption is on page 52

Changing the Snapshot copy schedule

You can change the Snapshot copy schedule for a specific volume by using the snap sched command.

Step

1. To change the Snapshot copy schedule for a specific volume, enter the following command:

snap sched vol_name weekly nightly hourly@n,n,....

vol_name is the name of the specific volume for the Snapshot copy.

weekly is the number of weekly Snapshot copies to keep.

nightly is the number of nightly Snapshot copies to keep.

hourly is the number of hourly Snapshot copies to keep.

n,*n*,... specifies the hours at which to create the hourly Snapshot copies.

Note: A zero in any of the three schedules (weekly, nightly, hourly) disables Snapshot copies for that interval.

Default snap sched command results

This is the default automatic Snapshot copy schedule:

snap sched volx 0 2 6 @8,12,16,20

The following example lists the Snapshot copies created using the default schedule (where January 11 is a Sunday):

```
ls -lu .snapshot
total 64
drwxrwsrwx 1 root 4096 Jan 14 12:00 hourly.0
drwxrwsrwx 1 root 4096 Jan 14 08:00 hourly.1
drwxrwsrwx 1 root 4096 Jan 13 20:00 hourly.2
drwxrwsrwx 1 root 4096 Jan 13 16:00 hourly.3
drwxrwsrwx 1 root 4096 Jan 13 12:00 hourly.4
drwxrwsrwx 1 root 4096 Jan 13 08:00 hourly.5
drwxrwsrwx 1 root 4096 Jan 14 00:00 nightly.0
drwxrwsrwx 1 root 4096 Jan 13 00:00 nightly.1
```

Note: Daily Snapshot copies are created at midnight of each day except Sunday, and weekly Snapshot copies are created at midnight on Sunday. Only one Snapshot copy is created at a time. If a weekly Snapshot copy is being created, for instance, a daily or hourly Snapshot copy is not created even if one is scheduled.

Enabling or disabling automatic Snapshot copies

You can disable the creation of automatic Snapshot copies, without changing the Snapshot copy schedule. Later, you can enable the Snapshot copy schedule.

Steps

1. To view whether the creation of automatic Snapshot copies is disabled, enter the following command:

vol options vol_name

vol_name is the name of the volume for which you want to view the value of the nosnap option.

nosnap=off indicates that the creation of automatic Snapshot copies is enabled. off is the default value for the nosnap option.

nosnap=on indicates that the creation of automatic Snapshot copies is disabled.

2. To change the value of the nosnap option, enter the following command:

vol options vol_name nosnap {on off}

Creating Snapshot copies manually

You might need to create Snapshot copies outside a specified schedule. You can create Snapshot copies manually by using the snap create command.

Step

1. To create a Snapshot copy manually, enter the following command:

```
snap create vol_name snapshot_name
```

vol_name is the name of the volume on which you want to create the Snapshot copy.

snapshot_name is the name you want to give the Snapshot copy.

Note: The snap create command does not accept a Snapshot copy name containing a slash (/). Therefore, you cannot enter a specific path for the Snapshot copy.

Creating Snapshot copies using FilerView

You can manually create Snapshot copies using FilerView.

Steps

- 1. In FilerView, click Volumes in the list on the left.
- 2. In the list under Volumes, click **Snapshots**.
- 3. In the list under Snapshots, click Add.

Why you might need to access a particular Snapshot copy

You might need to access an earlier version of a file, because a file was changed, corrupted, or erased. You might notice the issue only after one or more Snapshot copies were created with the incorrect copy of the file. When looking for the version of the file you need, you should look for it by means of the creation time of the Snapshot copy.

The *version* of a file refers to the last time the file was modified before a Snapshot copy was created. The *access time* of a file refers to the Snapshot copy creation time for a file, regardless of whether any modifications were made to that file.

The best way to find all versions of a particular file preserved in Snapshot copies is to use the ls command from an NFS client, or to use the search functionality in Windows.

Next topics

Finding the Snapshot copy you need from an NFS client on page 51 *Determining access times from an NFS client* on page 51 *Finding the Snapshot copy you need from a CIFS client* on page 52 *How to determine access times from a CIFS client* on page 52

Finding the Snapshot copy you need from an NFS client

The best way to find all versions of a particular file preserved in Snapshot copies is to use the ls command from an NFS client.

Step

1. To find all the versions of a particular file in the Snapshot copies, enter the following command:

```
ls -l filename .snapshot/*/file_name
```

A list is displayed of all versions of the requested file.

Accessing Snapshot copies from an NFS client

ls -l myfile.txt .snapshot/*/myfile.txt

-rw-r--r- 1 smith 0 Jan 14 09:40 myfile.txt
-rw-r--r- 1 smith 0 Jan 13 18:39 .snapshot/nightly.0/myfile.txt
-rw-r--r- 1 smith 0 Jan 12 19:17 .snapshot/nightly.1/myfile.txt

The version of myfile.txt in the active file system was last modified on January 14, but the old versions available in the Snapshot copies were modified on January 13 and January 12. You can use standard UNIX commands to read the earlier versions of myfile.txt. However, you cannot modify or delete these older versions because everything in the .snapshot directory is read-only.

Determining access times from an NFS client

When Data ONTAP creates a Snapshot copy, the access time of each file in the Snapshot copy is updated to the Snapshot copy creation time. You can use the ls -lu command to see the access times.

Step

1. To determine when Snapshot copies were created, from an NFS client, enter the following command:

```
ls -lu filename .snapshot/*/file_name
```

ls -lu myfile.txt .snapshot/*/myfile.txt

```
-rw-r--r- 1 smith 0 Jan 14 09:40 myfile.txt
-rw-r--r- 1 smith 0 Jan 14 00:00 .snapshot/nightly.0/myfile.txt
-rw-r--r- 1 smith 0 Jan 13 00:00 .snapshot/nightly.1/myfile.txt
```

Note: On a UNIX client, the times listed by the ls -l command reflect the modification times of the directory at the time of each Snapshot copy, and are not related to the times at which the Snapshot copies are created.

Finding the Snapshot copy you need from a CIFS client

The best way to find all versions of a particular file preserved in Snapshot copies is to use the search functionality in Windows.

Steps

 To find all the versions of a particular file in the Snapshot copies, click Start > Search > For Files or Folders... in the Windows Start menu.

A search window opens prompting you for a directory and file name.

2. In the search window, enter the file name to search for in the ~snapshot directory.

CIFS client example

If you map the home share to drive F: and want to find all versions of myfile.txt in Snapshot copies, you can use the Windows search functionality to search for myfile.txt in the f:\~snapshot folder.

How to determine access times from a CIFS client

You can determine the access time of a file from a CIFS client by checking the file's properties.

What Snapshot disk consumption is

Data ONTAP preserves pointers to all the disk blocks currently in use at the time the Snapshot copy is created. When a file is changed, the Snapshot copy still points to the disk blocks where the file existed before it was modified, and changes are written to new disk blocks.

Next topics

How Snapshot copies consume disk space on page 53 How changing file content consumes disk space on page 53 Monitoring Snapshot copy disk consumption on page 54 Displaying Snapshot copy disk consumption statistics on page 55 What the Snapshot copy reserve is on page 58 Changing the Snapshot copy reserve on page 61

How Snapshot copies consume disk space

Snapshot copies minimize disk consumption by preserving individual blocks rather than whole files. Snapshot copies begin to consume extra space only when files in the active file system are changed or deleted. When this happens, the original file blocks are still preserved as part of one or more Snapshot copies.

In the active file system the changed blocks are rewritten to different locations on the disk or removed as active file blocks entirely. As a result, in addition to the disk space used by blocks in the modified active file system, disk space used by the original blocks is still reserved to reflect the status of the active file system before the change.

The following illustration shows disk space usage for a Snapshot copy.



Figure 3: How Snapshot copies consume disk space before and after you delete a file myfile.txt.

How changing file content consumes disk space

A given file might be part of a Snapshot copy. The changes to such a file are written to new blocks. Therefore, the blocks within the Snapshot copy and the new (changed or added) blocks both use space within the volume.

Changing the contents of the myfile.txt file creates a situation where the new data written to myfile.txt cannot be stored in the same disk blocks as the current contents because the Snapshot copy is using those disk blocks to store the old version of myfile.txt. Instead, the new data is

written to new disk blocks. As the following illustration shows, there are now two separate copies of myfile.txt on disk—a new copy in the active file system and an old one in the Snapshot copy.



Figure 4: Two separate copies of myfile.txt on disk

Monitoring Snapshot copy disk consumption

You can monitor Snapshot copy disk consumption using the df command, which displays the amount of free space on a disk. The df command treats Snapshot copies as a partition different from the active file system.

Step

1. To display information about Snapshot copy disk consumption, enter the following command:

df

Example

 systemA> df

 Filesystem
 kbytes used avail capacity

 /vol/vol0
 3000000
 2000000
 1000000
 65%

 /vol/vol0/.snapshot
 1000000
 500000
 50%

In the example, the kbytes column shows that the vol0 volume contains 3,000,000 KB (3 GB) of disk space for the active file system and 1,000,000 KB (1 GB) of disk space reserved for Snapshot copies, for a total of 4,000,000 KB (4 GB) of disk space. In this example, 66 percent of the active disk space is used (which means that 34 percent is available). Note that the capacity

percentage is rounded to 65 percent. The 1,000,000 KB (1 GB) of disk space for Snapshot copies represents 25 percent of the volume capacity, of which 500,000 KB (0.5 GB) is used and 500,000 KB (0.5 GB) is available, so that the space for Snapshot copies is at 50 percent capacity.

Note: The 50 percent figure is not 50 percent of disk space, but 50 percent of the space allotted for Snapshot copies. If this allotted space is exceeded, this number will be over 100 percent.

It is important to understand that the /vol/vol0/.snapshot line counts data that exists only in a Snapshot copy. The Snapshot copy calculation does not include Snapshot copy data that is shared with the active file system.

Displaying Snapshot copy disk consumption statistics

You can use the snap list command to view the disk space utilization by Snapshot copies. This information helps you determine an appropriate Snapshot copy reserve. The snap list command also shows whether a Snapshot copy is required for a replication operation, such as SnapMirror.

Step

1. To display the Snapshot copy statistics for a volume, enter the following command:

snap list vol_name

vol_name is the name of the volume for which you want statistics.

If you do not specify a volume name in the command, the output contains statistics about each volume in the system.

Example

The following example gives a sample output of the snap list command.

systemA> snap list vol0

Vol	lur	ne vol	.0						
%/used			%∕t	%/total			9		name
0%	(0%)	0%	(0%)	Jan	19	08:01	hourly.0
1%	(1%)	1%	(1%)	Jan	19	00:01	nightly.0
2%	(2%)	2%	(2%)	Jan	18	20:01	hourly.1
38	(28)	2%	(28)	Jan	18	16:01	hourly.2
3%	(2%)	3%	(2%)	Jan	18	12:01	hourly.3
5%	(3%)	4%	(3%)	Jan	18	00:01	nightly.1
7%	(4%)	6%	(4%)	Jan	17	00:00	nightly.2
8%	(4%)	7%	(4%)	Jan	16	00:01	nightly.3
10%	5 (5%)	9%	(48)	Jan	15	00:01	nightly.4

Next topics

How the snap list output is calculated on page 56 How to use cumulative Snapshot copy values on page 57 Displaying Snapshot copy use and dependencies on page 57 snap list performance after a snap restore file operation on page 58

How the snap list output is calculated

The snap list output is calculated in a number of ways.

The %/used column shows space consumed by Snapshot copies as a percentage of disk space being used in the volume. The first number is cumulative for all Snapshot copies listed so far, and the second number is for the specified Snapshot copy alone.

• The first number is equal to

100% x cumulative Snapshot copy space cumulative Snapshot copy space + file system space

• The second number is equal to

The %/total column shows space consumed by Snapshot copies as a percentage of total disk space (both space used and space available) in the volume.

• The first number is equal to

Cumulative Snapshot copy space is the total space used by this Snapshot copy and all other more recent Snapshot copies (the ones preceding this Snapshot copy in the snap list output).

• The second number is equal to

Summary of the snap list command output: The **%/used** number is useful for planning the Snapshot copy reserve because it is more likely to remain constant as the file system fills.

The information shows a volume that keeps five nightly Snapshot copies and four hourly Snapshot copies.

The sample output shows that the overhead for Snapshot copies is only 10 percent, so the default Snapshot copy reserve of 20 percent seems to be a waste of disk space. If this pattern of change holds, a reserve of 12 percent to 15 percent provides a safe margin to ensure that deleting files frees disk space when the active file system is full.

The values in parentheses, which show the space used by an individual Snapshot copy, are useful in identifying a particular Snapshot copy to delete when the file system is full. However, deleting a particular Snapshot copy does not necessarily release the amount of disk space indicated, because other Snapshot copies might be referring to the same blocks.

Related concepts

What the Snapshot copy reserve is on page 58

How to use cumulative Snapshot copy values

If you want the amount of disk space consumed by all Snapshot copies not to exceed a certain percentage of the used disk space, you can use the cumulative values in the snap list command output to determine which Snapshot copies to delete.

For example, if you do not want more than 5 percent of used disk space to be spent by Snapshot copies, you can delete all Snapshot copies listed below nightly.1 in the snap list output; that is, nightly.2, nightly.3, and nightly.4. After deleting the Snapshot copies, nightly.1 and all the other more recent Snapshot copies consume 5 percent of the used disk space.

Displaying Snapshot copy use and dependencies

The output of the snap list command shows whether a Snapshot copy is being actively used by an application or a replication operation.

Step

1. To view the list of Snapshot copies, enter the following command:

snap list vol_name

Example

A sample output is given in the following example.

systemA> snap list vol1

7% (4%) 6% (4%) Jan 17 00:00 nightly.2 8% (4%) 7% (4%) Jan 16 00:01 nightly.3 10% (5%) 9% (4%) Jan 15 00:01 nightly.4

The snap list command displays the name of an application next to a Snapshot copy name if the application needs the Snapshot copy currently or at a later time. For example, backup is displayed next to the Snapshot copy name to show that the Snapshot copy is the result of a dump command transfer that was interrupted but is restartable. The number following backup is the backup ID assigned by the backup status command. The notation snapmirror next to the Snapshot copy name means that SnapMirror is retaining the Snapshot copy to maintain a sourcedestination relationship.

Note: Ownership information for a busy Snapshot copy is useful for determining whether to stop the activity in progress. For example, if the snap list command output displays a locked Snapshot copy that is imposing a resource constraint, you can delete that Snapshot copy and free up space.

snap list performance after a snap restore file operation

If you restore files with the snap restore command, and then issue the snap list command, the snap list command can take up to several minutes to complete.

This condition persists until the Snapshot copy from which you restored the file is purged from the system after reaching the end of its normal Snapshot retention cycle.

Related tasks

Reverting a file to a selected Snapshot copy on page 77

What the Snapshot copy reserve is

The Snapshot copy reserve sets a specific percent of disk space for Snapshot copies. By default, the Snapshot copy reserve is 20 percent of disk space. The Snapshot copy reserve can be used only by Snapshot copies, not by the active file system.

If the active file system runs out of disk space, any disk space still remaining in the Snapshot copy reserve is not available for use by the active file system.

Note: Although the active file system cannot consume disk space reserved for Snapshot copies, Snapshot copies can exceed the Snapshot copy reserve and consume disk space normally available to the active file system.

Managing the Snapshot copy reserve involves the following tasks:

- Ensuring that enough disk space is set aside for Snapshot copies so that they do not consume active file system space
- Keeping disk space consumed by Snapshot copies below the Snapshot copy reserve
- Ensuring that the Snapshot copy reserve is not so large that it wastes space that could be used by the active file system

Next topics

Use of deleted active file disk space on page 59 Snapshot copies can exceed reserve on page 60 Recovery of disk space for file system use on page 60

Use of deleted active file disk space

When enough disk space is available for Snapshot copies in the Snapshot copy reserve, deleting files in the active file system frees disk space for new files, while the Snapshot copies that reference those files consume only the space in the Snapshot copy reserve.

If Data ONTAP created a Snapshot copy when the disks were full, deleting files from the active file system does not create any free space because everything in the active file system is also referenced by the newly created Snapshot copy. Data ONTAP has to delete the Snapshot copy before it can create any new files.

Example

The following example shows how disk space being freed by deleting files in the active file system ends up in the Snapshot copy:

If Data ONTAP creates a Snapshot copy when the active file system is full and there is still space remaining in the Snapshot reserve, the output from the df command—which displays statistics about the amount of disk space on a volume—is as follows:

 Filesystem
 kbytes
 used
 avail
 capacity

 /vol/vol0/
 3000000
 3000000
 0
 100%

 /vol/vol0/.snapshot
 1000000
 500000
 50%

If you delete 100,000 KB (0.1 GB) of files, the disk space used by these files is no longer part of the active file system, so the space is reassigned to the Snapshot copies instead.

Data ONTAP reassigns 100,000 KB (0.1 GB) of space from the active file system to the Snapshot reserve. Because there was reserve space for Snapshot copies, deleting files from the active file system freed space for new files. If you enter the df command again, the output is as follows:

Filesystemkbytesusedavailcapacity/vol/vol0/3000000290000010000097%/vol/vol0/.snapshot100000060000040000060%

Snapshot copies can exceed reserve

There is no way to prevent Snapshot copies from consuming disk space greater than the amount reserved for them; therefore, it is important to reserve enough disk space for Snapshot copies so that the active file system always has space available to create new files or modify existing ones.

Example

Consider what happens in the following example if all files in the active file system are deleted. Before the deletion, the df output is as follows:

 Filesystem
 kbytes
 used
 avail
 capacity

 /vol/vol0/
 3000000
 3000000
 100%

 /vol/vol0/.snapshot
 1000000
 500000
 50%

After the deletion, the df command generates the following output:

Filesystemkbytesusedavailcapacity/vol/vol0/3000000250000050000083%/vol/vol0/.snapshot100000035000000350%

The entire 3,000,000 KB (3 GB) in the active file system is still being used by Snapshot copies, along with the 500,000 KB (0.5 GB) that was being used by Snapshot copies before, making a total of 3,500,000 KB (3.5 GB) of Snapshot copy data. This is 2,500,000 KB (2.5 GB) more than the space reserved for Snapshot copies; therefore, 2.5 GB of space that would be available to the active file system is now unavailable to it. The post-deletion output of the df command lists this unavailable space as used even though no files are stored in the active file system.

Recovery of disk space for file system use

Whenever Snapshot copies consume more than 100% of the Snapshot reserve, the system is in danger of becoming full. In this case, you can create files only after you delete enough Snapshot copies.

Example

If 500,000 KB (0.5 GB) of data is added to the active file system, a df command generates the following output:

Filesystemkbytesusedavail capacity/vol/vol030000003000000100%/vol/vol0/.snapshot10000003500000350%

As soon as Data ONTAP creates a new Snapshot copy, every disk block in the file system is referenced by some Snapshot copy. Therefore, no matter how many files you delete from the active file system, there is still no room to add any more. The only way to recover from this situation is to delete enough Snapshot copies to free more disk space.

Related tasks

Displaying Snapshot copy disk consumption statistics on page 55

Changing the Snapshot copy reserve

You can change the percent of disk space reserved for Snapshot copies by using the snap reserve command.

Before you begin

You should understand how Snapshot copies consume disk space before changing the Snapshot copy reserve.

Step

1. To change the percent of disk space used for the Snapshot copy reserve, enter the following command:

snap reserve vol_name percent

vol_name is the name of the volume.

percent is the percent of disk space you want to reserve for Snapshot copies.

Example

snap reserve vol1 25

What file folding means and how it saves disk space

File folding describes the process of checking the data in the most recent Snapshot copy, and if this data is identical to the Snapshot copy currently being created, by referencing the previous Snapshot copy instead of taking up disk space writing the same data in the new Snapshot copy.

File folding saves disk space by sharing unchanged file blocks between the active version of the file and the version of the file in the latest Snapshot copy, if any.

The system must compare block contents when folding a file, so file folding might affect system performance.

If the folding process reaches a maximum limit on memory usage, it is suspended. When memory usage falls below the limit, the processes that were halted are restarted.

Enabling file folding

You can enable or disable file folding as required, by using the cifs.snapshot_file_folding.enable option.

About this task

This option is only available for CIFS, and not for NFS.

Step

1. As required, choose one of the actions from the following table.

If you want to turn file folding	Then enter the following command
On	options cifs.snapshot_file_folding.enable on
Off	options cifs.snapshot_file_folding.enable off

Displaying the rate of change between Snapshot copies

You can use the snap delta command to view the rate of change between two Snapshot copies as well as the rate of change between a Snapshot copy and the active file system. This information can help you determine a suitable Snapshot copy schedule and Snapshot copy reserve.

About this task

Data ONTAP displays the rates of change in two tables. The first table displays rates of change between successive Snapshot copies. The second table displays a summary of the rate of change between the oldest Snapshot copy and the active file system. For details, see the na_snap(1) man page.

Step

1. To display data change rates on a volume, enter the following command:

snap delta vol_name

vol_name is the name of the volume containing the Snapshot copies.

Note: You can display change rates for all volumes by omitting the volume name.

Example

The following command lists the rates of change for the vol0 volume:

```
system> snap delta vol0
Volume vol0 working...
From Snapshot To
```

Active File System	149812	0d 03:43	40223.985
hourly.0	326232	0d 08:00	40779.000
hourly.1	2336 ld	12:00	64.888
hourly.2	1536 Od	04:00	384.000
hourly.3	1420 Od	04:00	355.000
hourly.4	1568 Od	12:00	130.666
nightly.0	1400 Od	04:00	350.000
hourly.5	10800 201d	21:00	2.229
То	KB changed	Time	Rate (KB/hour)
Active File System	495104	204d	20:43 100.697
	Active File System hourly.0 hourly.1 hourly.2 hourly.3 hourly.4 nightly.0 hourly.5 To Active File System	Active File System 149812 hourly.0 326232 hourly.1 2336 1d hourly.2 1536 0d hourly.3 1420 0d hourly.4 1568 0d hourly.5 10800 201d To KB changed Active File System 495104	Active File System 149812 Od 03:43 hourly.0 326232 Od 08:00 hourly.1 2336 1d 12:00 hourly.2 1536 0d 04:00 hourly.3 1420 0d 04:00 hourly.4 1568 0d 12:00 nightly.0 1400 0d 04:00 hourly.5 10800 201d 21:00 To KB changed Time

Displaying rates of change between Snapshot copies

You can display the rate of change between Snapshot copies.

Step

1. To display data change rates between two Snapshot copies, enter the following command:

```
snap delta vol_name snap1 snap2
```

vol_name is the name of the volume containing the Snapshot copies.

snap1 and snap2 are the names of the two Snapshot copies.

Example

The following command lists the rate of change between nightly.0 and hourly.1 of the vol0 volume:

Deleting Snapshot copies automatically

You can define and enable a policy for automatically deleting Snapshot copies by using the snap autodelete command. Automatically deleting Snapshot copies can help you manage space utilization.

Step

1. To define and enable a policy to automatically delete Snapshot copies, enter the following command:

snap autodelete vol_name on option value

on enables the Snapshot copy autodelete policy.

To define which Snapshot copies to delete, enter the following options and their values:

Option	Value			
commitment	Specifies whether a Snapshot copy is linked to data protection utilities (SnapMirror or NDMPcopy) or data backing mechanisms, (volume or LUN clones).			
	• try—Delete only Snapshot copies that are not linked to data protection utilities and data backing mechanisms.			
	• disrupt—Delete only Snapshot copies that are not linked to data backing mechanisms.			
trigger	Defines when to begin automatically deleting Snapshot copies.			
	 volume—Begin deleting Snapshot copies when the volume reaches 98% capacity and the volume Snapshot copy reserve has been exceeded. snap_reserve—Begin deleting Snapshot copies when the Snapshot copy reserve reaches 98% capacity. space_reserve—Begin deleting Snapshot copies when the space reserved in the volume reaches 98% capacity and the volume Snapshot copy reserve has been exceeded. 			
target_free_space	Determines when to stop deleting Snapshot copies. Specify a percentage. For example, if you specify 30, then Snapshot copies are deleted until 30% of the volume is free space.			
delete_order	Specifies which Snapshot copies to delete first—newest or oldest.			
	 newest_first—Delete the most recent Snapshot copies first. oldest_first—Delete the oldest Snapshot copies first. 			

Option	Value
defer_delete	Delete one of the following types of Snapshot copies last:
	 user_created—Snapshot copies that are not autoscheduled prefix—Snapshot copies with the specified prefix string
prefix	Delete Snapshot copies with a specific prefix last. You can specify up to 15 characters (for example, sv_snap_week). Use this option only if you specify prefix for the defer_delete option.

Next topics

Deleting Snapshot copies automatically without options on page 65 Viewing settings for the automatic deletion of Snapshot copies on page 65 Restoring the default settings for the automatic deletion of Snapshot copies on page 66 Disabling a policy for automatic deletion of Snapshot copies on page 66

Deleting Snapshot copies automatically without options

You can define and enable a policy for automatically deleting Snapshot copies when there are no options.

Step

1. Enter the following command:

snap autodelete vol_name on

on enables the Snapshot copy autodelete policy.

Note: Do not specify on when using options.

Example

snap autodelete vol0 on

Related tasks

Restoring the default settings for the automatic deletion of Snapshot copies on page 66

Viewing settings for the automatic deletion of Snapshot copies

You can view the settings for the automatic deletion of Snapshot copies by using the snap autodelete command.

Step

1. To view the settings for the automatic deletion of Snapshot copies for a given volume, enter the following command:

snap autodelete vol_name show

The snap autodelete settings for Snapshot copies revert to the following defaults:

- state—off
- commitment—try
- trigger—volume
- target_free_space—20%
- delete_order—oldest_first
- defer_delete—user_created
- prefix-no prefix specified

Restoring the default settings for the automatic deletion of Snapshot copies

You can restore the default settings for the automatic deletion of Snapshot copies by using the snap autodelete command.

Step

1. To restore the default settings for the automatic deletion of Snapshot copies, enter the following command:

```
snap autodelete vol_name reset
```

vol_name is the name of the volume.

The snap autodelete settings for Snapshot copies revert to the following defaults:

- state—off
- commitment—try
- trigger-volume
- target_free_space—20%
- delete_order—oldest_first
- defer_delete—user_created
- prefix—no prefix specified

Disabling a policy for automatic deletion of Snapshot copies

You can disable a policy for automatic deletion of Snapshot copies by using the snap autodelete command.

Step

1. To disable a policy for automatic deletion of Snapshot copies, enter the following command:

```
snap autodelete vol_name off
```

vol_name is the name of the volume.

After you disable the policy, Snapshot copies are not automatically deleted when the volume is nearly full.

Displaying space reclaimed from deleted Snapshot copies

You can display the amount of space you can reclaim by deleting one or more Snapshot copies in a volume by using the snap reclaimable command. The amount of space displayed is an approximation because writing to the volume, creating Snapshot copies, or deleting Snapshot copies causes the reclaimed amount to change.

Step

1. To display the amount of space you can reclaim by deleting Snapshot copies, enter the following command:

```
snap reclaimable vol_namesnap1 [snap2 ...]
```

vol_name is the volume which contains the Snapshot copies you might delete.

snap1 [snap2 ...] are the names of Snapshot copies you might delete. The names are separated by a space.

Note: It might take a while for Data ONTAP to display the amount of freed space. You can press Ctrl-C to interrupt the command.

Example

The following command displays the amount of space reclaimed by deleting the hourly.4, hourly.5, and nightly.0 Snapshot copies in the vol1 volume:

```
system> snap reclaimable vol1 hourly.4 hourly.5 nightly.0
Processing (Press Ctrl-C to exit) ...
snap reclaimable: Approximately 240 kbytes would be freed.
```

How to determine which Snapshot copies to delete on the basis of size

You can use the snap list command output to determine which Snapshot copies to delete to free the most disk space.

Before trying to conserve space by deleting a large Snapshot file, you should examine the cumulative values in the snap list output. If two adjacent Snapshot files show little difference in their

cumulative values, most of the data referenced by these Snapshot copies is the same. In this case, deleting only one of the Snapshot copies does not free much disk space.

In many cases, you can use the default Snapshot schedule and the default Snapshot reserve because these settings are appropriate for most environments. When you create a new volume, the new volume inherits the Snapshot schedule from the root volume. After you use the volume for several days, check how much disk space the Snapshot copies are consuming in the volume. If the amount seems high, you can decrease the amount of time that Snapshot copies are kept or increase the Snapshot reserve.

As you use Snapshot copies, you should continue to watch the statistics change over time. The statistics help you gain a better understanding of how Snapshot copies use disk space.

Attention: As a general rule, you should avoid deleting Snapshot copies that are not the product of the snap sched command (for example, Snapshot copies generated by SnapMirror or SnapVault commands). Deleting these Snapshot copies could halt the SnapMirror or SnapVault processes. An exception would be Snapshot copies left over from old SnapMirror relationships that you no longer want to maintain.

Next topics

Deleting a Snapshot copy manually on page 68 Manual deletion of a busy or locked Snapshot copy on page 69

Related tasks

Displaying Snapshot copy disk consumption statistics on page 55

Deleting a Snapshot copy manually

You can use the snap delete command to delete a Snapshot copy before the preset interval to free disk space or because it is a manual Snapshot copy that is no longer needed but is not going to be automatically deleted.

Step

1. To delete a Snapshot copy manually from a specific volume, enter the following command:

snap delete vol_name snapshot_name

vol_name is the name of the volume that contains the Snapshot copy to delete.

snapshot_name is the specific Snapshot copy to delete.

Note: To delete all Snapshot copies on a volume, use the -a parameter:

snap delete -a vol_name

Manual deletion of a busy or locked Snapshot copy

You can use the snap delete command to view ownership information of busy Snapshot copies. Before you can delete a busy Snapshot copy, you need to release the Snapshot copy from the application that is using it.

This information is useful for determining why a particular Snapshot copy is busy, and whether to stop the activity in progress. For example, if the snap delete command output displays a locked Snapshot copy that is imposing a resource constraint, you can delete that Snapshot copy and free up space.

If a Snapshot copy is locked, the snap delete operation fails until you execute a snapmirror release or snapvault release command to unlock the Snapshot copy. Snapshot copies are locked because SnapMirror or SnapVault is maintaining these copies for the next update.

Attention: Deleting a locked Snapshot copy would prevent SnapMirror or SnapVault from correctly replicating a file or volume as specified in the schedule you set up.

How to delete a locked SnapMirror Snapshot copy

The following example shows how to delete a SnapMirror Snapshot copy that is locked because SnapMirror requires it for an update:

```
systemA> snap delete vol0 oldsnap
Can't delete oldsnap: snapshot is in use by snapmirror.
Use 'snapmirror destinations -s' to find out why.
systemA> snapmirror destinations -s vol0
Path Destination
/vol/vol0 systemB:vol0
systemA> snapmirror release vol0 systemB:vol0
systemA> snap delete vol0 oldsnap
```

How to delete a locked SnapVault Snapshot copy

The following example shows how to delete a SnapVault Snapshot copy that is locked because SnapVault requires it for an update:

```
systemA> snap delete vol0 oldsnap
Can't delete oldsnap: snapshot is in use by snapvault.
Use 'snapvault status -1' to find out why.
systemA> snapvault status -1
SnapVault client is ON.
Source: systemA:/vol/vol0/qt3
Destination systemB:/vol/sv_vol/qt3...
systemA> snapvault release /vol/vol0/qt3
```

```
systemB:/vol/sv_vol/qt3
systemA> snap delete vol0 oldsnap
```

Related tasks

Releasing Snap Vault relationships on page 277 Releasing partners from a SnapMirror relationship on page 181

Renaming Snapshot copies

You might want to rename a Snapshot copy generated by the snap sched command if it contains data that you want to save. The snap sched command overwrites and deletes regularly scheduled Snapshot copies. You can use the snap rename command to save a Snapshot copy that contains data you want.

Step

1. To rename a Snapshot copy, enter the following command:

snap rename vol_name from_name to_name

vol_name is the name of the volume that contains the Snapshot copy to rename.

from_name is the current name of the Snapshot copy to rename.

to_name is the new name you want to give to the Snapshot copy.

Attention: When renaming a Snapshot copy, you should use a name that does not begin with one of the following standard prefixes: weekly, nightly, or hourly. Otherwise, Data ONTAP deletes the renamed Snapshot copy as per the schedule.

Example

snap rename vol0 hourly.2 MyDataSave

Data recovery using SnapRestore

SnapRestore uses Snapshot technology and enables you to recover data from any one of the Snapshot copies stored on the file system in case of a disaster.

Next topics

What SnapRestore is on page 71 When to use SnapRestore on page 72 Installing the SnapRestore license on page 74 Reverting a volume to a selected Snapshot copy on page 75 Reverting a file to a selected Snapshot copy on page 77 Obtaining correct incremental backups after reversion on page 80

What SnapRestore is

You can use the SnapRestore feature to recover data that is no longer available or if you are testing a volume or file and want to restore that volume or file to pre-test conditions.

Note: SnapRestore is a licensed feature. You must purchase and install the license code before you can use it.

What SnapRestore does

SnapRestore enables you to quickly revert a local volume or file to the state it was in when a particular Snapshot copy was taken. In most cases, reverting a file or volume is much faster than restoring files from tape or copying files from a Snapshot copy to the active file system.

How SnapRestore works: After you select a Snapshot copy for reversion, the Data ONTAP reverts the specified file or the volume to the data and timestamps that it contained when the selected Snapshot copy was taken. Data that was written after the selected Snapshot copy was taken is lost.

Note: If the volume you select to revert is a root volume, the system reboots.

What SnapRestore reverts: SnapRestore reverts only the file contents. It does not revert attributes of a volume. For example, the Snapshot copy schedule, volume option settings, RAID group size, and maximum number of files per volume remain unchanged after the reversion.

When to use SnapRestore: You use SnapRestore to recover from data corruption. If a primary system application corrupts data files in a volume, you can revert the volume or specified files in the volume to a Snapshot copy taken before the data corruption.

Why use SnapRestore rather than copying from a Snapshot copy: SnapRestore performs Snapshot copy restoration more quickly, using less disk space, than an administrator can achieve by manually copying volumes, qtrees, directories, or large files to be restored from the Snapshot copy system to the active file system. A large volume directory restore can be carried out in a few seconds using the SnapRestore feature.

SnapRestore can restore large volumes or files even if space limitations would prevent restoring by copying from a Snapshot copy.

When to use SnapRestore

You can use SnapRestore to recover from data corruption. If a primary system application corrupts data files in a volume, you can revert the volume or specified files in the volume to a Snapshot copy taken before the data corruption.

You must take into account certain considerations, prerequisites, and general cautions before deciding whether to use SnapRestore to revert a file or volume.

Next topics

Considerations before using SnapRestore on page 72 Prerequisites for using SnapRestore on page 73 General cautions for using SnapRestore on page 73 Caution about reverting the root volume on page 74 Preserving configuration files on page 74 Reverting a root volume before using SnapRestore on page 74

Related tasks

Reverting a file to a selected Snapshot copy on page 77 Reverting a volume to a selected Snapshot copy on page 75 Obtaining correct incremental backups after reversion on page 80

Considerations before using SnapRestore

You must take into account certain considerations before deciding whether to use SnapRestore to revert a file or volume.

- If the volume that you need to restore is a root volume, it is easier to copy the files from a Snapshot copy or restore the files from tape than to use SnapRestore, because you can avoid rebooting. If you need to restore only a corrupted file on a root volume, however, a reboot is not necessary.
- If you revert the entire root volume, the system reboots with configuration files that were in effect when the Snapshot copy was taken.
- If the amount of data to be recovered is large, SnapRestore is the preferred method, because it takes a long time to copy large amounts of data from a Snapshot copy or to restore from tape.
- If a file to be recovered needs more space than the amount of free space in the active file system, you cannot restore the file by copying from the Snapshot copy to the active file system. For example, if a 10-GB file is corrupted and only 5 GB of free space exists in the active file system, you cannot copy the file from a Snapshot copy to recover the file. However, SnapRestore can
quickly recover the file in these conditions. You do not have to spend time making the additional space available in the active file system.

Attention: SnapRestore lets you revert to a Snapshot copy from a previous release of Data ONTAP. However, doing so can cause problems because of potential version incompatibilities and can prevent the system from booting completely.

Prerequisites for using SnapRestore

You must meet certain prerequisites before using SnapRestore.

- SnapRestore must be licensed on your storage system.
- There must be at least one Snapshot copy on the system that you can select to revert.
- The volume to be reverted must be online.
- The volume to be reverted must not be in use for data replication.

General cautions for using SnapRestore

SnapRestore overwrites data permanently and might disrupt a SnapMirror relationship.

- SnapRestore overwrites all data in the file or volume. After you use SnapRestore to revert to a selected Snapshot copy, you cannot undo the reversion.
- If you revert to a Snapshot copy created before a SnapMirror Snapshot copy, Data ONTAP can
 no longer perform an incremental update of the data using the snapmirror update command.
 However, if there is any common Snapshot copy (SnapMirror Snapshot copy or other Snapshot
 copy) between the SnapMirror source and SnapMirror destination, then you can use the
 snapmirror resync command to resynchronize the SnapMirror relationship.
 If there is no common Snapshot copy between the SnapMirror source and SnapMirror source and SnapMirror destination,
 the you should reinitialize the SnapMirror relationship.
- Snapshot copy deletions are irrevocable. If you delete a Snapshot copy, you cannot recover the Snapshot copy by using SnapRestore.
- After you revert a volume to a selected Snapshot copy, you lose all the Snapshot copies that were taken after the selected Snapshot copy.
- Between the time you enter the snap restore command and the time when reversion is completed, Data ONTAP stops deleting and creating Snapshot copies.
- If you are reverting a file from a Snapshot copy, you can delete other Snapshot copies, except for the Snapshot copy you are reverting from.
- If you use the SnapRestore feature to restore a FlexClone from its base Snapshot copy, the space optimization relationship between the FlexClone and its parent is lost. For more information, see the *Data ONTAP Storage Management Guide*.

Caution about reverting the root volume

Because the /etc directory of the root volume contains configuration information about the system, reverting the root volume might change the configuration.

In addition, restoring the root volume restores the options for the entire system to the settings that were in effect when the Snapshot copy was taken. Reverting a root volume requires rebooting the system.

Preserving configuration files

To preserve the data, you must store all configuration file data in a volume other than a root volume.

Step

1. Store all data that needs to be reverted in a volume other than the root volume. This ensures that you never need to revert the root volume.

Reverting a root volume before using SnapRestore

If the data you want to revert resides in the root volume, you should back up the /etc directory to another volume or another system before using SnapRestore. After you revert the root volume, you can then restore the /etc directory and reboot.

Step

1. If you back up the /etc directory to another volume, use the following command to make the system reboot with that volume as the root volume:

vol options volume root

In this way, when the system reboots during a revert, it can use the correct settings in the /etc directory.

Installing the SnapRestore license

You must purchase and install the license code before you can use SnapRestore.

Step

1. On the server, enter the following command:

license add xxxxxxx

xxxxxxx is the license code you purchased.

This setting persists across reboots.

Reverting a volume to a selected Snapshot copy

In certain situations, you need to revert a volume to a selected Snapshot copy using SnapRestore. You should notify the users of the volume before you revert a volume.

Before you begin

Ensure that you notify the users of the volume that you are going to revert a volume, and that the current data in the volume will be replaced by the selected Snapshot copy.

Note: NFS users should unmount the files and directories in the volume before the reversion. If they do not unmount the files and directories, they might get a "stale file handle" error message after the volume reversion.

Steps

1. As required, choose one of the actions from the following table.

If	Then
You know the name of the Snapshot copy for each volume you want to revert	Go to Step 5.
You want to choose a Snapshot copy from the list of Snapshot copies available for reversion	Go to Step 2.

2. Enter the following command:

snap restore [-f] -t vol vol_name

-t vol specifies the volume name to revert.

vol_name is the name of the volume to be reverted. Enter the name only, not the complete path. You can enter only one volume name.

Use the -f option to avoid warning messages and prompts to confirm your decision to revert the volume. For more information, see the na_snap(1) man page.

3. Press y to confirm that you want to revert the volume.

Data ONTAP displays a list of Snapshot copies.

4. Enter the name of the Snapshot copy for reverting the volume, then go to Step 8.

Data ONTAP displays the name of the volume to be reverted and the name of the Snapshot copy to be used for the reversion.

5. Enter the following command:

snap restore [-f] -t vol -s snapshot_name vol_name

-t vol specifies the volume name to revert.

-s snapshot_name specifies the name of the Snapshot copy from which to revert the data. You can enter only one Snapshot copy name.

6. Press y to confirm that you want to revert the volume.

Data ONTAP displays the name of the volume and the name of the Snapshot copy for the reversion and, if you have not used the -f option, prompts you to decide whether to proceed with the reversion.

Note: To cancel volume reversion, press Ctrl-C at any time before you enter y in Step 8.

7. As required, choose one of the actions from the following table.

If	Then
You want to continue with the	Press y.
reversion	Result : The system reverts the volume from the selected Snapshot copy. If you are reverting the root volume, the system reboots.
You do not want to proceed with	Press n or Ctrl-C.
the reversion	Result : The volume is not reverted and you are returned to a prompt.

Example

system> snap restore -t vol -s nightly.0 /vol/vol1 system> WARNING! This will restore a volume from a snapshot into the active file system. If the volume already exists in the active file system, it will be overwritten with the contents from the snapshot. Are you sure you want to do this? y You have selected file /vol/vol1, snapshot nightly.0 Proceed with restore? y

Result: Data ONTAP restores the volume called vol1 at /vol/vol1.

After a volume is reverted with SnapRestore, all user-visible information (data and attributes) for that volume in the active file system is identical to that contained in the Snapshot copy.

Reverting a file to a selected Snapshot copy

Using snap restore to revert a single file to a selected Snapshot copy is practical when the file is so large that you cannot copy the previous file version from the Snapshot copy to the active file system.

Before you begin

Ensure that you notify the network users before reverting a file so that they know that the current data in the file will be replaced by that of the selected Snapshot copy.

Note: NFS users who try to access a reverted file without first reopening it might get a stale file handle error message after the volume reversion.

About this task

When you use snap restore for file reversion, note the following:

- You cannot use SnapRestore for single file reversion on files with NT streams, or on directories.
- If you restore single files with the snap restore command, and then issue the snap list command, the snap list command might take up to several minutes to complete. You can minimize the amount of time required to complete by using the snap list -n command. For more details, see the manual (man) pages.

Note: To cancel file reversion, press Ctrl-C at any time before you press y in Step 8.

Steps

1. As required, choose one of the actions from the following table.

If	Then
You know the name of the Snapshot copy for the file you want to revert	Go to Step 5.
You want to choose a Snapshot copy from the list of Snapshot copies available for reversion	Go to Step 2.

2. Enter the following command:

snap restore [-f] -t file -r restore_as_new_path path_and_file_name

-t file specifies that you are entering the name of a file to revert.

-r restore_as_new_path restores the file to a location different from (but in the same volume as) the location in the Snapshot copy. For example, if you specify /vol/vol0/vol3/ myfile as the argument to -r, SnapRestore reverts the file called myfile to the location /vol/ vol0/vol3 instead of to the path in vol3 indicated by path_and_file_name.

78 | Data ONTAP 7.3 Data Protection Online Backup and Recovery Guide

path_and_file_name is the complete path to the name of the file to be reverted. You can enter only one path name.

A file can be restored only to the volume where it was originally located. The directory structure to which a file is to be restored must be the same as that specified in the path. If this directory structure does not exist, you must create it before restoring the file.

Note: Use the -f option to avoid warning messages and prompts to confirm your decision to revert the volume. For more information, see the na_snap(1) man page.

Result: Data ONTAP displays a warning message and prompts you to confirm your decision to revert the file.

3. Press y to confirm that you want to revert the file.

Result: Data ONTAP displays a list of Snapshot copies.

4. Enter the name of the Snapshot copy for reverting the file, then go to Step 8.

Result: Data ONTAP displays the name of the file to revert and the name of the Snapshot copy to be used for the reversion.

5. Enter the following command:

snap restore [-f] -t file -s snapshot_name -r restore_as_path path_and_file_name

-t file specifies that you are entering the name of a file to revert.

-s snapshot_name specifies the name of the Snapshot copy from which to revert the data.

-r restore_as_path restores the file to a location different from the location in the Snapshot copy. For example, if you specify /vol/vol0/vol3/myfile as the argument to -r, SnapRestore reverts the file called myfile to the location /vol/vol0/vol3 instead of to the file structure indicated by the path in path_and_file_name.

path_and_file_name is the complete path to the name of the file to be reverted. You can enter only one path name.

A file can be restored only to the volume where it was originally. The directory structure to which a file is to be restored must be the same as specified in the path. If this directory structure does not exist, you must create it before restoring the file.

Unless you enter -r and a path name, only the file at the end of the *path_and_file_name* is reverted. You can enter only one path name.

Note: Use the -f option to avoid warning messages and prompts that confirm your decision to revert the file. For more information, see the na_snap(1) man page.

6. Press y to confirm that you want to revert the file.

Result: Data ONTAP displays the name of the file and the name of the Snapshot copy for the reversion and, if you have not used the -f option, prompts you to decide whether to proceed with the reversion.

7. As required, choose one of the actions from the following table.

If	Then
You want to continue with the reversion	Press y.
	Result : The system reverts the file from the selected Snapshot copy.
You do not want to continue with the reversion and	Press n or Ctrl-C.
want to choose another Snapshot copy from the list of Snapshot copies available for reversion	Result : The file is not reverted and you are returned to a prompt.

Example

system> snap restore -t file /vol/voll/users/jim/myfile -s nightly.0
system> WARNING! This will restore a file from a snapshot into the active file
system. If the file already exists in the active file system, it will be
overwritten with the contents from the snapshot. Are you sure you want to do
this? y
You have selected file /vol/voll/users/jim/myfile, snapshot nightly.0
Proceed with restore? y

Result: Data ONTAP restores the file called myfile to the existing volume and directory structure /vol/vol1/users/jim.

Example

```
system>snap restore -t file -s nightly.0 -r /vol/vol2/archive/eng/myfile /vol/
vol2/users/jim/myfile
```

system>WARNING! This will restore a file from a snapshot into the active file system. If the file already exists in the active file system, it will be overwritten with the contents from the snapshot. Are you sure you want to do this? y You have selected file /vol/voll/users/jim/myfile, snapshot nightly.0 Proceed with restore? y

Result: Data ONTAP restores the file called myfile to a new location at /vol/vol2/ archive/eng.

After a file has been reverted with SnapRestore, check whether all user-visible information (data and file attributes) for that file in the active file system is identical to that contained in the Snapshot copy.

Obtaining correct incremental backups after reversion

All files in a reverted volume have timestamps that are the same as those when the Snapshot copy was created. After a revert operation, incremental backup and restore operations on the file or volume cannot rely on the timestamps to determine what data needs to be backed up or restored.

Steps

- 1. Perform a base-level backup of the volume after you restore it.
- 2. When restoring data from tape, use only the backups that were created after the volume was restored.

Data protection using SnapMirror

SnapMirror is a feature of Data ONTAP that enables you to replicate data. SnapMirror enables you to replicate data from specified source volumes or qtrees to specified destination volumes or qtrees, respectively. You need a separate license to use SnapMirror.

After the data is replicated to the destination storage system, you can access the data on the destination to perform the following actions:

- Provide users immediate access to mirrored data in case the source goes down.
- Restore the data to the source to recover from disaster, data corruption (qtrees only), or user error.
- Archive the data to tape.
- Balance resource loads.
- Back up or distribute the data to remote sites.

You can configure SnapMirror to operate in one of the following modes:

- Asynchronous mode: SnapMirror replicates Snapshot copies to the destination at specified, regular intervals.
- Synchronous mode: SnapMirror replicates data to the destination as soon as the data is written to the source volume.
- Semi-synchronous mode: SnapMirror replication at the destination volume lags behind the source volume by 10 seconds. This mode is useful for balancing the need for synchronous mirroring with the performance benefit of asynchronous mirroring.

SnapMirror can be used with traditional volumes and FlexVol volumes.

Note: SnapMirror is supported for V-Series systems also. Information about SnapMirror in this chapter applies to both FAS systems and V-Series systems, unless specified otherwise.

Next topics

How SnapMirror works on page 82 Applications of SnapMirror on page 83 What synchronous SnapMirror is on page 84 Enabling SnapMirror by entering license codes on page 87 Considerations for the use of SnapMirror on page 88 Considerations before using synchronous SnapMirror on page 93 Deployment of SnapMirror on page 96 Initialization of a SnapMirror destination on page 121 Maximum number of concurrent replication operations on page 128 Methods for specifying destination systems on the SnapMirror source on page 132 What the snapmirror.conf file does on page 135 Scheduled updates for volumes or gtrees on page 144 Manual update of a SnapMirror destination on page 146 SnapMirror over multiple paths on page 149 Compression for SnapMirror transfers on page 151 Checking SnapMirror data transfer status on page 154 Adjusting the TCP window size for a SnapMirror relationship on page 162 Setting a maximum transfer rate for all transfers on page 164 Changing the maximum transfer rate for a single SnapMirror transfer on page 165 Use of SnapMirror to copy a volume to a tape on page 165 About moving SnapMirror sources on page 167 Migrating data between volumes by using SnapMirror on page 171 Conversion of a destination to a writable volume or gtree on page 172 Resizing a SnapMirror source and destination volume pair on page 174 Converting asynchronous SnapMirror replication to synchronous on page 178 Stabilizing destinations before a Snapshot copy on page 178 Aborting a SnapMirror transfer on page 180 Releasing partners from a SnapMirror relationship on page 181 SnapMirror data transfer logs on page 183 Listing SnapMirror Snapshot copies on page 186 What SnapMirror restarts and retries are on page 189 What the snapmirror resync command does on page 189 Operation of SnapMirror with other features and products on page 195 SnapMirror over Fibre Channel on page 207 Troubleshooting of SnapMirror issues on page 223

Related concepts

What synchronous SnapMirror is on page 84 SnapMirror over Fibre Channel on page 207

How SnapMirror works

SnapMirror replicates data from a source volume or qtree to a partner destination volume or qtree, respectively, by using Snapshot copies. Before using SnapMirror to copy data, you need to establish a relationship between the source and the destination.

You can specify a SnapMirror source and destination relationship between volumes or qtrees by using one of the following options.

- The /etc/snapmirror.conf file
- The snapmirror.access option
- The /etc/snapmirror.allow file

The SnapMirror feature does the following:

- 1. Creates a Snapshot copy of the data on the source volume.
- 2. Copies it to the destination, a read-only volume or qtree.
- **3.** Updates the destination to reflect incremental changes on the source, as per the schedule you specify.

The result of this process is an online, read-only volume or qtree that contains the same data as the source at the time of the most recent update.

Each volume SnapMirror replication, qtree SnapMirror replication, or SnapVault replication consists of a pair of operations. There is one operation each at:

- The source storage system
- The destination storage system

Therefore, if a storage system is the source for one replication and the destination for another replication, it uses two replication operations. Similarly, if a storage system is the source as well as the destination for the same replication, it uses two replication operations.

Applications of SnapMirror

SnapMirror is used to replicate data. Its qualities make SnapMirror useful in several scenarios, including disaster recovery, data backup, and data restoration.

You can copy or use the data stored on a SnapMirror destination. The additional advantages of SnapMirror make it useful in data retrieval situations such as those described in the following table.

Situation	How to use SnapMirror
Disaster recovery: You want to provide immediate access to data after a disaster has made a qtree, volume, or system unavailable.	You can make the destination writable so clients can use the same data that was on the source volume the last time data was copied.
Disaster recovery testing: You want to test the recovery of data and restoration of services in the event of a disaster.	You can use FlexClone technology on the SnapMirror destination, and test for disaster recovery, without stopping or pausing other replication operations.
Data restoration: You want to restore lost data on a qtree or volume source from its mirrored qtree or volume SnapMirror partner.	You can temporarily reverse the roles for the source and destination qtrees or volumes and copy the mirrored information back to its source.

Situation	How to use SnapMirror
Application testing: You want to use an application on a database, but you want to test it on a copy of the database in case the application damages the data.	You can make a copy of the database to be used in the application testing to ensure that the data on the source cannot be lost.
Load balancing: A large number of users need read-only access to a qtree or volume.	You can copy the data in a qtree or volume to multiple volumes or systems to distribute the load.
Off-loading tape backups: You need to reserve all processing and networking resources on a system for serving NFS and CIFS requests.	After copying data on the source system, you can back up the data in the destination to tape. This means that the source system does not have to allocate resources for performing backups.
Access to remote data: Users who need read access to a volume are distributed over a large geographical area.	You can copy the source volume to other systems that are geographically closer to the users. Users accessing a local system can read the data using less resource time than if they connected to a distant system.

What synchronous SnapMirror is

In the synchronous mode, SnapMirror enables you to replicate data to the destination as soon as it is written to the source volume.

Synchronous SnapMirror is a feature of SnapMirror. You can use synchronous SnapMirror to replicate data between systems, or systems in an active/active configuration, situated at remote sites, using either an IP or a Fibre Channel connection.

In addition to a standard SnapMirror license, the synchronous SnapMirror feature requires a separate license key. You must enter this key and receive important product information prior to enabling the synchronous SnapMirror feature. To enable the synchronous SnapMirror feature, use the license add command.

You can use synchronous SnapMirror only with volumes, not with qtrees. The source and destination volumes must be of the same type, as given in the following list.

- Traditional volumes
- FlexVol volumes

Next topics

Synchronous SnapMirror modes on page 85 How SnapMirror replicates data synchronously on page 86 How synchronous SnapMirror handles network issues on page 86 *Guidelines for growing an aggregate with a synchronous SnapMirror destination volume* on page 86

Synchronous SnapMirror modes

There are two modes available for synchronous SnapMirror replication: sync and semi-sync. The semi-sync mode helps in achieving a balance between the benefits of synchronous and asynchronous replication.

Note: You cannot set up a synchronous or semi-synchronous SnapMirror relationship between the two nodes of an active/active configuration.

You can specify either of the following two modes, when defining a SnapMirror relationship in the snapmirror.conf file.

- sync: The source system acknowledges a client write operation only after both the source and destination systems have completed the write operation. The sync option provides a recovery point objective of almost 0 seconds.
- semi-sync: The source system acknowledges the client write operation immediately after the source receives the data. The destination system is synchronized with the source at intervals of approximately 10 seconds. The semi-sync option provides a recovery point objective of about 10 seconds. This means that if the source becomes unavailable, you might lose up to 10 seconds worth of data changes. The semi-sync mode provides a performance advantage over the sync mode.

Note: If neither of these two modes is specified, then the SnapMirror relationship is set as asynchronous.

To enable the sync or semi-sync mode for a volume SnapMirror relationship, you need to specify the mode in the snapmirror.conf file entry for the relationship, as given in the following line:

src_system:src_path dst_system:dst_path - {sync|semi-sync}

src_system is the name of the SnapMirror source system.

dst_system is the name of the SnapMirror destination system.

src_path is the path of the SnapMirror source volume.

dst_path is the path of the SnapMirror destination volume.

Example of an entry for a SnapMirror relationship with semi-sync mode

systemA:volA systemB:volB - semi-sync

Related references

Syntax for snapmirror.conf file entries on page 137

How SnapMirror replicates data synchronously

Before Data ONTAP saves data to the disk, it collects written data in NVRAM. Then, at a point in time called a consistency point, it sends the data to disk.

As data is added or changed, the data is not directly written to the disk. Changes or additions to data are temporarily stored in the NVRAM. Then, at a consistency point, SnapMirror writes the data to the disks on the destination system. When the synchronous SnapMirror feature is enabled, the source system forwards data to the destination system as it is written in NVRAM. Then, at the consistency point, the source system sends its data to disk and tells the destination system to also send its data to disk. Finally, the source system waits for the destination system to acknowledge that it sent data to disk before continuing with the next write.

How synchronous SnapMirror handles network issues

If SnapMirror encounters any network issues that restrict the operation of synchronous replication, SnapMirror goes into the asynchronous mode.

The source and destination systems communicate with each other continuously. If a network failure disrupts the communication, SnapMirror initiates the following process:

- 1. SnapMirror sets the replication to the asynchronous mode.
- **2.** In the asynchronous mode, the source system tries to communicate with the destination system once a minute.
- **3.** When the source system reestablishes communication with the destination system, the source system asynchronously replicates data to the destination.
- 4. SnapMirror gradually transitions the replication relationship to the synchronous mode.

If the latest common Snapshot copy is deleted from the source, SnapMirror does not transition back from asynchronous to synchronous mode. The SnapMirror relationship should be broken and resynchronized, by using the snapmirror break and snapmirror resync commands. However, you can avoid this situation and let the relationship go to synchronous mode automatically, by setting the replication.volume.use_auto_resync option to on. The default value of this option is off.

Related references

SnapMirror options on page 103

Guidelines for growing an aggregate with a synchronous SnapMirror destination volume

When increasing the size of an aggregate that contains a synchronous SnapMirror destination volume, you need to follow several guidelines.

- Add a minimum of four disks.
- Ensure that any new RAID group created by the addition of new disks has at least four data disks.

• Ensure that the RAID group size is 16 or fewer disks.

Enabling SnapMirror by entering license codes

Before any SnapMirror replication process can begin, you must add the SnapMirror license on the system and enable SnapMirror.

About this task

In addition to a SnapMirror license, a special license key is required to enable synchronous SnapMirror. You need to enter the license code on the SnapMirror source and destination systems.

Step

1. To add the SnapMirror license code, enter the following command:

license add xxxxxxx

xxxxxx is the license code you purchased.

Related concepts

What synchronous SnapMirror is on page 84

Turning SnapMirror on

Before using SnapMirror you need to enable the SnapMirror license on both the source and the destination systems.

Steps

1. To turn SnapMirror on, enter the following command on both the source system and destination system.

options snapmirror.enable on

Alternatively, you can use the snapmirror on command to turn SnapMirror on.

Note: This setting persists across reboots.

2. Depending on whether you use the snapmirror.access option or the /etc/ snapmirror.allow file to specify allowed destinations, choose one of the actions from the following table.

If you choose	Then
snapmirror.access option	On the source, enter the following command as a single line.
	<pre>options snapmirror.access host=[dest_system1,dest_ system2,]</pre>
	The default value for the snapmirror.access option is legacy, which lets the /etc/snapmirror.allow file define the access permissions. This option persists across reboots.
/etc/snapmirror.allow file	Add the names of the destination systems, each on a separate line, in the /etc/snapmirror.allow file.

Considerations for the use of SnapMirror

When planning to use SnapMirror for replication, you need to be aware of the prerequisites, restrictions, points of caution, and recommended actions.

You also need to understand issues related to deduplication, adjusting the TCP window size for SnapMirror, possible conflicts between SnapMirror and Snapshot copy schedules, and destination accessibility when using CIFS with SnapMirror.

Next topics

Prerequisites for SnapMirror on page 88 Restrictions while using SnapMirror on page 89 Points of caution while using SnapMirror on page 90 Symmetrical disk geometry on page 90 Recommended actions while using SnapMirror on page 91 Deduplication with volume SnapMirror on page 91 Possible conflicts between SnapMirror operation and Snapshot copy schedule on page 92 Destination accessibility when using CIFS with SnapMirror on page 92

Prerequisites for SnapMirror

You need to fulfill a set of prerequisites before you can use SnapMirror.

- You must purchase and enable the SnapMirror license. If the SnapMirror source and destination are on different systems, you must enable the SnapMirror license on each system.
- For SnapMirror volume replication, you must create a restricted volume to be used as the destination volume. SnapMirror does not automatically create a volume. For information about how to create volumes, see the section on organizing data using volumes and qtrees in the *Data ONTAP Storage Management Guide*.
- For SnapMirror volume replication, the destination system must use a version of Data ONTAP that is the same as or later than that of the SnapMirror source system. If you configure volume

SnapMirror to support replication for the purpose of disaster recovery, both the source and destination system must use the same version of Data ONTAP.

Note: If you upgrade your system to a later version of Data ONTAP, you must upgrade the SnapMirror destination before upgrading the SnapMirror source.

- For SnapMirror qtree replication, you must not create a qtree to be used as a destination qtree; the snapmirror initialize command creates the destination qtree automatically.
- For SnapMirror qtree replication, the destination system must be using Data ONTAP 6.2 or later.
- The name and IP address of the source system must be in the /etc/hosts file of the destination system or must be resolvable through the DNS or by using the yp command.

Related tasks

Enabling SnapMirror by entering license codes on page 87

Restrictions while using SnapMirror

When planning the configuration of SnapMirror, you need to consider the relevant restrictions.

- The source volume must be online. For information about how to put a volume online, see the *Data ONTAP Storage Management Guide*.
- For SnapMirror volume replication, the capacity of the destination volume must be greater than or equal to the capacity of the source volume. For information about how to add disks to a volume, see the *Data ONTAP Storage Management Guide*.
- To support SnapMirror qtree replication, the destination volume must contain 5 percent more free space than the source qtree consumes.
- The SnapMirror destination volume cannot be the root volume of a storage system. The SnapMirror source volume, however, can be the root volume.
- A destination qtree can be on the root volume, but the /etc qtree cannot be a destination qtree.
- A destination qtree name should not:
 - Contain "*", "-", ".", or "/etc"
 - Contain the character combination "->" This restriction applies to source gtrees as well.
 - Contain the tab character
 - Be longer than 64 characters
 - Be specified as "/vol/vol_name/" (with no qtree name)
 - Be specified as "vol_name/qtree_name" (without /vol/)

Note: When creating or specifying a qtree, you can use the space character in the qtree name. However, if you do so, you need to enclose the qtree name in double quotes. You can also use a double quote within a qtree name. The following example shows how to specify the qtree name, vol/voll/x y"z, with both a space character and a double quote.

"systemA:/vol/vol1/x y""z"

• There must be a functional network to transfer data between two different storage systems.

• Each storage system model supports a specified number of snapmirror update command or vol copy command operations at a time.

Related concepts

Initialization of a SnapMirror destination on page 121 *Initialization of a SnapMirror destination from tape* on page 122

Related tasks

Verifying the size of each volume on page 336 Initializing a destination for non-qtree data on page 125 Initializing a SnapMirror destination on page 122

Related references

Maximum number of concurrent replication operations on page 128

Points of caution while using SnapMirror

While using SnapMirror, you need to exercise adequate caution on several points, including the use of Snapshot copies, the use of SnapMirror commands, and the use of SnapMirror destination volumes.

- Do not delete Snapshot copies that SnapMirror creates in the source volume before copying the data to the destination. The most recent SnapMirror Snapshot copy is referred to as the newest common Snapshot copy (NCS). Incremental changes to the destination depend on the NCS. If SnapMirror cannot find the required Snapshot copy on the source, it cannot perform incremental changes to the destination.
- Do not use the snapmirror release or snapmirror break command on the destination volume or qtree unless you no longer need to copy incremental changes from the source. The destination must be actively functioning as a destination to receive incremental updates.
- Do not restrict or take the destination volume offline while SnapMirror is configured to transfer. Taking the destination offline prevents SnapMirror from performing updates to the destination.

Symmetrical disk geometry

When replicating data with FlexVol volumes, disk geometry is not an issue, unlike replicating data with traditional volumes.

With FlexVol volumes, the symmetry of disk geometry between the source and the destination volumes is less important. It is not necessary that the source and destination volumes have the same number of disks or have disks with identical sizes.

Related references

Recommended actions while using SnapMirror on page 91

Recommended actions while using SnapMirror

While using SnapMirror, you can increase the efficiency of data copying by performing certain actions. This includes the staggering of Snapshot copy schedules and SnapMirror update schedules.

• To optimize performance, stagger your Snapshot copy update schedules so that SnapMirror activity does not begin or end at the exact minute a snap sched command operation attempts to create a Snapshot copy.

If the SnapMirror feature is scheduled to perform Snapshot copy management at the same time as a snap sched activity, then the Snapshot copy management operations scheduled using the snap sched command might fail with syslog messages: "Skipping creation of hourly snapshot" and "Snapshot already exists."

- For optimum SnapMirror volume replication performance, ensure that the SnapMirror source volume and destination volume contain disks of the same size, organized in the same RAID configuration.
 - If the SnapMirror source and destination are FlexVol volumes, the RAID configurations do not make a difference.
 - If the SnapMirror source and destination are qtrees, volume size and configuration do not make any difference.

Related references

Firewall usage with SnapMirror on page 110

Deduplication with volume SnapMirror

Starting with Data ONTAP 7.3, the deduplication metadata for a volume is placed outside the volume, at the aggregate level. This can improve the space savings achieved through the use of deduplication.

When replicating data using volume SnapMirror, the deduplication metadata for the volume is not replicated along with the volume. The data in the volume is usable both on the source and the destination.

To achieve maximum space savings on the destination volume, you must scan the entire file system to re-create the deduplication metadata for the destination volume. Use the sis start -s command to do so.

Note: The destination volume is accessible for read-write operations when the deduplication scan is in progress.

If you use the sis start command without the -s option, the potential for space savings on the destination volume is reduced because only the new data written to the volume is scanned for deduplication.

For more information about deduplication, see the Data ONTAP Storage Management Guide.

Related references

Considerations before using synchronous SnapMirror on page 93

Possible conflicts between SnapMirror operation and Snapshot copy schedule

Some of the operations of SnapMirror might conflict with the actions of a Snapshot copy management schedule. Certain steps enable you to avoid these conflicts.

If the SnapMirror feature is scheduled to perform Snapshot copy management at the same time as a snap sched activity, then the Snapshot copy management operations scheduled using the snap sched command might fail, generating syslog messages such as:Skipping creation of hourly snapshot and Snapshot already exists.

To avoid this situation, you should stagger the Snapshot copy update schedules so that SnapMirror activity does not begin, or end at the exact minute that a snap sched command operation attempts to create a Snapshot copy.

Destination accessibility when using CIFS with SnapMirror

Before copying a directory on a SnapMirror volume that supports CIFS clients, you should ensure that the directories are in the Unicode format. This ensures that the read-only directory copied on the destination is in the Unicode format. This also enables requests through CIFS to access the directory and its files on the destination, and prevents Access denied errors.

You can ensure that both source volume and destination volume directories are in the Unicode format by using one of the following methods.

Method 1

On the system console for the source volume, enter these two commands.

vol options vol_name convert_ucode on

Use this command to convert any existing directories in a volume to the Unicode format.

vol options vol_name create_ucode on

Use this command to ensure that any new directories created in a volume are in the Unicode format.

Method 2

Alternatively, ensure that all directories on the source volume that will be accessed by CIFS clients are accessed by a CIFS client before initial replication to a destination. Such access on a writable source volume automatically converts that directory to the Unicode format.

Considerations before using synchronous SnapMirror

You need to consider certain issues when planning to use SnapMirror for synchronous replication.

The following configurations do not support synchronous SnapMirror:

- One source system cannot have synchronous SnapMirror relationships to multiple destinations.
- You cannot create a synchronous SnapMirror relationship between FlexVol volumes within the same system or within the same active/active configuration.
- You cannot use synchronous SnapMirror to replicate volumes that use deduplication.
- A volume with a synchronous SnapMirror relationship should not be placed in the same aggregate as a volume using deduplication.

For more information about deduplication, see the Data ONTAP Storage Management Guide.

For synchronous SnapMirror, the disks comprising the source and destination volumes should be of the same type—either ATA or Fibre Channel. However, you can have the other type of disks attached to the source or destination system. The possible combinations are given in the following list.

- Fibre Channel disks in a synchronous SnapMirror relationship, with ATA disks on the source or destination system
- ATA disks in a synchronous SnapMirror relationship, with Fibre Channel disks on the source or destination system

Although such configurations are supported, their use is not recommended. The combined load on Fibre Channel and ATA disks, might negatively impact the performance of synchronous SnapMirror. This performance impact might cause the synchronous SnapMirror relationship to fall out of sync. If you face such performance issues, you might need to reduce the load on the system to resolve the issue.

The following table indicates the support for bidirectional synchronous and semi-synchronous SnapMirror replication with the two different types of volumes.

Type of volume	Data ONTAP version
Traditional volumes	7.2 or later
FlexVol volumes	7.2.2 or later

You can choose the data to be synchronously replicated. For example, you can synchronously replicate database data and asynchronously replicate home directories. If the home directories contain important log data, you can use the synchronous SnapMirror option to replicate the log data. For more details, see the na_snapmirror.conf(5) man page.

The source and destination systems should be adequately configured for the replication traffic. Synchronous SnapMirror is supported for traditional volumes only for configurations in which the source and destination systems are of the same type and have the same disk geometry.

Note: There is no disk geometry restriction for FlexVol volumes.

The type of system and the disk configuration on the destination system affects the performance of the source system. Therefore, the destination system should have the bandwidth required for the increased traffic and for message logging. The NVLOG files are stored in the parent aggregate of the volume being replicated.

The network transport should be optimized for SnapMirror replication. You must use a dedicated, high-bandwidth, low-latency network between the source and destination systems. Synchronous SnapMirror can support traffic over Fibre Channel and IP transports. SnapMirror also allows multipathing, enabling you to either balance the load between two paths or to reserve the second path for failover. For optimizing performance, you can use the best route available between the source and destination systems, and you can restrict the route to the traffic between the two systems.

You should keep well below the maximum number of Snapshot copies. Synchronous SnapMirror needs three Snapshot copies to get into synchronization. Therefore, you should limit the combined total of Snapshot copies retained on any one volume to 252 or fewer.

Synchronous SnapMirror license key required: In addition to a standard SnapMirror license, the synchronous SnapMirror feature requires a license key. Before enabling the synchronous SnapMirror feature, you must enter this key and receive important product information. To enable the synchronous SnapMirror feature, use the license add command, and enter one of the following license keys.

Platform	License key
Data ONTAP storage system	KZZTWOJ
NearStore system	TXKMEAK
V-Series system	PLFQNUJ

Next topics

Disk types for a V-Series system using synchronous and semi-synchronous SnapMirror on page 95 Estimating aggregate size for synchronous SnapMirror destination volumes on page 95

Related concepts

What the snapmirror.conf file does on page 135 *SnapMirror over multiple paths* on page 149

Related references

Syntax for snapmirror.conf file entries on page 137

Disk types for a V-Series system using synchronous and semisynchronous SnapMirror

Mixed-disk systems are not supported with synchronous and semi-synchronous SnapMirror, including between a FAS system and a V-Series system.

The same general guidelines about disk types with synchronous and semi-synchronous SnapMirror apply to both FAS systems and V-Series systems. The following additional information about disk types is specific to V-Series systems:

- The storage array that is presenting storage to a V-Series system must be using only one type of disk (FC or SATA), and the source or destination must use the same type of disks as the storage array.
- To replicate between a V-Series system and a FAS system, the controllers must be the same platform type (for example, both the V-Series and FAS systems must be 3170 systems), and the disks on the FAS system must be the same type as the disks on the storage array that provides storage for the V-Series system.

Estimating aggregate size for synchronous SnapMirror destination volumes

For synchronous replication of volumes using SnapMirror, the aggregates that contain destination volumes should have enough free space to store the NVLOG data.

Before you begin

You need to ascertain the model name of the system that is the synchronous SnapMirror source.

Steps

1. Determine the size of the source system NVRAM.

For information, see the System Configuration Guide.

Example 6070 has 2 GB of NVRAM.

2. Multiply the NVRAM size by 20 to determine the estimated free space size.

Example

2 GB times 20 is 40 GB; therefore, 40 GB of free space is required on the aggregate containing the destination volume.

Deployment of SnapMirror

A basic deployment of SnapMirror consists of source volumes and qtrees, and destination volumes and qtrees.

Source volumes or qtrees: In a SnapMirror configuration, source volumes and qtrees are the data objects that need to be replicated. Normally, users of storage can access and write to source volumes and qtrees.

Destination volumes or qtrees: In a SnapMirror configuration, destination volumes and qtrees are data objects to which the source volumes and qtrees are replicated. The destination volumes and qtrees are read-only, and usually placed on a separate system than the source. The destination volumes and qtrees can be accessed by users in case the source becomes unavailable. The administrator can use SnapMirror commands to make the replicated data at the destination accessible and writable.

Note: Destination volumes have to be writable when using qtree SnapMirror for replication.



The following illustration depicts a basic SnapMirror deployment.

Figure 5: Basic SnapMirror deployment

Next topics

Supported SnapMirror configurations on page 97 Comparison between volume SnapMirror and qtree SnapMirror on page 97 SnapMirror deployment variations on page 98 Migration from traditional volumes to FlexVol volumes on page 100 SnapMirror commands on page 100 SnapMirror options on page 103 SnapMirror files on page 105 SnapMirror support for IPv6 on page 107 Setting up a basic SnapMirror operation on page 108 Firewall usage with SnapMirror on page 110 Data replication from one destination to another in a series (cascading) on page 110 SnapMirror source to tape to destination scenario on page 116

Related concepts

SnapMirror deployment variations on page 98

Supported SnapMirror configurations

You can use SnapMirror to replicate both traditional and FlexVol volumes. However, there are certain configurations that are not supported for replication.

Volume SnapMirror

Volume SnapMirror only supports replication between the same type of volumes. The source and destination volumes must be of the same type, as given in the following list.

- Traditional volumes
- FlexVol volumes

Note: Volume SnapMirror does not support replication between different volume types.

Qtree SnapMirror

Qtree SnapMirror supports replication between different volume types. The source and destination volumes can be any of the following types.

- Traditional volumes
- FlexVol volumes

Note: Starting with Data ONTAP 7.3.2, qtree SnapMirror does not support the use of source Snapshot copies created by releases prior to Data ONTAP 6.5.

Comparison between volume SnapMirror and qtree SnapMirror

You can configure SnapMirror replication for either entire volumes or individual qtrees on a volume. You should consider the differences between the two options.

Volume SnapMirror	Qtree SnapMirror
Synchronous or asynchronous replication is supported for volumes.	Only asynchronous replication is supported for qtrees.
Destination volume is read-only.	Destination qtree is read-only. However, the volume on which the qtree is located must be online and writable.
Source and destination volumes must both be either traditional volumes or FlexVol volumes.	Source and destination qtrees can be on any type of volumes, traditional volumes or FlexVol volumes.

The following table describes the characteristics of SnapMirror replication.

Volume SnapMirror	Qtree SnapMirror
Replicates Snapshot copies of a source volume and all its qtrees, to the destination volume	Replicates only the contents of an individual qtree to a destination
You need to set a destination volume to restricted, read-only status, before setting it up for replication.	The destination volume for qtree replication is writable, and must not be read-only.
Replication of a volume on the destination takes up the space allocated to the source volume, irrespective of how much of the volume is used for storing data.	If you need to mirror only the data stored on an individual qtree, then SnapMirror replication of that individual qtree uses only as much disk space on the destination volume as the source qtree.
Replication can be set up to a destination volume from only one source volume. This implies that one destination volume cannot be used for replicating multiple source volumes.	Replication can be set up for a maximum of 255 qtrees on any one volume.
Block-for-block replication	Logical replication
It transfers the file system verbatim. Therefore, older releases of Data ONTAP cannot understand file system transfers from a later release of Data ONTAP.	All the files and directories in the source file system are created in the destination file system. Therefore, you can replicate data between a storage system running an older version of Data ONTAP and a storage system running a newer version.
	Note: If the source file system contains a file type that cannot be represented on the destination file system, the replication will fail. For example, Data ONTAP 7.0 supports files up to 16 TB in size, whereas earlier versions of Data ONTAP support files up to 4 TB. If the source system is running Data ONTAP 7.0, the qtree you want to replicate contains a file greater than 4 TB, and the destination system is running an earlier version of Data ONTAP, the replication will fail.

Related references

Prerequisites for SnapMirror on page 88

SnapMirror deployment variations

There are several variations possible while deploying SnapMirror. These variations allow you to customize the solution to suit your requirements.

Source to destination to tape variation: A common variation to the basic SnapMirror backup deployment adds a tape backup of the destination volume. By running a tape backup off the

SnapMirror destination volume (as shown in the following illustration), you do not subject the heavily-accessed source volume to the performance degradation and complexity of a direct tape backup.



Figure 6: SnapMirror deployment: Source to destination to tape

Source to tape to destination variation: A SnapMirror deployment that supports SnapMirror replication over low-bandwidth connections accommodates an initial mirroring between a source and destination volume using physically-transported tape (as shown in the following illustration). After the large base Snapshot copy has been replicated, smaller, incremental Snapshot copy updates can be carried out over a low-bandwidth connection.



Figure 7: SnapMirror deployment: Source to tape to destination

Cascading destinations variation: A variation on the basic SnapMirror deployment and function involves a writable source volume replicated to multiple read-only destinations. The function of this deployment is to make a uniform set of data available on a read-only basis to users from various locations throughout a network and to allow for updating that data uniformly at regular intervals.

Note: The cascade deployment (as shown in the following illustration) is supported for volume SnapMirror only. It is not supported for qtree SnapMirror.



Figure 8: SnapMirror deployment: Cascade

Related concepts

Data replication from one destination to another in a series (cascading) on page 110

Migration from traditional volumes to FlexVol volumes

You can use only qtree SnapMirror to migrate data from traditional volumes to FlexVol volumes, if you use SnapMirror for this type of migration. You cannot use volume SnapMirror for this purpose because it cannot replicate to a different type of volume.

Related concepts

Comparison between volume SnapMirror and qtree SnapMirror on page 97

SnapMirror commands

You can use SnapMirror commands to perform different SnapMirror operations for a volume, qtree, or system.

Table 1: SnapMirror commands

The following table lists the commands for using SnapMirror, along with their corresponding operations.

Command	Operation
snapmirror on	Enable SnapMirror on the system, in order to enable the system to work as both a source and a destination for SnapMirror transfers.
	Note: Alternatively, you can use the options snapmirror.enable on command.
	Attention: After using the snapmirror off command, you should wait for at least 60 seconds before using the snapmirror on command. This ensures that all subsequent SnapMirror transfers work properly.
vol create	Use these commands together to create a restricted, read-only volume, which is required as a destination for volume SnapMirror replication.
vol restrict	Attention: You should not use the vol restrict command for a qtree SnapMirror destination volume.
snapmirror initialize	Start the initial, complete SnapMirror (baseline) transfer from a source volume or qtree to a destination.
snapmirror status	View the status of SnapMirror data transfers.
snapmirror update	Perform a manual update of the SnapMirror destination.
snapmirror quiesce	Stabilize the contents of a destination before a Snapshot copy is taken, by allowing active SnapMirror transfers to finish, and temporarily preventing new transfers. This action ensures a manual Snapshot copy of a stable database.
snapmirror resume	Resume normal data transfer to a destination after it has been quiesced.
snapmirror abort	Stop an active SnapMirror transfer.
snapmirror break	Break the SnapMirror relationship between the source and destination, and convert the destination to a writable volume or qtree.
snapmirror resync	Reestablish the SnapMirror relationship between the source and a former destination volume or qtree. Use this command after the snapmirror break command to resynchronize the contents of the source and destination volumes or qtrees, without repeating the initial transfer.
snapmirror release	Release SnapMirror Snapshot copies on former source volumes or qtrees so that the Snapshot copies can be deleted.
snapmirror off	Turn off SnapMirror functionality for a specified system.
	Note: Alternatively, you can use the options snapmirror.enable off command.
snapmirror store and	Copy a volume to local tape, and continue the backup on subsequent tapes if necessary.
snapmirror use	

Command	Operation
snapmirror retrieve	Initialize or restore a volume from a local tape.
and snapmirror use	
snapmirror destinations	Set up a cascading series of SnapMirror destinations. Use this command to make a uniform set of data available on a read-only basis to users from various locations throughout a network.

For more information about SnapMirror commands, see the na_snapmirror man page.

Related concepts

Initialization of a SnapMirror destination on page 121 Initialization of a SnapMirror destination from tape on page 122 Manual update of a SnapMirror destination on page 146 Scheduled updates for volumes or qtrees on page 144 Conversion of a destination to a writable volume or qtree on page 172 What the quiesce command does on page 179 What the snapmirror resync command does on page 189 How the snapmirror resync command helps minimize data loss on page 192 Use of SnapMirror to copy a volume to a tape on page 165

Related tasks

Enabling SnapMirror by entering license codes on page 87 Turning SnapMirror on on page 87 Initializing a SnapMirror destination on page 122 Initializing a destination for non-qtree data on page 125 Aborting a SnapMirror transfer on page 180 Turning off SnapMirror updates on page 145 Checking SnapMirror data transfer status on page 154 Performing a manual SnapMirror update on page 147 Changing scheduled updates for one volume or gtree on page 144 Turning off scheduled updates for one volume or qtree on page 146 Converting a SnapMirror destination to a writable volume or gtree on page 173 *Releasing partners from a SnapMirror relationship* on page 181 Stabilizing destinations before a Snapshot copy on page 178 Resuming transfers after quiescing a destination on page 179 *Resynchronizing a SnapMirror relationship* on page 190 Initializing a SnapMirror destination by using tape on page 119

Related references

Methods for specifying destination systems on the SnapMirror source on page 132 What SnapMirror status check shows on page 155 Information messages in the SnapMirror status check on page 158 Quota restrictions on page 172 After using the snapmirror break command on page 174 Considerations before using SnapMirror copy to tape as a backup method on page 166

SnapMirror options

You can use the SnapMirror options to specify different SnapMirror options for a system.

You can view the values for the different SnapMirror options by using the options command.

Viewing SnapMirror options

system_A> options snapmirror.enable on

You can use the options snapmirror command to view the values for the SnapMirror options.

Similarly, you can use the options replication command to view the values for the replication options.

Table 2: SnapMirror options

The following table lists the SnapMirror options that you can use, along with their corresponding functions.

Option	Function	Default value
<pre>snapmirror.enable {on off}</pre>	Specifies whether SnapMirror is enabled for the system.	off
	Alternatively, you can use the snapmirror on and snapmirror off commands.	
	Attention: After changing the snapmirror.enable option to off, you should wait for at least 60 seconds before changing the option back to on. This ensures that all subsequent SnapMirror transfers work properly.	

104 | Data ONTAP 7.3 Data Protection Online Backup and Recovery Guide

Option	Function	Default value
<pre>snapmirror.access host=list list is a comma-separated list of the host names of allowed systems.</pre>	Specifies the SnapMirror destinations that are allowed to copy from the system. You can also use the /etc/ snapmirror.allow file to specify the allowed destinations. However, using the snapmirror.access option is the preferred method. When the option is set to legacy, access is controlled by the /etc/ snapmirror.allow file. Note: If both the snapmirror.access	legacy
	option and the /etc/ snapmirror.allow file are used, the snapmirror.access option takes precedence. This can affect the initialization of SnapMirror relationships.	
<pre>snapmirror.log.enable {on off}</pre>	Specifies whether SnapMirror activity is logged in the /etc/log/snapmirror.x files.	on
replication.volume. use_auto_resync {on off}	Specifies automatic resynchronization for synchronous SnapMirror relationships.	off
replication.volume. reserved_transfers <i>n</i>	Specifies the number of reserved transfers for SnapMirror volumes. <i>n</i> is a variable number, and depends on the system model.	0
replication.logical. reserved_transfers <i>n</i>	Specifies the number of reserved transfers for SnapVault and SnapMirror qtrees. <i>n</i> is a variable number, and depends on the system model.	0
replication.volume.transfer_ limits {current previous}	 Specifies the maximum number of transfers that volume SnapMirror can run concurrently. current: sets the limit as per Data ONTAP 7.3.x. previous: sets the limit as per Data ONTAP 7.2.x, which is a lower number. 	current

Option	Function	Default value
replication.logical.transfer_ limits {current previous}	 Specifies the maximum number of transfers that qtree SnapMirror and SnapVault can run concurrently. current: sets the limit as per Data ONTAP 7.3.x. previous: sets the limit as per Data ONTAP 7.2.x, which is a lower number. 	current

For more information about SnapMirror options, see the na_options man page.

Related concepts

How synchronous SnapMirror handles network issues on page 86

Related tasks

Turning SnapMirror logging on on page 184

Related references

Methods for specifying destination systems on the SnapMirror source on page 132

SnapMirror files

SnapMirror uses configuration files, log files, and other files for its operation.

Table 3: SnapMirror files

The following table lists the files used by SnapMirror, along with their corresponding functions.

File	Function
/etc/snapmirror.conf	 Enables you to specify SnapMirror source and destination relationships, along with the following settings: SnapMirror update schedules for a relationship Type of relationship; single path, multipath, or failover Other options for a given SnapMirror relationship

File	Function
/etc/ snapmirror.allow	Enables you to specify the SnapMirror destinations that are allowed to copy from the system.
	Note: You can also use the options snapmirror.access command to specify the allowed destinations. However, if both the options snapmirror.access command and the /etc/ snapmirror.allow file are used, options snapmirror.access takes precedence. This can affect the initialization of SnapMirror relationships.
/etc/log/ snapmirror.x	Records the SnapMirror data transfer history.
	Note: There might be one or more SnapMirror log files.
	The latest logs are stored in the file named snapmirror. The older logs are named snapmirror.0 and snapmirror.1.
/etc/hosts	SnapMirror uses the entries in this file to resolve host names.

For more information about SnapMirror files, see the following man pages:

- na_snapmirror.conf
- na_snapmirror.allow
- na_hosts

Related concepts

What the snapmirror.conf file does on page 135 SnapMirror data transfer logs on page 183 Scheduled updates for volumes or qtrees on page 144 Data replication from one destination to another in a series (cascading) on page 110

Related tasks

Turning SnapMirror logging on on page 184 *Changing scheduled updates for one volume or qtree* on page 144 *Turning off scheduled updates for one volume or qtree* on page 146 *Listing SnapMirror destinations for a volume in a cascading series* on page 113 *Restructuring a cascade* on page 114

Related references

Methods for specifying destination systems on the SnapMirror source on page 132 Format of SnapMirror log files on page 184

SnapMirror support for IPv6

SnapMirror supports the use of IPv6 addresses to specify source and destination systems. However, there are some differences between the specification of IPv6 and IPv4 addresses.

The SnapMirror commands that are affected by the support for IPv6 addresses are given in the following list.

- snapmirror initialize
- snapmirror update
- snapmirror resync

Note: Before using IPv6 functionality for a system, ensure that the ip.v6.enable option is set to on.

When using an IPv6 address to specify a SnapMirror source or destination system, you need to enclose the IPv6 address within square brackets. The usage is shown in the following examples.

Use of IPv6 address with the snapmirror initialize command

```
systemB> snapmirror initialize -S
[fd20:8b1e:b255:4166:2a0:98ff:fe07:23f3]:src_vol dst_vol
```

Use of IPv6 address in a snapmirror.conf file entry

```
[fd20:8b1e:b255:4166:2a0:98ff:fe07:23f3]:src_vol dst_system:dst_vol -
15 2 * *
```

Use of IPv6 addresses in a snapmirror.conf file entry for a multipath SnapMirror relationship

```
relation_1=multi(fd20:8ble:b255:4166:2a0:98ff:fe07:23f3,dst_system)
(2001:0:0:0:0:ffd3:0:57ab,dst_system)
```

```
relation_1:src_vol dst_system:dst_vol - * * * *
```

Use of IPv6 address in the snapmirror.allow file

fd20:8b1e:b255:4166:2a0:98ff:fe07:23f3

Use of IPv6 address with the snapmirror.access option

When using an IPv6 address with the snapmirror.access option, it is optional to enclose the access specification within double quotes. The usage is shown in the following example.

```
systemB> options snapmirror.acess
"host=fd20:8ble:b255:4166:2a0:98ff:fe07:23f3"
```

Related references

Syntax for snapmirror.conf file entries on page 137

Setting up a basic SnapMirror operation

Before initiating SnapMirror operations, you must enable the appropriate licenses on the source and destination systems. Also, you need to specify the destination systems that can access the source system for updates.

Before you begin

If your source volumes contain directories that are accessed by CIFS clients, you should ensure that those directories are in the Unicode format before replicating the volume using SnapMirror.

Also, you should ensure that you have appropriate SnapMirror licenses for both the source and destination systems.

Steps

1. For both the source and the destination system consoles, use the license add command to enable the SnapMirror license on the source and destination systems.

```
license add snapmirror_license_code
```

2. On the source system console, use the options snapmirror.access command to specify the host names of systems that are allowed to copy data directly from the source system.

Example

options snapmirror.access host=d_systemA

3. On the destination system, create or edit the /etc/snapmirror.conf file to specify the volumes and qtrees to be copied and the schedule (*minute hour day_of_month day_of_week* or sync) on which the destination is updated.

Example

The following entry specifies Snapshot copy replication from vol0 of s_systemA to vol1 of d_systemA at a maximum of 2,000 kilobytes per second 15 minutes past every hour, Monday through Friday.

s_systemA:vol0 d_systemA:vol1 kbs=2000,restart=always 15 * * 1,2,3,4,5

To synchronously mirror vol0 to vol1, use an entry similar to the following.

s_systemA:vol0 d_systemA:vol1 - sync

For more information about schedule entries in the /etc/snapmirror.conf file of the destination system, see the na_snapmirror.conf(5) man page.
- 4. On both the source and destination system consoles, use the snapmirror on command to enable SnapMirror on the source and destination systems.
- **5.** Prepare the destination system appropriately, depending on whether you are setting up SnapMirror volume or qtree replication.

If you are setting up a	Then
Volume SnapMirror relationship	On the destination system console, use the vol create command to create a destination volume, then use the vol restrict command to mark the volume as restricted.
Qtree SnapMirror relationship	Ensure that the volume on the destination system where you want to replicate a qtree with SnapMirror is online and not restricted. Do not manually create a destination qtree.

6. On the destination system console, use the snapmirror initialize command to create an initial complete (baseline) copy of the source on the destination and start the mirroring process.

For SnapMirror volume replication:

Example

Invoking the following command transfers a complete copy of the source volume (vol0 on systemA) to the destination volume (vol2 on systemB). The destination volume must be configured as restricted and read-only.

snapmirror initialize -S systemA:vol0 systemB:vol2

For SnapMirror qtree replication:

Example

The following command creates a destination qtree (qtree4 on vol1 on systemB) and transfers a complete copy of the source qtree (qtree4 on vol1 on systemA) to that destination qtree. The volume in which the destination qtree is created must be online and writable.

snapmirror initialize -S systemA:/vol/vol1/qtree4 systemB:/vol/vol1/ qtree4

After using the snapmirror initialize command, the scheduled Snapshot copy replication that you specified in Step 3 automatically updates the destination volume or qtree at the specified times.

After you finish

If the SnapMirror source volume or qtree becomes unavailable, you can use the snapmirror break command to make the destination volume or qtree writable. This enables you to provide continued access to data for the clients who are no longer able to access the unavailable source.

Related concepts

What the snapmirror.conf file does on page 135

Initialization of a SnapMirror destination on page 121 *Conversion of a destination to a writable volume or qtree* on page 172

Related tasks

Enabling SnapMirror by entering license codes on page 87 *Turning SnapMirror on* on page 87

Related references

Destination accessibility when using CIFS with SnapMirror on page 92 Methods for specifying destination systems on the SnapMirror source on page 132

Firewall usage with SnapMirror

SnapMirror uses the typical socket/bind/listen/accept sequence on a TCP socket.

SnapMirror source binds on port 10566. The destination storage system contacts the SnapMirror source storage system at port 10566 using any of the available ports assigned by the system. The firewall must allow requests to this port of the SnapMirror source storage system.

Synchronous SnapMirror requires additional TCP ports to be open. The source storage system listens on TCP ports 10566 and 10569. The destination storage system listens on TCP ports 10565, 10567, and 10568. Therefore, you should ensure that the firewall allows a range of TCP ports from 10565 to 10569.

Data replication from one destination to another in a series (cascading)

You can replicate data from a SnapMirror destination to another system using SnapMirror. Therefore, a system that is a destination for one SnapMirror relationship can act as the source for another SnapMirror relationship. This is useful when you need to copy data from one site to many sites.

Instead of replicating data from a single source to each of the destinations, you can replicate data from one destination to another destination, in a series. This is referred to as cascading.

Note: You can replicate data from a destination volume in the same way you replicate from a writable source volume.

Next topics

Supported cascade configurations for SnapMirror on page 111 Supported three-hop cascade configurations for SnapMirror on page 111 How SnapMirror handles Snapshot copies for cascading destinations on page 113 Listing SnapMirror destinations for a volume in a cascading series on page 113 Restructuring a cascade on page 114

Related concepts

Initialization of a SnapMirror destination on page 121

Related tasks

Setting up a basic SnapMirror operation on page 108 *Enabling SnapMirror by entering license codes* on page 87

Related references

Methods for specifying destination systems on the SnapMirror source on page 132 Maximum number of concurrent replication operations on page 128

Supported cascade configurations for SnapMirror

Only certain types of SnapMirror configurations support cascading.

The supported cascading configurations are listed in the following table. Any other configuration, such as extending the cascade beyond the number of cascades shown in the table, is not supported. This limitation does not apply to the strictly asynchronous volume SnapMirror cascading configuration, which can propagate to more than three systems.

Table 4: Supported two-hop cascade configurations for SnapMirror

The following table lists the two-hop cascade configurations that are supported for SnapMirror replication.

System A to system B	System B to system C
Synchronous SnapMirror	Asynchronous volume SnapMirror
Asynchronous volume SnapMirror	Asynchronous volume SnapMirror
Asynchronous volume SnapMirror	Qtree SnapMirror
Qtree SnapMirror	Asynchronous volume SnapMirror

This table should be read from left to right.

Example

The first line states that system A has a synchronous SnapMirror relationship with system B, and that system B has an asynchronous volume SnapMirror relationship with system C.

When the first hop in the cascade is synchronous SnapMirror, the synchronous replication can be to one destination system only. Subsequent SnapMirror replications cascading from that destination system must be asynchronous and can be to multiple destination systems.

Supported three-hop cascade configurations for SnapMirror

Only certain combinations of SnapMirror replication types are supported for three-hop cascade configurations.

Cascades of three SnapMirror relationships whose first relationship is a synchronous SnapMirror relationship are supported on Data ONTAP 7.1.2 and 7.2.1 releases, and later releases.

Table 5: Supported three-hop cascade configurations for SnapMirror

System A to system B	System B to system C	System C to system D
Synchronous SnapMirror	Asynchronous volume SnapMirror	Asynchronous volume SnapMirror
Synchronous SnapMirror	Asynchronous volume SnapMirror	Qtree SnapMirror
Asynchronous volume SnapMirror	Asynchronous volume SnapMirror	Asynchronous volume SnapMirror
Asynchronous volume SnapMirror	Asynchronous volume SnapMirror	Qtree SnapMirror
Asynchronous volume SnapMirror	Qtree SnapMirror	Asynchronous volume SnapMirror
Qtree SnapMirror	Asynchronous volume SnapMirror	Asynchronous volume SnapMirror

The following table lists the three-hop cascade configurations that are supported for SnapMirror replication.



Figure 9: Copying data between systems in a cascade configuration

To support a series of cascading volume destinations as shown in the preceding diagram, the entries in the /etc/snapmirror.conf file in each of the systems is the cascade should be similar to the following entries.

```
systemA:vol1 systemB:vol1 - 15 * * 1,2,3,4,5
systemA:vol1 systemL:vol1 - 15 * * 1,2,3,4,5
systemB:vol1 systemC:vol1 - 25 * * 1,2,3,4,5
systemC:vol1 systemD:vol1 - 35 * * 1,2,3,4,5
systemL:vol1 systemM:vol1 - 25 * * 1,2,3,4,5
systemM:vol1 systemX:vol1 - 35 * * 1,2,3,4,5
systemM:vol1:systemN:vol1 - 35 * * 1,2,3,4,5
```

systemX:vol1 systemZ:vol1 - 45 * * 1,2,3,4,5

Note: When specifying the destination update schedule in the snapmirror.conf file, you should stagger the update times instead of starting multiple destination updates at the same time. If SnapMirror does not have enough resources to perform all scheduled destination updates, it postpones some updates. As a result, SnapMirror might need to perform subsequent updates at times that are different from those you specify in the snapmirror.conf file.

How SnapMirror handles Snapshot copies for cascading destinations

SnapMirror retains on the original source volume the Snapshot copies needed for transfers to destinations further down the line. Snapshot copies that are still needed by a destination are labeled snapmirror in the output of the snap list command. SnapMirror deletes the Snapshot copies it no longer needs.

If you remove a destination from the cascade, you can use the snapmirror release command from the immediate source to tell SnapMirror to delete the Snapshot copies associated with that destination.

Listing SnapMirror destinations for a volume in a cascading series

You can use the snapmirror destinations command to display the destinations for a volume in a cascading series.

The snapmirror destinations command also displays entries related to vol clone command and dump command operations (if any) for SnapMirror source or destination volumes.

Step

1. From the system with the volume serving as the source, enter the following command:

snapmirror destinations [-s] [volume_name]

The -s option generates a list of the names of the Snapshot copies retained for each destination.

volume_name is the name of the source volume for which you want to see the destinations.

Listing SnapMirror destinations for a volume in a cascading series

Suppose that you have used the snapmirror destinations command for a cascade configuration depicted in the following figure.



about each destination volume on the system.

Restructuring a cascade

You might want to restructure a cascade to balance the load on your systems; to use a sytem or volume for a different purpose; or to perform upgrades, maintenance, or repairs.

For example, in the following cascade structure, you might want to make systemD:voll a destination of systemM:voll instead of a destination of systemC:voll.





Steps

1. On the destination system, change the /etc/snapmirror.conf file to indicate the new source for the destination.

Example

```
systemM:vol1 systemD:vol1 - 35 * * 1,2,3,4,5
```

2. As required, choose one of the actions from the following table.

If the newest Snapshot copy on the destination	Then			
Exists on the source	Use the following command to update the destination from the new source.			
	snapmirror update -S source_volume dest_system:dest_volume			
	For example:			
	<pre>snapmirror update -S systemM:vol1 systemD:vol1</pre>			
Does not exist on	Perform one of the following tasks.			
the source	 Update the new source from the original source using the snapmirror update command. Wait for the destination to update. Make the destination writable using the snapmirror break command. Then resynchronize the destination with the new source using the snapmirror resync command. 			

3. Release the former source using the following command:

```
snapmirror release source_volume [[dest_system:]dest_volume]
Example
```

```
systemC> snapmirror release systemC:vol1 systemD:vol1
```

Disconnecting a destination from a cascading series

The diagram depicts the change in the SnapMirror cascade configuration.



Note: If you want to permanently release a destination, you should delete the entry in the / etc/snapmirror.conf file. Alternatively, you can comment out the entry by preceding it with a pound sign (#). Otherwise, SnapMirror attempts to update the destination.

SnapMirror source to tape to destination scenario

In this scenario, you want to establish a SnapMirror relationship between a source system and a destination system over a low-bandwidth connection. Incremental Snapshot mirroring from the

source to the destination over the low bandwidth connection is feasible, but the initial base Snapshot mirroring is not.

In such a case, it might be faster to first transfer the initial base Snapshot image from source to destination using tape, and then set up incremental SnapMirror updates to the destination system through the low-bandwidth connection.

This scenario uses the following configuration.

- A low-bandwidth connection between the source and destination systems
- A local tape drive attached to the source system
- A local tape drive attached to the destination system

Note: To prevent extended tape-to-storage system transfer time, it is recommended that the destination system disks be the same size and in the same RAID configuration as the source system disks.

You must follow this sequence of activities to set up this arrangement.

- 1. On the source system, use the snapmirror store command to copy all volume Snapshot copies, including the base Snapshot copy, to tape, and use the snapmirror use command to continue the copy if more than one backup tape is necessary.
- 2. Physically transport the backup tapes from the source system to the destination system.
- **3.** On the destination system, use the vol create and vol restrict commands to set up a SnapMirror target volume.
- 4. Use the snapmirror retrieve command to copy the initial SnapMirror tape to the destination system and, if necessary, use the snapmirror use command to continue the copy if it is stored on more than one backup tape.
- 5. Either use the snapmirror update command to manually mirror an incremental update from the source to the destination system over the low-bandwidth connection, or edit the snapmirror.conf file to set up an incremental update schedule from the source to destination system.
- 6. After completing manual or scheduled incremental update over a connection, you can use the snapmirror release command to eliminate the source-to-tape relationship and associated Snapshot copy.

Next topics

Copying source to intermediate tape on page 118 Initializing a SnapMirror destination by using tape on page 119 Releasing a SnapMirror source-to-tape relationship on page 120

Copying source to intermediate tape

For an initial, baseline transfer for a SnapMirror relationship, you might want to do the transfer using a tape. To do this, you first need to copy the base data to tape.

Steps

1. If you do not know whether the disk geometries (that is, the size and number of disks) of the source volume and the ultimate destination volume match, use the snapmirror retrieve -g command to determine if this is so.

In the consoles of both the source and destination systems, enter the following command:

snapmirror retrieve -g vol_name

For both the source and the destination volume, the system displays the number and block size of the disks it includes.

Example

A return value of 7200000x10 7000x10 means that the specified volume consists of 10 disks of 720,000 blocks and 10 disks of 7,000 blocks.

- The most desirable result is for the source and destination volume disk geometries to match.
- If the geometries do not match, record the disk geometry of the destination volume for later use.
- 2. At the source system, start the data transfer to tape by entering the following command:

snapmirror store [-g dest_disk_geom] source_volume dest_tapedevices

-g dest_disk_geom applies if the disk geometry of the destination volume, as determined in Step 1, is different from the disk geometry of the source volume. If they are different, use the -g parameter to specify the destination volume disk geometry.

Example

-g 140000x10,7000x10

source_volume is the volume you are copying.

dest_tapedevices is a comma-separated list of tape devices to which you are copying the volume.

Example

snapmirror store -g 14000x10,7000X10 vol2 nrst0a,rst1a

SnapMirror starts transferring data to tape. This command creates a baseline transfer. If you have to use more than one tape, SnapMirror prompts you for another tape.

3. If SnapMirror prompts you for another tape, add another tape to the drive, and continue transfer of data to tape by entering the following command:

snapmirror use dest_tapedevices tape_drive

dest_tapedevices is the tape device or a comma-separated list of tape devices to which you are copying the volume.

tape_drive is the drive holding the new tape you supplied.

Note: The snapmirror use command does not support copying qtrees from tape. If you specify a qtree path as the source or destination, SnapMirror returns an error message.

4. Repeat the previous step until SnapMirror finishes copying the volume to tape.

Related concepts

Initialization of a SnapMirror destination on page 121 *Initialization of a SnapMirror destination from tape* on page 122

Related tasks

Initializing a SnapMirror destination by using tape on page 119

Initializing a SnapMirror destination by using tape

For an initial, baseline transfer for a SnapMirror relationship, you might want to do the transfer using a tape. After copying the base data to the tape, you need to copy the data from the tape to the intended destination, and initialize the destination.

Steps

- 1. Create a volume on the SnapMirror destination system. For information about how to create a volume, see the *System Administration Storage Management Guide*.
- 2. Put the volume in the restricted state. For information about how to restrict a volume, see the *System Administration Storage Management Guide*.
- 3. Load the tape (made with snapmirror store) into the destination system's local tape device.
- 4. Start the initialization by entering the following command on the destination system.

snapmirror retrieve dest_volume tape_drive

dest_volume is the volume that you are initializing.

tape_drive is a tape device or a comma-separated list of devices from which you are restoring the volume.

SnapMirror starts transferring data from the tape. If data is stored on more than one tape, SnapMirror prompts you for the next tape.

5. If the system prompts you for another tape, add the next tape to the drive and continue the initialization by entering the following command:

snapmirror use volume tape_list

volume is the volume to which you are restoring.

tape_list is the tape device from which you are restoring the volume.

- 6. Repeat the previous step until SnapMirror finishes initializing the volume from the tape.
- 7. If you need to, you can update the data online manually with the following command:

snapmirror update [-k n] -S source_system:source_volume [dest_system:]dest_volume

-k n sets the maximum transfer speed to n kilobytes per second. This option has the same effect as the kbs argument in the /etc/snapmirror.conf file.

-S source_system:source_volume specifies the source system and volume for the migration. source_volume is the volume you want to copy.

dest_system is the name of the destination system.

dest_volume is the destination volume.

Note: Alternatively, you can update the baseline transfer automatically with the schedule you set in the /etc/snapmirror.conf file.

Related concepts

Manual update of a SnapMirror destination on page 146 What the snapmirror.conf file does on page 135 Initialization of a SnapMirror destination on page 121 Initialization of a SnapMirror destination from tape on page 122

Related tasks

Editing the snapmirror.conf file on page 136

Releasing a SnapMirror source-to-tape relationship

After a successful update of the destination system over a connection, you no longer require the source-to-tape SnapMirror relationship and its associated Snapshot copy that you established with the snapmirror store command.

Steps

1. In the source system console, enter the following command:

snapmirror status

The system displays at least two SnapMirror relationships.

- The source-to-tape relationship, established when you used the snapmirror store command
- The source-to-destination relationship, established when you used the snapmirror update command

Example

>snapmirror status source destination state s_system:vol1 snapmirror_tape_01_15_03_20:05:32 ... s_system:vol1 d_system:vol1 snapmirrored ...

2. Release the source-to-tape SnapMirror relationship. For this particular operation, the snapmirror release syntax is unique.

```
snapmirror release source_vol tape_snapshot_name
```

Example

snapmirror release vol1 snapmirror_tape_01_15_03_20:05:32

Note: Do not release any other SnapMirror relationship. Those relationships are necessary to continue your incremental updates over the low-bandwidth connection.

Initialization of a SnapMirror destination

You must use the snapmirror initialize command to perform a complete (baseline) transfer of information whenever you start a SnapMirror source-destination relationship for the first time. This process is known as initializing a destination. You need to consider a number of issues when initializing a destination.

Next topics

Quotas for SnapMirror destination qtrees on page 122 Guidelines for creating a qtree SnapMirror relationship on page 122 Initialization of a SnapMirror destination from tape on page 122 Initializing a SnapMirror destination on page 122 Space guarantee for a volume SnapMirror destination on page 124 Initializing a destination for non-qtree data on page 125 How the snapmirror initialize command copies volumes on page 125 How the snapmirror initialize command copies qtrees on page 126 What happens after SnapMirror makes the initial copy to the destination on page 126 How to check the initialization of a volume on page 126 Checking the initialization of a qtree on page 127 How the snapmirror initialize command matches source and destination volume size on page 127 What you can do if an initial SnapMirror transfer fails on page 127

Quotas for SnapMirror destination qtrees

Qtree quotas apply to qtrees in a SnapMirror relationship.

If a destination qtree is limited to 100 GB, transfers from a source qtree greater than 100 GB will fail unless the source qtree drops to less than 100 GB or the quota for the destination qtree is increased.

Note: User quotas also apply to SnapMirror destination qtrees. However, a SnapMirror qtree update does not fail if the user exceeds the quotas.

Guidelines for creating a qtree SnapMirror relationship

When creating a qtree SnapMirror relationship, you should follow certain guidelines.

The following considerations apply to qtree naming and language settings, when setting up a qtree SnapMirror relationship.

- Establish a qtree SnapMirror relationship between volumes that have the same vol lang settings.
- After establishing a qtree SnapMirror relationship, do not change the language assigned to the destination volume.
- Avoid white space (space or tab characters) in names of source and destination qtrees.
- Do not rename volumes or qtrees after establishing a qtree SnapMirror relationship.

Initialization of a SnapMirror destination from tape

You can initialize a SnapMirror destination volume from tape using the snapmirror store command on the source volume and the snapmirror retrieve command on the destination volume.

The snapmirror store and snapmirror retrieve functions are valid only for volumes, but not for qtrees in a SnapMirror relationship.

Related concepts

Use of SnapMirror to copy a volume to a tape on page 165

Related tasks

Initializing a SnapMirror destination by using tape on page 119

Initializing a SnapMirror destination

You can initialize a SnapMirror destination by using the snapmirror initialize command.

Before you begin

If your source volumes contain directories that are accessed by CIFS clients, you should ensure that those directories are in the Unicode format before carrying out the initial SnapMirror replication of that volume.

Steps

1. If the destination is a volume, is online, and has not been initialized before, enter the following command from the destination system.

vol restrict dest_volume

dest_volume is the destination volume.

Note: Do not use the vol restrict command on a qtree. If you are initializing a qtree, go to step 2.

2. From the destination system, enter the following command:

```
snapmirror initialize [options] [dest_system:] {dest_volume|qtree_path}
```

options can be one or more of the following:

- -k n sets the maximum transfer speed to n kilobytes per second. This option has the same effect as the kbs argument in the /etc/snapmirror.conf file.
- -S [source_system:]{source_volume | source_qtree_path} specifies the source system and volume or qtree to copy.

source_volume is the volume you want to copy.

Note: The source specified must match an entry for *source_volume* in the /etc/ snapmirror.conf file, if one exists. If an entry exists but does not match, the operation displays an error message and terminates. If there is no entry for the specified source, the command runs.

source_qtree_path is the path to the qtree you want to copy. If the -S option is not set, the source must be specified in the /etc/snapmirror.conf file. If it is not specified, the operation displays an error message and terminates.

Note: The *source_qtree_path* can be a qtree in a SnapMirror destination volume.

• -c snapshot_name creates a Snapshot copy (with the name snapshot_name) of a qtree on the destination after the next update (so that it does not compete with any ongoing updates). SnapMirror does not lock or delete this Snapshot copy.

Note: *snapshot_name* cannot be minutely.*x*, hourly.*x*, nightly.*x*, or weekly.*x*, because these names are reserved for scheduled Snapshot copies.

Note: This option is valid only for a qtree.

-s snapshot_name specifies an existing source qtree Snapshot copy to be transferred. This
prevents the normal action of the source creating a Snapshot copy to transfer. SnapMirror
does not lock or delete this Snapshot copy.

Note: This option is valid only for a qtree SnapMirror replication.

dest_system is the name of the destination system. The destination can reside on the same system as the source or on another system.

dest_volume or qtree_path specifies the destination volume or qtree. If it is associated with a local source specified in the /etc/snapmirror.conf file, SnapMirror uses that source. If the

destination volume or qtree specified is not in a scheduled relationship, then the -S option must be used to provide a source.

The snapmirror initialize command creates the destination qtree, but you must specify the destination qtree name at the end of the path as though it already existed.

Note: If the destination qtree exists before the command runs, the command fails.

Example

Using the following command, SnapMirror transfers a complete copy of the source volume (vol0 on systemA) to the destination volume (vol2 on systemB).

```
systemB> snapmirror initialize -S systemA:vol0 systemB:vol2
```

Example

Using the following command, SnapMirror transfers a complete copy of the qtree source (qtree4 on vol1 on systemA) to the destination qtree (qtree4bak on vol1 on systemB).

```
systemB> snapmirror initialize -S systemA:/vol/vol1/qtree4 systemB:/vol/
vol1/qtree4bak
```

Related references

Destination accessibility when using CIFS with SnapMirror on page 92

Space guarantee for a volume SnapMirror destination

You can reserve space for a FlexVol volume in an aggregate by specifying the appropriate option for the volume. You can use this feature to ensure a space guarantee for a volume SnapMirror destination.

You can view the settings for a given volume using the vol options command.

The default setting for a volume is: guarantee=volume. This indicates that space for the volume is reserved in the aggregate. You can change this setting using the following command.

```
vol options vol_name guarantee {none | file | volume}
```

For more information about volume space guarantees, see the *Data ONTAP Storage Management Guide*.

Note: In a volume SnapMirror relationship, you can have guarantee=volume for both the source and the destination volumes.

Initializing a destination for non-qtree data

Non-qtree data is any data on a system that is not contained in its qtrees. Non-qtree data can include configuration and logging directories on the system (for example, /etc or /logs) that are not normally visible to system clients. You can use SnapMirror to replicate non-qtree data.

Step

1. To use SnapMirror to replicate non-qtree data from a source to a destination, enter the following command on the destination system:

```
snapmirror initialize -S source_system:/vol/source_volume/-
dest_system:/vol/dest_volume/qtree_name
```

The dash (-) character indicates all non-qtree data in the specified volume.

Note: The snapmirror initialize command creates a destination qtree with the name specified, and then transfers the non-qtree data in the volume. The qtree must not exist before using the command, otherwise the command fails.

Example

Using the following command, SnapMirror transfers to the destination qtree (non_qtree_data_in_vol3 on vol4 on systemB) a complete copy of all the data in vol3 (of systemA) that is not a part of a qtree.

```
systemB> snapmirror initialize -S systemA:/vol/vol3/- systemB:/vol/vol4/
non_qtree_data_in_vol3
```

Note: The non-qtree data can only be a SnapMirror source, never a destination. Although you can copy data from the non-qtree data to another qtree, you cannot perform a vice versa operation.

Initializing a destination for non-qtree data

If you run the snapmirror quiesce or the snapmirror break command on the destination volume (/vol/vol4/non_qtree_data_in_vol3), you can resynchronize the destination volume to the source volume.

```
systemB> snapmirror resync -S /vol/vol3/- /vol/vol4/
non_qtree_data_in_vol3
```

Note: You cannot resynchronize the SnapMirror relationship in the opposite direction.

How the snapmirror initialize command copies volumes

When the snapmirror initalize command copies a volume, it creates a Snapshot copy of all the data on the source and transfers it to the destination. The destination is a volume that you have already created and marked restricted. After SnapMirror finishes transferring the data, it brings the

destination online in a read-only state. This version of the destination is the baseline for the first incremental update.

While the initial data transfer is taking place, the destination is marked invalid in the output of a vol status command. The volume becomes valid and goes online after the initial transfer is complete.

Note: Any manual attempt to bring this volume online will only succeed after the initial transfer is complete.

How the snapmirror initialize command copies qtrees

To use SnapMirror to copy a qtree, you do not create a destination qtree because the snapmirror initialize command creates it. The volume where you want the destination qtree to be must be online. After the destination qtree is initialized, it is no longer writable. However, the rest of the volume where that qtree resides is still writable.

The destination Snapshot copy created by qtree initialization is marked busy in the output of the snap list command until the next transfer is complete.

What happens after SnapMirror makes the initial copy to the destination

After you initialize a SnapMirror volume replication, the files and Snapshot copies in the source volume are available on the destination. After you initialize a SnapMirror qtree replication, the files on the source qtree are available on its destination qtree.

You can export the destination for NFS mounting or add a share corresponding to the destination for CIFS sharing.

How to check the initialization of a volume

To check that a destination volume has been initialized, you can use the snapmirror status command.

If you specify no options or arguments, the snapmirror status command displays the status of the volumes in the system, as shown in the following example. You also can use the vol status or the qtree command to check whether the volume or qtree is a SnapMirror destination.

```
systemA> snapmirror status
Snapmirror is on.
Source Destination State Lag Status
systemA:vol0 systemA:vol0bak Snapmirrored 00:56:58 Idle
systemA:vol1 systemB:vol6 Source 23:69:26 Transferring (126 MB done)
```

Related tasks

Checking SnapMirror data transfer status on page 154

Checking the initialization of a qtree

You can check the creation and initialization of a destination qtree by using the qtree command.

Step

1. Enter the following command:

qtree

Note: If you specify no options or arguments, the qtree command displays the status of all the qtrees in the system.

Example

```
systemA> qtree
```

```
VolumeTreeStyleOplocksStatusvol0unixenablednormalqtree24unixenablednormalsystemB_vol0unixdisablednormalsystemB_vol0qt1mixedenabledsnapmirroredsystemB_vol0qt2unixdisablednormalsystemB_vol0qt3ntfsenabledsnapmirrored
```

How the snapmirror initialize command matches source and destination volume size

When you use the snapmirror initialize command to initialize a volume replication, SnapMirror sets the vol options fs_size_fixed option to on. This option forces the file system on the destination volume to remain the same size as the file system on the source volume.

What you can do if an initial SnapMirror transfer fails

If an initial SnapMirror transfer fails, you can resume the transfer by re-entering the snapmirror initialize command, under certain conditions.

If the following conditions are met, you can resume the initialization of a SnapMirror relationship.

- The value for restart mode in the /etc/snapmirror.conf file is set to always or is set to the default, and the next scheduled update has not begun.
- The output of the snapmirror status command displays that the process has a restart checkpoint.
- The Snapshot copy used for the initial SnapMirror transfer still exists.
- The disk geometry has not changed.

Note: If these conditions are not satisfied, you cannot resume the initial transfer. You need to start the initial SnapMirror transfer again.

SnapMirror does not automatically retry to initialize a destination.

Maximum number of concurrent replication operations

There is a limit on the number of concurrent replication operations, which depends on your system model. The system resources are shared between SnapMirror and SnapVault replication operations. Therefore, the limit for a particular type of replication is reduced if there are any other types of replication operations being performed concurrently.

A SnapMirror or a SnapVault replication consists of two replication operations: one operation on the source system and the other on the destination. Therefore, if a system is both a source and a destination, it uses two replication operations.

Each value listed in the following two tables indicates the maximum number of concurrent replication operations allowed for a specific model, when using a single type of replication. The values are exclusive, and not cumulative. For example, if you are using the maximum number of synchronous SnapMirror replication operations for a system as a source, you cannot use any more replication operations of any type for the system.

Model	Volume Synchrono SnapMirror SnapMirro		ronous Airror	nous Qtree rror SnapMirror		SnapVault		Open Systems SnapVault	
	SRC	DSTN	SRC	DSTN	SRC	DSTN	SRC	DSTN	DSTN
FAS250	4	4	4	4	4	4	4	4	4
FAS270 /	8	8	8	8	8	8	8	8	8
GF270c									
920	8	8	8	8	8	8	8	8	8
940	16	16	16	16	16	16	16	16	16
960	16	16	16	16	16	16	16	16	16
980	50	50	16	16	64	64	64	64	16
FAS2020	8	8	8	8	16	16	16	16	8
FAS2040	50	50	16	16	64	64	64	64	16
FAS2050	8	8	8	8	32	32	32	32	8
3020	50	50	16	16	64	64	64	64	16

Table 6: Maximum number of concurrent replication operations without NearStore personality license

Model	Volt SnapN	ume ⁄Iirror	Synch: SnapN	ronous ⁄Iirror	Qtree SnapMirror		Snap	Vault	Open Systems SnapVault
3040	50	50	16	16	64	64	64	64	16
3050	50	50	16	16	64	64	64	64	16
3070	50	50	16	16	64	64	64	64	16
3140	50	50	16	16	64	64	64	64	16
3160	50	50	16	16	64	64	64	64	16
3170	50	50	16	16	64	64	64	64	16
6030	100	100	24	24	96	96	96	96	24
6040	100	100	24	24	96	96	96	96	24
6070	150	150	32	32	128	128	128	128	32
6080	150	150	32	32	128	128	128	128	32
R200	64	64	16	16	64	128	64	128	128

SRC—Source; DSTN—Destination

Table 7: Maximum number of concurren	t replication operations with NearStore
personality license	

Model	Vol SnapN	ume ⁄Iirror	Synchronous SnapMirror		Qtree SnapMirror		SnapVault		Open Systems SnapVault
	SRC	DSTN	SRC	DSTN	SRC	DSTN	SRC	DSTN	DSTN
FAS2020	8	8	8	8	40	40	40	40	16
FAS2040	50	100	16	16	160	160	160	160	64
FAS2050	8	16	8	8	80	80	80	80	32
3020	50	100	16	16	80	80	80	80	32
3040	50	100	16	16	160	160	160	160	64
3050	50	100	16	16	120	120	120	120	64
3070	50	100	16	16	320	320	320	320	128
3140	50	100	16	16	160	160	160	160	64

Model	Volume SnapMirror		Synchronous SnapMirror		Qt SnapN	ree ⁄Iirror	Snap	Vault	Open Systems SnapVault
3160	50	100	16	16	320	320	320	320	128
3170	50	100	16	16	384	384	384	384	128
6030	100	200	24	24	384	384	384	384	96
6040	100	200	24	24	384	384	384	384	96
6070	150	300	32	32	512	512	512	512	128
6080	150	300	32	32	512	512	512	512	128

SRC—Source; DSTN—Destination

Note:

- R200 is a NearStore system. Therefore, it does not require the NearStore personality license.
- The NearStore personality license is not supported on the following systems:
 - FAS200 series/GF270c
 - 900 series

The following factors affect the maximum number of concurrent replication operations that a system can achieve.

- Heavy use of system resources, such as CPU, memory, disk bandwidth, or network bandwidth, might reduce the resources available for SnapMirror or SnapVault operations.
- The use of TCP window sizes higher than the default value, for SnapMirror relationships, might restrict the system from achieving the maximum limit specified.
- A system with the NearStore personality enabled is optimized as a destination for SnapMirror and SnapVault replication operations. Therefore, a system with the NearStore personality can handle more replication operations than a system without the NearStore personality.

Next topics

Enabling the NearStore personality on page 131 *Maximum number of concurrent replication operations in an active/active configuration* on page 132

Related concepts

Limit on the number of concurrent SnapVault targets on page 236 *Use of SnapMirror with S Family storage systems* on page 206

Enabling the NearStore personality

You can enable the NearStore personality to optimize your system as a destination for replications.

Before you begin

You must meet certain requirements before you can use the NearStore personality:

• The following storage systems, with the corresponding Data ONTAP version, support the NearStore personality.

Model	Data ONTAP version
FAS20xx	7.2.4L1 or later
FAS30xx	7.1 or later
V30xx	7.2.2 or later
3140	7.2.5.1 or later
3160	7.2.6 or later
3170	7.2.5.1 or later
FAS60xx	7.2 or later
V60xx	7.2.2 or later

• If you want to use the SnapVault feature, the storage system must have a SnapVault secondary license enabled.

About this task

When enabled, the nearstore_option license does the following.

- Allows a higher number of concurrent SnapMirror and SnapVault replication operations when the system is used as a destination.
- Allows SnapVault for NetBackup to be enabled on 30xx, 31xx, and 60xx systems.

Step

1. On the system console, enter the following command:

license add xxxxxxx

xxxxxx is the NearStore personality license code.

Maximum number of concurrent replication operations in an active/active configuration

If a failover occurs, the available system cannot process more than the maximum number of concurrent replication operations specified for that system. These operations can be those that were scheduled for the surviving system, the failed-over system, or both.

If a failover occurs during data backup, all of the concurrent transfers happening at the time of the failure on the failed node are aborted, and are rescheduled by the partner node. Conversely, if a giveback occurs during data backup, all of the concurrent transfers happening at the time of the giveback on behalf of the partner node are aborted, and are rescheduled by the partner node.

If more than the maximum number of SnapMirror volume or qtree replications are scheduled to run concurrently, each additional transfer generates an error message stating that resource limits have been reached. Each transfer beyond the maximum is re-attempted once per minute until it succeeds, SnapMirror is turned off, or the update is terminated.

Methods for specifying destination systems on the SnapMirror source

There are two methods of specifying destination systems on the source systems.

The two methods to specify destination systems on the source are:

- By using the snapmirror.access option
- By using the snapmirror.allow file

Next topics

Specifying SnapMirror destinations using the snapmirror.access option on page 132 Specifying SnapMirror destinations using the snapmirror.allow file on page 133 Resolving host names to their IP addresses on page 134

Specifying SnapMirror destinations using the snapmirror.access option

You can specify the destination systems that are allowed access to the source system by using the snapmirror.access option. This option specifies which SnapMirror destination system can initiate transfers, and which network interfaces they can use. This is the preferred method for controlling SnapMirror access on SnapMirror source system.

Step

1. To specify the SnapMirror destinations that are allowed access to the SnapMirror source using the snapmirror.access option, on the source system, enter the following command:

options snapmirror.access access_specification

The syntax is the same for SNMP, Telnet, and rsh, and is described in the na_protocolaccess(8) man page. For more information about the options command, see the na_options(1) man page.

Note: This option setting persists across reboots.

Example

If you want a destination (systemB) to have access to the source (systemA), for a SnapMirror relationship to copy data from systemA to systemB, enter the following at the prompt on systemA.

```
systemA> options snapmirror.access host=systemA,systemB
```

Specifying SnapMirror destinations using the snapmirror.allow file

You can create a snapmirror.allow file in the /etc/ directory on the source system. You can create entries in the snapmirror.allow file to specify the destination systems that are allowed to copy data directly from the source system. If the snapmirror.access option is set to legacy (the default setting), the snapmirror.allow file defines the access permissions.

Steps

- If the snapmirror.allow file...
 Then...

 Does not exist in the /etc/ directory on the root volume of the source system.
 Create the snapmirror.allow file in the / etc/directory on the root volume of the source system by using a text editor.

 Exists in the /etc/ directory on the root volume of the source system.
 Go to the next step.
- 1. As required, choose one of the actions from the following table.

2. Specify the SnapMirror destinations that can access the SnapMirror source using the snapmirror.allow file. Add the name of each allowed system on a separate line in the snapmirror.allow file.

Note: You do not need to add the name of the local system.

3. Save edits to the file.

Example

If you want SnapMirror to copy data locally on systemA and to other systems named systemB and systemC, add the following entries in the /etc/snapmirror.allow file on systemA.

systemB

systemC

Note: Entries in the snapmirror.allow file are case-sensitive. You can use the hostname command on the destination systems to find the correct entries for the snapmirror.allow file.

Resolving host names to their IP addresses

By default, SnapMirror checks host names in the /etc/snapmirror.allow file against the host name sent from the destination system. Alternatively, you can set SnapMirror to resolve the host names in the /etc/snapmirror.allow file to their IP addresses, and compare them with the IP address of the destination system.

Before you begin

The /etc/snapmirror.allow file entry must map to the IP address of the originating network interface on the destination system. For example, if the request comes from the IP address of a Gigabit Ethernet interface el0 named systemA-el0, then the /etc/snapmirror.allow file must contain systemA-el0 or systemA-el0.acme.com so that the name resolves to the correct IP address.

About this task

The snapmirror.checkip.enable option controls how the host names are checked. When the option is off, which is the default, the entries in the /etc/snapmirror.allow file must match the host name of the destination system reported by the hostname command. When the option is on, the source system resolves the names in the snapmirror.allow file to IP addresses and then checks for a match with the IP address of the requesting destination system. In this mode, literal IP addresses (for example, 123.45.67.89) and fully qualified names (for example, systemA.acme.com) can be valid entries in the /etc/snapmirror.allow file.

Note: A local SnapMirror relationship, between two volumes on the same system, does not require an entry in the /etc/snapmirror.allow file.

Step

1. To configure SnapMirror to resolve host names to their IP addresses, enter the following command on the source system:

options snapmirror.checkip.enable on

What the snapmirror.conf file does

The snapmirror.conf file is used to specify the details related to the copying of data, by using SnapMirror, from the source to the destination. This file resides on the destination storage system.

The /etc/snapmirror.conf file defines:

- The relationship between the source and the destination.
- The schedule used by the destination to copy data.
- The arguments that control SnapMirror when copying data.

Next topics

Distribution of the snapmirror.conf file on page 135 Limit on entries in the snapmirror.conf file on page 135 Editing the snapmirror.conf file on page 136 Syntax for snapmirror.conf file entries on page 137

Distribution of the snapmirror.conf file

You can create a single snapmirror.conf file for your site and copy it to all the storage systems that use SnapMirror. The snapmirror.conf file can contain entries pertaining to other storage systems.

For example, the /etc/snapmirror.conf file on systemB can contain an entry for copying a volume from system C to system D. When systemB reads the /etc/snapmirror.conf file, it ignores the entries for other storage systems.

Note: Each time the file is read, a warning message is displayed on the storage system console for each line that is ignored.

Limit on entries in the snapmirror.conf file

The limit on the number of entries for each system in the /etc/snapmirror.conf file is 1,024. Entries beyond this limit are ignored and the system console displays a warning message.

If you have an active/active configuration, the limit on the number of entries is shared between the two systems in the active/active configuration. For example, in an active/active configuration, if one system uses 480 entries, the other system has 544 entries available for its use.

Note: This limitation is different from the maximum number of concurrent replications you can have on a system.

Related references

Maximum number of concurrent replication operations on page 128

Editing the snapmirror.conf file

You should edit the snapmirror.conf file only when there are no active SnapMirror transfers for the source and destination relationships that you want to change. If SnapMirror is enabled, changes take effect within two minutes. If SnapMirror is not enabled, changes take effect immediately after you enable SnapMirror.

Before you begin

Check whether the snapmirror.conf file exists in the /etc/ directory on the root volume of the destination system. If the file does not exist, create the snapmirror.conf file by using a text editor.

About this task

If you change the host name in a snapmirror.conf file entry when there are more than one active SnapMirror transfers for that specific source and destination relationship, one or more of the transfers might be aborted.

Steps

1. For each destination volume or qtree on this system, type an entry specifying the source, destination, characteristics, and schedule of the data transfer on one line using the following syntax.

source_system:{source_volume | /vol/volume_name/qtree_name} dest_system:
{dest_volume | /vol/volume_name/qtree_name} arguments schedule

Note: When using a qtree as a source, you should not create the destination qtree manually. SnapMirror automatically creates the destination qtree for you, using the name you specify. However, you must specify the name and path of the destination qtree, either in the /etc/ snapmirror.conf file or in the snapmirror initialize command.

Note: You can only specify up to 254 destination qtrees for a specific volume in the /etc/ snapmirror.conf file.

If you want to replicate all non-qtree data within a volume to a destination qtree, use the following syntax.

source_system:/vol/source_volume/- dest_system:/vol/dest_volume/ qtree_name

The hyphen (-) character indicates all non-qtree data in the specified volume.

Note: The data in /vol/source_volume/- qtree can only be a SnapMirror source, not a destination.

2. If you want to add comments to the /etc/snapmirror.conf file, precede the comment with a pound (#) sign.

Example

Replicating from systemA

3. Save edits to the file.

Related references

Syntax for snapmirror.conf file entries on page 137

Syntax for snapmirror.conf file entries

You need to use the correct syntax for each entry in the snapmirror.conf file, to specify the source, the destination, the options, and the schedule for a SnapMirror relationship.

The syntax for entries in the snapmirror.conf file is as follows.

src_system:/vol/src_vol[/src_qtree] dest_system:/vol/dest_vol[/dest_qtree] arguments schedule

The parameters in the entries are listed in the following table.

Parameter	Description
src_system	The name of the system from which you are copying data. SnapMirror uses the /etc/hosts file or the database used by DNS and NIS for name resolution. When SnapMirror searches for the source system name, it should find the IP address for the source system on the network over which you want the transfer to occur.
	Example: Suppose you created a private network for connecting source and destination systems, and you name the interface on the source as systemA-e0. You must enter the interface name systemA-e0 in the <i>src_system</i> field.
src_vol/src_qtree	The name of the volume or qtree that you are copying. Use only the volume name for volumes. Use the full path name for qtrees.
	Example: If the name of the volume is vol1, enter vol1 in the <i>src_vol/src_qtree</i> field. If the name of the qtree is qtree3, and it is contained in the volume vol3, enter the full path, / vol/vol3/qtree3, in the <i>src_vol/src_qtree</i> field.

Parameter	Description			
dest_system	The host name of the system to which the data is copied. The name you use must be the exact host name of the destination system.			
	Note: The <i>dest_system</i> field is case-sensitive.			
	You can use the hostname command on the destination system to determine what you can enter in this field.			
	Example: If the name of the destination system is systemA, enter systemA in the <i>dest_system</i> field.			
-	Using a hyphen (-) specifies the use of default parameters.			
dest_vol[/dest_qtree]	The name of the destination volume or qtree to which you are copying data. Use only the volume name for volumes. Use the full path name for qtrees.			
	Example: If the name of the volume is vol1, enter vol1 in the dest_vol[/dest_qtree] field. If the name of the qtree is qtree4, and it is in vol2, enter the full path, /vol/vol2/ qtree4, in the dest_vol[/dest_qtree] field.			
kbs= <i>kbs</i>	Maximum transfer speed, in kilobytes per second, that Data ONTAP can use to transfer data.			
	The kbs and restart arguments are expressed as a comma- separated list of <i>name=value</i> pairs, with no spaces. For example:			
	kbs=2000,restart=always			

Parameter	Description
restart={ never always default }	Restart mode that SnapMirror uses to continue an incremental transfer from a checkpoint, if it is interrupted. The three options are:
	 never—Transfers are always restarted from the beginning of a transfer and never from where they were before an interruption. This mode is useful if you must have the latest data on the destination. always—Transfers are always restarted, if possible, from where they were before an interruption. This mode is useful for copying large volumes. default—Transfers are restarted if they do not conflict with a scheduled transfer. This is the recommended option. SnapMirror always restarts from where the transfer was before an interruption. However, the transfer restarts only if the restart occurs before a scheduled transfer.
<pre>cksum= { none crc32c crc32c_header_only }</pre>	Selects the checksum algorithm that is used to check the data replicated by SnapMirror.
	none specifies that SnapMirror does not perform a checksum for the data. This option assumes that the network used by SnapMirror is reliable and delivers the data packets from the source to the destination without errors.
	crc32c specifies that the data being transferred by SnapMirror is checked by an application-level 32-bit cyclic redundancy check. This check ensures that any transmission errors not corrected at the network level are detected by SnapMirror. However, using this option adds to the processing load of both the source and destination systems.
	crc32c_header_only specifies that only the headers of the SnapMirror data packets are checked by the cyclic redundancy check. This check enhances the reliability of volume SnapMirror, with negligible performance impact. This option can be used only with asynchronous volume SnapMirror. crc32c_header_only is the default option for volume SnapMirror.

Parameter	Description
<pre>visibility_interval={xs xm xh}</pre>	Determines the amount of time before an automatic Snapshot copy is created on the source volume that is being replicated using synchronous or semi-synchronous SnapMirror. When replicating synchronously or semi-synchronously, using SnapMirror, changes to the source volume do not show immediately on the destination volume, even though the changes have been replicated. The changes are shown only after the source system takes an automatic Snapshot copy of the source volume. This event occurs every three minutes by default. You can change the interval for automatic Snapshot copies, but performance can degrade if you set smaller intervals because Snapshot copies are taken more often. The smallest interval you can set is 30 seconds. Note: The s, m, and h suffixes specify seconds, minutes, and
	hours, respectively.
wsize= <i>size</i>	Determines the TCP window size used by the SnapMirror relationship. Note: The TCP window size is specified in bytes.
compression=enable	Enables the compression feature for the SnapMirror relationship. This option can only be used for asynchronous volume SnapMirror. This option is not supported for qtree SnapMirror or synchronous SnapMirror. To disable compression, remove this option from the
	SnapMirror relationship entry in the snapmirror.conf file.
<pre>connection_mode={inet6 inet unspec}</pre>	Specifies whether the relationship uses an IPv6 or IPv4 connection.
	 unspec is the default value of the connection_mode option. An IPv6 connection is attempted first. If an IPv6 connection is not established, then an IPv4 connection is attempted. inet6 specifies the use of an IPv6 connection. inet specifies the use of an IPv4 connection.

schedule specifies the time and frequency of the SnapMirror updates on the destination. Specifying the schedule is mandatory.

schedule consists of four space-separated fields in the following order:

minute hour dayofmonth dayofweek

- *minute* can be a value from 0 to 59.
- *hour* can be a value from 0 to 23.
- *dayofmonth* can be a value from 1 to 31.
- *dayofweek* can be a value from 0 (Sunday) to 6 (Saturday).

Multiple values, separated by commas, can be entered for any field.

All possible values for a field can be applied with an asterisk (*). If you specify an asterisk in each field of the schedule, SnapMirror updates the destination every minute.

A single dash (-) in any field means "never" and prevents this schedule entry from executing. (This option is useful if you want the server to appear in the /etc/snapmirror.conf file so that snapmirror update can find it, but you do not want the SnapMirror scheduler to run automatically.)

A range of values for any field can be indicated with a low value and a high value separated by a dash. For example, you can indicate that you want an update every hour from 8:00 a.m. to 5:00 p.m by entering this value in the hour field:

8-17

A range of values followed by a slash and a number indicates the frequency of the update. For example, you can indicate that you want an update every five minutes by entering this value in the minutes field:

0-59/5

Using sync or semi-sync instead of the four space-separated fields specifies synchronous or semisynchronous replication, respectively. For more information, see the na_snapmirror.conf(5) man page.

Example of snapmirror.conf file schedule entries

Suppose you create a private network between systemA and systemB. In the /etc/hosts file on systemA, you specify the host name for the interface as systemA-e0, and you ensure that systemA-e0 is also in the /etc/hosts file on systemB, the destination system. You want to copy vol0 of systemA to vol1 of systemB over the private network every Monday, Wednesday, and Friday at 11 p.m. You also want to use the default for the arguments field.

To copy the data over the private network every Monday, Wednesday, and Friday at 11 p.m., you would enter the following in the /etc/snapmirror.conf file.

systemA-e0:vol0 systemB:vol1 - 0 23 * 1,3,5

The following table indicates what the entry in each field in the example means.

Entry	Description
systemA-e0	Name of the source system, along with the interface to be used
vol0	Source volume

142 | Data ONTAP 7.3 Data Protection Online Backup and Recovery Guide

Entry	Description
systemB	Name of the destination system
voll	Destination volume
-	Use default values for arguments
0	Starts a SnapMirror update at the 0th minute of the hour(s), as specified in the next (hour) field
23	Starts a SnapMirror update at 11p.m.
*	Starts a SnapMirror update on all applicable days of the month, as specified in the next field
1,3,5	Starts a SnapMirror update on Monday, Wednesday, and Friday

Example of a snapmirror.conf file entry with an IPv6 address

The following snapmirror.conf file entry shows the use of an IPv6 address to specify the source system for a given SnapMirror relationship. Note that the IPv6 address is enclosed within square brackets.

[fd20:8b1e:b255:4166:2a0:98ff:fe07:23f3]:src_vol dst_system:dst_vol 15 2 * *

Example of a snapmirror.conf file entry with space character and double quotes in qtree name

The following snapmirror.conf file entry shows how qtree names with space character and double quotes must be specified. This entry indicates that data from the qtree vol/vol1/x y"z on systemA is replicated to the qtree vol/vol2/x y"z on systemB.

"systemA:/vol/vol1/x y""z" "systemB:/vol/vol2/x y""z" - * * * *

Note: Qtree names with space character or double quotes might not work with previous versions of Data ONTAP. Therefore, when reverting to a previous version of Data ONTAP, you must check for compatibility.

Example setting maximum update speed

The following line in an /etc/snapmirror.conf file sets the speed to 2000 kilobytes per second.

systemA:vol0 systemA:vol1 kbs=2000 15 * * 1,2,3,4,5

Note: The specified transfer speed might not be achievable because transfer speed is limited by factors such as network bandwidth.

Example specifying always restart

The following line in an /etc/snapmirror.conf file sets the restart value to always.

```
systemA:vol0 systemA:vol1 kbs=2000,restart=always 15 * * 1,2,3,4,5
```

Example specifying default values for maximum speed and restart

If you set the value of only one argument (kbs or restart), the other argument uses the default value. If you want to use the default argument for both values, enter a dash (-).

The following line in an /etc/snapmirror.conf file sets both arguments to the default value.

systemA:vol0 systemA:vol1 - 15 * * 1,2,3,4,5

Example specifying 15-minute interval updates during specific hours

If you want to schedule an update every afternoon at 1:00, 1:15, 1:30, 1:45, 5:00, 5:15, 5:30, 5:45, 7:00, 7:15, 7:30, and 7:45, you should enter the following in the schedule field.

```
systemA:vol0 systemA:vol1 - 0,15,30,45 13,17,19 * *
```

Note: An update is started when the current time matches a value in all four fields. Be careful that a value in the day of the month field does not exclude a value in the day of the week field. For example, the following schedule updates the destination every afternoon at 1:00, 1:30, 3:00, 3:30, 5:00, and 5:30, but only when the first day of the month falls on a Monday.

```
systemA:vol0 systemA:vol1 - 0,30 13,15,17 1 1
```

Note: The schedule represents the goal of the SnapMirror feature. Factors that might prevent SnapMirror from updating every minute include resource limitations or network connectivity. If an update is in progress when another is scheduled to occur, SnapMirror will start another transfer as soon as the first transfer is complete. However, if more than one update gets queued while a transfer is in progress, SnapMirror only perfoms the update queued last. The other updates become obsolete due to the later update.

Related concepts

Compression for SnapMirror transfers on page 151 Synchronous SnapMirror modes on page 85 SnapMirror support for IPv6 on page 107

Related tasks

Adjusting the TCP window size for a SnapMirror relationship on page 162

Scheduled updates for volumes or qtrees

You can edit the destination's snapmirror.conf file to change or turn off scheduled updates for a particular volume or qtree if you decide that there is no need to update the destination.

You might want to change the time or frequency of scheduled updates if the pattern of use or the configuration of the systems has changed. Or, if you want to use the volume for a different purpose, you can change the destination to a writable volume.

Note: Editing entries in the destination's snapmirror.conf file to turn off scheduled updates does not change the destination to a writable volume. If you want to change the destination to a writable volume or qtree, you use the snapmirror break command to turn the destination into a writable volume or qtree and the snapmirror release command to allow SnapMirror to delete the Snapshot copies it no longer needs on the source.

You can edit the destination's snapmirror.conf file to turn off or change scheduled updates at any time, even when data transfer is underway. The destination remains the same as before the transfer. The Snapshot copy taken in the source for the data transfer remains, but it can be deleted and replaced by a new Snapshot copy the next time the destination is updated.

Next topics

Changing scheduled updates for one volume or qtree on page 144 *Turning off SnapMirror updates* on page 145 *Turning off scheduled updates for one volume or qtree* on page 146

Related concepts

Conversion of a destination to a writable volume or qtree on page 172

Related tasks

Converting a SnapMirror destination to a writable volume or qtree on page 173 *Releasing partners from a SnapMirror relationship* on page 181

Changing scheduled updates for one volume or qtree

You can change scheduled updates for one volume or qtree, by editing the snapmirror.conf file on the destination.

Step

1. In the destination's snapmirror.conf file, edit the destination volume or the schedule information to specify the configuration you want.
Example

Original update schedule:

```
systemA:vol0 systemA:vol1 - 0 23 * 1,3,5
systemA:vol1 systemB:vol6 - 0 23 * 1,3,5
Changed update schedule
systemA:vol0 systemA:vol2 - 0 23 * 1,3,5
systemA:vol1 systemB:vol6 - 0 23 * 2,4,6
```

Turning off SnapMirror updates

You can use the snapmirror off command to turn off updates, both scheduled and manual, for the entire system at any time, even when copying is underway. Any active transfer is aborted when you turn off SnapMirror for the system. The destination remains unchanged after you turn off updates.

About this task

This process affects all SnapMirror transfers for the system, whether the system is the source or the destination of the SnapMirror relationship.

Steps

1. Enter the following command on both the source system and destination system to disable SnapMirror.

options snapmirror.enable off

Alternatively, you can use the snapmirror off command to turn off SnapMirror.

If SnapMirror is currently transferring data from one volume or qtree to another, the transfer aborts immediately. The destination remains the same as before the transfer. The Snapshot copy taken in the source volume for the data transfer remains.

SnapMirror stops monitoring the /etc/snapmirror.conf file for changes.

Entering the snapmirror off command on the destination system alone does not affect SnapMirror on the source system. Other systems can continue to copy data from the source system.

Note: Both the snapmirror off command and the snapmirror.enable off option are persistent across reboots.

2. If the snapmirror on command is in the /etc/rc file, remove the command (to keep the current setting after reboot). Otherwise, the setting in the /etc/rc file overrides the command you entered.

Turning off scheduled updates for one volume or qtree

You can turn off scheduled updates for one volume or qtree, by editing the snapmirror.conf file.

Step

- 1. Either delete the entry in the snapmirror.conf file or change the entry by:
 - Commenting out the entry by preceding it with a pound sign (#).

Example

```
systemA:vol0 systemA:vol1 - 0 23 * 1,3,5
systemA:vol1 systemB:vol6 - 0 23 * 1,3,5
#systemB:vol1 systemC:vol2 - 0 23 * 1,3,5
```

• Putting a dash (-) in one of the schedule fields (minute/hour/dayofmonth/dayofweek).

Note: Deleting or commenting out a destination or putting a dash in one of the schedule fields of a destination in the /etc/snapmirror.conf file does not prevent you from performing manual updates to that destination.

Example

systemA:vol0 systemA:vol1 - 0 23 * 1,3,5 systemA:vol1 systemB:vol6 - 0 23 * 1,3,5 systemB:vol1 systemC:vol2 - 23 * 1,3,5

Manual update of a SnapMirror destination

SnapMirror automatically updates the destination according to the update schedule specified in the snapmirror.conf file. You can also initiate updates manually by using the snapmirror update command.

You might need a manual update to prevent data loss due to an upcoming power outage, scheduled maintenance, or data migration. You can also include the snapmirror update command in an external script if you want to drive updates using that script.

Next topics

Performing a manual SnapMirror update on page 147 Creating extra backup Snapshot copies for SnapMirror qtrees on page 148 What happens after SnapMirror makes incremental updates to the destination on page 149

Performing a manual SnapMirror update

You can perform an unscheduled SnapMirror incremental update, independent of the schedule in the /etc/snapmirror.conf file.

Step

1. From the destination system, enter the following command:

snapmirror update [options] [dest_system:] {dest_volume | /vol/ dest_volume/qtree_path}

options can be one or more of the following:

- -k *n* sets the maximum transfer speed to *n* kilobytes per second. This option has the same effect as the kbs argument in the /etc/snapmirror.conf file.
- -s snapshot_name specifies an existing (qtree only) source Snapshot copy to be transferred, rather than a Snapshot copy taken by the source. SnapMirror does not lock or delete this Snapshot copy.

Note: *snapshot_name* cannot be minutely.*x*, hourly.*x*, nightly.*x*, weekly.*x*, snapshot_for_backup.*x* or snapshot_for_volcopy.*x*. You must rename such Snapshot copies on the source and then copy them.

• -c snapshot_name creates a Snapshot copy named snapshot_name of a qtree on the destination after the next update (so that it does not compete with any ongoing updates). SnapMirror does not lock or delete this Snapshot copy.

Note: *snapshot_name* cannot be minutely.*x*, hourly.*x*, nightly.*x*, or weekly.*x*, because these names are reserved for scheduled Snapshot copies.

-S [source_system:]source_volume | qtree_path specifies the source system and volume for the update. source_volume is the volume you want to copy.
 The source specified by the -S option must match an entry for source_volume in the /etc/ snapmirror.conf file. If an entry exists but does not match, the operation displays an error message and terminates. If there is no entry for the specified source volume, the command runs.

Note: If the -S option is not set, the source must be specified in the /etc/ snapmirror.conf file. If it is not specified, the operation displays an error message and terminates.

dest_system specifies the name of the destination system.

dest_volume specifies the destination volume. If it is a scheduled destination of a local source volume, as specified in the /etc/snapmirror.conf file, that source volume is considered to be the source. If the destination volume specified is not in a scheduled relationship, then the -S option must be used to provide a source.

Example

Using the following command, SnapMirror updates the destination (vol2 on systemB) from the source specified in the /etc/snapmirror.conf file.

systemB> snapmirror update systemB:vol2

Example

Using the following command, SnapMirror updates the qtree destination on systemB:/vol/vol2/usersbak from the source qtree on systemA:/vol/vol1/users.

```
systemB> snapmirror update -S systemA:/vol/vol1/users systemB:/vol/vol2/
usersbak
```

Creating extra backup Snapshot copies for SnapMirror qtrees

You might want to create an extra backup Snapshot copy on the source and destination in a qtree SnapMirror relationship. You can create a manual Snapshot copy using the snap create command. Then, you can use the -c and -s options of the snapmirror update command together to establish an extra backup Snapshot copy on both sides of a SnapMirror qtree relationship.

About this task

These Snapshot copies can serve as the newest common Snapshot copy in case a base Snapshot copy is accidentally deleted. You can also use them to resynchronize a SnapMirror qtree relationship to an earlier resynchronization point.

Steps

1. Enter the following command:

```
snap create vol_name snapshot_name
```

vol_name is the name of the volume whose Snapshot copy you want to create.

snapshot_name is the name of the Snapshot copy.

Example

systemB> snap create vol2 my_snap

This creates a Snapshot copy, my_snap, of the volume vol2.

2. Enter the following command:

snapmirror update -S source -s src_snap -c dest_snap destination
source is the name of the source.

src_snap is the name of the Snapshot copy on the source.

dest_snap is the name of the Snapshot copy on the destination.

destination is the name of the destination.

Example

```
systemA> snapmirror update -S systemB:/vol/vol2/qtree1 -s my_snap -c
my_dest_snap vol/vol4/qtreeSafe
```

What happens after SnapMirror makes incremental updates to the destination

The destination reflects the changes on the source after SnapMirror completes the transfer. If the SnapMirror transfer is incomplete or interrupted, the changes on the destination are not visible till the transfer is complete. After SnapMirror completes the destination update, you can see the changes when you open the file.

Note: SnapMirror automatically deletes old Snapshot copies that are no longer necessary for updating data.

SnapMirror over multiple paths

You might want more than one physical path for a SnapMirror relationship. SnapMirror supports up to two paths for a particular SnapMirror relationship.

When using multiple paths, you need to set up the configuration in one of the following ways.

- Set up static routes to ensure different routes are used for different IP connections.
- Use different subnets for the two connections.

The paths can be Ethernet, Fibre Channel, or a combination of Ethernet and Fibre Channel. The two paths can be used in one of these two modes.

- Multiplexing mode—SnapMirror uses both paths at the same time, essentially load balancing the transfers. If one path fails, the transfers occur on the remaining path. After the failed path is repaired, the transfers resume using both paths.
- Failover mode—SnapMirror uses the first specified path as the desired path and uses the second specified path only after the first path fails.

Next topics

Setting up a multipath SnapMirror relationship on page 150 Converting a single-path SnapMirror relationship to multipath on page 150

Related concepts

Compression for SnapMirror transfers on page 151

Setting up a multipath SnapMirror relationship

You can use multiple paths between the source and destination systems for baseline initialization.

About this task

You can set up SnapMirror to use multiple paths at the outset. You can also convert a single path SnapMirror relationship to use multiple paths.

Steps

- 1. Ensure that you have two valid paths using the ping command from the source system to each of the IP addresses on the destination system.
- 2. On the source system console, use the options snapmirror.access command to specify the host names of systems that are allowed to copy data directly from the source system.

Example

options snapmirror.access host=d_systemA

3. Edit the snapmirror.conf file on the destination system to add an entry that defines the mode of the connection and what the two connections are. The format of the entry is as follows:

name=mode(src_system-e0,dst_system-e0)(src_system-e1,dst_system-e1)

where mode is either multi or failover. See the na_snapmirror.conf(5) man page for details.

4. Edit the /etc/snapmirror.conf file on the destination system to specify the volumes and qtrees to be copied and the schedule (*minute*, *hour*, *day_of_month*, *day_of_week*, sync, or semi-sync) on which the destination is updated. Use the connection name specified in previous step as the source system.

Related references

Methods for specifying destination systems on the SnapMirror source on page 132

Converting a single-path SnapMirror relationship to multipath

You can set up SnapMirror to use multiple paths at the outset. You can also convert a single-path SnapMirror relationship to use multiple paths.

Steps

- 1. Ensure that you have two valid paths using the ping command from the source system to each of the IP addresses on the destination system.
- 2. Edit the snapmirror.conf file on the destination system to add a connection line that defines the mode of the connection and what the two connections are. The format of the line is as follows:

name=mode(src_system-e0,dst_system-e0)(src_system-e1,dst_system-e1)

mode is either multi or failover. See the na_snapmirror.conf(5) man page for details.

3. In the same snapmirror.conf file, edit the schedule entry to reflect the new connection name as the source system.

Note: Multiple paths are supported by SnapMirror running asynchronously and synchronously. The following are examples of implementing multiple paths using synchronous SnapMirror.

Example

You want to synchronously replicate volume vol1 on a system called NYC to volume vol1 on a system called Newark. You require two physical paths between source and destination systems for each synchronously-mirrored volume. You have two network interface cards on each system. You named the two interfaces on the NYC system NYC-pri and NYC-sec, and the two on the Newark system Newark-pri and Newark-sec. To implement multiple paths in the failover mode, you edit the snapmirror.conf file on Newark to include the following two lines.

```
NYC-Newark=failover(NYC-pri,Newark-pri)(NYC-sec,Newark-sec)
```

NYC-Newark:vol1 Newark:vol1 - sync

Example

If NYC-pri and Newark-pri are Fibre Channel NIC adapters and you want to replicate data using both connections, you follow the procedure to configure Fibre Channel NIC adapters for SnapMirror. Then, you edit the snapmirror.conf file on Newark to include the following two lines to implement multiple paths in multi mode.

NYC-Newark=multi(NYC-pri,Newark-pri)(NYC-sec,Newark-sec)

NYC-Newark:vol1 Newark:vol1 - sync

Related tasks

Configuring SnapMirror over Fibre Channel on page 212

Compression for SnapMirror transfers

The compression feature of SnapMirror enables data compression over the network for SnapMirror transfers. It helps in optimizing network bandwidth utilization between SnapMirror source and destination systems. This feature can be especially useful for connections that have relatively low bandwidth, such as WAN connections. This feature is supported only for asynchronous volume SnapMirror.

The compression feature of SnapMirror increases resource utilization on both the SnapMirror source and destination systems. Therefore, you need to evaluate the usage and benefits before deploying compression. For example, compression might not be useful for a high-bandwidth, low-latency connection.

Next topics

Enabling compression for a SnapMirror transfer on page 152 *Viewing the compression ratio for a SnapMirror transfer* on page 153

Related concepts

SnapMirror over multiple paths on page 149

Related references

Syntax for snapmirror.conf file entries on page 137

Enabling compression for a SnapMirror transfer

You can enable compression for a volume SnapMirror relationship by specifying the option in the snapmirror.conf file.

Before you begin

- The compression feature is supported only for asynchronous volume SnapMirror.
- The SnapMirror destination system should be using Data ONTAP 7.3.2.

About this task

There are certain considerations for enabling and disabling compression for SnapMirror relationships.

- You can enable or disable compression for both initial and incremental SnapMirror transfers.
- You need to enable the compression feature on a per-relationship basis.
- You cannot enable or disable compression for an active SnapMirror transfer. However, after you edit the snapmirror.conf file, transfers activated after a minute would use the updated option. If you want to ensure that the change in the option is applied immediately, you can use the snapmirror on command.

Steps

- 1. Open the snapmirror.conf file for editing.
- 2. You can enable compression for existing or new SnapMirror relationships, defined in the snapmirror.conf file. Ensure that the SnapMirror relationship is specified as shown in the following syntax.

connection_name=multi(src_system,dst_system)

connection_name:src_vol dst_system:dst_vol - * * * *

connection_name is the name of this specific connection between a SnapMirror source and destination system pair.

src_system is the name of the SnapMirror source system.

dst_system is the name of the SnapMirror destination system.

src_vol is the path of the SnapMirror source volume.

dst_vol is the path of the SnapMirror destination volume.

3. To enable the compression feature for a SnapMirror relationship, add the compression=enable option to the relationship specification.

The syntax of the entries in the snapmirror.conf file required for enabling compression is given in the following lines.

```
connection_name=multi(src_system,dst_system)
```

```
connection_name:src_vol dst_system:dst_vol compression=enable - * * * *
```

Example

```
conxn_1=multi(src_system,dst_system)
conxn_1:src_vol dst_system:dst_vol compression=enable - * * * *
```

Note: To disable compression, remove the compression=enable option from the SnapMirror relationship entry in the snapmirror.conf file.

Compression for a multipath SnapMirror relationship

The syntax of the entries in the snapmirror.conf file required for enabling compression for a multipath SnapMirror relationship is given in the following lines.

```
connection_name=multi(src_ip1,dst_ip1)(src_ip2,dst_ip2)
```

```
connection_name:src_vol dst_system:dst_vol compression=enable - * * *
*
```

src_ip1 and src_ip2 are the IP addresses of the SnapMirror source system.

dst_ip1 and dst_ip2 are the IP addresses of the SnapMirror destination system.

```
conxn_2=multi(10.72.146.74,10.79.106.40)(10.72.147.74,10.79.107.40)
conxn_2:src_vol dst_system:dst_vol compression=enable - * * * *
```

Viewing the compression ratio for a SnapMirror transfer

For a SnapMirror relationship that uses the compression feature, you can view the compression ratio for an active transfer.

Before you begin

Ensure that the compression feature is enabled. To enable the compression feature, you need to add the appropriate option to the SnapMirror relationship details, in the snapmirror.conf file.

About this task

- The compression ratio is displayed only for a SnapMirror transfer that is in progress.
- The compression ratio is also logged in the SnapMirror log file.

Step

1. To view the compression ratio for an active SnapMirror transfer, enter the following command on the SnapMirror destination:

snapmirror status -1 dst_vol

Viewing the compression r	atio for a SnapMirror transfer
dst_system> snapmirror s Snapmirror is on.	status -l dst_vol
Source: Destination: Status: Progress: Compression Ratio: State: Lag: Mirror Timestamp: Base Snapshot: Current Transfer Type: Current Transfer Type: Contents: Last Transfer Type: Last Transfer Size:	<pre>src_system:src_vol dst_system:dst_vol Transferring 24 KB 4.5 : 1 Initialize - Initialize 132 MB</pre>
Last Transfer Duration: Last Transfer From:	00:00:27 src_system:src_vol

Checking SnapMirror data transfer status

You need to check the data transfer status, by using the snapmirror status command, to determine the status of all existing SnapMirror relationships on the system.

Step

1. Enter the following command:

```
snapmirror status [options] [[system:] [path] ...]
```

options can be one of the following.

- -1 displays the long format of the output, which contains more detailed information.
- -q displays which volumes or qtrees are quiesced or quiescing.

• -t displays which volumes or qtrees are active.

system is the name of the source system.

path is the name of the source volume or the path to and name of the source qtree.

Note: When you use the -t option, the output displays the active relationships. A relationship is considered active if the source or destination is involved in one of the following:

- Data transfer to or from the network.
- Reading or writing to a tape device.
- Waiting for a tape change.
- Performing local on-disk processing or cleanup.

Result

If no arguments or options are given, SnapMirror displays a message that indicates whether a transfer is in progress, how much of the data transfer has been completed, the state of the destination, and the amount of time since the last Snapshot copy was created and transferred successfully.

Next topics

What SnapMirror status check shows on page 155 *Information messages in the SnapMirror status check* on page 158

Related tasks

Stabilizing destinations before a Snapshot copy on page 178

What SnapMirror status check shows

The SnapMirror status check shows information about SnapMirror transfers for volumes or qtrees.

If you check the status and you enabled SnapMirror, messages similar to the following are displayed.

systemA> snapmirror st	atus						
Snapmirror is on.							
Source	Destination	State	Lag	Status			
systemA:vol0	systemA:vol1	Snapmirrored	02:25:11	Transferring	(60 1	MВ	done)
systemB:/vol/vol1/qt3	systemB:/vol/vol3/qt3	Quiesced	00:01:15	Idle			

You see a status report for any SnapMirror source that contains the base Snapshot copy, and for any destination in a current SnapMirror relationship or listed in the /etc/snapmirror.conf file. Destinations that were broken through the snapmirror break command but still contain the base Snapshot copy are listed.

If you check the status of data transfer and you did not enable SnapMirror, the following message is displayed.

systemA> snapmirror status

Snapmirror is off.

Note: The status of SnapMirror relationships, if any, are still displayed, as shown in the preceding example.

Example

With no options, the information displayed by the snapmirror status command looks similar to the following:

```
systemB> snapmirror status

Snapmirror is on.

Source Destination State Lag Status

systemA:vol0 systemB:vol2 Broken-off 29:09:58 Idle

systemC:vol0 systemB:vol3 Snapmirrored 00:09:53 Idle with restart

checkpoint (23 MB done)

systemC:vol4 systemB:vol5 Snapmirrored 00:04:58 Transferring (36 MB

done)

systemA:/vol/vol1/qt5 systemB:/vol/vol4/qt5 Quiesced 00:05:12 Idle

systemC:vol/vol2/qt1 systemB:/vol/vol1/qt2 Snapmirrored 00:02:33 Quiescing
```

Example

With the -1 option, the configuration described in the previous example looks similar to the following:

```
systemB> snapmirror status -1
```

```
Snapmirror is on.
Source:
Destination:
Status:
                                            systemA:vol0
                                           systemB:vol2
                                            Syncing
                                            60 KB
Progress:

      State:
      Source

      Lag:
      00:01:17

      Mirror Timestamp:
      Sat Jul 15 00:50:02 GMT 2000

      Base Snapshot:
      tpubs-f720(0016791363)_vol2.1249

 Current Transfer Type:
 Current Transfer Error: -
 Contents:
Last Transfer Type: Update
Last Transfer Size: 1052 KB
 Last Transfer Duration: 00:00:02
 Last Transfer From: systemA:vol0
Source:systemC:vol0Destination:systemB:vol3Status:Idle with restart checkpointProgress23552 KB doneState:SnapmirroredLag:00:09:53Mirror Timestamp:Sun Jul 16 05:50:07 GMT 2000Base Snapshot:system2(0016778780)_vol3.985Current Transfer Type:-Current Transfer Error:Abort b
 Current Transfer Error: Abort by user
Contents: Replica
Last Transfer Type: Update
Last Transfer Size: 432000 KB
Last Transfer Duration: 00:01:23
```

```
Last Transfer From: systemC:vol0
Source:
                              systemC:vol4
Destination:
                             systemB:vol5
                             Transferring
Status:
                             36864 KB done
Progress
                             Snapmirrored
State:
Lag: 00:04:58
Mirror Timestamp: Sun Jul 16 05:55:02 GMT 2000
Base Snapshot: systemB(0016778780)_vol5.57843
Current Transfer Type: Scheduled
Current Transfer Error: -
Contents: Replica
Last Transfer Type: Scheduled
Last Transfer Size: 345000 KB
Last Transfer Duration: 00:03:23
Last Transfer From: systemB:vol4
                        systemC:/vol/vol1/qt5
systemB:/vol/vol4/qt5
Idle
Source:
Destination:
Status:
Progress
State:
                             Quiesced
                             0:05:12
Lag:

        Mirror Timestamp:
        Sun Jul 16 05:56:12 GMT 2000

        Base Snapshot:
        systemB(0016778780)_vol_vol4_qt5.54

Current Transfer Type:
Current Transfer Error: -
                             Replica
Contents:
Last Transfer Type: Scheduled
Last Transfer Size: 45000 KB
Last Transfer Duration: 0:00:12
Last Transfer From: systemC:/vol/vol1/qt5
Source: systemC:/vol/vol2/qt1
Destination: systemB:/vol/vol4/qt2
Status: Quiescing
Progress
                           Snapmirrored
State:
                             0:02:33
Lag:
Mirror Timestamp: Sun Jul 16 05:58:20 GMT 2000
Base Snapshot: systemB(0016778780)_vol_vol4
Base Snapshot: systemB(0016778780)_vol_vol4_qt2.122
Current Transfer Type: -
Current Transfer Error: -
Contents: Transition
Last Transfer Type: Scheduled
Last Transfer Size: 80 KB
Contents:
                               Transitioning
Last Transfer Duration: 0:00:08
Last Transfer From: systemC:/vol/vol2/qt1
```

Example

With the -q option, the output looks similar to the following:

```
systemC> snapmirror status -q
Snapmirror is on.
vol3 is quiesced
vol2 has quiesced/quiescing qtrees:
```

/vol/vol2/qt1 is Quiescing /vol/vol2/qt2 is Quiesced

Related references

Information messages in the SnapMirror status check on page 158

Information messages in the SnapMirror status check

The snapmirror status command displays information messages.

The SnapMirror status entries on the source category are as follows.

Source entry	Description	
system:vol	The source system and source volume	
system:qtree_path	The source system and qtree path	
-	Either the SnapMirror destination is an imported volume without an entry in the /etc/snapmirror.conf file, or a Data ONTAP upgrade is in progress	
<pre>system:tape_device</pre>	The source tape device; transfer from this tape device is still in progress	
base snapshot	The name of the base Snapshot copy from which a completed transfer was made, if the source is a tape device and there is no entry in the / etc/snapmirror.conf file for the destination	

The SnapMirror status entries on the destination category are as follows.

Destination entry	Description	
system:vol	The destination system and volume	
system:qtree_path	The destination system and qtree path	
system:tape_device	The destination tape device; transfer to this tape device is in progress	
tape_destination	Displayed after a transfer to tape is finished. The snapmirror destinations command also displays this information	

State entry	Description
Uninitialized	The destination is listed in the /etc/snapmirror.conf file, but the volume or qtree has not been initialized or the destination is being initialized.
Snapmirrored	The volume or qtree is in a SnapMirror relationship.

State entry	Description
Broken-off	The destination was in a SnapMirror relationship, but a snapmirror break command made the volume or qtree writable. This state is reported as long as the base Snapshot copy is still present in the volume. If the Snapshot copy is deleted, the state is listed as "uninitialized" if the destination is in the /etc/ snapmirror.conf file or is no longer listed if it is not. A successful snapmirror resync command restores the snapmirrored status.
Quiesced	SnapMirror is in a consistent internal state and no SnapMirror activity is occurring. In this state, you can create Snapshot copies knowing that all destinations are consistent. The snapmirror quiesce command brings the destination into this state. The snapmirror resume command restarts all SnapMirror activities.
Unknown	The destination volume or the volume that contains the destination qtree is in an unknown state. It might be offline or restricted.
Source	When the snapmirror status command is run on the source system and the destination is on another system, the state of the destination is unknown, so the source status is reported.

Lag entry	Description
hh:mm:ss	Indicates the difference between the current time and the timestamp of the Snapshot copy last successfully transferred to the destination.
-	The destination is not initialized.

The SnapMirror status entries on the status category are as follows.

Status entry	Description
Idle	No data is being transferred.
Idle with restart checkpoint (<i>n x</i> B done)	 No data is being transferred. The last transfer attempt was aborted, but the transfer saved a restart checkpoint and thus can be restarted at the next attempt. Transfer sizes are reported in the following units: KB up to 10,240 KB MB up to 10,240 MB GB up to 10,240 GB TB
Transferring	Transfer has been initiated but has not yet started, or is just finishing.

Status entry	Description	
Transferring (<i>n</i> XB done)	Data transfer is in progress. Transfer sizes are reported in the following units:	
	 KB up to 10,240 KB MB up to 10,240 MB GB up to 10,240 GB TB 	
Pending	The destination was not updated because of a transfer failure; the transfer will be retried automatically.	
Pending with restart checkpoint (nXB done)	The destination was not updated because of a transfer failure. The transfer will be retried automatically from the restart checkpoint. Transfer sizes are reported in the following units:	
	 KB up to 10,240 KB MB up to 10,240 MB GB up to 10,240 GB TB 	
Aborting	A transfer is being aborted and cleaned up.	
Quiescing	The specified volume or qtree is waiting for all existing transfers to complete. The destination is being brought into a stable state.	
Resyncing	The specified volume or qtree is being matched with data in the common Snapshot copy.	
Waiting	SnapMirror is waiting for a new tape to be put in the tape device.	

Additional SnapMirror status entries are as follows.

Additional -1 option entries	Description	
Progress	Displays the amount of data (in KB) transferred by the current transfer. Displays the restart check point if the status is Idle or Pending.	
Mirror Timestamp	The timestamp of the last Snapshot copy successfully transferred from the source to the destination.	
	Note: A resynchronization might change the base Snapshot copy to a Snapshot copy with an older timestamp.	

Additional -1 option entries	Description	
Base Snapshot	The name of the base Snapshot copy for the destination.	
сору	For volumes in a SnapMirror relationship, this field is the same on the source side and the destination side. For qtrees in a SnapMirror relationship, the destination side lists the name of the exported Snapshot copy for that qtree on the destination.	
	Note: A resynchronization might change the name of the base Snapshot copy.	
Current Transfer Type	Indicates the kind of transfer now in progress: scheduled, retry, resync, update, initialize, store, or retrieve. This field applies only to the destination side.	
Current Transfer Error	Displays an error message if the latest transfer attempt failed.	
Contents	Indicates whether the contents of the destination volume or qtree in the active file system are up-to-date replicas or in transition. The field applies only to the destination side.	
	 Under SnapMirror volume replication, the contents are always a replica. Under SnapMirror qtree replication, the contents are usually a replica, but sometimes are transitioning. 	
Last Transfer Type	Indicates the kind of transfer previously performed: scheduled, retry, resync, update, initialize, store, or retrieve. This field applies only to the destination side.	
Last Transfer Size	Displays the amount of data (in KB) transferred in the last successful transfer.	
Last Transfer Duration	Displays the elapsed time for the last successful transfer to complete. If the transfer failed and restarted, this includes time waiting to restart the transfer. If a transfer aborted and was retried from the beginning, it includes only the time required for the final successful attempt.	
Last Transfer From	This field applies only to the destination side and shows the name of the source system and volume or qtree. This field is useful if you have changed the source in the /etc/snapmirror.conf file but the data is actually from the old source. For snapmirror retrieve (from tape) operations, this field lists the tape device used in the retrieve operation.	

Related concepts

Data replication from one destination to another in a series (cascading) on page 110

Related tasks

Listing SnapMirror destinations for a volume in a cascading series on page 113

Related references

What SnapMirror status check shows on page 155

Adjusting the TCP window size for a SnapMirror relationship

The TCP window size for SnapMirror might have an impact on SnapMirror performance. You can change the default value the TCP window size to suit the network configuration. You can specify the window size for a particular SnapMirror relationship by modifying the corresponding entry in the snapmirror.conf file.

Before you begin

- Ascertain the round-trip time between the source and the destination for a SnapMirror relationship.
- Determine the bandwidth available for the SnapMirror relationship.
- The default TCP window size for a SnapMirror relationship is 1,994,752 bytes.
- Adjustment of the TCP window size is applicable only for asynchronous SnapMirror relationships.
- For qtree SnapMirror relationships, TCP window sizes higher than the default value are not supported.

Note: You should only adjust the TCP window size for a SnapMirror relationship if there are throughput issues related to bandwidth utilization.

About this task

The TCP window size specifies the amount of data that a source can send through a connection before it requires an acknowledgement from the destination for the data received. A larger TCP window size can increase SnapMirror throughput in certain scenarios. You can change the TCP window size to optimize SnapMirror transfers for the network in use. Thereofre, you can change the TCP window size to optimize SnapMirror transfers.

Note: When using higher TCP window sizes than the default, the system might not be able to achieve the maximum concurrent replication operations specified for the system. This is due to increased resource utilization by the higher TCP window sizes.

The maximum TCP window size that you can specify for a SnapMirror relationship depends on the connection type, as given in the following table.

Connection type	Default TCP window size	Maximum TCP window size
Single path	1,994,752 bytes	7,340,032 bytes (7 MB)
Multipath	1,994,752 bytes	14,680,064 bytes (14 MB)

Note: To limit the network bandwidth used by a particular SnapMirror relationship, use the kbs parameter for the relationship entry in the snapmirror.conf file.

Steps

1. Calculate a TCP window size that works well for a particular SnapMirror relationship by using the following formula:

window size = (round-trip time) × (available bandwidth)

Example

If the average round trip delay is 130 milliseconds and the available bandwidth is 200 Mbps, the equation is:

window size = $(((0.13 \text{ sec}) \times (200,000,000 \text{ bps})) / 8)$ bytes = 3,250,000 bytes

Therefore, you should set the TCP window size for the SnapMirror relationship to 3,250,000 bytes.

Example

Similarly, you can calculate the optimal TCP window size for different round-trip time and bandwidth values. The following table provides a few examples.

Round-trip time	Available bandwidth	TCP window size to maximize throughput
120 ms	400 Mbps	6,000,000 bytes
100 ms	1000 Mbps	12,500,000 bytes (can be used only for a multi-path SnapMirror relationship)
50 ms	155 Mbps	968,750 bytes

2. Specify the required TCP window size by adding the following option to the SnapMirror relationship entry in the snapmirror.conf file.

wsize=value

value is the required TCP window size (in bytes), as calculated in the preceding step.

Example

The following entry specifies a TCP window size of 3,250,000 bytes for the SnapMirror relationship.

src_system:src_vol dst_system:dst_vol wsize=3250000 * * * *

Example

The following entries specify a TCP window size of 3,250,000 bytes for the multipath SnapMirror relationship.

conxn_2=multi(10.72.146.74,10.79.106.40)(10.72.147.74,10.79.107.40)

conxn_2:src_vol dst_system:dst_vol wsize=3250000 * * * * *

Related references

Maximum number of concurrent replication operations on page 128 *Syntax for snapmirror.conf file entries* on page 137

Setting a maximum transfer rate for all transfers

Setting a maximum transfer rate for all transfers enables you to limit the total bandwidth used by all transfers at any time. You can set a maximum rate for transfers coming into a system, and a maximum rate for transfers going out of a system.

About this task

You can configure a maximum transfer rate for a system and set maximum transfer rates for each transfer using the /etc/snapmirror.conf file. When both the rates are configured, the system-level maximum is applied only if the combined bandwidth of transfers goes above the system-level maximum rate.

Note: This setting applies to all SnapMirror and SnapVault transfers.

Steps

1. Enable the ability to set system-level maximum transfer rates using the following command:

options replication.throttle.enable on

2. You can specify the maximum transfer rate used by transfers for a system, as a source or destination. As required, choose one of the actions from the following table.

If you want to specify the maximum Then enter the following command... rate for...

Outgoing SnapMirror transfers (applied at the SnapMirror source).	options replication.throttle.outgoing.max_kbs value
Incoming SnapMirror transfers (applied at the SnapMirror destination).	options replication.throttle.incoming.max_kbs value

value is the maximum transfer rate in kilobytes per second. Valid transfer rate values are 1 to 125000. The default value is unlimited.

For more information about the replication.throttle.enable option, the replication.throttle.incoming.max_kbs option, and the replication.throttle.outgoing.max_kbs option, see the options(1) man page.

Changing the maximum transfer rate for a single SnapMirror transfer

You can specify the maximum transfer rate for scheduled SnapMirror transfers using the kbs option in the snapmirror.conf file. However, you can change the maximum transfer rate for the current transfer by using the snapmirror throttle command. The change applies to the current transfer only, and the next scheduled transfer uses the maximum transfer rate that is specified in the snapmirror.conf file.

About this task

If you change the maximum transfer rate while the current transfer is active, the new maximum transfer rate takes effect within two minutes.

Step

1. To change the maximum transfer rate and apply it to the current transfer, enter the following command on either the source or destination system.

snapmirror throttle n [system:]path

n is the new maximum transfer rate in kilobytes per second. A value of zero (0) disables throttling.

system is the destination system. Use this variable if you are executing the command on the source system.

path is the destination path. The path can be the /volume_name or /vol/volume_name/
qtree_name.

Use of SnapMirror to copy a volume to a tape

You might want to use SnapMirror to copy a volume from a source system to a local tape for several reasons.

- Network transfer time of the baseline transfer between a SnapMirror source and a SnapMirror destination is prohibitive.
- You are backing up SnapVault secondary storage data to tape for offline storage or to protect against the possible loss of the SnapVault secondary.

The snapmirror store command does not support SnapMirror qtree replication to tape. If you specify a qtree path as the source or destination, SnapMirror returns an error message.

Related references

SnapMirror source to tape to destination scenario on page 116 *SnapVault destination to tape backup* on page 200

Considerations before using SnapMirror copy to tape as a backup method

SnapMirror replication to copy data to local tape is meant as an initialization process for SnapMirror relationships and has limitations when used as a backup method.

Limitation	Description
Disk geometry	When SnapMirror replicates data to tape, it optimizes writing of the file system based on the disk geometry of either the source or destination file system. The disk geometry of the source file system is used by default, but you can specify the disk geometry of the destination file system using the snapmirror store -g command. See the na_snapmirror(1) man page for more information.
	If you retrieve a backup tape into a file system that does not match the disk geometry of the system used when writing the data on to tape, the retrieval process can be extremely slow.
	Note: If FlexVol volumes are in place, disk geometry mismatch when restoring the volume is not an issue.
File system version	Data written to tape is a block by block copy of the file system associated with a particular version of Data ONTAP; therefore, when you retrieve data from tape, the destination of the retrieval must use the same or a later version of Data ONTAP used when storing data to tape. If you attempt to retrieve data to a destination that uses an older version of Data ONTAP, the retrieval will fail.
Snapshot copy issues	Because the intended purpose of SnapMirror to tape is to initialize mirrors for SnapMirror relationships, it maintains Snapshot copies that future SnapMirror relationships use to perform updates. When used as a backup method, Snapshot copies are created, but never used; therefore, Snapshot copy and data resources are wasted.
	Note: You can manage and delete Snapshot copies using the snapmirror release and snapmirror destinations commands. See the na_snapmirror(1) man page for more information.
Volume types	The traditional volume file system format is different from the FlexVol volume file system format. Because of this difference, a backed-up FlexVol volume cannot be retrieved on a traditional volume. Likewise, a backed-up traditional volume cannot be retrieved on a FlexVol volume.

Consider the following limitations before using SnapMirror to tape as a backup method.

Limitation	Description
Lack of archive support	Some backups are made to be kept for a long time; as such, they become long term storage of old versions of data. Backups using SnapMirror replication to tape are in a format that is only readable by Data ONTAP and the WAFL file system, and is not meant for long term storage. Keeping a set of tapes for extended periods of time risks the ability to restore them in the future.
Effects of a bad tape	When retrieving a file system from tape, the format of the SnapMirror replication to tape requires the entire file system to be retrieved before the file system is usable. A bad tape prevents some of the file system from being retrieved, which means that none of the data is usable.
Lack of backup features	SnapMirror replication to tape does not have features that backup software does. For example, features like individual file restore, backup indexing, and incremental backups are not supported.

About moving SnapMirror sources

Whether you are moving a volume SnapMirror source or qtree SnapMirror source to new systems or newer drives, as long as there is a Snapshot copy in common on the source and destination, the transition goes smoothly.

Volume SnapMirror transfers all of the Snapshot copies as part of the SnapMirror replication process.

For qtree SnapMirror, the source and destination have only one Snapshot copy in common. Different qtree SnapMirror destinations have no common Snapshot copy, unless the Snapshot copy is specifically replicated.

In a production environment, you should perform the process of moving SnapMirror relationships from one volume or system to another only in a maintenance or out-of-service window. You should also ensure that new data is not added to the original source during the move.

Next topics

Moving volume SnapMirror sources on page 167 *Moving qtree SnapMirror sources* on page 169

Moving volume SnapMirror sources

You can move a volume SnapMirror source volume to another source volume.

Before you begin

Ensure that the Data ONTAP release on the destination system is from a release family that is the same, or later than, the Data ONTAP release on the new source system.

About this task

The following terms are used in the task description for moving volume SnapMirror sources.

- oldsource—The original system on which the source resides.
- newsource—The system to which you are moving the source.
- destination—The system to which the source is replicated.
- oldsourcevol—The original source volume.
- newsourcevol-The new source volume to which you are moving.
- destinationvol—The volume to which the source is replicated.

Steps

1. Copy the original source to the new source using the following command:

```
newsource> snapmirror initialize -S oldsource:oldsourcevol
newsource:newsourcevol
```

Note: This might take some time to finish.

- 2. You should make oldsource read-only before continuing.
- 3. Create a manual Snapshot copy on the oldsource system by using the following command:

oldsource> snap create oldsourcevol common_Snapshot

4. Update newsource and destination based on oldsource using the following commands.

```
newsource> snapmirror update -S oldsource:oldsourcevol
newsource:newsourcevol
```

destination> snapmirror update -S oldsource:oldsourcevol destination:destinationvol

Note: The common_Snapshot copy is on all volumes because all Snapshot copies are mirrored using volume SnapMirror.

5. Quiesce and break the SnapMirror relationship between oldsource and destination, and oldsource and newsource by using the following commands.

destination> snapmirror quiesce destinationvol destination> snapmirror break destinationvol newsource> snapmirror quiesce newsourcevol newsource> snapmirror break newsourcevol

6. Using an editor, update the /etc/snapmirror.conf file on the destination for the new relationship by replacing the oldsource information with newsource information.

Before edit

```
oldsource:oldsourcevol destination:destinationvol restart=always 0 * * *
*
```

After edit

```
newsource:newsourcevol destination:destinationvol restart=always 0 * * *
*
```

7. Establish the new SnapMirror relationship by using the following command:

destination> snapmirror resync -S newsource:newsourcevol destination:destinationvol

Note: The SnapMirror relationship discards any Snapshot copies older than common_Snapshot: namely, the ones used for the last SnapMirror update. This is expected and no data is lost if you ensure that no new data was added to the original source volume during the move.

The new SnapMirror relationship automatically picks the newest Snapshot copy in common to mirror. This is the common_Snapshot.

8. Verify that the SnapMirror relationship is resynchronizing by using the following command:

```
destination> snapmirror status
```

Related references

Considerations for the use of SnapMirror on page 88 Prerequisites for SnapMirror on page 88

Moving qtree SnapMirror sources

With qtree SnapMirror you must create a Snapshot copy on the source and force its propagation to the destination and new source. This behavior is unlike volume SnapMirror, in which all of the Snapshot copies from the source are replicated to the destination.

Before you begin

Ensure that the Data ONTAP release on the destination system is from a release family that is the same, or later than, the Data ONTAP release on the new source system.

About this task

The process to move the qtree SnapMirror source is to create a Snapshot copy on the original source and then replicate the Snapshot copy on the destinations, both the new source and the existing destination. After this is done, the Snapshot copy is common on all volumes, allowing for the SnapMirror relationship to be broken from the original source and established between the new source and the existing destination.

The following terms are used in the task description for moving qtree SnapMirror sources.

- oldsource—The original system on which the source resides.
- newsource—The system to which you are moving the source.
- destination—The system to which the source is replicated.

- oldsourcevol—The original source volume.
- newsourcevol—The new source volume to which you are moving.
- destinationvol—The volume to which the source is replicated.

Steps

1. Copy the original source to the new source by using the following command.

newsource> snapmirror initialize -S oldsource:/vol/oldsourcevol/qtree
newsource:/vol/newsourcevol/qtree

Note: This might take some time to finish.

2. Make the old source volume read-only, by using the following command.

oldsource> vol restrict oldsourcevol

3. Create a manual Snapshot copy on the oldsource system by using the following command:

oldsource> snap create oldsourcevol common_Snapshot

4. Update the destinations by using the following commands.

newsource> snapmirror update -c common_Snapshot -s common_Snapshot -S
oldsource:/vol/oldsourcevol/qtree newsource:/vol/newsourcevol/qtree

destination> snapmirror update -c common_Snapshot -s common_Snapshot -s oldsource:/vol/oldsourcevol/qtree destination:/vol/destinationvol/qtree

The -s option of the snapmirror update command synchronizes newsource with oldsource and destination with oldsource based on common_Snapshot. The -c option of the snapmirror update command creates the common_Snapshot Snapshot copy on the destination systems.

5. Quiesce and break the SnapMirror relationship between oldsource and destination, and oldsource and newsource, using the following commands.

destination> snapmirror quiesce /vol/destinationvol/qtree destination> snapmirror break /vol/destinationvol/qtree newsource> snapmirror quiesce /vol/volnewsourcevol/qtree newsource> snapmirror break /vol/volnewsourcevol/qtree

6. Using an editor, update the /etc/snapmirror.conf file on the destination for the new relationship by replacing the oldsource information with newsource information.

Before edit

```
oldsource:/vol/oldsourcevol/qtree destination:/vol/destinationvol/qtree
restart=always 0 * * * *
```

After edit

```
newsource:/vol/newsourcevol/qtree destination:/vol/destinationvol/qtree
restart=always 0 * * * *
```

7. Establish the new SnapMirror relationship using the following command on the destination system:

snapmirror resync -S newsource:/vol/newsourcevol/qtree destination:/vol/
destinationvol/qtree

Note: SnapMirror discards any Snapshot copies older than the common Snapshot copy, namely, the ones used for the latest SnapMirror update. This is expected and no data is lost if you ensure that no new data was added to the original source volume during the move.

The new SnapMirror relationship automatically picks the newest common Snapshot copy for replication. This is the common Snapshot copy.

8. Verify that the SnapMirror relationship is resynchronizing by using the following command:

destination> snapmirror status

Related references

Considerations for the use of SnapMirror on page 88 Prerequisites for SnapMirror on page 88

Migrating data between volumes by using SnapMirror

SnapMirror can migrate data between volumes and redirect NFS clients to the new volume without rebooting the system or remounting to volume on NFS clients.

About this task

The migration must be run on two volumes which are currently the source volume and destination volume in a SnapMirror relationship. When you start the migration process, SnapMirror does the following:

- Performs a SnapMirror incremental transfer to the destination volume.
- Stops NFS and CIFS services on the entire system with the source volume.
- Migrates NFS file handles to the destination volume.
- Makes the source volume restricted.
- Makes the destination volume read-write.

SnapMirror does not transfer IP addresses, license keys, or quota information. You must remount on the NFS clients unless one of the following is true.

- The IP address of the source system is transferred to the destination system independently after the migration.
- The source and destination volumes reside on the same system, in which case, the IP address to access either volume is the same.

SnapMirror does not migrate CIFS clients. You must reestablish CIFS client sessions after migrating data to the destination volume.

Step

1. Enter the following command:

snapmirror migrate [src_system:]src_vol [dst_system:]dst_vol

src_system is the source system.

src_vol is the source volume.

dst_system is the destination system.

dst_vol is the destination volume.

Conversion of a destination to a writable volume or qtree

You can use the snapmirror break command to convert a SnapMirror destination, with read-only status, to a writable volume or qtree.

You might want to convert a destination to a writable volume or qtree to perform one of the following tasks.

- Data migration—Moving data from one volume or qtree (original source) to another volume or qtree (present destination) and make the data on the destination accessible and writable.
- Disaster recovery—If your source becomes unavailable, and you want your present destination to substitute as the users' retrieval and input source.
- Application testing—You want to make your current destination volume or qtree writable to test a new application on a mirrored replication of your current data rather than risk corruption of original data on the source volume or qtree.

Converting the destination to a writable volume or qtree enables you to use data on the destination, especially when the original source is unavailable.

Next topics

Quota restrictions on page 172 *Converting a SnapMirror destination to a writable volume or qtree* on page 173 *After using the snapmirror break command* on page 174

Quota restrictions

Quotas are always disabled on a SnapMirror volume destination, regardless of whether quotas are enabled on the source volume. If you try to enable quotas on a volume destination, SnapMirror displays an error message. Quotas are not disabled on SnapMirror destination qtrees.

If the source volume or qtree and the destination reside on different storage systems, and you want the same quota restrictions to be applied after you make the destination writable, the destination system must have an /etc/quotas file that includes all the entries from the /etc/quotas file used by the source system.

- If you use SnapMirror replication for data migration, you can copy the /etc/quotas entries from the source system to the /etc/quotas file of the destination system before you use the snapmirror break command to make the destination writable.
- If you use SnapMirror replication for backup and potential disaster recovery, you must keep a copy on the destination system of all /etc/quotas entries used by the source system at all times. That way, you can apply the quota entries to the destination volume or qtree if the source system becomes unavailable.

Converting a SnapMirror destination to a writable volume or qtree

You can convert a SnapMirror destination to a writable volume or qtree.

Steps

- 1. On the destination system, use the snapmirror break command to make the destination volume or qtree writable.
 - To make a destination volume writable, enter the following command on the destination system.

snapmirror break volume_name

• To make a destination qtree writable, enter the following commands on the destination system.

snapmirror quiesce /vol/volume_name/qtree_name

snapmirror break /vol/volume_name/qtree_name

- 2. If you want to enable quotas on the former destination volume, carry out the following steps.
 - **a.** Edit the /etc/quotas file on the former destination system so that, after the conversion, the former destination includes the same quota restrictions as the source volume.

If the original source volume uses per-volume quotas, replace the original source volume name with the former destination name in the quota entries.

b. Enter the following command to enable quotas on the former destination.

quota on volume_name

- 3. Consider the following optional measures.
 - If you want to stop a SnapMirror source from trying to update a broken-off destination, you can delete or comment out the entry in the /etc/snapmirror.conf file. Otherwise, SnapMirror continues to try to update the destination.
 - You might also want to use the options fs_size_fixed off command to turn off the option that restricts the size of the file system on a destination volume.

Note: If you set options fs_size_fixed off, the ability of the destination and source volumes to resync is not guaranteed.

After using the snapmirror break command

After using the snapmirror break command to temporarily break a SnapMirror relationship between a source and destination, you can use other SnapMirror commands to either make the break permanent, or restore or redefine the SnapMirror relationship.

- Use the snapmirror release command to make the break permanent.
- Use the snapmirror resync command to restore or redefine the SnapMirror relationship.

Related concepts

What the snapmirror resync command does on page 189 *How the snapmirror resync command helps minimize data loss* on page 192

Related tasks

Releasing partners from a SnapMirror relationship on page 181 *Resynchronizing a SnapMirror relationship* on page 190

Resizing a SnapMirror source and destination volume pair

You can increase the size of a volume SnapMirror source. In the next update, volume SnapMirror makes an identical increase to the size of the destination file system, if the destination volume is large enough to contain the larger file system.

About this task

Before increasing the size of a source volume, you must compare the size of the source and destination volumes for a specific volume SnapMirror relationship. If the destination volume is not large enough to contain the larger source volume, you must manually resize the destination volume.

Note: In an active volume SnapMirror relationship, the size of the source and destination file systems is identical. However, the size of the destination volume can be larger than or equal to the size of the source volume. The size of the file systems must be identical to allow the source to be restored from the destination.

Steps

1. On the source system, enter the following command to check whether the fs_size_fixed option is off.

vol status vol_name -v

vol_name is the name of the source volume for SnapMirror.

Example

```
SRC_A> vol status sm_src -v
Volume State
```

Status

```
sm src online
                       raid dp, flex
                                          nosnap=off, nosnapdir=off,
                                          minra=off, no_atime_update=off,
                                          nvfail=off,
                                          ignore_inconsistent=off,
                                          snapmirrored=off,
                                          create_ucode=off,
                                          convert_ucode=off,
                                          maxdirsize=31457,
                                          schedsnapname=ordinal,
                                          fs size fixed=off,
                                          compression=off,
                                          guarantee=volume,
                                          svo enable=off,
                                          svo_checksum=off
                                          svo_allow_rman=off,
                                          svo_reject_errors=off,
                                          no_i2p=off,
                                          fractional_reserve=100,
                                          extent=off,
                                          try_first=volume_grow,
                                          read_realloc=off,
                                          snapshot_clone_dependency=off
       Containing aggregate: 'aggr1'
       Plex /aggr1/plex0: online, normal, active
           RAID group /aggr1/plex0/rg0: normal
```

Note: If the fs_size_fixed option is set to off, the size of the destination volume's file system might be different from that of the source volume. To restore the source volume from the SnapMirror destination, the size of the source and destination file systems should be identical. When the status of a volume SnapMirror relationship is Idle, and the size of the destination volume is increased, then you can change the fs_size_fixed option to on.

2. Depending on the value of the fs_size_fixed option, choose one of the actions from the following table.

If the fs_size_fixed option is set to	Then
off	Go to the next step.
on	On the source system, enter the following command:
	vol options vol_name fs_size_fixed off
	vol_name is the name of the source volume for SnapMirror.
	Example
	SRC_A> vol options sm_src fs_size_fixed off

3. On the source system, enter the following command to find the size of the source volume and the size of the file system.

```
vol status vol_name -b
```

vol_name is the name of the SnapMirror source volume.

Example

SRC_A> vol status s Volume	n_src -b Block Size (bytes)	Vol Size (blocks)	FS Size (blocks)
sm_src	4096	256000	256000

4. On the destination system, enter the following command to find the size of the destination volume and the size of file system.

vol status vol_name -b

vol_name is the name of the SnapMirror destination volume.

Example

```
DST_B> vol status sm_dst -b
Volume Block Size (bytes) Vol Size (blocks) FS Size (blocks)
sm_dst 4096 512000 256000
```

5. Compare the size of the volume and the file system, for the source and destination pair of a SnapMirror relationship.

Note: If you want to increase the size of the source volume beyond the present size of the destination volume, you must manually increase the size of the destination volume.

Example

In the example shown in the previous steps, the destination volume size is greater than the source volume size.

6. Depending on the present size of the SnapMirror destination volume, choose one of the actions from the following table.

If the size of the destination volume	Then
Is enough to contain the intended increase in the size of the source volume	Go to step 8.
Needs to be increased to contain the intended increase in the size of the source volume	Go to the next step.

7. To increase the size of the SnapMirror destination volume, depending on the type of the volume, choose one of the actions from the following table.

FlexVol volume	vol size vol_name size
	vol_name is the name of the SnapMirror source volume.
	size is the required size of the SnapMirror source volume.
	Note: When you use this command for a SnapMirror destination FlexVol volume, the system displays the following message.
	Warning: Volume 'sm_dst' is a replica. This will not change the file system size in the replica, which is set from the replica source. The new volume size will be used to set the upper limit on replica transfers into this volume.
	vol size: Flexible volume 'sm_dst' size limit set to new_size.
Traditional volume	vol add vol name disks

If the volume is a... Then use this command to increase the size of the volume...

voi uuu voi_name uibhb

vol_name is the name of the SnapMirror source volume.

disks is the number of disks that you want to add to the traditional volume.

You can use the vol status command to confirm the increase in the source volume size.

Note: You can increase the destination volume size to be equal to or greater than the intended increased size of the source volume.

- 8. Perform step 7 for the source volume to increase the size of the SnapMirror source volume.
- 9. On the source system, enter the following command.

```
vol options vol_name fs_size_fixed on
```

vol_name is the name of the source volume for SnapMirror.

This option ensures that the size of the SnapMirror source and destination file systems are identical.

Note: SnapMirror updates the size of the destination file system to match the source in the next SnapMirror transfer for the volume pair.

Converting asynchronous SnapMirror replication to synchronous

You can change an asynchronous volume SnapMirror relationship to replicate data synchronously by editing the snapmirror.conf file on the destination system.

About this task

Synchronous replication is not supported for qtree SnapMirror relationships. Therefore, a qtree SnapMirror relationship cannot be converted to a synchronous SnapMirror relationship.

Step

1. To convert an asynchronous SnapMirror relationship to a synchronous SnapMirror relationship, on the administration host, edit the snapmirror.conf file on the destination system to change the schedule to sync.

Stabilizing destinations before a Snapshot copy

You might need to temporarily stop transfers to a destination, by using the snapmirror quiesce command. For example, if you want to create a Snapshot copy of a SnapMirror destination volume or qtree that contains a database, you need to ensure that its contents are stable during the Snapshot copy.

Step

1. Enter the following command on the system on which you want to block transfers.

snapmirror quiesce {dest_volume | /vol/volume_name/qtree_name}

dest_volume is the name of the destination volume.

gtree_name is the name of a qtree in volume_name.

Example

systemA> snapmirror quiesce vol1

snapmirror quiesce: in progress.
snapmirror quiesce: voll: successfully quiesced

SnapMirror stops any further data transfers to vol1.

Example

systemA> snapmirror quiesce vol2

snapmirror quiesce: in progress.

SnapMirror waits for a transfer to finish and stops any further data transfers to vol2.

Example

systemA> snapmirror quiesce /vol/vol1/qtree1

SnapMirror stops data transfers to qtree1 in vol1.

If you use the snapmirror break command on a destination that is quiesced, the quiesce condition is automatically cleared when the destination becomes writable.

Note: If you decide to abort a SnapMirror quiesce operation, press Ctrl-C or enter the snapmirror resume command at any time.

A SnapMirror destination volume might have been deleted after the volume was quiesced. If you want to create a SnapMirror destination volume with the same name as the deleted volume, first use the snapmirror release command. This step would ensure that the SnapMirror relationship is set up properly.

Next topics

What the quiesce command does on page 179 *Resuming transfers after quiescing a destination* on page 179

What the quiesce command does

The snapmirror quiesce command waits for all volume and qtree SnapMirror transfers to complete, and blocks any further updates. If a qtree is not in a stable state (is in transition), the snapmirror quiesce command forces it into a stable state.

You can quiesce only volumes and qtrees that are online and that are SnapMirror destinations. You cannot quiesce a restricted or offline volume or a qtree in a restricted or offline volume.

The snapmirror quiesce command stops a volume or qtree from acting as a SnapMirror destination, but does not prevent it from acting as a SnapMirror source.

Note: The quiesced state persists across reboots.

Resuming transfers after quiescing a destination

You can use the snapmirror resume command to restore the capability for data transfer to a volume or qtree you have quiesced.

Step

1. Enter the following command for the system on which you want to resume transfers.

snapmirror resume {dest_volume | /vol/vol_name/qtree_name}

dest_volume is the name of the destination volume.

gtree_name is the name of a qtree in vol_name.

Example

systemA> snapmirror resume vol2
snapmirror resume: vol2: Successfully resumed

SnapMirror resumes normal data transfer capability for vol2.

Aborting a SnapMirror transfer

You can use the snapmirror abort command to abort a volume or qtree replication operation before the transfer is complete. You can abort a scheduled update, a manual update, or an initial SnapMirror transfer.

About this task

You should consider the following issues before aborting a SnapMirror transfer.

- If you abort a copy operation, data transfer stops and SnapMirror is put in a restartable mode.
- If you abort a transfer that has been aborted before, you cannot restart that transfer again.
- If you use the -h (hard abort) option with the snapmirror abort command, you cannot restart the transfer.

Step

1. From either the source or the destination system, enter the following command:

```
snapmirror abort [-h] {[dest_system:]dest_volume | [dest_system:]/vol/
volume_name/qtree_name ...}
```

-h specifies a hard abort; the transfer cannot be restarted. SnapMirror stops the transfer and clears the restartable transfer log. This option applies only to the SnapMirror destination.

dest_system is the name of the destination system.

dest_volume is the destination volume.

/vol/volume_name/qtree_name is the path name of a destination qtree.

Note: If no destination system is specified, the local host's name is used for the system name. You can enter more than one destination volume.

You can obtain the destination system and volume from the snapmirror status output.

Note: If no destination volume or qtree is specified, the command returns an error message; it does not abort all transfers. To abort all transfers, use the snapmirror off command.
If you enter an invalid SnapMirror destination (one that is not displayed in the output of the snapmirror status command), the command fails and displays an error message.

Example

```
systemA> snapmirror abort vol1 systemB:vol2 systemC:/vol3/qtree3
```

```
snapmirror abort: Aborting transfer to vol1 systemB:vol2 systemC:/
vol3/qtree3
```

SnapMirror aborts the transfer to voll on systemA, where the command was entered, and aborts the transfer to vol2 on systemB and the transfer to qtree3 in vol3 on systemC.

Releasing partners from a SnapMirror relationship

To permanently end a SnapMirror relationship between a source and destination pair of volumes or qtrees, you need to use the snapmirror release command on the source and the snapmirror break command on the destination.

About this task

Releasing a source from a destination volume or qtree allows the source to delete its base Snapshot copy for the SnapMirror relationship. After breaking the relationship, you need to take additional steps to scrub the destination. Unless these extra steps are performed, the Snapshot copies associated with the broken relationship remain stored on the destination system, and a snapmirror status command continues to list the former destination object as a current destination object.

Steps

1. On the source system, enter the following command:

```
snapmirror release {source_volume | qtree_path}[dest_system:]
{dest_volume | qtree_path}
```

source_volume or *qtree_path* is the name of the source volume or path to the qtree that you want to release from the destination.

dest_system is the name of the system where the destination is located.

dest_volume or qtree_path is the name of the volume or path to the qtree that is the destination.

If you do not enter the name of the destination system, SnapMirror uses the name of the system on which you entered the command.

Example

For a SnapMirror volume relationship:

systemA> snapmirror release vol0 systemB:vol2

Example

For a SnapMirror qtree relationship:

systemA> snapmirror release vol/vol1/qtree2 systemB:/vol/vol2/qtree5

SnapMirror frees all resources on the source system that had been dedicated to the SnapMirror relationship.

2. On the destination system, enter the following command to break the SnapMirror relationship between the source and destination objects.

snapmirror break {vol_name | qtree_path}

vol_name is the name of the volume that you want to release from the relationship.

qtree_path is the path of the qtree that you want to release from the relationship.

- **3.** On the destination system, use the snapmirror status -1 command to determine which Snapshot copy basename is associated with the SnapMirror relationship that you just broke.
 - For a broken SnapMirror volume relationship:

snapmirror status -1 dest_vol

• For a broken SnapMirror qtree relationship:

snapmirror status -1 /vol/dest_vol/dest_qtree

In the detailed output that is displayed, note the Snapshot copy basename associated with the SnapMirror relationship that you just broke.

4. On the destination system, use the following command to delete the Snapshot copy set that you displayed in the previous step.

snap delete dest_vol snapshot_basename

5. Through the Adminhost client, edit the /etc/snapmirror.conf file on the destination system. Locate and delete the entry that specifies the SnapMirror relationship you want to end.

Related concepts

What the snapmirror.conf file does on page 135 *Conversion of a destination to a writable volume or gtree* on page 172

Related tasks

Editing the snapmirror.conf file on page 136 *Converting a SnapMirror destination to a writable volume or gtree* on page 173

Related references

Syntax for snapmirror.conf file entries on page 137

SnapMirror data transfer logs

You can use the options snapmirror.log.enable command to check SnapMirror data transfer logs. You can find out whether transfers are occurring as planned, how long the transfers take, and how well the system setup works. You find this information in the SnapMirror log file.

The SnapMirror log file provides the following information.

- The start time and the end time of the SnapMirror logging process.
- The start time, end time, and size of each transfer.
- Any abnormal termination and restart of a transfer.
- Other SnapMirror-related activities.

You can use the raw information provided to do the following.

- Calculate the average transfer size.
- Calculate the average transfer time.
- Look at the number of successful transfers and the failure rate.
- Tune the schedule.
- Create a notifier for aborted transfers.
- Monitor performance on a per-volume level.
- Be assured that things are working as planned.

Next topics

Checking for SnapMirror logging on page 183 Turning SnapMirror logging on on page 184 Format of SnapMirror log files on page 184 Turning SnapMirror logging off on page 186

Checking for SnapMirror logging

SnapMirror logging is on, by default. However, you can find out whether SnapMirror logging is on by using the snapmirror.log.enable option.

Step

1. Enter the following command on the system for which you want the information.

options snapmirror.log.enable

SnapMirror reports whether logging is enabled.

Example

```
systemA> options snapmirror.log.enable
snapmirror.log.enable on
```

Turning SnapMirror logging on

You can turn on SnapMirror logging by setting the snapmirror.log.enable option to on.

About this task

SnapMirror keeps the current log on the root volume of the system as /etc/log/snapmirror.0. A new log file is generated every week as /etc/log/snapmirror.0. Older log files are renamed / etc/log/snapmirror.[1-5] and the oldest log file is deleted. You can read the log files using a text editor.

Step

1. Enter the following command on the system for which you want the log.

options snapmirror.log.enable on

Note: This setting is persistent across reboots.

Result

SnapMirror enables the logging of transfers for the system.

Format of SnapMirror log files

Understanding the format of SnapMirror log files can help you better handle issues related to SnapMirror transfers.

The log file is in the following format. type timestamp source_system:source_path dest_system:dest_path event_info

type can be one of the following: src, dst, log, cmd. type specifies whether the record is for the source side (src) or destination side (dst) of the transfer. Certain events apply to only one side. The type log indicates a record about the logging system itself, for example, Start_Logging and End_Logging. The type cmd indicates a record of user commands, for example, Release_command and Resync_command.

timestamp is expressed in ctime format, for example:
Fri Jul 27 20:41:09 GMT.

event_info includes the following event names.
Request (IP address | transfer type) Start Restart (@ num KB) End (num KB
done) Abort (error_msg) Defer (reason) Rollback_start Rollback_end
Rollback_failed Start_Logging End_Logging Wait_tape New_tape Snapmirror_on

```
Snapmirror_off Quiesce_start Quiesce_end Quiesce_failed Resume_command
Break_command Release_command Abort_command Resync_command Migrate_command
```

The Request event on the source side includes the IP address of the system that made the transfer request; the Request event on the destination side includes the type of transfer. At the end of each successful transfer, the End event also reports the total size of the transfer in KB. Error messages are included with the Abort and Defer events.

Example

The following is an example of a log file from the source side.

log Fri Jul 27 20:00:01 GMT - - Start_Logging cmd Fri Jul 27 20:00:20 GMT - - Snapmirror_on src Fri Jul 27 20:41:09 GMT system1:vol1 system2:vol1 Request (10.56.17.133) src Fri Jul 27 20:41:32 GMT system1:voll system2:voll Abort (Destination not allowed) src Fri Jul 27 20:45:31 GMT systeml:vol0 systeml:vol1 Request (10.56.17.132) src Fri Jul 27 20:45:35 GMT system1:vol0 system1:vol1 Start src Fri Jul 27 20:51:40 GMT system1:vol0 system1:vol1 End (26200 KB) src Fri Jul 27 22:41:09 GMT system1:/vol/vol1/qtA system2:/vol/vol1/qtB Request (10.56.17.133) src Fri Jul 27 22:41:12 GMT system1:/vol/vol1/qtA system2:/vol/vol1/qtB Start src Fri Jul 27 22:41:13 GMT system1:/vol/vol1/qtA system2:/vol/vol1/qtB Abort (Nonunicode directory found in source qtree.) src Fri Jul 27 22:45:53 GMT system1:/vol/vol1/qtb system2:/vol/vol1/qsmb Request (10.56.17.133)src Fri Jul 27 22:45:56 GMT system1:/vol/vol1/qtb system2:/vol/vol1/qsmb Start src Fri Jul 27 22:45:59 GMT system1:/vol/vol1/qtb system2:/vol/vol1/qsmb End (3800 KB) cmd Fri Jul 27 22:50:29 GMT system1:/vol/vol1/qtb system2:/vol/vol1/qsmb Release_command

Example

The following is an example of a log file from the destination side.

```
dst Fri Jul 27 22:50:18 GMT systeml:vol0 systeml:vol1 Request (Initialization)
dst Fri Jul 27 22:50:20 GMT systeml:vol0 systeml:vol1 Abort (Destination is not
restricted)
dst Fri Jul 27 22:57:17 GMT systeml:/vol/vol1/qtA system2:/vol/vol1/qtB Request
(Initialize)
dst Fri Jul 27 22:57:24 GMT systeml:/vol/vol1/qtA system2:/vol/vol1/qtB Start
dst Fri Jul 27 22:57:36 GMT systeml:/vol/vol1/qtA system2:/vol/vol1/qtB End (55670 KB)
dst Fri Jul 27 23:10:03 GMT system1:/vol/vol1/qtA system2:/vol/vol1/qtB Request
(Scheduled)
dst Fri Jul 27 23:10:07 GMT system1:/vol/vol1/qtA system2:/vol/vol1/qtB Start
dst Fri Jul 27 23:10:07 GMT system1:/vol/vol1/qtA system2:/vol/vol1/qtB End (12900 KB)
cmd Sat Jul 28 00:05:29 GMT - system2:/vol/vol1/qtB Quiesce_start
cmd Sat Jul 28 00:05:29 GMT - system2:/vol/vol1/qtB Break_command
cmd Sat Jul 28 00:41:05 GMT system1:/vol/vol1/qtA system2:/vol/vol1/qtB Resync_command
log Sat Jul 28 00:41:10 GMT - End_Logging
```

Example

The following is an example of a log file from a retrieve (from tape) request.

```
dst Fri Jun 22 03:07:34 GMT filer_1:rst0l filer_1:bigtwo Request (retrieve)
dst Fri Jun 22 03:07:34 GMT filer_1:rst0l filer_1:bigtwo Start
dst Fri Jun 22 05:03:45 GMT filer_1:rst0l filer_1:bigtwo Wait_tape
dst Fri Jun 22 15:16:44 GMT filer_1:rst0l filer_1:bigtwo New_tape
dst Fri Jun 22 17:13:24 GMT filer_1:rst0l filer_1:bigtwo Wait_tape
```

```
dst Fri Jun 22 17:56:43 GMT filer_1:rst0l filer_1:bigtwo New_tape
dst Fri Jun 22 18:10:37 GMT filer_1:rst0l filer_1:bigtwo End (98602256 KB)
```

Turning SnapMirror logging off

You can turn off the SnapMirror log process by setting the snapmirror.log.enable option to off.

Step

1. Enter the following command on the system for which you want to disable SnapMirror logging.

```
options snapmirror.log.enable off
```

Listing SnapMirror Snapshot copies

You can use the snap list command to list all Snapshot copies, including the SnapMirror-specific Snapshot copies, that are stored on the system.

Step

1. In the console of either your source or destination system, enter the following command:

snap list vol_name

Result

A list of all Snapshot copies stored on your system is displayed. SnapMirror Snapshot copies are distinguished from system Snapshot copies by a more elaborate naming convention and the label "snapmirror" in parentheses.

Next topics

Naming conventions for Snapshot copies used by SnapMirror on page 186 Use of the snap list command to display SnapMirror updates on the destination volume on page 187

Naming conventions for Snapshot copies used by SnapMirror

When you run the snap list command, you can distinguish SnapMirror Snapshot copies from the regular system Snapshot copies by their naming conventions.

For volume replication, SnapMirror creates a Snapshot copy of the entire source volume that is copied to the destination volume.

A SnapMirror volume Snapshot copy name is in the following format.

```
dest_system(sysid)_name.number
```

- *dest_system* is the host name of the destination system.
- *sysid* is the destination system ID number.
- *name* is the name of the destination volume.
- *number* is the number of successful transfers for the Snapshot copy, starting at 1. Data ONTAP increments this number for each transfer.

Note: In the output of the snap list command, SnapMirror Snapshot copies are followed by the SnapMirror name in parentheses.

Volume example

```
systemA(0016791363)_vol0.9 (snapmirror)
```

For qtree replication, SnapMirror creates Snapshot copies of one or more source qtrees on the source volume that are copied to a qtree on the destination volume.

A qtree SnapMirror Snapshot copy name is in the following format.

dest_system(sysid)_name-src|dst.number

- *dest_system* is the host name of the destination system.
- *sysid* is the destination system ID number.
- *name* is the name of the destination volume or qtree path.
- *src*|*dst* is the source or destination name.
- *number* is an arbitrary start point number for the Snapshot copy. Data ONTAP increments this number for each transfer.

Qtree example

systemA(0016789302)_vol1_qtree3-dst.15 (snapmirror)

Attention: You should not delete manually-created Snapshot copies marked snapmirror in the output of the snap list command. Otherwise, later SnapMirror updates might fail.

Use of the snap list command to display SnapMirror updates on the destination volume

The snap list command displays information for each Snapshot copy on a storage system. Along with the name of the Snapshot copy, it displays when the Snapshot copy was created and the size of the Snapshot copy.

Example

The following example describes SnapMirror Snapshot copies that are created on a source volume and copied to a destination volume. In this example, data is copied from vol1 of systemA (the source) to vol2 of systemB (the destination).

To create a baseline version of a destination volume, systemA creates a Snapshot copy named systemB(0016782130)_vol2.1 on systemA. All Snapshot copies in vol1 of systemA, including systemB(0016782130)_vol2.1, are transferred to vol2 of systemB. When replicating a qtree, SnapMirror transfers only the qtree's data in the Snapshot copy for the qtree.

If the administrator runs the snap list command on the destination systemB after the systemB(0016782130)_vol2.1 Snapshot copy is transferred from systemA to systemB, a listing similar to the following example is generated.

When it is time to update the destination, another Snapshot copy is created on systemA.

The snap list command on systemA generates the following display after the systemB(0016782130)_vol2.2 Snapshot copy is created on systemA.

After the systemB(0016782130)_vol2.2 Snapshot copy is transferred from systemA to systemB, both Snapshot copies exist on systemB. On systemA, however, systemB(0016782130)_vol2.1 is no longer needed and is deleted; only systemB(0016782130)_vol2.2 is retained to be used for the next transfer.

You can see a list of each SnapMirror Snapshot copy on the server, and the qtrees it contains, and the client sources of those qtrees and their timestamps by using the snap list -q command.

You can use the snap list -o command to display the names, timestamps, and sources (if they are copies) of the qtrees in a specified volume or at a path name.

What SnapMirror restarts and retries are

In SnapMirror, a retry is an automatic attempt to start the transfer process after an interruption, whether or not any data was successfully transferred. A restart is the resumption of a previous transfer process from a restart checkpoint.

SnapMirror sets a restart checkpoint every 5 minutes during a transfer. SnapMirror restarts the previous transfer where it left off, if the following conditions are met:

- A restart checkpoint exists.
- All Snapshot copies being transferred still exist.
- The value for the restart mode in the snapmirror.conf file is set to always or is not set, and the next scheduled update has not arrived.

If the conditions are not met, SnapMirror creates a new Snapshot copy and starts a new transfer.

If a scheduled transfer fails (for example, due to network failure), SnapMirror automatically retries the transfer the next minute. If a transfer fails due to an error that renders it unfit for a retry (for example, if a user aborts the transfer), or if the source denied the transfer for any reason, the transfer is not retried the next minute. In such cases, an update is always attempted according to the schedule specified in the snapmirror.conf file.

Note: If a manual update fails, the update is not tried automatically and the user is informed. The user needs to reissue the command if an update is required.

After a reboot, SnapMirror does not automatically retry a transfer that was interrupted; however, the next scheduled or manual transfer restarts it at that restart checkpoint, if the checkpoint is still valid.

An initial transfer can be restarted but will not be retried automatically. To restart an initial transfer, enter the snapmirror initialize command again. Scheduled incremental updates automatically retry the transfer.

What the snapmirror resync command does

After the snapmirror break command, you can apply the snapmirror resync command to either the original SnapMirror destination or the original source.

- Applied to the original destination—the snapmirror resync command puts a volume or qtree back into a SnapMirror relationship and resynchronizes its contents with the source without repeating the initial transfer.
- Applied to the source volume—the snapmirror resync command can turn the source volume into a copy of the original destination volume. In this way, the roles of source and destination can be reversed.

Note: If SnapMirror cannot find a common Snapshot copy on the source and destination to use as the basis for resynchronization, resynchronization is not possible. SnapMirror generates an error

message that states the problem and terminates the command. You must reinitialize the destination to establish the SnapMirror relationship.

Next topics

Resynchronizing a SnapMirror relationship on page 190 How the snapmirror resync command helps minimize data loss on page 192 Resynchronization of Flex Vol volumes on page 193 Testing database applications: A special use of snapmirror resync on page 193 Retrieving data for disaster recovery: A special use of snapmirror resync on page 194

Resynchronizing a SnapMirror relationship

You can use the snapmirror resync command to restore or redefine a SnapMirror source or destination relationship that was broken with the snapmirror break command.

About this task

You might want to resynchronize a source and a destination volume or qtree under the following circumstances.

- You are changing the current source to a different volume or qtree.
- You make a destination volume writable for application testing and then want to make it a SnapMirror destination again.
- You need to recover from a disaster that disabled the source.
- You want to reverse the functions of the source and the destination.

Note: When you perform resync for the destination system, the contents on the destination are overwritten by the contents on the source.

Steps

1. From the destination system, enter the following command:

snapmirror resync [options] [dest_system:]{dest_volume | /vol/ qtree_path}

options can be any of the following.

- -n does not execute the resynchronization, but displays what would be done if the snapmirror resync command is run. You can use this option to find whether you have a Snapshot copy on the source and on the destination that can be used as the newest common Snapshot copy (base Snapshot copy) so that you can resync a specific SnapMirror relationship.
- -f forces the operation to proceed without prompting you for confirmation.
- -k *n* sets the maximum transfer speed to *n* kilobytes per second. This option has the same effect as the kbs argument in the /etc/snapmirror.conf file.

• -S [source_system:]{source_volume | qtree_path} specifies the system and volume or qtree you want to use as the source for resynchronization.

The source specified by the -S option must match a source entry in the /etc/ snapmirror.conf file. If entries exist but the source does not match, the operation displays an error message and terminates. If there is no entry for the specified source, the command runs.

Note: If the -S option is not set, the source must be specified in the /etc/ snapmirror.conf file. If it is not specified, the operation displays an error message and terminates.

dest_system is the name of the destination system.

dest_volume or /vol/qtree_path is the destination volume or qtree. If it is a scheduled destination as specified in the /etc/snapmirror.conf file, that source volume or qtree is considered to be the source. If the destination volume or qtree specified is not in a scheduled relationship, then the -S option must be used to provide a source.

SnapMirror identifies the newest common Snapshot copy, which is to be used as the base for resynchronization, and generates a list of Snapshot copies on the destination volume that meet the following criteria.

- The Snapshot copies are newer than the base Snapshot copy and are deleted.
- The Snapshot copies are older than the base Snapshot copy and have already been deleted from the source.

Note: For qtree resynchronization, only the common Snapshot copy is displayed.

SnapMirror then prompts you to choose whether to continue.

2. As required, choose one of the actions from the following table.

If	Then
You want to:	Type y at the prompt.
 Reestablish the SnapMirror pair Delete the listed Snapshot copies on the destination volume (if you are resynchronizing volumes) 	 Result: SnapMirror: Deletes the listed Snapshot copies on the destination volume (if you are resynchronizing volumes). Makes the destination read-only. Initiates an update of the destination.
You do not want to lose the data in a Snapshot copy that was created after the common Snapshot copy on the destination, but you want to resynchronize the two volumes or qtrees after the data is saved	 Type n at the prompt. Manually copy the data you want to save to the source or other volume. Return to Step 1 to rerun the snapmirror resync command.

If	Then	
You do not want to reestablish the SnapMirror relationship	Type n at the prompt.	
	Result: SnapMirror terminates the command.	

Example

SnapMirror resynchronization for volumes

systemB> snapmirror resync systemB:vol2
The resync base snapshot will be vol2(0001234567)_d.4
These newer snapshots will be deleted from the destination:
hourly.0
hourly.1
These older snapshots have already been deleted from the source and
will be deleted from the destination:
vol2(0001234567)_d.3
Are you sure you want to resync the volume?

Example

SnapMirror resynchronization for qtrees

```
systemB> snapmirror resync -S systemA:/vol/vol2/qtreeBob systemB:/vol/
vol3/qtreeBak
The resync base snapshot will be vol2(0001234567)_d.4
Data could be lost as a result of this operation.
Are you sure you want to resync the volume?
```

How the snapmirror resync command helps minimize data loss

The snapmirror resync command enables you to reestablish a broken SnapMirror relationship without a lengthy baseline transfer.

This command offers the choice of either source or destination to serve as the source in the restarted SnapMirror relationship. It finds the newest common Snapshot copy (NCS) shared by the two volumes or qtrees, and removes all newer information on the system on which the command is run.

Note: The snapmirror resync command requires that the two volumes or qtrees have at least one Snapshot copy in common. You can resynchronize a volume or qtree to any other volume or qtree as long as both have at least one Snapshot copy in common.

Resynchronization causes the loss of all data written to the destination after the base Snapshot copy was made. The snapmirror resync command informs you what data might be lost during the resynchronization and requests permission to proceed. If you want to save the data on the destination, you can stop the resynchronization, manually copy the desired data elsewhere, and reissue the snapmirror resync command to the destination.

Resynchronization of FlexVol volumes

If there are any changes on the destination system that you need to replicate to the source, you can use qtree SnapMirror to resynchronize data.

You can resynchronize the data at the source with any changes made at the destination by using the snapmirror resync command. When resynchronizing data, the Data ONTAP version restrictions apply in case of volume SnapMirror.

Related concepts

Comparison between volume SnapMirror and qtree SnapMirror on page 97

Testing database applications: A special use of snapmirror resync

Testing software applications that run on a database can sometimes change or corrupt the database. To ensure that you do not lose data while testing such applications, you can copy the data to another volume for testing purposes, break the SnapMirror relationship and return the destination volume to writable state, and run the test application on it. Upon completion of the test, you can resynchronize the source and the destination volume.

Before you begin

Ensure that SnapMirror is enabled on the source and destination systems.

About this task

In the following procedure, you can use a combination of the snapmirror break and snapmirror resync commands to perform the following tasks:

- Make a destination volume writable for testing.
- Restore the newly writable volume to its original state if further testing is required.

Steps

- 1. Create or choose a volume or qtree to be used as a destination for the volume or qtree containing the database. (This example uses a volume called Test_vol.)
- 2. On the destination system, enter the following command to make the destination writable.

snapmirror break Test_vol

For a qtree, the path must be specified as shown in the following example. You must ensure that the qtree is quiesced before breaking the relationship.

dst> snapmirror quiesce /vol/dst_vol/Testqtree dst> snapmirror break /vol/dst_vol/Testqtree

- 3. Run the application on the data in the former destination (Test_vol).
- 4. Check the data in the former destination (Test_vol).

If	Then
The data has been altered in some way that is not useful and you want to import a fresh copy of the data for further testing.	From the destination system, enter the following command:
	snapmirror resync Test_vol
	Note: For a qtree, the path must be specified as shown in the following example:
	<pre>src> snapmirror resync dst_system:/vol/dst_vol/ Testqtree /vol/src_vol/Testqtree</pre>
	SnapMirror makes the former destination volume into a SnapMirror destination again and updates the destination with the latest data.
The data has not been altered adversely, or you wish to stop testing.	The task is completed.

5. As required, choose one of the actions from the following table.

6. Repeat steps 3, 4, and 5, until you are satisfied with the testing.

Retrieving data for disaster recovery: A special use of snapmirror resync

When disaster disables the source of a SnapMirror relationship, you can use the snapmirror resync command as part of a strategy to update the repaired source and reestablish the original configuration of the systems.

About this task

In the following example, the original source (the one disabled by the disaster) is systemA:vol/ volA and the original destination is systemB:/vol/volB. You use a combination of snapmirror break and snapmirror resync or snapmirror initialize commands to perform the following tasks:

- Temporarily make systemB:volB the source and systemA:volA the destination to restore mirrored data back to systemA:volA and to update systemA:volA.
- Restore systemA:/vol/volA and systemB:volB to their original roles as SnapMirror source and SnapMirror destination volume.

In this example, all data from the last scheduled SnapMirror Snapshot copy before the source was disabled and all the data written to systemB:vol/volB after it was made writable is preserved. Any data written to systemA:vol/volA between the last SnapMirror Snapshot copy and the time that systemA:vol/volA was disabled is not preserved.

Steps

1. After the source volume (in this case, systemA:volA) is disabled, use the snapmirror break command to make the destination volume, systemB:volB, writable.

snapmirror break systemB:volB

2. Redirect the clients of source systemA to source systemB.

The former clients of systemA are now accessing and writing to systemB.

- 3. Temporarily make the original source volume a read-only destination volume.
 - If systemA:volA is recoverable, and its data is intact, then use the snapmirror resync command on systemA to resynchronize systemA with systemB.

```
snapmirror resync -S systemB:VolB systemA:volA
```

• If systemA:volA is unrecoverable, make a new volA on systemA, and from systemA, initialize systemA:volA from systemB.

```
snapmirror initialize -S systemB:volB systemA:volA
```

This command also makes systemA:volA a read-only destination.

Note: These commands need to be performed on the original source system.

4. Redirect the clients from systemB to systemA.

The clients cannot access or write to systemA:volA, but they are no longer writing new data to systemB:volB.

5. Update systemA:volA from systemB to transfer the latest data from systemB.

Example

Perform the following step from systemA.

snapmirror update -S systemB:volB systemA:volA

6. Use the snapmirror break command to make systemA:volA writable. On systemA, enter the following command:

snapmirror break volA

7. On systemB, use the snapmirror resync command to make systemB, the original destination, the destination again.

snapmirror resync volB

Operation of SnapMirror with other features and products

Next topics

Comparison between SnapMirror and the vol copy command on page 196 Comparison between qtree SnapMirror and SnapVault on page 196 Transfer of LUN clones using qtree SnapMirror on page 197 Management of SnapMirror operations through Protection Manager on page 198 Use of SnapMirror with SnapDrive on page 198 SnapMirror and MultiStore on page 198 How FlexClone volumes impact SnapMirror on page 199 Guidelines for creating a clone of a qtree SnapMirror destination volume on page 200 SnapVault destination to tape backup on page 200 How SnapMirror works with the dump command on page 203 Protection of SnapVault secondaries using volume SnapMirror on page 204 Use of SnapMirror with S Family storage systems on page 206 SnapMirror and ACLs on page 207

Comparison between SnapMirror and the vol copy command

You can use SnapMirror or the vol copy command to copy volumes from the source to the destination. There are some similarities between them. However, there are significant differences too.

The following points list the similarities between SnapMirror and the vol copy command.

- Both enable you to copy Snapshot copies from a source to a destination volume.
- The source and destination volumes should both be either traditional volumes or FlexVol volumes.

Note: You can use qtree SnapMirror to replicate data between traditional volumes and FlexVol volumes.

• The volumes should be of the same type.

These commands differ in several important ways, as listed in the following table.

SnapMirror	The vol copy command
Supports scheduled, incremental updates of Snapshot copies, replicated from the source to the destination volumes	Does not support incremental updates
Supports qtree-level replication between the source and destination systems	Supports only volume replication and not qtree-level replication
Requires an appropriate license	Does not require an additional license

Comparison between qtree SnapMirror and SnapVault

There are several similarities between qtree SnapMirror and SnapVault. However, there are differences in how they are used.

The following table compares qtree SnapMirror with SnapVault.

Qtree SnapMirror	SnapVault
More suitable for providing immediate failover capability.	More suitable where data availability is less critical, and immediate failover is not required.
Uses the same software and licensing on the source and destination systems.	Uses SnapVault source system and SnapVault destination system, which provide different functionality.
Transfers can be scheduled at a maximum rate of once every minute.	Transfers can be scheduled at a maximum rate of once every hour.
Every qtree within a source volume uses one Snapshot copy each on the source system.	Only one Snapshot copy is used.
Snapshot copies are deleted by qtree SnapMirror when not required for replication.	Snapshot copies are retained and deleted on a specified schedule.
Relationships can be reversed. This allows the source to be resynchronized with changes made at the destination.	Relationships cannot be reversed. It provides the capability to transfer data from the destination to the source only to restore data. The direction of replication cannot be reversed.
Can be used to replicate data between systems running Data ONTAP only.	Can be used to back up both NetApp systems and open systems. However, the destination system should be a NetApp system.

Transfer of LUN clones using qtree SnapMirror

In versions earlier than Data ONTAP 7.3, SnapMirror considers each LUN clone as a new LUN. Therefore, during the initial transfer of the LUN clone, all data from the LUN clone and the original Data ONTAP LUN is transferred to the secondary system.

For descriptions of data backup and restore on volumes containing Data ONTAP LUNs, see the *Data* ONTAP Block Access Management Guide for iSCSI and FC.

The transfer of LUN clones using SnapMirror works the same way as the transfer of LUN clones using SnapVault in the non-optimized mode.

Note: Qtree SnapMirror transfers LUN clones in the non-optimized mode only. Qtree SnapMirror does not have the option of optimized transfers.

The use of snapmirror resync for restoring data to a source qtree with LUN clones is not supported. When you replicate qtrees with LUN clones, each LUN clone within the qtree is stored as a LUN within the destination qtree. Therefore, when you recover data from such a destination qtree, the original LUN clones are restored as complete LUNs.

Attention: If you attempt to recover data from the destination to a source qtree with LUN clones, using a snapmirror resync operation, the system displays the following error message.

```
cannot resync as qtree has one or more lun clones Aborting resync.
```

To recover data for a qtree with LUN clones, you can replicate the destination qtree to a new qtree.

Attention: For a qtree with LUN clones, ensure that the volume has enough free space to store the LUN clones as complete LUNs before you initiate data recovery using qtree SnapMirror.

Related concepts

About LUN clones and Snap Vault on page 262

Management of SnapMirror operations through Protection Manager

You can use the Protection Manager graphical user interface to perform some of the management tasks in a SnapMirror environment.

You can perform the following tasks by using Protection Manager.

- · Create and manage asynchronous and synchronous SnapMirror relationships.
- Create and manage policies for replication and failover.
- Report on relationships and lag times.
- Configure alerts about replication state changes.
- Schedule replica updates.
- Visualize relationships.
- Simplify data services recovery after a failure.

For more information, see the Protection Manager Administration Guide.

Use of SnapMirror with SnapDrive

If you are using SnapDrive software, you can use SnapMirror to replicate your data.

For more information, see the *SnapDrive Installation and Administration Guide* for your version of SnapDrive.

SnapDrive supports the use of volume SnapMirror in the asychronous mode. For volume SnapMirror in the sychronous or semi-synchronous mode, you need to use SnapDrive 5.0.

SnapDrive does not support qtree SnapMirror replication.

SnapMirror and MultiStore

If you are using MultiStore software, you can use SnapMirror to replicate data in the volumes of a vFiler unit, for disaster recovery. You can also use SnapMirror to migrate a vFiler unit from one system to another.

When performing online migration of a vFiler unit, any SnapMirror operations that start a transfer do not work for volumes owned by the vFiler unit. Therefore, any manual or scheduled SnapMirror

update operations for the vFiler unit will not work. SnapMirror commands that are not allowed for use during an online migration of a vFiler unit are given in the following list.

- snapmirror initialize
- snapmirror update
- snapmirror resync

For more information about vFiler units, see the Data ONTAP MultiStore Management Guide.

How FlexClone volumes impact SnapMirror

You can create FlexClone volumes from SnapMirror source or destination volumes. However, you should understand the behavior of the resulting FlexClone volume before creating it.

FlexClone volumes create a nearly instantaneous replica of a volume within the same aggregate. For information about FlexClone volumes, see the *Data ONTAP Storage Management Guide*.

Cloning SnapMirror volumes: You can clone SnapMirror volumes. A clone of a SnapMirror destination locks the Snapshot copy from which the clone was created. It also locks that Snapshot copy in the source volume and every volume in the cascade (if the volume is part of a SnapMirror cascade).

There are differences between cloning volumes that use qtree SnapMirror and cloning volumes that use volume SnapMirror. When you create a clone of a volume that is a destination for qtree SnapMirror replication, without specifying the Snapshot copy name, the clone is not writable until the SnapMirror relationship of the clone volume is broken. Specifying a Snapshot copy name when creating a clone breaks the SnapMirror relationship of the clone volume and makes the volume writable. The clone of a volume SnapMirror destination volume does not have this behavior.

Note: A FlexClone volume can be created from a Snapshot copy in a SnapMirror destination, but a FlexClone volume cannot be the destination of a SnapMirror relationship.

Cloning from a Snapshot copy that is not the most recent Snapshot copy: If a FlexClone volume is created from a Snapshot copy that is not the most recent Snapshot copy, and that Snapshot copy no longer exists on the source volume, then every update will need to delete the Snapshot copy on the destination. In this case, all SnapMirror updates to the destination volume fail until the clone is destroyed or split.

Note: This will not occur if the clone is created from the most recent Snapshot copy in the SnapMirror destination because that Snapshot copy is guaranteed to exist in the source volume.

Effects of deleting the source Snapshot copy: If a source Snapshot copy for a SnapMirror relationship is deleted, SnapMirror fails on subsequent update requests. The failures occur because FlexClone volumes are bound to the source Snapshot copy regardless of whether the FlexClone volume was created on the source or destination volume. Even if the FlexClone volume was created remotely on the destination system, deleting the source Snapshot copy on the source system results in a failed SnapMirror update when attempting to propagate the deletion to the destination. You can delete or split the FlexClone volume for the SnapMirror updates to continue.

Guidelines for creating a clone of a qtree SnapMirror destination volume

When using the vol clone create command to create a clone of a qtree SnapMirror destination volume, you need to consider the status of any qtree SnapMirror transfers for that volume.

You can check the status of SnapMirror transfers by using the snapmirror status -1 command.

The following table describes the different scenarios possible, when creating a clone of a qtree SnapMirror destination volume.

If the status of qtree SnapMirror transfers for a specific volume is	Then, for the creation of a clone of the qtree SnapMirror destination
Idle	The clone will be writable.
	Note: It is recommended that you create a clone of a qtree SnapMirror destination, when the status is idle.
Active	You can specify a Snapshot copy that is older than any of the Snapshot copies being used for transfers. In this case, the clone will be writable.
	If you do not specify a Snapshot copy for the creation, the clone will be read-only. For a read-only clone created when a SnapMirror transfer is active for the volume, the SnapMirror relationship is active for the clone. Therefore, for the clone, you need to quiesce the SnapMirror transfer and break the SnapMirror relationship. This would make the clone writable.
	Note: The original SnapMirror transfer or relationship need not be modified. You only need to change the status of the clone to make it writable.

Note: If you use the snapmirror quiesce or snapmirror break command on a writable clone, then the system displays an error message and proceeds without effect.

For more information about creating volume clones, see the *Data ONTAP Storage Management Guide*.

SnapVault destination to tape backup

You can use the snapmirror store command to replicate Snapshot copies from a SnapVault secondary system to an attached local tape drive for the purposes of backup. In case of data loss on the SnapVault secondary system, you can restore a specified Snapshot copy back to the SnapVault secondary system.

Note: This scenario applies only to backup of SnapVault secondary storage. Do not apply this configuration for any other purpose.

You need a SnapVault secondary system with an attached local tape drive. The volume on which the secondary storage qtrees reside must also be configured as a SnapMirror source volume.

You must follow this sequence of activities to set up this arrangement.

- 1. On the SnapVault secondary/SnapMirror source system, use the snapmirror store command to copy the volume to tape. Use the snapmirror use command to continue the copy if more than one backup tape is necessary.
- 2. Use the snapmirror update command to manually mirror incremental updates to tape.
- **3.** In the event of data loss on the SnapVault secondary system, convert the volume that holds the SnapVault data from its role as a SnapMirror source volume to a destination volume, mirror the tape contents back, and then convert the restored destination volume back to a regular volume again.

Next topics

Copying from a SnapVault volume to a local tape on page 201 *Restoring to SnapVault from a local tape* on page 202

Copying from a SnapVault volume to a local tape

You can use SnapMirror to back up a SnapVault volume to tape. The backup on the tape can later be used to restore data to the SnapVault volume.

Steps

- 1. At the SnapVault secondary system, load the tape into a local attached tape device.
- **2.** At the SnapVault secondary system, start the data transfer to tape by entering the following command:

snapmirror store sv_volume dest_tapedevices

sv_volume is the volume you are copying.

dest_tapedevices is a comma-separated list of tape devices to which you are copying the volume.

SnapMirror starts transferring data to tape. This command creates a baseline transfer. If you have to use more than one tape, SnapMirror prompts you for another tape.

3. If SnapMirror prompts you for another tape, add another tape to the drive, and continue transfer of data to tape by entering the following command:

snapmirror use dest_tapedevices tape_drive

dest_tapedevices is a comma-separated list of tape devices to which you are copying the volume.

tape_drive is the drive holding the new tape you supplied.

Note: The snapmirror use command does not support copying qtrees from tape. If you specify a qtree path as the source or destination, SnapMirror returns an error message.

4. Repeat the previous step until SnapMirror finishes copying the volume to tape.

If the volume on the SnapVault secondary system that you backed up ever needs to be restored, use the backup on the tape to restore the volume.

Related tasks

Restoring to Snap Vault from a local tape on page 202

Restoring to SnapVault from a local tape

In the event of data loss on a SnapVault secondary storage volume that is also configured as a SnapMirror source, data that has been replicated to tape using SnapMirror can be restored to the SnapVault secondary storage volume with the snapmirror retrieve command. The snapmirror retrieve command restores only full volumes. If you specify a qtree path as the source or destination, SnapMirror returns an error message.

Steps

- 1. On the SnapVault secondary system, use the vol restrict command to put the volume that you want to restore from tape into restricted state. See the *System Administration Storage Management Guide* for information about how to restrict a volume.
- 2. Load the tape (made with the snapmirror store command) into the local tape device.
- 3. Start the initialization by entering the following command on the SnapVault secondary system.

snapmirror retrieve [-h] rest_volume tape_drive

The -h option displays the headers of the source tapes but does not transfer data.

rest_volume is the volume that you are restoring.

tape_drive is a tape device or a comma-separated list of devices from which you are restoring the volume.

SnapMirror starts transferring data from the tape. If data is stored on more than one tape, SnapMirror prompts you for the next tape.

4. If Data ONTAP prompts you for another tape, add the next tape to the drive and restart the initialization by entering the following command:

snapmirror use rest_volume tape_list

rest_volume is the volume to which you are restoring.

tape_list is the tape device from which you are restoring the volume.

- 5. Repeat the previous step until SnapMirror finishes initializing the volume from the tape.
- 6. If incremental updates to the baseline Snapshot data that you just restored exist on tape, insert that tape into the local tape drive of the SnapVault secondary system, and enter the following command:

snapmirror update [options] [sv_secondary:]sv_volume

options can be one or more of the following:

- -k *n* sets the maximum transfer speed to *n* kilobytes per second. This option has the same effect as the kbs argument in the /etc/snapmirror.conf file.
- -S [source_system:]source_volume specifies the source system and volume for the migration. source_volume is the volume you want to copy.

sv_secondary is the name of the SnapVault secondary system that you want to restore to.

sv_volume is the volume that you want to restore.

7. After all tape-to-SnapVault retrieve and update operations are complete, use the snapmirror break command to return the restored volume to a writable state.

snapmirror break /vol/volume_name/

Related concepts

Manual update of a SnapMirror destination on page 146

How SnapMirror works with the dump command

You can use the dump command to back up data from a SnapMirror destination volume. The dump command picks the most recent Snapshot copy and copies that to tape.

Note: The use of the dump command to copy data from a synchronous SnapMirror destination volume is not supported.

You can back up any Snapshot copy displayed by the snap list command on the destination. You can also create a Snapshot copy on the source volume, copy the Snapshot copy to the destination, and use the dump command to back up this Snapshot copy from the destination to tape.

Effect of the dump command on the SnapMirror destination update schedule

Running the dump command on a SnapMirror destination affects SnapMirror operations on that destination in several ways.

• Scheduled incremental SnapMirror updates of a destination volume can occur concurrently with a dump command operation to tape; however, if a scheduled SnapMirror update to the destination volume involves the deletion of a Snapshot copy that the dump command operation is currently writing to tape, the SnapMirror update will be delayed until the dump command operation is complete.

Note: SnapMirror updates of a destination qtree are not affected by dump command operations under any circumstances.

• The operation of snapmirror break, snapmirror resync, and snapmirror migrate commands cannot be carried out concurrently with the operation of the dump command.

Protection of SnapVault secondaries using volume SnapMirror

Volume SnapMirror protects SnapVault secondaries by creating SnapMirror relationships to migrate data from the volumes on the SnapVault secondary system to volumes on a remote (tertiary) system running Data ONTAP. SnapMirror provides an exact replica of the SnapVault secondary data on the tertiary system.

The advantage of protecting SnapVault secondaries using volume SnapMirror is that soft lock support enables you to continue SnapVault relationships between the original SnapVault primary system and the tertiary system, without initial baseline transfers.

Note: To deploy this solution, you should ensure the use of the appropriate licenses for using SnapMirror and SnapVault. You cannot execute SnapVault commands on the new secondary system (which is also a SnapMirror destination) if you do not license sv_ontap_sec on this tertiary system.

For example, if your SnapVault secondary system becomes unusable because of a disaster, you can manually redirect your next SnapVault transfers to the tertiary system instead of the unusable secondary system. Your tertiary system becomes the new SnapVault secondary system, and your SnapVault transfers continue, using the latest Snapshot copy common to both the primary and the tertiary systems.

Migrating SnapVault data using volume SnapMirror

You can migrate a volume that contains SnapVault destination qtrees from one secondary system to a tertiary system without having to perform a baseline transfer. Ensure that you have Open Systems SnapVault baselines. For example, in the following procedure, consider a baseline of the bno:C:\500MB directory was backed up to system-old:/vol/ old_vol/bno_C_500MB.

Steps

1. Using SnapMirror, replicate the volume from the present secondary system to a volume on the new secondary system.

Example

To replicate the old_vol volume from the system-old secondary system to the new_vol volume on the system-new secondary system, complete the following steps on the new secondary system (system-new).

a. Create the new_vol volume.

system-new> vol create new_vol 3

b. Mark the new_vol volume restricted.

```
system-new> vol restrict new_vol
```

c. Transfer the old_vol volume to the new_vol volume.

```
system-new> snapmirror initialize -S system-old:old_vol new_vol
```

2. Quiesce and break the SnapMirror relationship between the old secondary system and the new secondary system.

Example

To quiesce and break the SnapMirror relationship between system-old and system-new, complete the following steps on system-new.

```
a. snapmirror quiesce new_vol
```

```
b. snapmirror break new_vol
```

3. Check the SnapMirror status and SnapVault status on the new secondary system. SnapMirror status should be Broken-off. SnapVault status should be Snapvaulted to the new volume on the new secondary system.

Example

Perform the following steps from system-new.

```
a. snapmirror status
```

Source Destination State system-old:old_vol system-new:new_vol Broken-off

 $\boldsymbol{b}.$ snapvault status

Source	Destination	State
bno:C:\500MB	<pre>system-new:/vol/new_vol/bno_C</pre>	_500MB Snapvaulted

4. Confirm that SnapVault configuration information is not present on the new secondary system by using the snapvault status -c command.

Example

Perform the following step from system-new.

snapvault status -c

Snapvault secondary is ON.

5. Enable access to the new SnapVault secondary system from the SnapVault primary system using the options snapvault.access command.

Example

Perform the following step from system-new.

options snapvault.access host=system-old

Note: When using SnapVault, access needs to be specified on both the primary and secondary systems.

6. Add SnapVault configuration information to the registry on the new secondary system using the snapvault start command.

Note: This does not start a new baseline, it updates the registry.

206 | Data ONTAP 7.3 Data Protection Online Backup and Recovery Guide

Example

Perform the following step from system-new.

snapvault start -S bno:C:\500MB system-new:/vol/new_vol/bno_C_500MB

```
SnapVault configuration for the qtree has been set.
Qtree /vol/new_vol/bno_C_500MB is already a replica.
```

7. Confirm that SnapVault configuration information is present on the new secondary system using the snapvault status -c command.

Example

Perform the following step from system-new.

snapvault status -c

SnapVault secondary is ON.
/vol/new_vol/bno_C_500MB source=bno:C:\500MB

8. Test the new SnapVault relationship by manually updating system-new.

If you are using the CLI to manage your environment, continue to the next step; otherwise, you have completed the task.

Example

Perform the following step from system-new.

```
snapvault update system-new:/vol/new_vol/bno_C_500MB
```

Transfer started. Monitor progress with 'snapvault status' or the snapmirror log.

9. Re-create any schedules used on the old secondary system to the new secondary system and ensure access permissions are in place.

Use of SnapMirror with S Family storage systems

You can use SnapMirror S Family Edition to replicate data between S Family storage systems.

S Family storage systems only support the following versions of Data ONTAP:

- Data ONTAP 7.2.1 S4
- Data ONTAP 7.2.1 S8
- Data ONTAP 7.2.1 S9

SnapMirror S Family Edition works only for asynchronous replication. When using S Family storage systems as source and destination, you can replicate data in the following ways:

- One to another
- One to many
- · Many to one

Note: When using SnapMirror S Family Edition for replicating volumes or qtrees, the maximum number of concurrent replication operations is four.

You can use the snapmirror resync command to restore data to an S Family storage system.

SnapMirror S Family Edition also supports replication from an S Family storage system to a FAS series storage system.

Note: In this replication relationship, the FAS series storage system is the destination. Therefore, the FAS series storage system should be using Data ONTAP 7.2.1 or later.

SnapMirror S Family Edition does not support replication from a FAS series storage system to an S Family storage system.

Related references

Maximum number of concurrent replication operations on page 128

SnapMirror and ACLs

SnapMirror and SnapVault support the replication of NFS version 4 access control lists (ACLs).

When replicating, the destination might not understand NFS version 4 ACLs.

- If the destination can understand NFS version 4 ACLs, the ACL is replicated as is from the source to the destination. The access rules applied for the data on the destination is identical to the access rules on the source.
- If the destination cannot understand NFS version 4 ACLs, a new ACL, which the destination can understand, is created. The access rules in this new ACL is equivalent or stricter than the original ACL. This is done to avoid any security issues.

Note: This alternative also implies that a user might not be given access to a set of data on the destination, although the user has access to the same data on the source.

SnapMirror over Fibre Channel

SnapMirror over Fibre Channel enables you to use the SnapMirror feature over Fibre Channel in a SAN environment.

SnapMirror over Fibre Channel includes all the features that are available with SnapMirror over Ethernet. The operational concepts and the command interfaces are identical for both. However, there are a few differences between them.

Next topics

Hardware requirements for SnapMirror over Fibre Channel on page 208 Supported Fibre Channel switches on page 208 SnapMirror over Fibre Channel topology on page 209 SnapMirror traffic zones on page 210 Requirements for deploying SnapMirror over Fibre Channel on page 211 Functionality supported by SnapMirror over Fibre Channel on page 212 Configuring SnapMirror over Fibre Channel on page 212 Troubleshooting issues related to SnapMirror over Fibre Channel on page 219

Hardware requirements for SnapMirror over Fibre Channel

You must install X1024 or X1124 Fibre Channel (FC) adapters on the system for using SnapMirror over Fibre Channel.

An X1024 adapter has two 2-GB FC ports and an X1124 adapter has two 4-GB FC ports. In a system, you can only use either an X1024 adapter or an X1124 adapter at a time. You can use either of the two cards on the source and destination systems.

For more information about the hardware supported with different storage system models, see the *System Configuration Guide*.

When using SnapMirror over Fibre Channel, these systems run in the switch fabric topology logged in to the FC switch as an F-Port. Each port discovers other FCVI-capable ports by querying the switch name server and then logs in to these ports as necessary. Each port is identified by its unique worldwide name (WWN).

The adapter operates as a standard network interface. Each port must be configured with its own IP address and network mask, just as an Ethernet adapter is configured. For an active/active configuration system, the administrator must also configure its partner IP address.

Note: Only the X1024 and X1124 support Fibre Channel Virtual Interface (FCVI) Architectural Mapping. FCVI functionality is not supported by other FC adapters or FC ports in the systems.

Related tasks

Configuring SnapMirror over Fibre Channel on page 212

Supported Fibre Channel switches

SnapMirror over Fibre Channel works with Fibre Channel switches from the following vendors.

- Brocade
- Cisco

You must have a homogeneous Fibre Channel SAN environment. To comply with SnapMirror over Fibre Channel certification, the switches in the SnapMirror data path must be from a single vendor.

SnapMirror over Fibre Channel topology

SnapMirror over Fibre Channel is normally installed in a SAN-based data center using Fibre Channel as its network transport protocol.

SnapMirror traffic between source and destination storage systems travels through the Fibre Channel network. The adapters installed in the storage systems translate the SnapMirror IP packets to and from Fibre Channel frames.

Topology choices

SnapMirror over Fibre Channel requires at least one Fibre Channel switch in the data path.

More complex topologies can involve multiple paths and switches between source and destination storage systems, as shown in the following illustration. The storage systems in each building are arranged in an active/active configuration. An active/active configuration removes single points of failure and provides a more practical, fault-tolerant setup.



Figure 13: SnapMirror deployment using multiple paths and switches

Multipath support

SnapMirror over Fibre Channel supports multipath configurations, just as SnapMirror over Ethernet. The configurations are identical.

Private subnet considerations

The storage system needs to direct Fibre Channel Virtual Interface (FCVI) traffic to the Fibre Channel NIC adapter. Therefore, you should configure the IP address of a Fibre Channel NIC port within the range of the private subnet, one that is not globally routable.

The private subnets include:

• 10/8

- 172.16/12
- 192.168/16

Multiple SAN islands support

You might partition your SAN into multiple SAN islands. SnapMirror over Fibre Channel currently supports the Cisco VSAN feature.

Note: The Brocade Multiprotocol Router feature is not supported in the SnapMirror over Fibre Channel data path.

Extended fabric considerations

A typical SnapMirror over Fibre Channel setup involves an extended fabric configuration to mirror data on the primary storage system to a remote site.

The remote site might be quite far from the primary site. To achieve proper operation and expected throughput of the SnapMirror data transfer, you need to follow these guidelines.

- Verify that proper equipment is used in the data path, including:
 - Long-distance, small form-factor pluggable (SFP) optical transceiver modules
 - Long-distance fiber optic cables and repeaters
- Verify that the Fibre Channel switch is extended-fabric capable and has the appropriate license for the capability.
- Configure the inter-switch links (ISLs) to be extended-fabric ports.

For information about how to check the extended license and configure a switch port to be in extended-fabric mode, see the documentation for the specific switch.

SnapMirror traffic zones

SnapMirror over Fibre Channel requires traffic to occur in a dedicated zone.

In a typical SAN, SnapMirror traffic shares the same physical network with other Fibre Channel traffic. Zoning enables you to partition the servers and storage into different groups, and to enforce access privileges to these groups.

When you have a multipath setup, you should use a separated zone for each data path, as shown in the following figure. SnapMirror over Fibre Channel supports both WWN zoning and physical port number zoning.



Figure 14: Zones for SnapMirror traffic

Note: For information about detailed zoning configuration procedures, see the documentation for the specific switch.

Related tasks

Configuring SnapMirror over Fibre Channel on page 212

Requirements for deploying SnapMirror over Fibre Channel

You need to ensure that the minimum requirements are met for deploying SnapMirror over Fibre Channel.

• You must use supported switches and compatible firmware.

Note: To view the supported switches and firmware versions, see *Requirements for SnapMirror over Fibre Channel transport (Asnychronous, Synchronous, and Semi-synchronous modes)* on the NOW site.

- There must be at least one switch between the source and destination.
- The SAN configuration must be homogenous with respect to the switch vendors.
- SnapMirror over Fibre Channel requires frame in-order delivery (IOD), even during the fabric reconfiguration process. Therefore, users need to turn on the IOD-related options on the Fibre Channel switch. For example, on a Cisco switch, configure VSAN with the in-order guarantee option.
- The path between the source and destination must use dedicated switches or the Storage Area Network (SAN) must be configured such that SnapMirror traffic occurs in a dedicated zone. For Cisco switches, you can use the dedicated VSAN capability.
- You must use the native Fibre Channel fabric deployment, that is, no Fibre Channel over IP protocol conversions in the SnapMirror data path.
- Brocade Fibre Channel router in the SnapMirror data path is not supported.

Related information

now.netapp.com/NOW/knowledge/docs/switches/sm_fc_switch_support/

Functionality supported by SnapMirror over Fibre Channel

SnapMirror over Fibre Channel supports port ID-based load-balancing algorithms, such as SID/DID. SnapMirror over Fibre Channel also supports distance extension using dark fibre or DWDM equipment.

The following functionalities are not supported.

- Exchange ID-based load-balancing algorithm
- Distance extension using protocols such as Fibre Channel over IP

Related tasks

Configuring SnapMirror over Fibre Channel on page 212

Related references

Troubleshooting issues related to SnapMirror over Fibre Channel on page 219

Configuring SnapMirror over Fibre Channel

To use SnapMirror over Fibre Channel, you must install the Fibre Channel NIC adapters, and set up the SnapMirror relationships. You can also set up multiple SnapMirror traffic zones on supported switches.

Before you begin

To view the supported switches and firmware versions, see the *Requirements for SnapMirror over Fibre Channel transport (Asnychronous, Synchronous, and Semi-synchronous modes)* on the NOW site.

About this task

For updated information about the hardware supported with different system models, see the *System Configuration Guide*.

Steps

1. Install the Fibre Channel NIC adapters in the source and destination systems.

See the System Configuration Guide for installation instructions.

2. Connect the systems to Fibre Channel switches.

For more information, see the hardware and service guide for your system.

3. Use the sysconfig command to identify the Fibre Channel NIC adapters on each system.

Example

In the following example, the sysconfig command shows that the system has a Fibre Channel NIC adapter installed in slot 4.

```
system_1> sysconfig
Release 7.3: Wed Mar 31 02:47:49 PST 2008
System ID: 0033587346 (system_1); partner ID 0033586737
(system_2)
System Serial Number: 1022105 (system_1)
System Rev: B0
Backplane Part Number: 0
Backplane Rev: B0
Backplane Serial Number: 1022105
slot 0: System Board
Processors:1
Memory Size:3072 MB
CIOB Revision ID:5
slot 0:FastEnet-10/100 Ethernet Controller
e0 MAC Address: 00:a0:98:00:f5:39 (auto-100x-fd-up)
slot 4: FCVI Host Adapter 4a
slot 4: FCVI Host Adapter 4b
slot 6: Gigabit Ethernet Controller IV
e6 MAC Address 00:02:b3:aa:19:d6: (auto-100x-fd-up)
slot 7: FC Host Adapter 7
28 Disks 1904.0GB
slot 8: FC Host Adapter 8
28 Disks: 1904.0GB
slot 10: VI Cluster Adapter
slot 11: NVRAM
Memory Size: 128 MB
```

4. Enter the sysconfig -v command to display port connection details, including host port ID acquired from the login to the switch, as well as the ID of the switch port to which it connects.

Example

The following command displays port connections for the adapter in slot 4.

```
system_1> sysconfig -v 4
slot 4: FCVI Host Adapter 4a (Dual Channel, QLogic 2312
(2352) rev. 2, 64
-bit,F-port, <UP>
Firmware rev: 3.1.18
Host Loop Id:0xffHost Port Id: 0xa01200
Cacheline size:8FC Packet Size: 2048
SRAM parity:yesExternal GBIC: No
Link Data Rate:2 Gbit
Switch Port brcd_sw_1: 2
slot 4: FCVI Host Adapter 4a (Dual Channel, QLogic 2312
(2352) rev. 2, 64
-bit,F-port, <UP>
Firmware rev: 3.1.18
Host Loop Id: 0xffHost Port Id: 0xa01200
Cacheline size:8FC Packet Size: 2048
SRAM parity: yesExternal GBIC: No
Link Data Rate:2 Gbit
Switch Port brcd_sw_2: 2
```

5. Determine the zoning configuration.

See the documentation for your switch configuration information. Ensure that the Fibre Channel switches are in fabric mode and support the Simple Name Service (SNS) protocol with support for symbolic names.

Steps 6 through 9 show you how to create new zones and a multipath setup between a source system and a destination system with a Brocade switch. Each system has one X1024 Fibre Channel NIC card installed. The port WWNs are listed in the following table.

Source system	Destination system
port a: 20:00:00:e0:8b:0a:aa:6d	port a: 20:00:00:e0:8b:14:70:af
port b: 20:01:00:e0:8b:2a:aa:6d	port b: 20:01:00:e0:8b:34:70:af

The primary path is formed by port a on the source and destination, and the secondary path is formed by port b on the source and destination. SnapMirror traffic needs a dedicated zone, and there should be separate zones for the two paths.

Steps 10 through 13 show you how to create new zones and a multipath setup with a Cisco switch. The zoning ideas are the same when using Cisco or Brocade switches. However, the command semantics are different. Use the config command to enter the switch configuration mode and set up the zoning configuration as shown in Steps 14 through 17.

6. (With a Brocade switch) Create a new zone with the zonecreate command.

Example

```
brcd_sw_1:root> zonecreate "sm_zone_1", "20:00:00:e0:8b:0a:aa:6d;
20:00:00:e0:8b:14:70:af"
brcd_sw_1:root> zonecreate "sm_zone_2", "20:01:00:e0:8b:2a:aa:6d;
20:01:00:e0:8b:34:70:af"
```

7. Create a new zoning configuration by using the cfgcreate command.

Example

brcd_sw_1:root> cfgcreate "sm_zone_cfg", "sm_zone_1; sm_zone_2"

8. Enable the zoning configuration by using the cfgenable command.

Example

```
brcd_sw_1:root> cfgenable "sm_zone_cfg"
zone config "sm_zone_cfg" is in effect
Updating flash ...
```

9. Check the zoning configuration by using the cfgshow command.

Example

```
brcd_sw_1:root> cfgshow
Defined configuration:
cfg: sm_zone_cfg
sm_zone_1; sm_zone_2
```

```
zone: sm_zone_1
20:00:00:e0:8b:0a:aa:6d; 20:00:00:e0:8b:14:70:af
zone: sm_zone_2
20:01:00:e0:8b:2a:aa:6d; 20:01:00:e0:8b:34:70:af
Effective configuration:
cfg: sm_zone_cfg
zone: sm_zone_1
20:00:00:e0:8b:0a:aa:6d
20:00:00:e0:8b:14:70:af
zone: sm_zone_2
20:01:00:e0:8b:2a:aa:6d
20:01:00:e0:8b:34:70:af
```

10. (With a Cisco switch) Define the two zones by using the following commands.

```
cisco_sw_1(config)# zone name sm_zone_1
cisco_sw_1(config-zone)# member pwwn 20:00:00:e0:8b:0a:aa:6d
cisco_sw_1(config-zone)# member pwwn 20:00:00:e0:8b:14:70:af
cisco_sw_1(config-zone)# zone name sm_zone_2
cisco_sw_1(config-zone)# member pwwn 20:01:00:e0:8b:2a:aa:6d
cisco_sw_1(config-zone)# member pwwn 20:01:00:e0:8b:34:70:af
```

11. Define the zoning configuration by using the following commands.

```
cisco_sw_1(config)# zoneset name sm_zone_cfg
cisco_sw_1(config-zoneset)# member sm_zone_1
```

12. Activate the zoning configuration by using the following commands.

```
cisco_sw_1(config-zoneset)# zoneset activate name sm_zone_cfg
Zoneset activation initiated. check zone status
```

```
cisco_sw_1(config-zoneset)# member sm_zone_2
```

13. Check zoning configuration status by using the following commands.

```
cisco_sw_1# show zoneset active
zoneset name sm_zone_cfg
zone name sm_zone_1
pwwn 20:00:00:e0:8b:0a:aa:6d
pwwn 20:00:00:e0:8b:14:70:af
zone name sm_zone_2
pwwn 20:01:00:e0:8b:2a:aa:6d
pwwn 20:01:00:e0:8b:34:70:af
```

14. Determine the IP address and net mask for each port. If you have an active/active configuration, you also need to decide the IP address of the partner port. These IP addresses must be within the private network IP address range.

Note: You should configure different private subnets for each Fibre Channel port on the system.

15. Use the setup command to configure the IP address. This ensures that the changes are committed to non-volatile storage and persist after a system reboot.

Example

The following example shows an abbreviated setup command output displayed when configuring a Fibre Channel NIC adapter.

```
system_1> setup
The setup command will rewrite the /etc/rc, /etc/exports, /etc/hosts, /
etc/hosts.equiv, /etc/dgateways, /etc/nsswitch.conf, and /etc/
resolv.conf files, saving the original contents of these files in .bak
files (e.g. /etc/exports.bak).
Are you sure you want to continue? [yes] yes
Release 7.3: Wed Mar 31 02:47:49 PST 2009
System ID: 0033587346 (system_1); partner ID: 0033586737 (system_2)
    System Serial Number: 1022105 (system_1)
    System Rev: B0
    Backplane Part Number: 0
    Backplane Rev: B0
    Backplane Serial Number: 1022105
    slot 0: System Board
        Processors: 1
        Memory Size: 3072 MB
        CIOB Revision ID: 5
    slot 0: FastEnet-10/100 Ethernet Controller
        e0 MAC Address: 00:a0:98:00:f5:39 (auto-100tx-fd-up)
slot 4: FCVI Host Adapter 4b
    slot 4: FCVI Host Adapter 4b
    slot 6: Gigabit Ethernet Controller IV
        e6 MAC Address: 00:02:b3:aa:19:d6 (auto-1000sx-fd-up)
    slot 7: FC Host Adapter 7
        28 Disks: 1904.0GB
        2 shelves with LRC
    slot 8: FC Host Adapter 8
        28 Disks: 1904.0GB
        2 shelves with LRC
slot 10: VI Cluster Adapter
    slot 11: NVRAM
        Memory Size: 128 MB
Please enter the new hostname [system_1]:
Do you want to configure virtual network interfaces? [n]: . .
Please enter the IP address for FC Network Interface ql4a [10.1.1.15]:
10.1.1.15
Please enter the netmask for FC Network Interface ql4a [255.255.255.0]:
255.255.255.0
Should interface ql4a take over a partner IP address during failover?
[y]: y
Please enter the IP address or interface name to be taken over by ql4a
[10.1.1.16]: 10.1.1.16
Please enter the IP address for FC Network Interface ql4b [10.1.2.15]:
10.1.2.15
Please enter the netmask for FC Network Interface ql4b [255.255.255.0]:
255.255.255.0
Should interface ql4b take over a partner IP address during failover?
[y]: y
```
Please enter the IP address or interface name to be taken over by ql4b $[10.1.2.16]\colon$ 10.1.2.16 . . .

You need to reboot the system for the changes to take effect. Use the reboot command.

16. Reboot the systems and Fibre Channel switches.

17. Use the ifconfig command to verify the IP address configuration.

Note: Ensure that IP addresses for the ports on the same fabric have the same net number.

A port on the FC NIC adapter is named as qlxa or qlxb, where:

- ql—Indicates the card vendor. (At this point it is ql only, which stands for QLogic)
- x—The slot number in which the card is installed in the system
- a/b—The FC port number on the card

Example

The following example shows an output for the ifconfig command.

```
system_1> ifconfig ql4a
ql4a: flags=840041<UP,RUNNING,LINK_UP> mtu 8160
    inet 10.1.1.15 netmask 0xffffff00 broadcast 0.0.0.0
    partner inet 10.1.1.16 (not in use)
    ether 00:00:00:00:00 (VIA Provider)
system_1> ifconfig ql4b
ql4b: flags=840041<UP,RUNNING,LINK_UP> mtu 8160
    inet 10.1.2.15 netmask 0xffffff00 broadcast 0.0.0.0
    partner inet 10.1.2.16 (not in use)
    ether 00:00:00:00:00 (VIA Provider)
```

Note: Under normal operation, you should see UP and LINK_UP in the command output. UP means that this interface has been enabled and is working. LINK_UP means that the physical connection of the interface is online.

18. Use the route command to verify the IP routing table setup. As this is on a private network, it has a special entry in the routing table.

Example

```
system_1> route -s
Routing tables
Internet:
Destination Gateway Flags Refs Use Interface
default harley-29-19.lab.n UGS 4 319989 e0
10.1.1/24 system_1-q14a U 0 0 q14a
10.1.2/24 system_1-q14b U 0 0 q14b
127 localhost UGS 0 0 lo
localhost localhost UH 1 0 lo
```

The preceding entries specify that for IPs within subnet 10.1.1.x, use the ql4a interface; for IPs within subnet 10.1.2.x, use the ql4b interface.

19. Use the ping command to verify the connectivity between two network interfaces.

Example

system_1> ping -s 10.1.1.26 64 bytes from 10.1.1.26 (10.1.1.26): icmp_seq=0 ttl=255 time=0.903 ms 64 bytes from 10.1.1.26 (10.1.1.26): icmp_seq=1 ttl=255 time=0.462 ms 64 bytes from 10.1.1.26 (10.1.1.26): icmp_seq=2 ttl=255 time=0.442 ms 64 bytes from 10.1.1.26 (10.1.1.26): icmp_seq=3 ttl=255 time=0.442 ms --- 10.1.1.26 ping statistics ---4 packets transmitted, 4 packets received, 0% packet loss round-trip min/avg/max = 0.439/0.561/0.903 ms

Note: After the zoning configuration has been changed, you need to verify the connectivity again.

Related information

now.netapp.com/NOW/knowledge/docs/switches/sm_fc_switch_support/

Troubleshooting issues related to SnapMirror over Fibre Channel

You might come across certain issues when setting up a SnapMirror over Fibre Channel connection. The following table lists some of the problems you might encounter and recommends ways to tackle these issues.

Problem	Solution
Connection failure	1. Check the cabling between the system and switch and make sure there are no loose connections.
	2. Ping the ports. If you have a good connection, you should see something similar to the following output.
	filer_1*> ping -s 10.1.1.26
	64 bytes from 10.1.1.26 (10.1.1.26): icmp_seq=0 ttl=255
	64 bytes from 10.1.1.26 (10.1.1.26): icmp_seq=1 ttl=255
	64 bytes from 10.1.1.26 (10.1.1.26): icmp_seq=2 ttl=255
	time=0.439 ms 64 bytes from 10.1.1.26 (10.1.1.26): icmp_seq=3 ttl=255 time=0.442 ms
	10.1.1.26 ping statistics
	4 packets transmitted, 4 packets received, 0% packet loss
	round-trip min/avg/max = 0.439/0.561/0.903 ms
	In case of a connection failure, you see something similar to this.
	filer_1*> ping 10.1.1.26
	no answer from 10.1.1.26

220 | Data ONTAP 7.3 Data Protection Online Backup and Recovery Guide

Problem	Solution	
The local Fibre Channel NIC port might be offline	 Enter the ifconfig command to exam the Fibre Channel NIC port state. If you do not see the UP state in the ifconfig output, as shown here, that means the interface has been taken down. 	
	<pre>filer_1*> ifconfig ql4a ql4a: flags=800040<running,link_up> mtu 8160 inet 1.1.1.15 netmask 0xffffff00 broadcast 0.0.0.0 partner inet 1.1.1.16 (not in use) ether 00:00:00:00:00 (VIA Provider)</running,link_up></pre>	
	3. Enter the following command to correct this problem. ifconfig interface_name up	
	4. If you do not see the LINK_UP state in the output, the physical connections is offline.	
	<pre>filer_1*> ifconfig ql4a ql4a: flags=40041<up,running> mtu 8160 inet 1.1.1.15 netmask 0xffffff00 broadcast 0.0.0.0 partner inet 1.1.1.16 (not in use) ether 00:00:00:00:00 (VIA Provider)</up,running></pre>	
	5. In this case, check the physical connections including the fiber-optic cable and the FC switch side configuration.	

Problem	Solution	
The remote Fibre Channel NIC port information might not be correct, and the local port might not "see" the remote port.	 Use the fcnic command to verify that a local FC NIC port can "see" remote ports that are zoned together. This command requires diagnostic privilege (priv set diag command). A sample output is shown here. 	
	<pre>filer_1*> fcnic show fabric q14a pid = 0xa01000 wwn = 20:00:00:e0:8b:14:67:af IP addr(s) = 10.1.1.13 * pid = 0xa01200 wwn = 20:00:00:e0:8b:0a:a8:6d IP addr(s) = 10.1.1.15 pid = 0xa01300 wwn = 20:01:00:e0:8b:2a:aa:6d IP addr(s) = 10.1.1.16 pid = 0xa01800 wwn = 20:00:00:e0:8b:14:70:af IP addr(s) = 10.1.1.26 Port Login database: pid = 0xfffffe, lid = 0x7e pid = 0xfffffe, lid = 0x7e pid = 0xa01800, lid = 0xee pid = 0xfffffa, lid = 0xef 2. The entry prefixed by an asterisk (*) is the local entry specified in the</pre>	
	command line. This output displays all the FC NIC ports (remote and local) that are zoned together. Any missing entries or any missing information within an entry indicates a connection problem.	

Problem	Solution		
The Fibre Channel switch might not be configured correctly.	To pinpoint a Fibre Channel connection problem, check the Fibre Channel switch and the fabric to which the system is connected. Under normal working conditions, an FC NIC port should communicate with the FC switch as an F-Port. The following example shows an output of the switchshow command on a Brocade switch.		
	<pre>brcd_sw_1:root> switchshow switchName: brcd_sw_1 switchType: 9.2 switchState: Online switchMode: Native switchRole: Principal switchDomain: 160 switchId: fffca0 switchId: fffca0 switchBeacon: 0FF Zoning: 0N (sm_zone_cfg)</pre>		
	port 0: id N2 Online F-Port 20:00:00:e0:8b:14:67:af port 1: id N1 Online F-Port 20:00:00:e0:8b:0a:16:6e port 2: id N2 Online F-Port 20:00:00:e0:8b:0a:a8:6d port 3: id N2 Online F-Port 20:00:00:e0:8b:0a:a8:6d port 4: id N2 No_Light F-Port 20:00:00:e0:8b:0a:a2:6e port 5: id N2 Online F-Port 21:00:00:e0:8b:0a:15:6e port 6: id N1 Online F-Port 20:00:00:e0:8b:14:7c:af port 7: id N2 Online F-Port 21:00:00:e0:8b:14:7c:af port 8: id N2 Online F-Port 20:00:00:e0:8b:14:7c:af port 9: id N2 No_Light F-Port 20:00:00:e0:8b:14:70:af port 10: id N2 No_Light F-Port 20:00:00:e0:8b:14:70:af port 11: id N2 No_Light L2 port 12: id N2 No_Light L2 port 13: id N2 No_Light L2 port 14: id N2 No_Light L2 port 15: id N2 No_Light L2		
	3. If the corresponding port does not show up as an F-Port, there are some negotiation problems between the Fibre Channel NIC port on the system and the switch port. Check the switch port configuration, in particular, the port topology and speed configurations.		
The switch name server might not have all the Fibre	1. Make sure that the switch name server database has all the FC NIC port entries. On a Brocade switch, use the nsallshow command, the output of which is shown here.		
Channel NIC port entries.	brcd_sw_1:root> nsallshow { a01000 a01100 a01200 a01300 a01500 a01600 a01700 a01800 8 Nx_Ports in the Fabric }		
	2. This command displays the 24-bit Fibre Channel addresses (PIDs) of all the devices in the fabric. Ensure that this list includes all the FC NIC ports.		

Note: If you observe very low throughput compared to the available physical network bandwidth, contact technical support.

Troubleshooting of SnapMirror issues

When using SnapMirror, you might face issues when you change the name of the destination volume. There can also be issues when SnapMirror Snapshot copies are deleted.

Next topics

What happens if you change a SnapMirror destination volume name on page 223 *Accidental deletion of SnapMirror Snapshot copies* on page 224

What happens if you change a SnapMirror destination volume name

If you change the name of a SnapMirror destination volume, you need to manually correct the SnapMirror relationships affected by the change. SnapMirror is unable to replicate source volume data to a newly named destination volume whose configuration information is incomplete.

In the following case, the destination, volJobak, was renamed to volStatbak. After the renaming, the snapmirror status command does not display the source. Instead, the entry is shown with a dash (-) in the source column.

systemB> vol rename volJobak volStatbak volJobak renamed to volStatbak you may need to update /etc/exports systemB> snapmirror status volJobak Snapmirror is on. systemB>snapmirror status volStatbak Snapmirror is on. Source Destination State Lag Status - systemB:volStatbak Snapmirrored -00:03:22 Idle

If you change the volume name of a SnapMirror source or destination, you need to make the following changes.

- 1. Update the snapmirror.conf file, if there is an old entry.
- 2. Use the snapmirror release command to update the old destination name, and SnapMirror releases the soft lock and the old Snapshot copy.
- **3.** Use the snapmirror update command on the new volume name, and status registry is updated with the new volume name.
- 4. Update the /etc/exports file.

Note: If a system is running at its limit of concurrent transfers, and you attempt to initiate more transfers through by using the snapmirror update command, the attempted transfer will fail.

Accidental deletion of SnapMirror Snapshot copies

SnapMirror Snapshot copies stored on either the SnapMirror source or destination location must not be deleted. If the base Snapshot copy (most recent common Snapshot copy) is accidentally deleted from either the source or destination location, you can attempt recovery.

You might be able to recover without reinitializing the destination by breaking the SnapMirror relationship and then resynchronizing the source and the destination.

As long as there is at least one Snapshot copy common to both the source and the destination, resynchronization will succeed.

If there is no Snapshot copy common to both the source and the destination, you need to use the snapmirror initialize command over the network. Or, if the source and destination are volumes, use the snapmirror store command to store the source volume on tape and then use the snapmirror retrieve command to restore the volume from the tape to the destination.

Related concepts

Conversion of a destination to a writable volume or qtree on page 172

Related tasks

Resynchronizing a SnapMirror relationship on page 190

Data protection using SnapVault

SnapVault protects data on a SnapVault primary system by maintaining a number of read-only versions of that data on a SnapVault secondary system and the SnapVault primary system.

Next topics

What Snap Vault is on page 226 Planning Snap Vault backups on page 233 Enabling Snap Vault on page 237 How to start a Snap Vault backup relationship on page 240 What Snap Vault Snapshot copy update schedules are on page 244 Unscheduling SnapVault Snapshot copies on page 251 Checking Snap Vault transfers on page 253 Displaying Snap Vault Snapshot copies on page 258 Guidelines for creating a clone of a Snap Vault destination volume on page 261 About LUN clones and Snap Vault on page 262 How to change SnapVault settings on page 264 Why you manually update a gtree on the secondary system on page 266 Why you create a Snapshot copy manually on page 268 Renaming a Snap Vault or Open Systems Snap Vault secondary volume on page 270 Restoring Snap Vault data to the primary system on page 271 How to abort Snap Vault transfers on page 275 Ending Snap Vault backups for a qtree on page 276 Releasing Snap Vault relationships on page 277 Turning Snap Vault off on page 278 Compression feature of Open Systems Snap Vault on page 278 Snap Vault secondary system protection on page 282 How to use SnapVault to protect a volume SnapMirror destination on page 287 Snap Vault and MultiStore on page 290 Error regarding language setting changes on volumes on page 291 About Snap Vault for NetBackup on page 291

What SnapVault is

SnapVault is a disk-based storage backup feature of Data ONTAP. SnapVault enables data stored on multiple systems to be backed up to a central, secondary system quickly and efficiently as read-only Snapshot copies.

In the event of data loss or corruption on a system, backed-up data can be restored from the SnapVault secondary system with less downtime and uncertainty than is associated with conventional tape backup and restore operations.

You must understand the following terms that describe the SnapVault feature:

- Primary system—a system whose data is to be backed up
- Secondary system—a system to which data is backed up
- Primary system qtree—a qtree on a primary system whose data is backed up to a secondary qtree on a secondary system
- Secondary system qtree—a qtree on a secondary system to which data from a primary qtree on a primary system is backed up
- Open systems platform—a server running AIX, Solaris, HP-UX, Red Hat Linux, SUSE Linux, or Windows, whose data can be backed up to a SnapVault secondary system
- Open Systems SnapVault agent—a software agent that enables the system to back up its data to a SnapVault secondary system
- SnapVault relationship—the backup relationship between a qtree on a primary system or a directory on an open systems primary platform and its corresponding secondary system qtree
- SnapVault Snapshot copy—the backup images that SnapVault creates at intervals on its primary and secondary systems

SnapVault Snapshot copies capture the state of primary qtree data on each primary system. This data is transferred to secondary qtrees on the SnapVault secondary system. The secondary system creates and maintains versions of Snapshot copies of the combined data for long-term storage and possible restore operations.

- SnapVault Snapshot basename—a name that you assign to a set of SnapVault Snapshot copies using the snapvault snap sched command. As incremental Snapshot copies for a set are taken and stored on both the primary and secondary systems, the system appends a number (0, 1, 2, 3, and so on) to the basenames to track the most recent and earlier Snapshot updates.
- SnapVault baseline transfer—an initial complete backup of a primary storage qtree or an open systems platform directory to a corresponding qtree on the secondary system
- SnapVault incremental transfer—a follow-up backup to the secondary system that contains only the changes to the primary storage data between the current and last transfer actions

Next topics

Advantages of using SnapVault on page 227 What data gets backed up and restored through SnapVault on page 227 Types of SnapVault deployment on page 228 How Snap Vault backup works on page 230 How Snap Vault backup works for open systems on page 231 Snap Vault support for IPv6 on page 232

Advantages of using SnapVault

The SnapVault disk-based backup and restore system enables you to perform fast and simple data restore operations.

You can also perform the following operations:

- Browse backed-up files online.
- Schedule frequent and efficient backup of large amounts of data.
- Minimize media consumption and system overhead through incremental backup.
- If tape backup is necessary, offload the tape backup task from the primary storage systems to the SnapVault secondary storage system, which centralizes the operation and saves resources.
- Configure and maintain a single storage system for backing up data stored on multiple platforms: Data ONTAP, AIX, Solaris, HP-UX, Linux, Windows, or VMware ESX server systems.

What data gets backed up and restored through SnapVault

The data structures that are backed up and restored through SnapVault depend on the primary system.

- On systems running Data ONTAP, the qtree is the basic unit of SnapVault backup and restore. SnapVault backs up specified qtrees on the primary system to associated qtrees on the SnapVault secondary system. If necessary, data is restored from the secondary qtrees back to their associated primary qtrees.
- On open systems storage platforms, the directory is the basic unit of SnapVault backup. SnapVault backs up specified directories from the native system to specified qtrees in the SnapVault secondary system.

If necessary SnapVault can restore an entire directory or a specified file to the open systems platform.

Note: You can back up the qtrees from multiple primary systems, or directories from multiple open systems storage platforms, to associated qtrees on a single SnapVault secondary volume.

The illustration shows the backup of qtrees and directories on different systems to a single secondary volume.



Figure 15: Primary and secondary systems

Note: Starting with Data ONTAP 7.3.2, SnapVault does not support the use of source Snapshot copies created by releases prior to Data ONTAP 6.5.

Types of SnapVault deployment

You can deploy SnapVault in three ways as per business requirements.

- Basic SnapVault deployment
- · Primary to secondary to tape backup variation
- · Primary to secondary to SnapMirror variation

Next topics

What basic Snap Vault deployment is on page 228 Primary to secondary to tape backup variation on page 229 Primary to secondary to SnapMirror variation on page 230

What basic SnapVault deployment is

The basic SnapVault backup system deployment consists of primary system and a secondary system.

Primary storage systems: Primary systems are the platforms that run Data ONTAP and open systems storage platforms to be backed up.

- On primary systems, SnapVault backs up primary qtree data, non-qtree data, and entire volumes, to qtree locations on the SnapVault secondary systems.
- Supported open systems storage platforms include Windows servers, Solaris servers, AIX servers, Red Hat Linux servers, SUSE Linux servers, and HP-UX servers. On open systems storage platforms, SnapVault can back up directories to qtree locations on the secondary system. SnapVault can restore directories and single files. For more information, see the *Open Systems SnapVault Installation and Administration Guide*.

Secondary storage system: The SnapVault secondary system is the central disk-based unit that receives and stores backup data from the system as Snapshot copies. Any system can be configured as a SnapVault secondary system; however, the recommended hardware platform is a NearStore system.



The following figure shows a basic SnapVault deployment.

Figure 16: Basic SnapVault deployment

Primary to secondary to tape backup variation

A common variation to the basic SnapVault backup deployment adds a tape backup of the SnapVault secondary system.

This deployment can serve two purposes:

- It enables you to store an unlimited number of network backups offline while keeping the most recent backups available online in secondary storage. This can help in the quick restoration of data. If you run a single tape backup off the SnapVault secondary storage system, the storage platforms are not subject to the performance degradation, system unavailability, and complexity of direct tape backup of multiple systems.
- It can be used to restore data to a SnapVault secondary system in case of data loss or corruption on that system.

Note: Some UNIX attributes are not preserved using this method; notably, UNIX access control lists (ACLs).

The following figure shows a basic SnapVault deployment with tape backup.



Figure 17: Basic SnapVault deployment with tape backup

Primary to secondary to SnapMirror variation

In addition to the basic SnapVault deployment, you can replicate the SnapVault secondary using SnapMirror. This protects the data stored on the SnapVault secondary against problems with the secondary system itself.

The data backed up to SnapVault secondary storage is replicated to a SnapMirror destination.

If the secondary system fails, the data mirrored to the SnapMirror destination can be converted to a secondary system and used to continue the SnapVault backup operation with minimum disruption.

How SnapVault backup works

Backing up qtrees using SnapVault involves starting the baseline transfers, making scheduled incremental transfers, and restoring data upon request.

How to start the baseline transfers:

- In response to command-line input, the SnapVault secondary system requests initial base transfers of qtrees specified for backup from a primary storage volume to a secondary storage volume. These transfers establish SnapVault relationships between the primary and secondary qtrees.
- Each primary system, when requested by the secondary system, transfers initial base images of specified primary qtrees to qtree locations on the secondary system.

How to make scheduled incremental transfers:

• Each primary system, in response to command-line input, creates sets of scheduled SnapVault Snapshot copies of the volumes containing the qtrees to be backed up. For tracking purposes, you might name according to frequency, for example, sv_hourly, sv_nightly, and so on.

For each Snapshot set, SnapVault saves the number of primary storage Snapshot copies you specify and assigns each Snapshot a version number (0 for most current, 1 for second most recent, and so on).

The SnapVault secondary system, in response to command-line input, carries out a specified set of scheduled data transfer and Snapshot actions. For each of its secondary gtrees on a given volume, SnapVault retrieves, from the Snapshot data of each corresponding primary qtree, the incremental changes to the primary gtrees made since the last data transfer. Then SnapVault creates a volume Snapshot copy of the changes in the secondary qtrees. For each transfer and Snapshot set, SnapVault saves the number of secondary storage Snapshot copies that you specify and assigns each Snapshot copy a version number (0 for most current, 1 for second most recent, and so on).

Restoration upon request:

If data needs to be restored to the primary system, SnapVault transfers the specified versions of the gtrees back to the primary system that requests them.

The following diagram illustrates SnapVault functionality.



Protects multiple gtrees/volumes on multiple primary storage systems on

Figure 18: SnapVault functionality

How SnapVault backup works for open systems

The process of SnapVault backup of open systems platform directories involves starting the baseline transfers, scheduling incremental transfers, and restoring data upon request.

Starting the baseline transfers:

In response to CLI input, the SnapVault secondary system requests from an open systems platform initial baseline image transfers of directories specified for backup. These transfers establish SnapVault relationships between the open systems platform directories and the SnapVault secondary gtrees.

• Each open systems platform, when prompted by the secondary system, transfers initial base images of specified directories to qtree locations on the secondary system.

Note: There are no primary-side Snapshot copies in Open Systems SnapVault.

Scheduling incremental transfers:

• The SnapVault secondary storage system, in response to CLI input, follows a set of scheduled data transfers (to which, for tracking purposes, you can assign names like "sv_hourly," "sv_nightly," and so on).

To each secondary qtree on a given volume, from a corresponding primary directory on the open systems storage platform, SnapVault transfers the files that have been added or modified since the previous data transfer.

For each set of scheduled data transfers, SnapVault creates a set of incremental Snapshot copies that capture the changes to the secondary qtrees after each transfer. For each set of Snapshot copies, the SnapVault secondary system saves the number of secondary storage Snapshot copies you specify and assigns each Snapshot copy in the set a version number (0 for most current, 1 for second most recent, and so on).

• Restore upon request: If directory or file data needs to be restored to the open systems storage platform, SnapVault retrieves the data from one of the retained Snapshot copies and transfers the data back to the open systems storage platform that requests it.

For more information, see the Open Systems Snap Vault Installation and Administration Guide.

SnapVault support for IPv6

SnapVault supports the use of IPv6 addresses to specify source and destination systems. However, there are some differences between the specification of IPv6 and IPv4 addresses.

When using an IPv6 address with the snapvault command, you need to enclose the IPv6 address within square brackets. The usage is shown in the following example.

Use of IPv6 address with the snapvault start command

```
snapvault start -S [2001:0:0:0:FFD3:0:57ab]:/vol/vol1/qtree2
system2:/vol/vol1/qtree2
```

You can also specify whether to use an IPv6 connection or an IPv4 connection, for the following three commands by using the -p option.

- snapvault start
- snapvault modify
- snapvault restore

The use of the -p option is shown in the following command entry.

snapvault start [-p { inet6 | inet | unspec }] -S system1:/vol/vol1/qtree3
system2:/vol/vol1/qtree3

- unspec is the default value. An IPv6 connection is attempted first. If an IPv6 connection is not established, then an IPv4 connection is attempted.
- inet6 specifies the use of an IPv6 connection.
- inet specifies the use of an IPv4 connection.

Use of IPv6 connection with the snapvault modify command

```
snapvault modify -p inet6 -S system1:/vol/vol1/qtree3 system2:/vol/
vol1/qtree3
```

Planning SnapVault backups

Before starting SnapVault backups, you need to plan your primary system qtrees or open systems directories and their corresponding secondary system qtrees. You also need to plan the SnapVault backup schedule and Snapshot copy retention, and estimate the initial backup time.

Next topics

Planning primary and secondary qtree locations on page 233 SnapVault primary and secondary on the same system on page 234 Planning SnapVault backup schedule and Snapshot copy retention on page 234 Estimating the initial backup time on page 236 Limit on the number of concurrent SnapVault targets on page 236

Planning primary and secondary qtree locations

Planning the location of primary system qtrees or open systems directories and their corresponding secondary system qtrees is helpful for better storage management. You can back up multiple qtrees from different volumes and multiple open systems directories to corresponding qtrees in a single volume.

Step

1. List the primary system qtrees or open systems directories and their corresponding secondary system qtrees in a table. For an example, see the following table.

Primary system qtree or directory location	Corresponding secondary system qtree location
systemA:/vol/vol1/qtreeAA	<pre>sv_secondary:/vol/sv_vol/qtreeAA</pre>
systemA:/vol/vol1/qtreeAB	<pre>sv_secondary:/vol/sv_vol/qtreeAB</pre>
systemB:/vol/vol1/qtreeBB	<pre>sv_secondary:/vol/sv_vol/qtreeBB</pre>
winsrvrA:c:\melzdir	<pre>sv_secondary:/vol/sv_vol/melzdir</pre>
ux_srvrB:/usrs/moz_acct	<pre>sv_secondary:/vol/sv_vol/moz_acct</pre>

Note: The maximum number of secondary system qtrees per volume is 255. SnapVault backup of qtrees over a Fibre Channel network is not supported.

SnapVault primary and secondary on the same system

In Data ONTAP 7.3 and later, the SnapVault primary and secondary features can be on the same system. However, you need to enable both the SnapVault primary and the SnapVault secondary licenses on the same system.

After enabling both the SnapVault primary and secondary licenses on one system, you can use the system in both of the following ways.

- SnapVault destination for one or multiple backup relationships.
- Both SnapVault source and SnapVault destination for the same backup relationship. For example, by using SnapVault you can back up data from FC aggregates to ATA aggregates connected to the same system.

Note: The source and destination qtrees cannot be within the same volume.

Planning SnapVault backup schedule and Snapshot copy retention

It is important to plan the SnapVault backup schedule and number of Snapshot copies to be retained.

Before you begin

Before you start SnapVault configuration, you should create a table to plan how many Snapshot copies you want per volume, when you want them updated, and how many of each you want to keep.

For example:

- Hourly (periodically throughout the day) Does the data change often enough throughout the day to make it worthwhile to create a Snapshot copy every hour, every two hours, or every four hours?
- Nightly
 - Do you want to create a Snapshot copy every night or just workday nights?
- Weekly

How many weekly Snapshot copies is it useful to keep?

On storage-based primary systems and SnapVault secondary systems, the data to be backed up is captured and preserved in Snapshot copies.

Steps

1. On the Data ONTAP primary systems, plan the intervals at which to create SnapVault Snapshot copies of your primary system qtrees. The maximum frequency at which you can take Snapshot copies is 1 hour.

Note: On primary storage platforms not running Data ONTAP, Snapshot copy planning and creation does not apply. For more information, see the Open Systems SnapVault documentation.

- 2. On the SnapVault secondary system, plan the intervals at which you want to update the secondary system qtrees with data transferred from primary storage platforms, and create SnapVault Snapshot copies to retain that information.
- **3.** Plan how to limit the combined total of Snapshot copies retained on any one volume of the SnapVault secondary system to 251 or fewer.

Attention: The combined total of Snapshot copies retained on each volume of the SnapVault secondary system cannot exceed 251. If the number of Snapshot copies per volume limit is reached and the old Snapshot copies are not deleted, SnapVault will not create new Snapshot copies on that volume.

Example

In this example, the user is supposed to have 12 qtrees on the secondary system volume.

Snapshot intervals	Primary storage: when created	Primary storage: Snapshot copies retained	Secondary storage: when created	Secondary storage: Snapshot copies retained
weekly	sat @19	4	sat @21	8
nightly	mon-fri @19	10	mon-fri @20	60
hourly	@7-18	11	@8-19	120
Total	n/a	21	n/a	188

On the secondary system, the user schedules the following:

- A weekly update every Saturday at 9:00 p.m. and keeps 8 of them
- A daily update every Monday through Friday at 8:00 p.m and keeps 60 of them
- An hourly update every hour from 8:00 a.m. to 7:00 p.m. and keeps 120 of them The result in this example is that 188 Snapshot copies are being kept in the SnapVault secondary system volume.

The limit on Snapshot copies per volume is 251, so the 188 Snapshot copies scheduled in this example do not exceed the volume limit.

If you need to retain more than 251 Snapshot copies on the SnapVault secondary system, you can configure additional volumes on the secondary system. Each additional volume can support 251 additional Snapshot copies.

Estimating the initial backup time

The backup time required for the initial transfer of data from the primary storage system to the secondary storage system depends on the inode count of the primary data to be backed up, size of the dataset, and capabilities of the network.

SnapVault can carry out initial backup at an approximate rate of 7 million inodes per hour (110,000 inodes per minute).

Limit on the number of concurrent SnapVault targets

Starting with Data ONTAP 7.3, the maximum possible number of concurrent SnapVault transfers of individual qtrees on a storage system is increased. However, the maximum number of concurrent SnapVault targets has not changed.

Before Data ONTAP 7.3, the maximum number of concurrent SnapVault targets supported by a system was equal to the maximum number of concurrent SnapVault transfers possible for the system.

A SnapVault target is a process that controls the creation of a scheduled SnapVault Snapshot copy on a SnapVault destination volume. For each SnapVault destination volume that has qtrees being updated, there is a SnapVault target.

The maximum number of concurrent SnapVault targets for each platform is described in the following table. At any point in time, no more than the listed number of volumes can have their qtrees updated concurrently. If the number of SnapVault targets exceeds the limit, the excess SnapVault targets are queued and executed after the active SnapVault targets complete their backups.

Model	Maximum number of concurrent SnapVault targets
FAS250	4
FAS270/GF270c	8
920	8
940	16
960	16
980	16
FAS2020	16
FAS2040	64
FAS2050	32
3020	32
3040	64
3050	64

Table 8: Maximum number of concurrent SnapVault targets

Model	Maximum number of concurrent SnapVault targets
3070	128
3140	64
3160	128
3170	128
6030	96
6040	96
6070	128
6080	128
R200	128

Note: These maximum numbers apply only to SnapVault targets, and therefore to SnapVault qtrees. There is no restriction on the number of volumes that can be updated concurrently for SnapMirror qtrees.

Although there is a maximum number of concurrent SnapVault targets, you can configure SnapVault relationships in as many volumes as required. However, at any point in time, only the qtrees in the limited number of volumes can be updated.

If the SnapVault backup throughput is limited by the bandwidth or latency of individual links between multiple primary systems and the secondary system, you can use the enhanced number of concurrent transfer limits available in Data ONTAP 7.3 to get higher overall throughput. You can use the higher limits by spreading the qtrees in the specified number of volumes.

Related references

Maximum number of concurrent replication operations on page 128

Enabling SnapVault

Setting up SnapVault backup on the primary systems means preparing the primary storage systems and SnapVault secondary storage systems to fulfill their backup tasks. In Data ONTAP 7.3 and later, you can install the SnapVault primary and SnapVault secondary licenses on the same storage system. You must license and prepare your storage systems before you can use SnapVault to back up data.

- 1. Enabling licenses for Snap Vault on page 238
- 2. Setting the snapvault.enable option on page 238
- 3. Setting the ndmpd option on page 239
- 4. Setting the snapvault.access option on page 239

Enabling licenses for SnapVault

You need to enable the appropriate licenses on the SnapVault primary and secondary systems. If you are using an active/active configuration, you can enable a SnapVault primary license on one node and a SnapVault secondary license on the other node.

Steps

1. To enter the SnapVault secondary system license code, on the system, enter the following command:

license add xxxxxxx

xxxxxx is the license code you purchased.

This setting persists across reboots.

2. To enter the SnapVault primary system license code, on the system, enter the following command:

license add xxxxxxx

xxxxxx is the license code you purchased.

This setting persists across reboots.

For more information about entering license codes, see the information about licensing in the *Data ONTAP Storage Management Guide*.

Setting the snapvault.enable option

You can set the snapvault.enable option to perform SnapVault data transfers and to create Snapshot copies.

Before you begin

You need to enter the SnapVault secondary system license code on the secondary system, and enter the separate SnapVault primary system license code on the primary system.

Step

1. On both the primary and secondary systems, enter the following command:

options snapvault.enable on

This option persists across reboots.

Setting the ndmpd option

The ndmpd option enables the NDMP service on each primary system to be backed up.

Step

1. To set the ndmpd option, enter the following command:

ndmpd on

This option persists across reboots. To enable and configure NDMP, see the NDMP management chapter in the *Data ONTAP Data Protection Tape Backup and Recovery Guide*.

Setting the snapvault.access option

The snapvault.access option controls which systems can request data transfers. This option persists across reboots.

Steps

1. On the primary system: To set the primary systems to grant access only to the secondary systems, enter the following command:

options snapvault.access host=snapvault_secondary

Note: In the snapvault.access option, upto 255 characters are supported after host=.

Setting this option on the SnapVault primary system determines which secondary system can access data from that primary system.

2. On the secondary system: To allow the primary systems to restore data from the secondary system, enter the following command:

options snapvault.access host=snapvault_primary1, snapvault_primary2,...

Setting this option on the SnapVault secondary system determines which SnapVault primary systems can access the secondary system.

The system must be able to resolve the host name entered as *snapvault_primary* to an IP address in the /etc/hosts file, or else the system needs to be running DNS or NIS. You can also use the literal IP address instead of the host name. The syntax for specifying which systems are allowed access to the secondary system is described in the na_protocolaccess(8) man page. For more information about the options command, see the na_options(1) man page.

The system must be able to resolve the host name entered as *snapvault_secondary* to an IP address in the /etc/hosts file, or else the system needs to be running DNS or NIS. You can also use the literal IP address instead of the host name. For details, see the na_protocolaccess(8) man page. For more information about the options command, see the na_options(1) man page.

Note: To grant access to any requester, enter options snapvault.access all. For security reasons, you should avoid using this option setting.

Example systemB> options snapvault.access host=systemA Example systemA> options snapvault.access host=systemB,systemC

How to start a SnapVault backup relationship

After you have enabled SnapVault on both the primary and secondary storage systems and have given primary and secondary storage systems access to each other, you must specify the qtrees or volumes whose data you want transferred from the primary storage system to the SnapVault secondary storage system. You must then perform a complete (baseline) transfer of data from the primary storage system to secondary storage system.

The snapvault start command configures the SnapVault relationship between primary qtrees or volumes and secondary qtrees by specifying the following:

- The primary storage system qtrees or volumes to be backed up on the storage system qtrees
- The parameters for the updates from the primary storage system qtrees to the secondary storage system qtrees

These parameters include transfer speed and try count. Try count is the number of times SnapVault will attempt to start a transfer before stopping the operation.

• The snapvault start command also initiates the baseline transfer of the primary storage system qtree data to the secondary storage system qtree

Next topics

Guidelines for creating a SnapVault relationship on page 240 Backing up qtree data on page 241 What non-qtree data is on page 241 Backing up non-qtree data on page 242 What volume data backup involves on page 242

Guidelines for creating a SnapVault relationship

You need to follow certain guidelines when creating a SnapVault relationship.

When creating a SnapVault relationship, remember the following guidelines for volumes and qtrees.

- Establish a SnapVault relationship between volumes that have the same vol lang settings.
- After you establish a SnapVault relationship, do not change the language assigned to the destination volume.
- Avoid white space (spaces and tab characters) in names of source and destination qtrees.

- Do not rename volumes or qtrees after establishing a SnapVault relationship.
- The qtree cannot exist on the secondary system before the baseline transfer.

Backing up qtree data

You should run a complete SnapVault transfer, by using the snapvault start command, for the qtrees to be backed up.

Step

1. At the console for the secondary system, enter the following command:

snapvault start -S prim_system:prim_qtree_path sec_host:sec_qtree_path

The –S option specifies the primary system and path. This option must be set the first time the command is run for each primary system qtree you want to copy. It is optional when restarting an initial transfer for a previously configured qtree.

prim_system:prim_qtree_path specifies the qtree on the primary system to be backed up.

sec_host is the name of the destination (secondary system) to which the data from the primary system qtree is transferred. If no secondary system is specified, the local host's name is used.

sec_qtree_path is the path to and includes the name of the qtree on the secondary system.

Note: The qtree specified for *sec_qtree_name* must not exist on the secondary system before you run the snapvault start command.

For information about snapvault start command options, see the na_snapvault(1) man page.

Backing up qtree data

systemB> snapvault start -S systemA:/vol/vol2/qtree3 /vol/vol1/qtree3

SnapVault creates a Snapshot copy of qtree3 in vol2 of the primary system (systemA), copies the data in it to the secondary system (systemB) at vol/vol1/qtree3, and configures the qtree for future updates.

Note: The time required for this baseline transfer is limited by the total amount of primary system data to be backed up and by the inode count. SnapVault can carry out initial backup at an approximate rate of 7 million inodes per hour (110,000 inodes per minute). In the first phase of a large transfer, SnapVault creates inodes, and it might appear that no activity is taking place.

What non-qtree data is

Non-qtree data is any data on a storage system that is not contained in its qtrees.

Non-qtree data can include the following items:

- Configuration and logging directories (for example, /etc or /logs) that are not normally visible to clients
- Directories and files on a volume that has no qtree configured

Backing up non-qtree data

You can replicate and protect non-qtree data in a primary system.

Step

1. From the secondary system, enter the following command:

```
snapvault start -S prim_system:/vol/vol_name/- /vol/vol_name/qtree_name
```

-S prim_system:/vol/vol_name/- specifies the volume on the primary system whose nonqtree data you want to backup.

The dash (-) indicates all non-qtree data in the specified volume.

/vol/vol_name/qtree_name specifies the qtree in the secondary system where you want to store this data.

Note: The qtree that is specified for /vol/vol_name/qtree_name must not exist on the secondary system before you run the snapvault start command.

Note: The non-qtree part of the primary system volume can be replicated only to the SnapVault secondary system. The data can be restored to a qtree on the primary system, but cannot be restored as non-qtree data.

Example

```
systemB> snapvault start -S systemA:/vol/vol1/- /vol/vol4/
non_qtree_data_in_vol7
```

SnapVault transfers the non-qtree data on primary systemA to the qtree called non_qtree_data_in_vol7 in vol4 on systemB (the secondary system). It also configures the qtree for future updates.

What volume data backup involves

When you back up a source volume using SnapVault, the volume is backed up to a qtree on the secondary system; therefore, any qtrees in the source volume become directories in the destination qtree.

Reasons for backing up a volume using SnapVault

- You want to back up a volume that contains many qtrees.
- You want the Snapshot copy management that SnapVault provides.
- You want to consolidate the data from several source volumes on a single destination volume.

Limitations to backing up a volume to a qtree

Before performing a volume-to-qtree backup, consider the following limitations:

- You lose the qtree as a unit of quota management. Quota information from the qtrees in the source volume is not saved when they are replicated as directories in the destination qtree.
- You lose qtree security information. If the qtrees in the source volume had different qtree security styles, those security styles are lost in the replication to the destination qtree and are replaced by the security style of the volume.
- The use of SnapVault for backing up volumes to atrees is not integrated with Protection Manager.
- It is not a simple process to restore data. SnapVault cannot restore the data back to a volume. When restoring data, the original source volume is restored as a qtree. Also, incremental restores are not supported.
- Volume-to-qtree backup is not supported for volumes containing Data ONTAP LUNs.

Next topics

Backing up volume data on page 243 *Restoring a qtree to the original volume structure* on page 244

Related concepts

How to start a SnapVault backup relationship on page 240

Related tasks

Unscheduling Snap Vault Snapshot copies on page 251

Backing up volume data

You can replicate and protect volume data in a primary system using SnapVault.

Step

1. From the secondary system, enter the following command:

snapvault start -S prim_system:/vol/volume_name /vol/volume_name/ gtree_name

-S prim_system:/vol/volume_name specifies the volume on the primary system whose data you want to backup.

/vol/volume_name/qtree_name specifies the qtree in the secondary system where you want to store this data.

Note: The qtree that is specified for /vol/volume_name/qtree_name must not exist on the secondary system before you run the snapvault start command.

Example

systemB> snapvault start -S systemA:/vol/vol1 /vol/vol4/vol1_copy

SnapVault transfers the data in vol1 on primary systemA to the qtree called vol1_copy in vol4 on systemB (the secondary system). It also configures the qtree for future updates.

Restoring a qtree to the original volume structure

You can use the snapvault restore command so that the source volume you backed up to a qtree is restored as a qtree on the primary system.

Steps

1. To restore the backed-up qtree to the original volume structure with multiple qtrees on the primary system, re-create all of the qtrees in the volume on the primary system by using the qtree create command.

pri_system> qtree create /vol/projs/project_x

2. Restore the data for each qtree using the ndmpcopy command. The following command restores data from the backed-up project_x directory on the secondary system to the re-created project_x qtree on the primary system.

pri_system> ndmpcopy -sa username:password sec_system:/vol/vol1/projs/ project_x /vol/projs/project_x

For more information about the ndmpcopy command, see the *Data ONTAP Data Protection Tape Backup and Recovery Guide*.

3. Stop qtree updates and remove the qtree on the secondary system by using the snapvault stop command. The following command removes the projs qtree from the secondary system:

sec_system> snapvault stop /vol/vol1/projs

4. Reinitialize a baseline copy of each qtree to the secondary system by using the snapvault start command. The following command reinitializes the SnapVault backup:

sec_system> snapvault start -S pri_system:/vol/projs /vol/vol1/projs

What SnapVault Snapshot copy update schedules are

After you have completed the initial baseline backup of qtrees on the primary system to qtrees on the SnapVault secondary system, you must use the snapvault snap sched command to schedule a set of Snapshot copies on the SnapVault primary system. You can also specify the volume to create Snapshot copies for, the Snapshot copy basename, how many versions of the Snapshot copies to retain, and the days and hours to create this set of Snapshot copies.

You must use the snapvault snap sched command to schedule the following tasks:

- Regular SnapVault Snapshot copy times of volumes on the primary system to capture new and changed data in the qtrees that have a SnapVault relationship configured through the snapvault start command
- Regular transport of new or modified data in the primary qtrees to their associated secondary qtrees on the SnapVault secondary system
- Regular Snapshot copies of the volume containing the updated secondary qtrees for archiving by SnapVault

Note: For descriptions and procedures pertaining to SnapVault backup of open systems drives and directories, see the Open Systems SnapVault documentation.

Next topics

How to avoid Snapshot copy schedule conflicts on page 245 Scheduling Snapshot copies on the SnapVault primary system on page 245 Scheduling Snapshot copy backups to the SnapVault secondary system on page 247 Scheduling Snapshot copies on the secondary system for archiving on page 248 Displaying the currently configured Snapshot copy schedule on page 249 Retaining older SnapVault Snapshot copies on SnapVault secondary volumes on page 249

How to avoid Snapshot copy schedule conflicts

If SnapVault is scheduled to perform Snapshot copy management at the same time as default snap sched activity, then the Snapshot copy management operations scheduled using the snap sched command might fail with syslog messages, Skipping creation of hourly snapshot, and Snapshot already exists.

To avoid this condition, you should disable the conflicting times using snap sched, and use the snapvault snap sched command to configure equivalent schedules to create Snapshot copies.

Note: You can disable the snap sched schedule and only use the snapvault snap sched command to create Snapshot copies. Therefore, to track the schedule for creating Snapshot copies, look at the snapvault snap sched output, and not the snap sched output.

Scheduling Snapshot copies on the SnapVault primary system

To schedule a set of Snapshot copies on the SnapVault primary system, you can use the snapvault snap sched command to specify the volume to create Snapshot copies for, the Snapshot copy basename, how many versions of the Snapshot copies to retain, and the schedule to execute this set of Snapshot copies.

Step

1. To set a schedule for the primary system, from the primary system, enter the following command:

snapvault snap sched vol_name snap_name schedule_spec

vol_name is the name of the volume on the primary system on which to create this set of Snapshot copies.

snap_name is the basename of a Snapshot copy set, for example, sv_nightly. The name of this Snapshot copy must be the same on the primary and secondary systems. The snap_name must not be hourly, nightly, or weekly to avoid conflict with snap_sched Snapshot copies.

schedule_spec is made up of count[@day_list][@hour_list].

count is the number of Snapshot copies to retain for this Snapshot copy set. A zero (0) in this field means no new instance of this Snapshot copy will be created.

Attention: The combined total of Snapshot copies retained cannot exceed 251 Snapshot copies per volume. If it does, SnapVault will not create new Snapshot copies.

@day_list is a comma-separated list that specifies the days on which a new Snapshot copy for this set is created. Valid entries are mon tue wed thu fri sat sun. They are not case-sensitive. You can specify a range using a dash (-), for example, mon_fri. The dash (-) by itself means no Snapshot copy will be created automatically. You can create the Snapshot copy manually. The default value is mon_sun.

@hour_list specifies the hours at which a new Snapshot copy is created for this set. Valid entries are whole numbers from 0 to 23. You can specify a range using a dash (-), or use a comma-separated list, for example, 7, 8–17, 19, 21, 23. The default is midnight (0).

Scheduling Snapshot copies on the SnapVault primary system

The following three snapvault snap sched command lines schedule three sets of SnapVault Snapshot copies on volume vol1 of the primary system:

systemB> snapvault snap sched vol1 sv_weekly 1@sat@19

systemB> snapvault snap sched voll sv_nightly 2@mon-fri@19

systemB> snapvault snap sched vol1 sv_hourly 11@mon-fri@7-18

Result: SnapVault primary system creates Snapshot copies on the specified volume, as follows:

- SnapVault creates *sv_weekly*. 0 every Saturday at 7:00 p.m., and keeps one copy.
- SnapVault creates *sv_nightly.0* every Monday through Friday at 7:00 p.m., and keeps two copies.
- SnapVault creates *sv_hourly.0* every Monday through Friday, every hour from 7:00 a.m. to 6:00 p.m., and keeps eleven copies.

Scheduling Snapshot copy backups to the SnapVault secondary system

You can schedule the backup of SnapVault Snapshot copies from the primary systems to the secondary system, by using the snapvault snap sched -x command.

Step

1. From the secondary system, enter the following command on a single line:

snapvault snap sched -x sec_vol snap_name schedule_spec

The -x portion of the command is required on the secondary system. This parameter specifies that the SnapVault secondary qtrees on the specified volume are updated from their associated primary system qtrees just before the new Snapshot copy of the specified volume is created.

sec_vol is the name of the volume on the secondary system for which this Snapshot copy is scheduled.

snap_name is the basename of the set of Snapshot copies to create, for example, sv_nightly. The basename of this Snapshot set must match the basename of the corresponding Snapshot copy set configured on the primary system volume. Each new Snapshot copy created for this set is numbered 0, the number of each previous Snapshot copy in this set is increased by 1, and the oldest Snapshot copy in this set is deleted. The snap_name must not be hourly, nightly, or weekly to avoid conflict with regular Data ONTAP snap_sched Snapshot copies.

schedule_spec is made up of count[@day_list][@hour_list].

• *count* is the number of Snapshot copies to retain for this set. A zero (0) in this field means no new secondary system Snapshot copy will be created for this set, although the qtrees on the secondary system will be updated by the transfers from the primary systems.

Attention: The combined total of Snapshot copies retained for this and other Snapshot sets cannot exceed 251 Snapshot copies per volume. If it does, SnapVault will not create new Snapshot copies.

- @day_list is a comma-separated list that specifies the days on which a new Snapshot copy is created for this set. Valid entries are mon, tue, wed, thu, fri, sat, sun. They are not case-sensitive. You can specify a range using a dash (-), for example, mon-sun. The dash (-) by itself means no Snapshot copy will be created automatically. You can create the Snapshot copy manually.
- @hour_list specifies the hours at which a new Snapshot copy is created for this set. Valid entries are whole numbers from 0 to 23. You can specify a range using a dash (-), or use a comma-separated list, for example, 6, 8-17, 19, 21, 23. The default is midnight (0).

Note: SnapVault transfers scheduled on the secondary system with the snapvault snap sched -x command are started five minutes after the hour you specify, to give SnapVault on the primary systems enough time to create Snapshot copies before the secondary system starts the update.

Note: You can turn off the SnapVault Snapshot copy schedule on the primary or secondary system at any time with the snapvault snap unsched command. You can also use the options snapvault.enable off to stop all SnapVault transfers.

Scheduling Snapshot copy backups to the SnapVault secondary system

The following three snapvault snap sched command lines schedule three sets of SnapVault updates and Snapshot copies on volume vol1 of the secondary systems.

```
systemA> snapvault snap sched -x vol1 sv_weekly 5@sat@21
```

```
systemA> snapvault snap sched -x vol1 sv_nightly 5@mon-fri@20
```

```
systemA> snapvault snap sched -x voll sv_hourly 4@mon-fri@8-19
```

Result: SnapVault transfers qtree data from the primary systems Snapshot copy as follows:

- SnapVault transfers *sv_weekly*. 0 to the secondary storage system every Saturday at 9:00 p.m., makes a new Snapshot copy with the same name containing all the transferred data, and keeps five copies.
- SnapVault transfers *sv_nightly.0* to the secondary system every Monday through Friday at 8:00 p.m., makes a new Snapshot copy with the same name containing all the transferred data, and keeps five copies.
- SnapVault transfers *sv_hourly.0* to the secondary storage system every hour from 8:00 a.m. to 7:00 p.m., Monday through Friday, makes a new Snapshot copy with the same name containing all the transferred data, and keeps four copies.

Scheduling Snapshot copies on the secondary system for archiving

You might want to schedule some Snapshot copies on the secondary storage system that do not require a transfer from the primary storage system. For example, you might want to maintain hourly and nightly Snapshot copies on the primary storage system and you might want to keep only weekly Snapshot copies on the secondary storage system.

Step

1. To schedule Snapshot copies on the secondary storage system without having to create a corresponding Snapshot copy on the primary storage system, enter the following command: snapvault snap sched sec_vol snap_name schedule_spec

Note: The snapvault snap sched command is used because it waits for any active SnapVault transfers to finish before creating the Snapshot copy.

Example

The following command schedules a weekly Snapshot copy at 11 p.m. on Saturdays and keeps the last five Snapshot copies:

snapvault snap sched vol2 sv_weekly 5@sat@22

Displaying the currently configured Snapshot copy schedule

You can use the snapvault snap sched command without a schedule specification to show the current schedule values.

Step

1. Enter the following command.

snapvault snap sched

Note: If no *snap_name* is given, the command displays the basenames of all Snapshot copy sets scheduled for the specified volume. If no *volume_name* is given, the command shows the basenames of all Snapshot copy sets for all SnapVault volumes.

Displaying the SnapVault Snapshot copy schedule

A sample output of the snapvault snap sched command is given in the following example.

```
systemA> snapvault snap sched
```

xfer voll sv_weekly 5@sat@21 xfer voll sv_nightly 5@mon-fri@20 xfer voll sv_hourly 4@mon-fri@8-19 xfer vol2 sv_nightly 5@mon-fri@20 xfer vol2 sv_hourly 4@mon-fri@8-19

Retaining older SnapVault Snapshot copies on SnapVault secondary volumes

In Data ONTAP 7.3.2 and later, you can retain older SnapVault Snapshot copies on SnapVault secondary volumes.

About this task

• In Data ONTAP 7.3.1 and earlier, when the maximum number of Snapshot copies is reached, SnapVault deletes older Snapshot copies of the specified SnapVault Snapshot copy schedule to make space available for new Snapshot copies.

For example, if you want to preserve 250 nightly Snapshot copies for five years. However, when the number of SnapVault scheduled Snapshot copies reaches 250, SnapVault automatically deletes the older Snapshot copies to make space available for the new Snapshot copy. This behavior is not what you want. You want to preserve all the Snapshot copies and a message to indicate that you have reached the limit of Snapshot copies for a specified SnapVault Snapshot copy schedule.

• In Data ONTAP 7.3.2 and later, you can enable the preservesnap or preserve option to retain all Snapshot copies on the SnapVault secondary volumes.

When you enable this option, SnapVault does not delete an older Snapshot copy to create a new Snapshot copy when the retention limit is reached. In such a case, you can manually delete some Snapshot copies to make space available for new Snapshot copies.

Note: Once the retention limit for the SnapVault Snapshot copies is reached, further SnapVault updates for the volume will not take place.

You can also clone the volume to create new Snapshot copies on this new volume while preserving the older Snapshot copies intact on the old base volume.

Step

1. You can prevent automatic deletion of Snapshot copies at the backup schedule level or at the global level in a storage system.

If you want to	Then enter the following command
Prevent deletion of the Snapshot copies at the backup schedule level	snapvault snap sched -x -o preserve=on,warn=warn_count vol_name snap_name n@-
	 -x—specifies the transfer of new data from all primary paths before creating the Snapshot copy. -o—sets user-configurable options for this Snapshot copy schedule. preserve—can take one of the following values.
	 on—Create and retain the SnapVault archive Snapshot copies until the retention limit is reached. Thereafter, SnapVault does not create archive Snapshot copies. off—Automatically delete the oldest archived Snapshot copy to create a new Snapshot copy.
	 default—The preserve setting depends on the global snapvault.preservesnap setting.
	 warn—sends out the warning EMS and SNMP alerts when the remaining number of SnapVault scheduled Snapshot copies for a target reaches the warn_count limit set. This option can have a value from 0 to n - 1. n@- is the number of Snapshot copies to retain along with the list of hours or days.
	Note: During an upgrade from an earlier Data ONTAP release, the preserve option at the individual backup schedule level is set to default and warn to 0. During a revert operation, all the unknown options are ignored and cleaned up from the registry.
	For information about snapvault sched command options, see the na_snapvault(1) man page.

If you want to	Then enter the following command
Prevent auto deletion ofoptions snapvault.preservesnap onSnapVault Snapshot copies at the global level in a systemon—After reaching the retention limit, SnapVault d oldest Snapshot copies to make space available for r SnapVault aborts the creation of new Snapshot copie according to this global option setting when the pre- or is set to default for a given SnapVault schedul off—SnapVault executes according to the option s schedule.Note: A backup schedule-level setting of the pre- precedence over the global snapvault.preset	options snapvault.preservesnap on on—After reaching the retention limit, SnapVault does not automatically delete the oldest Snapshot copies to make space available for new Snapshot copies. Instead, SnapVault aborts the creation of new Snapshot copies. SnapVault executes according to this global option setting when the preserve option is not specified or is set to default for a given SnapVault schedule.
	off—SnapVault executes according to the option set at the level of backup schedule.
	Note: A backup schedule-level setting of the preserve option takes precedence over the global snapvault.preservesnap setting.

Retaining SnapVault Snapshot copies on the SnapVault secondary system

In this example, you want to create and retain up to 250 Snapshot copies, after which the system does not create any backup Snapshot copies on the specified SnapVault Snapshot copy schedule. You want the system to issue a warning message when there is room for fewer than 10 Snapshot copies before the retention limit is reached (that is, 241 or more Snapshot copies have already been created). You would enter the following command:

snapvault snap sched -o preserve=on,warn=10 vol1 sv_nightly 250@-

Unscheduling SnapVault Snapshot copies

You can unschedule a set of SnapVault Snapshot copies if the data in the qtrees you are backing up has been migrated to another location or is no longer useful.

Step

1. To turn off the SnapVault schedule for a set of Snapshot copies and stop the Snapshot copy process for the SnapVault primary system or secondary system, enter the following command at the console of the primary or secondary system:

```
snapvault snap unsched [-f] [volume [snap_name]]
```

-f forces the command to run without showing the list of Snapshot copies to stop creating and without asking for confirmation.

volume is the name of the volume to stop creating Snapshot copies on.

snap_name is the basename of the Snapshot copy set to stop creating.

If no value for *snap_name* is provided, SnapVault turns off the SnapVault Snapshot copy schedule for all Snapshot copy sets in the volume and does not update any more SnapVault Snapshot copies in the volume. If no volume is provided, SnapVault deletes the schedules for all SnapVault Snapshot copy sets and does not update any more

SnapVault Snapshot copies in the system. If there is already a Snapshot copy being created in the volume, the command fails.

Example

systemB> snapvault snap unsched vol1 sv_nightly

Unless you used the -f option, SnapVault asks for confirmation. If you confirm the action, SnapVault unschedules all SnapVault Snapshot copies with the basename sv_nightly on voll of systemB.

Next topics

Disabling Snapshot copies temporarily without unscheduling on page 252 *Enabling Snapshot copies that are temporarily disabled* on page 252

Disabling Snapshot copies temporarily without unscheduling

You can temporarily disable Snapshot copies without having to unschedule using the snap sched command and the tries option.

Step

1. To disable Snapshot copies temporarily, enter the following command:

snapvault snap sched -o tries=0 volname snapname sched_spec

Data ONTAP never takes a Snapshot copy because the tries option is set to 0.

Enabling Snapshot copies that are temporarily disabled

You can enable Snapshot copies that are temporarily disabled.

Step

1. Enter the following command:

```
snapvault snap sched -o [tries=option] vol_name snap_name sched_spec
```

tries specifies the number of times SnapVault should try to create a Snapshot copy. You can specify a value between 1 and 120, or unlimited. The default is unlimited. For more information, see the na_snapvault(1) man page.
Checking SnapVault transfers

To ensure SnapVault transfers are taking place as expected, you can check the transfer status using the snapvault status command.

Step

1. To check the status of a data transfer and see how recently a qtree has been updated, enter the following command:

```
snapvault status [-1|-s|-c|-t] [[[system_name:]qtree_path] ...]
```

- -1 displays the long format of the output, which contains more detailed information.
- -s displays the SnapVault Snapshot copy basename, status, and schedule for each volume.
- -c displays the configuration parameters of all SnapVault qtrees on the system. This option can be run only from the secondary system.
- -t displays the relationships that are active.

Note: A relationship is considered *active* if the source or destination is involved in any one of the following activities: transferring data to or from the network, reading or writing to a tape device, waiting for a tape change, or performing local on-disk processing or clean-up.

system_name is the name of the system for which you want to see the status of SnapVault operations.

qtree_path is the path of the qtree or qtrees for which you want to see the status of SnapVault operations. You can specify more than one qtree path.

The system displays a message showing whether a transfer is in progress, how much data has been transferred, the state of the destination, and how long ago the last successful transfer took place.

- If [system_name:]qtree_path arguments are specified, then status is displayed only for the specified qtrees.
- If the -1 option is given, the output includes the more detailed information shown in Example 2.
- If the -s option is given, the output displays Snapshot copy creation status, as shown in Example 3.
- If the -c option is given, the output displays the parameter settings for the primary system configuration, as shown in Example 4.
- If the -t option is given, the output displays the list of relationships that have active transfers as shown in Example 5.

Data ONTAP allows you to set a maximum rate for transfers coming into a system and for transfers going out of a system.

Next topics

Examples for checking the status on page 254 *What the status fields mean* on page 256

Related tasks

Setting a maximum transfer rate for all transfers on page 164

Examples for checking the status

The following examples help you check the data transfer status pertaining to SnapVault backup of storage systems running Data ONTAP only.

Example 1

If you enter snapvault status with no option, you see SnapVault qtree relationships involved in the most recent transfer from a primary qtree to a secondary qtree:

```
systemA> snapvault status
Snapvault primary is ON.
Source Destination State
Lag Status
systemB:/vol/vol2/qtree3 systemA:/vol/sv_vol/qtree3 Snapvaulted 00:48:24 Idle
```

Note: If the SnapVault primary and secondary licenses are installed on the same storage system, the following message is displayed:

Snapvault is ON.

Example 2

The snapvault status -1 command displays more detailed information on the most recent SnapVault transfer and Snapshot copy activity:

```
systemA> snapvault status -1
```

SnapVault primary is ON	
Source:	systemA:/vol/vol2/qtree3
Destination	systemB:/vol/sv_vol/qtree3
Status	Idle
Progress:	-
State:	SnapVaulted
Lag:	2:09:58
SnapVault Timestamp:	Thu Jan 31 12:30:01 PST 2002
Base Snapshot:	systemB(0016778780)_sv_vol.59
Current Transfer Type	
Current Transfer Error:	
Contents:	Replica
Last Transfer Type:	Scheduled

```
Last Transfer Size: 1680 KB
Last Transfer Duration: 00:02:08-
```

Example 3

The snapvault status -s command lists all the Snapshot copies scheduled on the primary or secondary storage system. Information includes volume, Snapshot copy basename, current status, and Snapshot schedule:

Example 4

systemA> snapvault status -c

The snapvault status -c command lists all the secondary system qtrees, their corresponding primary system qtrees, maximum speed of scheduled transfers, and maximum number of times SnapVault attempts to start a scheduled transfer before skipping that transfer:

```
/vol/sv_vol/db_qtree1 source=systemB:/vol/db1/s1 kbs=unlimited tries=20
/vol/sv_vol/db_qtree2 source=systemC:/vol/db2/s2 kbs=unlimited tries=20
/vol/sv_vol/qtree1 source=systemC:/vol/users/qtree1 kbs=10000 tries=10
/vol/sv_vol/qtree2 source=systemC:/vol/users/qtree2 kbs=10000 tries=10
/vol/sv_vol/qtree3 source=systemD:/vol/users/qtree1 kbs=10000 tries=10
/vol/sv_vol/qtree4 source=systemD:/vol/db/db_1 kbs=7000 tries=10
systemA> snapvault status -c /vol/sv_vol/qtree3
/vol/sv_vol/qtree3 source=systemA:/vol/vol2/qtree3 kbs=10000 tries=10
```

Example 5

The snapvault status -t command lists all the SnapVault relationships that have active transfers:

```
systemA> snapvault status -t
Snapvault secondary is ON.
Source Destination State Lag
Status
systemB:/vol/vol2/qtree3 systemA:/vol/sv_vol/qtree3 Snapvaulted 00:48:24
Transferring
```

If there are no active transfers, the following message appears:

```
systemA> snapvault status -t
Snapvault secondary is ON.
There are no active transfers.
```

What the status fields mean

The information fields that SnapVault can display for the snapvault status and snapvault status -l commands are as follows.

Field	Possible values that might be displayed
Source	<i>system:qtree_path</i> —The source is the primary storage system and qtree path listed.
Destination	<i>system:qtree_path</i> —The destination is the secondary storage system and qtree path listed.
State	Uninitialized—The destination SnapVault secondary storage qtree is not yet initialized through the snapvault start command.
	Snapvaulted—The qtree is a SnapVault secondary destination.
	Unknown—It is not known what state the secondary storage system qtree is in; the volume that contains the secondary storage system qtree could be offline or restricted.
	Source—This state is reported when the snapvault status command is run on the primary storage system. It also appears if snapvault status is run on secondary storage systems after the snapvault restore command was run on an associated primary storage system.
Lag	<i>hh:mm:ss</i> indicates the time difference between the data currently on the primary storage system and the latest data stored on the secondary storage systemstorage system; that is, the difference between the current time and the timestamp of the Snapshot copy last successfully transferred to the secondary storage system.
	A dash (-) in this field means that the secondary storage system is not initialized.

Field	Possible values that might be displayed
Status	Aborting—A transfer is being aborted and cleaned up.
	Idle—No data is being transferred.
	Pending—The secondary storage system cannot be updated because of a resource issue; the transfer is retried automatically.
	Quiescing—The specified qtree is waiting for all existing transfers to complete. The destination is being brought into a stable state.
	Resyncing—The specified qtree is resynchronizing.
	Transferring—Transfer has been initiated, but has not yet started, or is just finishing.
Progress	Shows the number of KB transferred by the current transfer, or the restart check point if the status is Idle or Pending.
Mirror Timestamp	<i>hh:mm:ss</i> indicates the timestamp of the last Snapshot copy successfully transferred from the primary storage system to the secondary storage system.
	Note: A resynchronization (snapvault start -r) might change the base Snapshot copy to a Snapshot copy with a timestamp older than the original base.
Base Snapshot copy	The name of the base Snapshot copy for the corresponding qtree on the secondary storage system.
	For qtrees in a SnapVault relationship, the secondary storage side lists the name of the exported Snapshot copy for that qtree on the storage side. A resynchronization (snapvault start -r) might change the name of the base Snapshot copy.
Current Transfer Type	Indicates the type of the current transfer: scheduled, retry, resync, update, initialize, store, or retrieve. This field applies only to the destination side.
Current Transfer Error	Displays an error message if the latest transfer attempt failed.
Contents	Indicates whether the contents of the destination volume or qtree in the active file system are up-to-date replicas or are in transition. The field applies only to the destination side. Since a destination is read-only, its contents are always a replica.
Last Transfer Type	Indicates the type of the previous transfer: scheduled, retry, resync, update, initialize, store, or retrieve. This field applies only to the secondary storage side.
Last Transfer Size	Shows the number of KB transferred in the last successful transfer.

Field	Possible values that might be displayed
Last Transfer Duration	Shows the elapsed time for the last successful transfer. If the transfer failed and restarted, the time includes time waiting to restart the transfer. If a transfer aborted and was retried from the beginning, it includes only the time required for the final successful attempt.
Last Transfer From	This field applies only to the secondary storage side and shows the name of the primary system and volume or qtree (where the content is transferred from).

Note: If snapvault status displays a negative lag time, that means the clock on the destination storage system is ahead of the clock on the source storage system. The solution is to synchronize the clocks on both the storage systems.

Displaying SnapVault Snapshot copies

You can use the snap list command to display a list of Snapshot copies to confirm what versions of your primary qtree data have been backed up, or to locate by date or time a particular version of a qtree to retrieve.

Using the snap list -q command, you can see the following:

- A list of all Snapshot copies on the secondary storage system (not just SnapVault Snapshot copies)
- The qtrees in the Snapshot copies
- The primary storage system sources of those qtrees
- The timestamp of the primary storage system Snapshot copy that was the source for the data in the secondary storage system Snapshot copy

Using the snap list -o command, you can also list the Snapshot copy timestamps, primary qtree origin (if applicable), and Snapshot copy names stored for an individual qtree.

Note: The descriptions and procedures pertain to SnapVault backup of storage systems running Data ONTAP only. For descriptions and procedures pertaining to SnapVault backup of open systems drives and directories, see the Open Systems SnapVault documentation.

Next topics

Displaying SnapVault Snapshot copies on a volume on page 259 *Listing Snapshot copies for qtrees* on page 261

Displaying SnapVault Snapshot copies on a volume

You can display a list of the Snapshot copies and qtrees on your volumes, by using the snap list command.

Step

1. On the system for which you want to see the Snapshot copy information, enter the following command:

snap list -q [vol_name]

vol_name is the name of the volume for which you want to list the Snapshot copies.

If no volume name is given, the Snapshot copies on all this system's volumes are displayed.

Note: If the deduplication feature is enabled on the SnapVault secondary volume, a deduplication operation is run on the volume after the SnapVault target snapshot has been created. This operation eliminates duplicate data blocks from the volume. After the completion of the deduplication operation, the SnapVault target Snapshot copy that was created earlier is deleted and a new Snapshot copy with the same name is created.

For more information about deduplication, see the Data ONTAP Storage Management Guide.

Displaying SnapVault Snapshot copies on a volume

Primary storage output example

If you specify the primary volume name, the command lists the information for each Snapshot copy in the volume. This output is from a volume used as a SnapVault primary system:

```
systemA> snap list -q vol2
Volume vol2
working...
                                                 contents timestamp source
       qtree
        ____
sv_hourly.0 (Jan 22 20:00)

        Original
        Jan 22 20:00 -

        Original
        Jan 22 20:00 -

        Original
        Jan 22 20:00 -

       qtreel
       qtree2
       qtreeZ
sv_hourly.1 (Jan 22 16:00)

        Original
        Jan 22 16:00 -

        Original
        Jan 22 16:00 -

        Original
        Jan 22 16:00 -

      qtreel
      qtree2
      qtreeZ
sv_hourly.2 (Jan 22 12:00)

        Original
        Jan 22 12:00 -

        Original
        Jan 22 12:00 -

        Original
        Jan 22 12:00 -

       qtreel
      qtree2
      qtreeZ
sv_hourly.3 (Jan 22 08:00)

        Original
        Jan 22 08:00 -

        Original
        Jan 22 08:00 -

        Original
        Jan 22 08:00 -

      qtree1
       gtree2
       qtreeZ
sv_nightly.0 (Jan 22 00:00)
```

```
Original
Original
Original
                                                            Jan 22 00:00 -
     qtreel
                                                            Jan 22 00:00
     qtree2
                                                            Jan 22 00:00 -
     qtreeZ
                  Original Jan 21 20:00 -
Original Jan 21 20:00 -
qtreeZ Original
sv_hourly.4 (Jan 21 20:00)
     qtree1
     qtree2
                                          Original Jan 21 20:00 -
sv_hourly.5 (Jan 21 16:00)

        Original
        Jan 21 16:00 -

        Original
        Jan 21 16:00 -

        Original
        Jan 21 16:00 -

        Original
        Jan 21 16:00 -

     qtreel
     gtree2
     qtreeZ
sv_nightly.1 (Jan 21 00:00)
                               Original Jan 21 00:00 -
Original Jan 21 00:00 -
Original Jan 21 00:00 -
     qtreel
     qtree2
     qtreeZ
```

This output displays which qtrees were writable and therefore have original content (the timestamp in these cases is the same as for the Snapshot copy as a whole). It also displays whether any qtrees were transitioning and are therefore neither a faithful replica nor original content. Instead of a timestamp, transitioning qtrees are shown with a dash (-).

Secondary storage output example

If you specify the volume name (in this example, sv_vol) and are running the command from a system used as a SnapVault secondary system, you see a list of all the SnapVault Snapshot copies retained on volume sv_vol and the details of the qtrees contained in those Snapshot copies:

```
systemB> snap list -q sv_vol
Volume sv_vol
working...
qtree contents date source
             _____
                         _____
                                         _____
sv_hourly.0 (Jan 31 20:00)
  qtree1ReplicaJan 22 20:40systemA:/vol/vol2/qtree1qtree2ReplicaJan 22 20:40systemA:/vol/vol2/qtree2qtreeZReplicaJan 22 20:40systemA:/vol/vol2/qtreeZ
sv_hourly.1 (Jan 22 16:00)
  gtreel Replica Jan 22 16:00 systemA:/vol/vol2/gtreel
   qtree2 Replica Jan 22 16:00 systemA:/vol/vol2/qtree2
qtreeZ Replica Jan 22 16:00 systemA:/vol/vol2/qtreeZ
sv_hourly.2 (Jan 22 12:00)
   qtree1 Replica Jan 22 12:00 systemA:/vol/vol2/qtree1
   qtree2 Replica Jan 22 12:00 systemA:/vol/vol2/qtree2
   qtreeZ Replica Jan 22 12:00 systemA:/vol/vol2/qtreeZ
. . . . .
```

This output displays which qtrees are replicas of another qtree, and the timestamp of the source Snapshot copy.

Note: Qtrees that are transitioning appear with a dash (-) instead of a timestamp. In general, you should not attempt to restore Snapshot copy versions of qtrees that are transitioning.

If you specify no arguments, the output displays the information for each Snapshot copy in each volume.

Listing Snapshot copies for qtrees

You use the snap list command to see a list of Snapshot copies associated with a qtree and, if applicable, the Snapshot copies' primary qtree origins.

Step

1. On the system for which you want to see the information, enter the following command:

```
snap list -o [qtree_path]
```

qtree_path displays one qtree. If no qtree name is given, information about all the qtree names on the volume is displayed.

Sample snap list -o output

If you specify the -o parameter with a qtree path, the snap list output includes the dates, sources (if any), and names of associated SnapVault Snapshot copies that are retained on the system, for example:

```
systemB> snap list -o /vol/sv_vol/qtree3
working...
Qtree /vol/sv_vol/qtree3
date
      source
                                           name
_____
              _____
Jan 31 18:00 systemA:/vol/vol2/qtree3 hourly.0
Jan 31 17:00 systemA:/vol/vol2/qtree3 hourly.1
Jan 31 16:00 systemA:/vol/vol2/qtree3 hourly.2
Jan 31 15:00 systemA:/vol/vol2/qtree3 hourly.3
Jan 30 14:00 systemA:/vol/vol2/qtree3 hourly.4
Jan 30 13:00 systemA:/vol/vol2/qtree3 hourly.5
Jan 30 12:00 systemA:/vol/vol2/gtree3 hourly.6
Jan 30 11:00 systemA:/vol/vol2/gtree3 hourly.7
Jan 31 10:00 systemA:/vol/vol2/qtree3 hourly.8
Jan 31 9:00 systemA:/vol/vol2/qtree3 hourly.9
Jan 31 8:00 systemA:/vol/vol2/qtree3 hourly.10
Jan 30 20:00 systemA:/vol/vol2/qtree3 nightly.0
Jan 29 20:00 systemA:/vol/vol2/gtree3 nightly.1
Jan 26 16:00 systemA:/vol/vol2/qtree3 weekly.0
```

Guidelines for creating a clone of a SnapVault destination volume

When creating a clone of a SnapVault destination volume, you should make sure that no SnapVault transfers are active for that volume.

You can check the status of SnapVault transfers by using the snapvault status -l command. If you find that the status is idle, you can proceed with the creation of the volume clone. The vol

clone create command for a SnapVault destination volume creates a writable clone if the SnapVault transfer status is idle.

If a clone is not created, wait till the status becomes idle and then create the clone. The status should be idle between scheduled SnapVault transfers.

If the clone is already created and is in the read-only state, then convert the relationship into a SnapMirror relationship by using the snapvault convert command. You should quiesce the relationship using the snapmirror quiesce command, and then break the relationship.

About LUN clones and SnapVault

A LUN clone is a space-efficient copy of another LUN. Initially, the LUN clone and its parent share the same storage space. More storage space is consumed only when one LUN or the other changes.

In releases prior to Data ONTAP 7.3, SnapVault considers each LUN clone as a new LUN. Therefore, during the initial transfer of the LUN clone, all data from the clone and the backing LUN is transferred to the secondary system.

Note: LUNs in this context refer to the LUNs that Data ONTAP serves to clients, not to the array LUNs used for storage on a storage array.

For descriptions of data backup and restore on volumes containing LUNs, see the *Data ONTAP Block Access Management Guide for iSCSI and FC.*

Starting with Data ONTAP 7.3, SnapVault can transfer LUN clones in an optimized way by using SnapDrive for Windows. To manage this process, SnapDrive for Windows creates two Snapshot copies:

- Backing Snapshot copy, which contains the LUN to be cloned
- · Backup Snapshot copy, which contains both the LUN and the clone

Modes of transfer

Starting with Data ONTAP 7.3, a SnapVault transfer with LUN clones can run in two modes:

- In non-optimized mode, a LUN clone is replicated as a LUN. Therefore, a LUN clone and its backing LUN get replicated as two separate LUNs on the destination. SnapVault does not preserve space savings that come from LUN clones.
- In optimized mode, a LUN clone is replicated as a LUN clone on the destination. Transfers of LUN clones to the secondary system in optimized mode are possible only with SnapDrive for Windows.

These modes apply to newly created LUN clones. On successive update transfers, only the incremental changes are transferred to the destination in both modes.

Note: A single relationship must either be optimized or non-optimized. Switching between the two modes is not allowed.

Next topics

LUN clone transfer in non-optimized mode on page 263 LUN clones transfer in optimized mode using SnapDrive for Windows on page 263

LUN clone transfer in non-optimized mode

LUN clones are typically replicated as complete LUNs on the destination system, because nonoptimized transfers are used by default.

Consider the following points before transferring the data in non-optimized mode:

- The backing Snapshot copy must be present on the source system.
- There is no space saving on the destination. The entire LUN clone, and all of the data from the backing LUN, is replicated to the destination.
- The destination qtree should not have LUN clones. If there are LUN clones on the destination qtree, the transfer fails. Before you perform the transfer again, you must delete the LUN clones.

Data restoration for qtrees with LUN clones

If the source qtree has LUN clones, SnapVault does not support in-place restores.

To recover data from the destination qtree using SnapVault, you can use one of the following options for a qtree with LUN clones.

• Delete the LUN clones within the source qtree, and then perform an in-place restore, using the snapvault restore command.

Note: If you attempt an in-place restore for a qtree with LUN clones, the system displays the following error message.

Qtree has lun clones

• Restore the data to a new qtree, by using the snapvault restore command.

Attention: For a qtree with LUN clones, ensure that the volume has enough free space to store the LUN clones as complete LUNs before you initiate data recovery using SnapVault.

LUN clones transfer in optimized mode using SnapDrive for Windows

LUN clones can be replicated in optimized mode when you use SnapDrive for Windows to manage the SnapVault relationship. You need to keep certain conditions in mind when transferring LUN clones.

Consider the following requirements before using SnapVault with LUN clones in optimized mode:

• SnapDrive for Windows only transfers optimized LUN clones supporting Microsoft Windows Volume Shadow Copy Services for TxF Recovery. Volume Shadow Copy Services helps create application-consistent Snapshot copies. SnapDrive for Windows automatically creates these Snapshot copies if the backup application requests TxF Recovery data during Snapshot creation.

For more information on Volume Shadow Copy Services and TxF Recovery data, see the Microsoft documentation.

- SnapVault creates LUN clones on the secondary system if both the primary and secondary storage systems are running Data ONTAP 7.3 or later, and optimized transfers occur using SnapDrive for Windows.
- After the relationship has been handed off to SnapDrive for Windows, the relationship must be left there.

Attention: Do not run manual updates from the CLI. This might cause data corruption.

- When creating a SnapVault relationship, ensure that the SnapVault primary system does not contain any LUN clones.
- An initial LUN clone transfer requires the backing LUN and the cloned LUN to be in the same qtree.
- The backing Snapshot copy must exist on the secondary system. It can be missing on the primary system for optimized transfers.
- If the backing Snapshot copy is deleted on the secondary storage system, the transfer of the backup Snapshot copy fails.
 If the backing Snapshot copy is deleted on the primary system, but exists on the secondary system, the transfer succeeds.
- The backing Snapshot copies on the secondary system are locked after the backup Snapshot copy is transferred. To delete a backing Snapshot copy, you should first delete any backup Snapshot copy that depends on it. You can run the lun snap usage command to find the backup Snapshot copy.
- During a restore operation, the backing Snapshot copy must be on the primary storage system before the backup Snapshot copy can be transferred. If the backing Snapshot copy already exists on the primary system, there is no need to transfer it again.
- If you have a SnapVault transfer followed by a volume SnapMirror cascade, you should not run volume SnapMirror updates while backing Snapshot copies are being transferred. This might cause SnapVault to use a sub-optimal backing Snapshot copy when the LUN clones are created. No data corruption is possible, but extra Snapshot copies might be locked.
- You should not create Snapshot copies manually when backing Snapshot copies are being transferred.

Note: If SnapDrive for Windows uses the SnapVault relationship running Data ONTAP 7.3 for Volume Shadow Copy Service-based backup, you should not revert to a release earlier than Data ONTAP 7.3.

For more details on how to transfer Volume Shadow Copy Service-based Snapshot copies using SnapDrive for Windows, see the *SnapDrive for Windows Installation and Administration Guide*.

How to change SnapVault settings

You can use the snapvault modify command to change the primary system (source) qtree that you specified using the snapvault start command. You can change the SnapVault settings for

transfer speed and number of tries before quitting. You might need to make these changes if there are hardware or software changes to the systems.

The meaning of the options is the same as for the snapvault start command. If an option is set, it changes the configuration for that option. If an option is not set, the configuration of that option is unchanged.

Note: The descriptions and procedures in this section pertain to SnapVault backup of systems running Data ONTAP only. For descriptions and procedures pertaining to SnapVault backup of open systems drives and directories, see the Open Systems SnapVault documentation.

The snapvault modify command is available only from the secondary system. You can also use this command to modify the tries count after the relationship has been set up. This is useful when there is a planned network outage.

You use the snapvault modify command to change the source if the primary system, volume, or qtree is renamed. This ensures the continuity of the existing SnapVault relationship between the primary and secondary systems. However, you cannot copy a primary qtree to another volume or system and use this command to take backups from that new location.

If you need to change the SnapVault schedule, use the snapvault snap sched command.

Related concepts

How to start a SnapVault backup relationship on page 240

Changing settings for SnapVault backup relationships

You can change the settings for SnapVault backup relationships that you entered with the snapvault start command, by using the snapvault modify command.

Step

1. From the secondary system, enter the following command on a single line:

```
snapvault modify [-k kbs] [-t n] [-o options] [-S
[pri_system:]pri_qtree_path] [sec_system:]sec_qtree_path
```

-k *kbs* specifies a value in kilobytes per second for the throttle (transfer speed) for the primary system. A value of unlimited lets the transfer run as fast as it can. Other valid values are whole positive numbers.

-t *n* specifies the number of times to try the transfer before giving up. The default is 2.

If set to 0, the secondary system does not update the qtree. This is one way to temporarily stop updates to a qtree.

options is opt_name=opt_value [[, opt_name=opt_value]...]. For more details about the available options, see the SnapVault man page.

Note: To ensure that SnapVault does not transfer inodes because of access time changes when a client reads a file, set -o ignore_atime=on.

-S [pri_system:] pri_qtree_path specifies the primary storage system for the qtree. [sec_system:] sec_qtree_path specifies the secondary system for the update.

The following example shows how to modify SnapVault so that it continues backing up the data on the primary system systemB:/vol2/qtree3 after the name qtree3 on the primary system changes to qtreeBob.

```
systemA> snapvault status
Snapvault server is ON.
                                                  State
                                                              Lag
Source
                       Destination
                                                                         Status
systemB:/vol/vol2/qtree3 systemA:/vol/sv_vol/qtree3 Snapvaulted 02:48:24 Idle
[The qtree3 on systemB is renamed qtreeBob.]
systemA> snapvault modify -S systemB:/vol/vol2/qtreeBob /vol/sv_vol/qtree3
systemA> snapvault status
Snapvault server is ON.
                                                    State Lag
                         Destination
                                                                          Status
Source
systemB:/vol/vol2/qtreeBob systemA:/vol/sv_vol/qtree3 Snapvaulted 0:00:31 Idle
```

Why you manually update a qtree on the secondary system

You can use the snapvault update command to manually update the SnapVault qtree on the secondary system from a Snapshot copy on the primary system. You might want to update at an unscheduled time to protect the primary system data.

Manual updates are useful in the following situations:

- A disk failed on the primary system and you want extra protection for the data.
- The nightly backup failed due to a network problem.
- The primary system hardware is going to be reconfigured.
- You want to transfer a Snapshot copy of a quiesced database.

Note: The descriptions and procedures in this section pertain to SnapVault backup of systems running Data ONTAP only. For SnapVault backup of opens systems drives and directories, see the Open Systems SnapVault documentation.

Next topics

Manually updating individual secondary system qtrees on page 267 *Examples of how to update the Snapshot copy on the secondary system* on page 267

Manually updating individual secondary system qtrees

You can manually update a SnapVault qtree on the secondary system, by using the snapvault update command.

Step

1. To manually update a SnapVault qtree on the secondary system, enter the following command from the secondary system:

```
snapvault update [options] [sec_system:]sec_qtree_path
```

options can be one or more of the following:

- -k *kbs* overrides the configured rate and specifies a value in kilobytes per second for the throttle (transfer speed) for this Snapshot copy transfer. The default value, unlimited, lets the transfer run as fast as it can.
- -s *snapname* enables you to specify a primary system Snapshot copy that is more recent than the current base Snapshot copy.

[sec_system:]sec_qtree_path is the name of the secondary system qtree that you want to update.

Examples of how to update the Snapshot copy on the secondary system

The following examples show how to update the Snapshot copy on the secondary system.

Example 1

systemB> snapvault update /vol/vol2/qtree3

SnapVault updates the qtree on the secondary system (systemB) with the data from a new Snapshot copy of the qtree it creates on the primary system (systemA). You do not have to specify where the primary system data is, because you already did so when you set up the SnapVault relationship using the snapvault start command.

Example 2

To update gtree3 on the secondary system (systemB) with a particular Snapshot copy of gtree3 on the primary system (systemA), you would use the -s option to specify the Snapshot copy on the primary system, as in the following example.

systemB> snapvault update -s my_snap systemB:/vol/vol0/qtree3

SnapVault updates the qtree on the secondary system (systemB) with the data from a Snapshot copy of qtree3 (my_snap) that you created earlier on the primary system (systemA).

Note: The snapvault update command does not create a new Snapshot copy on the secondary system. You need to use the snapvault snap create command if you want to create a new Snapshot copy on the secondary system.

Related concepts

Why you create a Snapshot copy manually on page 268

Why you create a Snapshot copy manually

In certain cases, you might want to create a manual (unscheduled) Snapshot copy.

Creating a manual Snapshot copy is useful in these situations:

- You anticipate planned downtime or you need to recover from downtime (during which a Snapshot copy was not taken on time).
- You have just carried out a manual update of a secondary qtree, and you want to immediately incorporate that update into the retained Snapshot copies on the secondary system.

Note: The descriptions and procedures pertain to SnapVault backup of systems running Data ONTAP only. For descriptions and procedures pertaining to SnapVault backup of open systems drives and directories, see the Open Systems SnapVault documentation.

Next topics

Creating a Snapshot copy manually on page 268 *Specifying a single try for SnapVault Snapshot copy creation* on page 269

Related tasks

Manually updating individual secondary system qtrees on page 267

Creating a Snapshot copy manually

You can manually create a SnapVault Snapshot copy on the SnapVault primary or secondary system by using the snapvault snap create command.

Step

1. To create a manual Snapshot copy of a volume, from the primary system or secondary system, enter the following command:

snapvault snap create vol_name snap_name

vol_name is the name of the volume where the Snapshot copy to be created will reside.

snap_name is the basename of the Snapshot copy to create.

If there is already a Snapshot copy being created in the volume at the time this command is invoked, this command is carried out after the other Snapshot copy is completed.

systemB> snapvault snap create vol1 sv_nightly

SnapVault creates a new Snapshot copy and, based on the specified Snapshot copy basename, numbers it just as if that Snapshot copy had been created by the SnapVault schedule process. SnapVault names the new Snapshot copy sv_nightly.0, renames the older Snapshot copies, and deletes the oldest sv_nightly Snapshot copy.

Note: Unlike the snapvault snap sched -x command, the snapvault snap create command does not update the data in the secondary qtree from the data in the primary qtree prior to creating the new Snapshot copy. If you want to update your secondary qtrees before using the snapvault snap create command, use the snapvault update command.

Related tasks

Manually updating individual secondary system qtrees on page 267

Specifying a single try for SnapVault Snapshot copy creation

When a Snapshot copy creation fails because the volume is out of space, SnapVault puts the request for the Snapshot copy in a queue and keeps trying until the attempt is successful. Some applications have a limited backup time frame and cannot support retries. In such a case, you might want to stop SnapVault from trying to create the Snapshot copy again.

Step

1. To stop SnapVault from repeatedly trying to create a Snapshot copy, enter the following command:

snapvault snap create -o tries=1 vol_name snap_name

vol_name is the name of the volume where the Snapshot copy is created.

snap_name is the basename of the Snapshot copy that is created.

tries=1 tries to create a Snapshot copy only once.

For more information about the tries option, see the na_snapvault(1) man page.

Renaming a SnapVault or Open Systems SnapVault secondary volume

If you rename a volume involved in a SnapVault relationship, you need to update the SnapVault configurations with the new name.

About this task

If the qtree is not configured in SnapVault, using the snapvault start command to configure the qtree gives the following error:

Error: Not a snapvaulted qtree, ignoring.

This error means that the qtree is not a SnapVault replica.

Steps

1. To rename a volume, enter the following command:

```
vol rename oldvolname newvolname
```

2. To verify the changes, enter the following command:

snapvault status -c

snapvault status -c does not show the new path yet.

3. Enter the following command:

snapvault start -S pri_filer:pri_qtree sec_filer:sec_qtree

Snapvault configuration for the qtree has been set. Qtree /vol/ newvolname/sec_qtree is already a replica.

4. Enter the following command:

snapvault status -c

snapvault status -c now shows the new path.

5. Enter the following command to verify that the change was successful:

```
snapvault update sec_qtree
```

6. The output of snapvault status -c will contain entries referencing the old volume name in addition to the new volume name. Remove the old entries by using the snapvault stop command:

```
snapvault stop /vol/oldvolname/sec_qtree
```

Snapvault configuration for the qtree has been deleted. Could not delete qtree: destination qtree does not exist

The output reflects that the configuration information is deleted and that the qtree did not exist on disk. This is normal because the volume name has changed.

Restoring SnapVault data to the primary system

In the event of data loss on a primary system, you might need to restore data from the secondary system.

About this task

Restoring data from the SnapVault secondary system involves the following command-line operations.

- You use the snapvault restore command to restore a backed-up qtree saved to the secondary system. Starting with Data ONTAP 7.3, you can restore the data to an existing qtree on the primary system using baseline restore or incremental restore.
 - **Baseline restore**: The primary system must be running Data ONTAP 7.3 or later, and the secondary system can be running any Data ONTAP version.
 - **Incremental restore**: Both the primary and secondary systems must be running Data ONTAP 7.3 or later.

Note: Starting with Data ONTAP 7.3, the SCSI connectivity of applications to all LUNs within the qtree being restored is maintained throughout the restore process in order to make the restore operation nondisruptive to applications. However, I/O operations are not allowed during the restore operation. Only in-place baseline restores and incremental restores can be nondisruptive.

- After successfully restoring data, you use the snapvault start -r command to resume the SnapVault relationship between the restored qtree and its backup qtree on the secondary system (assuming you want to continue SnapVault protection of the data). If you do not want to continue the backup relationship, you use the snapvault release command to cancel any further backups of the restored qtree and to release the resources on the secondary system that were used in the SnapVault relationship.
- When several SnapVault restore operations to different primary qtrees in the same volume are running concurrently, Snapshot cleanup might fail due to Snapshot copies being locked by various restore operations. Therefore, some unwanted Snapshot copies might be left behind in the volume.

You can manually delete these unwanted Snapshot copies. Make sure that you do not delete the base Snapshot copy of the restore operation.

Note: If you do not want to resume the relationship, you can delete the Snapshot copy created by the restore operation. If you want to resume the SnapVault operation, you can delete the Snapshot copy after you have successfully resumed the relationship.

Steps

- 1. If you intend to restore a primary qtree to the exact qtree location on the primary system from which you backed it up, you can perform any one of the following steps:
 - Baseline restore—The baseline restore can be to an existing qtree or to a non-existing qtree.

Note: In case of a baseline restore to an existing qtree, the restore operation overwrites the qtree data.

• Incremental restore—The restore operation transfers only incremental changes from the secondary qtree to the specified primary qtree.

Note: When restoring an existing primary qtree, an incremental restore is more efficient. If the incremental restore fails, then you can attempt an in-place baseline restore.

2. Enter the following command on the primary system on a single line:

```
snapvault restore [-f] [-k n] [-r] [-w] [-s snapname] -
S sec_system:sec_qtree_path [prim_system:]prim_qtree_path
```

-S [sec_system:]sec_qtree_path specifies the secondary system and qtree path from which you want to restore the data.

The -f option forces the command to proceed without first asking for confirmation from the user.

The -k option sets the maximum transfer rate in kilobytes per second.

The -r option attempts an incremental restore. The incremental restore can be used to revert the changes made to a primary qtree since any backed-up version on the secondary system.

The -w option causes the command not to return after the baseline transfer starts. Instead, it waits until the transfer completes (or fails). At that time, it prints the completion status and then returns.

The -s option specifies that the restore operation must use the specified(*snapname*) Snapshot copy on the secondary system.

prim_system is the name of the primary system that you want to restore to. If specified, this name must match the name of the host system.

prim_qtree_path is the name of the primary system qtree that you want to restore to.

For more information about the snapvault restore command options, see the na_snapvault(1) man page.

3. As required, choose one of the actions from the following table.

If you want to	Then
Resume SnapVault backups of the newly restored qtree on the primary system	Go to Step 4.
Discontinue the backup relationship between the newly restored qtree on the primary system and its secondary qtree partner	Go to Step 6.

4. To resume the SnapVault relationship between the restored qtree and its backup qtree on the secondary system, enter the following command on a single line:

snapvault start -r [options] -S [prim_system:]prim_qtree_path
[sec_system:]sec_qtree_path

-r is required to restart backups from a restored primary system. For details about the snapvault command syntax, see the na_snapvault(1) man page.

5. To discontinue the SnapVault relationship between the restored qtree and its backup qtree on the secondary system, enter the following command on the secondary system:

```
snapvault release sec_qtree_path [prim_system:]prim_qtree_path
```

sec_qtree_path is the name of the secondary system qtree that you want to release from a SnapVault relationship.

[prim_system:]prim_qtree_path is the name of the primary system qtree that you want to release.

Next topics

Examples of restoring SnapVault data on page 273 *Deleting the residual Snapshot copy* on page 274

Examples of restoring SnapVault data

The following examples show how to restore SnapVault data on the primary storage system.

Example 1

This example shows the primary storage system (systemA) requesting data to be restored from the secondary storage system (systemB) and then the secondary storage system restarting the SnapVault backup relationship.

For incremental restore:

```
systemA> snapvault restore -r -s sv_backup.0 -S systemB:/vol/sv_vol/
qtree3 systemA:/vol/vol1/qtree3
Restore will overwrite existing data in /vol/vol1/qtree3.
Are you sure you want to continue? yes
Transfer started.
Monitor progress with 'snapvault status' or the snapmirror log.
```

For baseline restore:

```
systemA> snapvault restore -s sv_backup.0 -S systemB:/vol/sv_vol/
qtree3 systemA:/vol/vol1/qtree3
Restore will overwrite existing data in /vol/vol1/qtree3.
Are you sure you want to continue? yes
Transfer started.
Monitor progress with 'snapvault status' or the snapmirror log.
```

After the restore operation, you can restart the SnapVault relationship:

```
systemB> snapvault start -r -S systemA:/vol/vol1/qtree3 systemB:/vol/
sv_vol/qtree3
The resync base snapshot will be: sv_backup.0
Resync may alter the data in this qtree.
Are you sure you want to resync the qtree? yes
Transfer started.
Monitor progress with 'snapvault status' or the snapmirror log.
```

Example 2

This example shows the primary storage system (systemA) requesting data to be restored from the secondary storage system (systemB) and then the secondary storage system canceling the SnapVault backup relationship on both storage systems to release the resources used.

```
systemA> snapvault restore -S systemB:/vol/sv_vol/qtree3 /vol/vol1/
qtree3
```

```
systemB> snapvault release /vol/sv_vol/qtree3 systemA:/vol/vol1/qtree3
```

Deleting the residual Snapshot copy

When you use the snapvault restore command to restore a primary qtree, SnapVault places a residual SnapVault Snapshot copy on the volume of the restored primary qtree. This Snapshot copy is not automatically deleted. However, you can delete this Snapshot copy manually.

About this task

If you have configured this volume to retain the maximum 251 Snapshot copies allowed by Data ONTAP, you must manually delete this residual Snapshot copy, or else no new Snapshot copies can be created.

Steps

1. To display all Snapshot copies (including the residual Snapshot copy) on the volume of the restored qtree, enter the following command:

snap list primaryvolume

The residual Snapshot copy is distinguished by the following syntax:

primaryhost (nvram_id)_primaryvolume_restoredqtree-dst.x

Example

prim_system (1880911275)_vol1_mytree-dst.2

2. To delete the residual Snapshot copy, enter the following command:

snap delete primaryvolume extrasnapshotname

Example

```
snap delete vol1 prim_system (1880911275)_vol1_mytreedst.2
```

How to abort SnapVault transfers

You can use the snapvault abort command to halt an ongoing SnapVault transfer if a later transfer is more useful or if an immediate shutdown or restart is necessary.

This command can halt ongoing SnapVault transfers from primary to secondary storage system (invoked by the snapvault start or snapvault update commands or scheduled through the snapvault snap sched -x command), or from secondary back to primary storage system (invoked by the snapvault restore command).

You can enter the snapvault abort command at the primary storage system or at the secondary storage system.

Next topics

Aborting primary-to-secondary storage transfers on page 275 Aborting secondary-to-primary storage transfers on page 276 Aborting SnapVault Snapshot copy creation on page 276

Aborting primary-to-secondary storage transfers

The snapvault abort command can halt ongoing SnapVault transfers from the primary to the secondary storage system that were invoked by the snapvault start or snapvault update commands, or that were scheduled through the snapvault snap sched -x command.

Step

1. To abort a primary to secondary storage system transfer, at the console of either the primary or secondary storage system, enter the following command:

```
snapvault abort [-f] [-h] [sec_system:]/vol/volx/sec_qtree
```

The -f option forces the abort command to proceed without first asking confirmation from the user.

The -h option causes a hard abort. It is only effective on the SnapVault secondary storage system. This option is ignored if specified on the primary storage system.

Note: If you use the -h (hard abort) option with the snapvault abort command, you cannot restart the transfer.

sec_system is the name of the SnapVault secondary storage system.

sec_qtree is the name of the secondary qtree to which the data is being transferred through SnapVault start or update commands.

Aborting secondary-to-primary storage transfers

The snapvault abort command can halt ongoing SnapVault transfers from secondary to primary storage that were invoked by the snapvault restore command.

Step

1. To abort a secondary to primary storage transfer, at the console of either the primary or secondary storage system, enter the following command:

```
snapvault abort [prim_system:]/vol/volx/prim_qtree
```

prim_system is the name of the primary storage system.

prim_qtree is the name of the primary qtree to which the data is being restored.

Note: If you use the -h option (hard abort) with the snapvault abort command, you cannot restart the transfer.

Aborting SnapVault Snapshot copy creation

You can abort the ongoing creation of a SnapVault Snapshot copy on the secondary system.

About this task

You can obtain the secondary system volume and Snapshot copy basenames from the snapvault status -s output.

Step

1. At the console of the secondary system, enter the following command:

snapvault abort -s volx sv_snapname

The -s option aborts the attempt to create a Snapshot copy with basename of *sv_snapname* on volume *volx*.

Ending SnapVault backups for a qtree

You can use the snapvault stop command to end the SnapVault backup process for a qtree when you no longer need the data in the primary system qtree to be protected.

About this task

After you use the snapvault stop command, SnapVault stops updating the qtree on the secondary system and deletes the qtree. Existing Snapshot copies on the secondary system are unaffected, but as

new Snapshot copies replace the old ones, the data from the qtree whose backup was stopped disappears.

Step

1. From the secondary system, enter the following command:

```
snapvault stop [sec_system:]sec_qtree_path
```

[sec_system:]sec_qtree_path is the qtree that you no longer want to back up.

Example

systemB> snapvault stop systemB:/vol/sv_vol/qtree3

Note: After you end the backup process from a SnapVault secondary system, you might want to release the obsolete Snapshot copies on the primary system.

Related tasks

Releasing Snap Vault relationships on page 277

Releasing SnapVault relationships

There are two methods of releasing a SnapVault relationship between a primary qtree and its secondary qtree backup (originally defined through the snapvault start command) after the relationship is no longer needed.

About this task

You can release SnapVault relationships in the following scenarios:

- On a primary storage system, as a part of shutting down a SnapVault relationship after a snapvault stop command was completed on the secondary storage system.
- On the secondary storage system, after data is restored to a primary storage system and you do not want to reactivate the backup relationship between the primary and secondary qtrees.

Steps

1. On a primary storage system console, enter the following command:

snapvault release prim_qtree_path sec_system:sec_qtree_path

2. On the secondary storage system console, enter the following command:

snapvault release sec_qtree_path prim_system:prim_qtree_path
Example

systemB> snapvault release /vol/sv_vol/qtree3 systemA:/vol/vol1/qtree3

Turning SnapVault off

You can turn SnapVault off by using the snapvault.enable option if the files on the primary or secondary storage system are no longer important or current or have been moved to another location.

Step

1. To turn off SnapVault on a primary storage system or secondary storage system, enter the following command:

options snapvault.enable off

This option persists across reboots.

Compression feature of Open Systems SnapVault

The compression feature of Open Systems SnapVault enables data compression over the network. This feature helps optimize bandwidth usage for Open Systems SnapVault data transfers.

Data ONTAP 7.3 and later support bandwidth optimization for Open Systems SnapVault through the compression feature. However, SnapVault primary systems do not support bandwidth optimization.

To specify whether to compress the network data, you can use global or local compression options.

Next topics

Enabling the compression feature globally for Open Systems SnapVault relationships on page 279 Enabling the compression feature for a new Open Systems SnapVault relationship on page 279 Enabling the compression feature for an existing Open Systems SnapVault relationship on page 280 Disabling the compression feature globally for Open Systems SnapVault relationships on page 280 Disabling the compression feature for a new Open Systems SnapVault relationship on page 281 Disabling the compression feature for an existing Open Systems SnapVault relationship on page 281 Disabling the compression feature for an existing Open Systems SnapVault relationship on page 281 Setting the default value for compression feature on page 282 Viewing the compression status for Open Systems SnapVault relationships on page 282

Enabling the compression feature globally for Open Systems SnapVault relationships

You can enable the compression feature globally, to optimize bandwidth usage for all Open Systems SnapVault relationships.

Step

1. To enable the compression feature globally, enter the following command on the secondary system:

options snapvault.ossv.compression on

options snapvault.ossv.compression on enables compression for the Open Systems SnapVault relationships for which the compression feature was not specified locally.

Note: The compression feature is enabled only for those relationships for which the compression option is not individually specified.

Enabling the compression feature for a new Open Systems SnapVault relationship

You can enable the compression feature for bandwidth optimization for a new Open Systems SnapVault relationship.

Step

1. On the secondary storage system, when setting up an Open Systems SnapVault relationship, enter the following command to specify the compression option:

snapvault start -o compression=on -S prim_host:dirpath
sec_host:sec_qtree_path

-o compression=on enables the compression feature for bandwidth optimization.

-S specifies the primary storage system and path. It must be given the first time to configure the qtree. It is optional when restarting an initial transfer for a previously configured qtree.

Example

```
snapvault start -o compression=on -S systemA:C:\dir3 systemB:/vol/vol1/
qtree3
```

Enabling the compression feature for an existing Open Systems SnapVault relationship

You can enable the compression feature for bandwidth optimization for an existing Open Systems SnapVault relationship.

Step

1. To enable the compression feature for an existing Open Systems SnapVault relationship, enter the following command:

snapvault modify -o compression=on [-S [prim_host:]dirpath]
[sec_system:]sec_qtree_path

-S specifies the primary storage system and path. It must be given the first time to configure the qtree. It is optional when restarting an initial transfer for a previously configured qtree.

compression=on enables the compression feature for bandwidth optimization.

Example

snapvault modify -o compression=on /vol/vol1/qtree3

Note: A snapvault modify command becomes effective only with the next transfer. Ongoing SnapVault transfers are not affected.

Disabling the compression feature globally for Open Systems SnapVault relationships

You can disable the compression feature for bandwidth optimization globally for all Open Systems SnapVault relationships.

Step

1. To disable the compression feature *globally*, enter the following command on the secondary storage system:

options snapvault.ossv.compression off

options snapvault.ossv.compression off disables compression for those Open Systems SnapVault relationships for which the compression feature is not specified locally.

Note: The compression feature will be disabled only for those relationships for which per relationship compression option is not enabled or disabled.

Disabling the compression feature for a new Open Systems SnapVault relationship

You can disable the compression option locally for a new Open Systems SnapVault relationship.

Step

1. To disable the compression option locally for each SnapVault relationship, enter the following command on the secondary storage system:

snapvault start -o compression=off -S prim_host:dirpath
sec_host:sec_qtree_path

-S specifies the primary storage system and path. It must be given the first time to configure the qtree. It is optional when restarting an initial transfer for a previously configured qtree.

-o compression=off disables the compression feature.

Example

```
snapvault start -o compression=off -S systemA:C:\dir3 systemB:/vol/vol1/
qtree3
```

Disabling the compression feature for an existing Open Systems SnapVault relationship

You can disable the compression feature for an existing Open Systems SnapVault relationship.

Step

1. To disable the compression feature for an existing Open Systems SnapVault relationship, enter the following command on the secondary storage system:

snapvault modify -o compression=off [-S [prim_system:]prim_qtree_path]
[sec_system:]sec_qtree_path

-S specifies the primary storage system and path.

compression=off disables the compression feature.

Disabling the compression feature for an existing Open Systems SnapVault relationship

snapvault modify -o compression=off /vol/vol1/qtree3

Note: To use the default value of the global compression option for an existing SnapVault relationship, enter the following command:

snapvault modify -o compression=default sec_qtree_path

A snapvault modify command becomes effective only with the next transfer. Ongoing SnapVault transfers are not affected.

Setting the default value for compression feature

You can set the default value for the compression feature for an existing Open Systems SnapVault relationship that has compression enabled or disabled locally.

Step

1. To set the default value, enter the following command:

```
snapvault modify -o compression=default sec_qtree_path
```

- If the compression feature is enabled globally, then compression is enabled for this Open Systems SnapVault relationship.
- If the compression feature is disabled globally, then compression is disabled for this Open Systems SnapVault relationship.

Viewing the compression status for Open Systems SnapVault relationships

You can view the compression status for Open Systems SnapVault relationships.

Step

1. To view the compression status for each Open Systems SnapVault relationship, enter the following command:

snapvault status -c

Either compression=on or compression=off is displayed for each relationship, if it is configured for that relationship.

Note: To view the compression ratio for each Open Systems SnapVault relationship, run the snapvault status -l command.

SnapVault secondary system protection

By setting up a SnapMirror relationship between the SnapVault secondary storage system and a SnapMirror destination storage system, NearStore system, or tape backup unit, you can provide backup and standby service or backup and restore protection for the SnapVault secondary storage system data.

• SnapMirror backup and standby service for SnapVault uses the SnapMirror destination device as a standby device to be activated as an alternate SnapVault secondary storage system if the original secondary storage system goes down.

• SnapMirror backup and restore protection for SnapVault uses the SnapMirror destination device as a source from which you can restore backup data to a SnapVault secondary storage system that has suffered data loss or corruption.

Next topics

How to use SnapMirror to replicate SnapVault data on page 283 *Using backup and standby service for SnapVault* on page 283

How to use SnapMirror to replicate SnapVault data

The SnapVault secondary storage system carries out SnapVault operations on its sources as usual. Then on a scheduled per-volume basis, the system replicates the SnapVault data to its SnapMirror destination partner or tape backup unit.

In this configuration, SnapVault does not delete any Snapshot version on the primary storage systems until that version has been successfully replicated from the SnapVault secondary storage unit to its SnapMirror destination. This guarantees that a Snapshot version will always be retrievable even if the SnapVault secondary storage system is disabled.

Related concepts

Protection of Snap Vault secondaries using volume SnapMirror on page 204

Using backup and standby service for SnapVault

You can set up SnapMirror to protect a SnapVault secondary system.

Steps

- 1. Use the license command to confirm that the SnapVault secondary storage device has both SnapVault secondary storage and SnapMirror features licensed.
- 2. Use the license command to confirm that the SnapMirror destination device has both the SnapVault secondary storage and SnapMirror features licensed.
- **3.** Set up SnapMirror replication from the active SnapVault secondary system to a disk-based destination device (another system or NearStore system).
- 4. If the active SnapVault secondary system is damaged or destroyed, convert the SnapMirror destination device to an alternate SnapVault secondary system to carry on the task of backing up data from the primary systema.

Example

You have the following configuration: systemA (primary) has a SnapVault backup to systemB (secondary) and systemB has a SnapMirror backup to systemC (tertiary). If systemB fails, break the SnapMirror backup between systemB and systemC, and re-create the SnapVault relationship such that systemA has a SnapVault backup to systemC.

5. After evaluating the reason for the data loss or corruption, either return the secondary system to the path as a tertiary system or add a new system as tertiary storage.

After adding systemB, you have the following configuration: systemA (primary) has a SnapVault backup to systemC (secondary) and systemC has a SnapMirror backup to systemB (tertiary). If you added a new system, it would replace systemB in the configuration.

6. As an optional step, you might want to return the systems to their original configuration.

Next topics

Re-creating the SnapVault relationship on page 284 *Adding back the tertiary system for SnapMirror backup* on page 285 *Returning systems to the original configuration* on page 286

Related tasks

Setting up a basic SnapMirror operation on page 108

Re-creating the SnapVault relationship

You can re-create the SnapVault relationship from the primary system to the tertiary system. The original configuration in the following procedure is a SnapVault relationship between systemA:/vol/vol1/qtree3 and systemB:/vol/vol2/qtree3 with the SnapVault destination volume systemB:vol2 backed up to another volume on a third system, systemC:vol3.

Steps

1. Break the volume SnapMirror relationship to the volume on the new secondary system (systemC) to make it writable.

Example

Perform the following steps from systemC.

snapmirror quiesce vol3 snapmirror break vol3

2. Check the status of the SnapMirror relationship on the new secondary system (systemC), by using the snapmirror status command.

The SnapMirror relationship should be broken-off.

3. Check the status of the SnapVault relationship on the new secondary system (systemC).

Example

snapvault status

The SnapVault relationship should be snapvaulted.

4. Add SnapVault configuration information to the new secondary system using the snapvault start command.

Note: This does not start a new baseline, it updates the registry.

Example

Perform the following step from systemC (the new secondary system).

snapvault start -S systemA:/vol/vol1/qtree3 systemC:/vol/vol3/qtree3

5. Check that the new SnapVault configuration is present.

Example

Perform the following step from systemC.

snapvault status -c

6. Test the new SnapVault relationship by manually updating systemC.

Example

Perform the following step from systemC.

snapvault update systemC:/vol/vol3/qtree3

- 7. Re-create any schedules used on the old secondary system on the new secondary system, and ensure access permissions are in place.
- 8. Release resources locked on systemA for the removed systemA to systemB SnapVault relationship.

Example

Perform the following step from systemA.

snapvault release /vol/vol1/qtree3 systemB:/vol/vol2/qtree3

Adding back the tertiary system for SnapMirror backup

Depending on the kind of backup interrupt you encountered, the old secondary system might be usable again as the tertiary system.

After adding the tertiary system, systemA has a SnapVault backup to systemC, and systemC has a SnapMirror backup to system B (or systemD if systemB cannot be reused).

Steps

1. Resynchronize the SnapMirror relationship between the new secondary system, systemC, and the new tertiary system, systemB.

Example

snapmirror resync -S systemC:vol3 systemB:vol2

2. Release resources that are locked for the old secondary system to tertiary system SnapMirror relationship.

Example Perform the following s

Perform the following step from systemB.

```
snapmirror release vol2 systemC:vol3
```

Related concepts

How the snapmirror resync command helps minimize data loss on page 192

Related tasks

Setting up a basic SnapMirror operation on page 108 Restoring to SnapVault from a local tape on page 202

Returning systems to the original configuration

In certain scenarios, you can return the systems to the original configuration after using backup and standby.

This is an optional procedure because you might not need to return the backup configuration to the original systemA to systemB to systemC configuration.

Steps

1. Ensure that the secondary system and tertiary system have the same data by temporarily blocking SnapVault updates to the secondary system.

Example

Perform the following step from the secondary system, systemC.

snapvault snap unsched vol3

2. Allow updates to the secondary system to be propagated to the tertiary system.

Example

Perform the following step from the tertiary system, systemB.

snapmirror update systemB:vol2

3. Reverse the roles of the secondary and tertiary systems, so that what was the secondary becomes the tertiary system and what was the tertiary becomes the secondary system.

Example

Perform the following step from the new tertiary system, systemC:

snapmirror resync systemC:vol3

4. Release resources locked on systemC for the removed systemC to systemB SnapMirror relationship.

Example

Perform the following step from the new tertiary system, systemC.

snapmirror release vol3 systemB:vol2

5. Break the SnapMirror relationship on systemB to allow further SnapVault updates from the original SnapVault relationship.

Example

Perform the following step from systemB.

snapmirror break vol2

6. Re-create the original SnapVault relationship from systemA to systemB.

Example

Perform the following step from systemB (the original secondary system).

snapvault update /vol/vol2/qtree3

7. Re-create the original SnapMirror relationship from systemB to systemC.

Example

Perform the following step from systemC (the original tertiary system).

snapmirror update vol3

8. Remove the SnapVault configuration from systemC.

Example

Perform the following step from systemC.

```
snapvault stop [-f] /vol/vol3/qtree3
```

This command is available on the secondary system only. The command unconfigures the qtree to ensure that are no more updates of the qtree and then deletes the qtree from the active file system.

The -f option forces the snapvault stop command to proceed without first asking for confirmation.

How to use SnapVault to protect a volume SnapMirror destination

You can use SnapVault to protect a volume SnapMirror destination. You can perform SnapVault transfers from the volume SnapMirror destination when you want to retain the data for a longer period.

In this deployment scenario, data from various primary systems is replicated to a remote site for disaster recovery. Volume SnapMirror ensures identical data at the source and destination systems. If you want to retain the data for a longer duration (that is, 90 days and more) at the disaster recovery site, you can use SnapVault to back up data from the SnapMirror destination.

When using SnapVault to back up a SnapMirror destination volume or qtree, you need to ensure the following:

• The SnapMirror license is installed on the SnapMirror primary system. Both the SnapMirror and SnapVault primary licenses are installed on the SnapMirror destination system (also the

SnapVault primary system). The SnapVault secondary license is installed on the SnapVault secondary system.

- The SnapVault operation occurs between SnapMirror scheduled updates. SnapMirror updates fail if a SnapVault operation is initiated or in progress.
- Before performing a SnapVault transfer for a particular Snapshot copy, this Snapshot copy must be preserved on the primary system.

To use the SnapVault backup schedule, you need to configure the SnapVault primary schedule at the volume SnapMirror primary system. Therefore, you need the SnapVault license installed on the volume SnapMirror primary system.

For more information about disaster recovery and long-term backup of data, see the *SnapMirror Async Overview & Best Practices Guide*.

Next topics

Preserving a Snapshot copy on page 288 *Unpreserving a Snapshot copy* on page 289

Related tasks

Planning SnapVault backup schedule and Snapshot copy retention on page 234

Related information

SnapMirror Async Overview & Best Practices Guide: http://media.netapp.com/documents/ tr-3446.pdf

Preserving a Snapshot copy

The snapvault snap preserve command enables you to preserve the required Snapshot copy. This command prevents Data ONTAP features (such as snap autodelete) from deleting the Snapshot copy.

About this task

You might want to preserve the Snapshot copies in the following scenarios:

- You do not want a Snapshot copy created by SnapVault to be recycled.
- You want to preserve some Snapshot copies from getting deleted by the snap autodelete command.

Note: The Snapshot copies locked by data replication features like SnapMirror and SnapVault are not deleted when the snap autodelete command uses the commitment try option.

• You want to preserve an application-consistent Snapshot copy.
Steps

1. To preserve a Snapshot copy at the volume SnapMirror primary system, enter the following command:

snapvault snap preserve vol_name snapshot_name [tag_name]

tag_name is the name of the preserve operation. It uniquely identifies this preserve operation. When the tag name is not specified, a tag name is added automatically.

snapshot_name is the name of the Snapshot copy.

Note: This command does not need a SnapVault license.

Example

snapvault snap preserve vol1 snap1 tag1

2. To list all preserved Snapshot copies, enter the following command:

snapvault snap preservations [vol_name] [snap_name]

If *snap_name* is not specified, then all the preserved Snapshot copies are listed.

If snap_name is specified, then all preservations on the specified Snapshot copy are displayed.

Example

snapvault snap preservations vol1 snap1

Unpreserving a Snapshot copy

If you do not want to retain the Snapshot copy that you have preserved, you need to unpreserve the Snapshot copies.

Before you begin

You can list all preserved Snapshot copies by using the snapvault snap preserve command.

Step

1. To unpreserve a Snapshot copy, run the following command:

snapvault snap unpreserve vol_name snapshot_name [tag_name|-all]

If tag_name is specified, then the Snapshot copy that is preserved with the specified tag name is removed.

If tag_name is not specified, then Snapshot copies preserved without a tag name are removed.

If the -all option is specified, all preservations on a specified Snapshot copy are removed.

Example

snapvault snap unpreserve vol1 snap1 -all

SnapVault and MultiStore

If you are using MultiStore software, you can use SnapVault to replicate the data for a vFiler unit.

There are certain limitations when using SnapVault with vFiler units. These limitations apply to all Data ONTAP releases that support vFiler units.

The management of SnapVault secondary (creation or modification of SnapVault relationships and schedules at the SnapVault secondary) is only supported from the default vFiler unit (vfiler0). The management of SnapVault secondary is not supported from a nondefault vFiler context. If the volume containing the SnapVault destination qtree is owned by a nondefault vFiler unit, the SnapVault secondary needs to be managed through the default vFiler unit (vfiler0).

Table 9: vFiler unit support with SnapVault secondary volumes

Management of SnapVault secondary volume	Ownership of SnapVa	ult secondary volume
	Default vFiler unit (vfiler0)	Nondefault vFiler unit
Default vFiler unit (vfiler0)	Yes	Yes
Nondefault vFiler unit	No	No

The following table shows vFiler unit support with SnapVault secondary volumes for different combinations.

The management of SnapVault primary in a vfiler context is supported.

Table 10: vFiler unit support with SnapVault primary volumes

The following table shows vFiler unit support with SnapVault primary volumes for different combinations.

Management of SnapVault primary volume	Ownership of SnapV	ault primary volume
	Default vFiler unit (vFiler0)	Nondefault vFiler unit
Default vFiler unit (vFiler0)	Yes	Yes
Nondefault vFiler unit	Yes	Yes (From a nondefault vFiler context, you can only manage volumes owned by that nondefault vFiler unit.)

For more information about vFiler units, see the Data ONTAP MultiStore Management Guide.

When performing online migration of a vFiler unit, any SnapVault operations that start a transfer do not work for volumes owned by the vFiler unit. Therefore, any manual or scheduled SnapVault update operations for the vFiler unit will not work. SnapVault commands that are not allowed for use during an online migration of a vFiler unit are given in the following list.

- snapvault start
- snapvault snap sched
- snapvault update
- snapvault restore

DataFabric Manager support for the management of SnapVault relationships

DataFabric Manager supports the management of SnapVault relationships for volumes through the default vFiler (vFiler0) context only. When using DataFabric Manager, the following limitations apply for SnapVault relationships involving nondefault vFiler units.

- You can only view SnapVault relationships configured through the default vFiler unit (vfiler0). You cannot view any SnapVault relationships configured through nondefault vFiler units.
- You can configure new SnapVault relationships for a volume only through the default vFiler unit (vfiler0), even if the volume belongs to a nondefault vFiler unit.

Error regarding language setting changes on volumes

A warning message is displayed on the Data ONTAP CLI when the language setting of the volume changes on volumes containing SnapVault or qtree SnapMirror destinations.

The following EMS error message appears if you update the SnapVault or qtree SnapMirror relationship after changing the language setting of the volume in the secondary system on which qtree replicas are configured.

```
Volume language was changed before this transfer or it was changed before
some previous update transfer for the relationship of qtree %s. This might
create some problems with further replication transfers of the
relationship.
```

To correct this problem, you need to create a new baseline for the transfer.

About SnapVault for NetBackup

The SnapVault for NetBackup feature provides an optimized disk-based backup repository for heterogeneous primary systems, by using NearStore secondary systems. By using NetBackup software with NearStore systems, you get a backup and recovery solution that can store a large number of backup images using a minimal amount of secondary storage and can quickly and easily recover data.

You can easily integrate SnapVault for NetBackup without changes to existing backup topologies. All components in the topology, such as the NetBackup master server, media servers, and clients, are unchanged. Adding SnapVault for NetBackup is similar to adding a traditional NetBackup tape unit or disk storage unit (DSU).

Note: This supposes you are using NetBackup 6.0 MP5 or later.

Data ONTAP allows a NetBackup media server to back up data from multiple platforms or clients, to a system that has the NearStore personality enabled.

These clients include: AIX, Solaris, HP-UX, Linux, Novell, Windows Server 2000, or Windows 2003 Server systems. This configuration is shown in the following illustration. For more information, see the NetBackup documentation.



Figure 19: SnapVault for NetBackup

Next topics

Snap Vault for NetBackup benefits on page 293 Snap Vault for NetBackup terminology on page 293 What the NetBackup catalog is on page 294 How NetBackup transfers data to the NearStore system on page 294 How to back up file system data using NetBackup on page 294 How the NearStore secondary system processes the data on page 295 Limitations of SnapVault for NetBackup on page 297 SnapVault for NetBackup usage recommendations on page 298 How to configure SnapVault for NetBackup transfers on page 301 Management of data transfers on page 302 Data restore from the exported file system on page 313 Removing SnapVault for NetBackup volumes on page 314 SnapVault for NetBackup with active/active configurations on page 317 *Disaster recovery using volume SnapMirror on NearStore systems* on page 318 *How to access backups on the replica of a SnapVault for NetBackup volume* on page 327 *Troubleshooting for SnapVault for NetBackup* on page 327

SnapVault for NetBackup benefits

The SnapVault for NetBackup feature provides benefits such as GUI administration and optimizing secondary storage.

The SnapVault for NetBackup feature provides the following benefits.

- You use a GUI-based backup and recovery administration.
- You can back up heterogeneous primary storage systems, such as UNIX, Linux, Novell, Solaris, and Windows systems. For systems supported by the NetBackup application, see the NetBackup documentation.
- You can create a large number of backup images online for immediate access.
- The feature creates point-in-time images of the primary file system in archival Snapshot copies.
- You can drag and drop files and directories to recover them over NFS and CIFS protocols.

Note: This feature requires NetBackup Enterprise 6.0 MP5 or later.

- The feature optimizes secondary storage by eliminating redundant data blocks between backup images.
- You can achieve high availability for SnapVault for NetBackup by directing backups to NearStore systems in an active/active configuration.
- You can configure a disaster recovery solution for SnapVault for NetBackup using volume SnapMirror on NearStore systems.

SnapVault for NetBackup terminology

You should know the following terms that describe the SnapVault for NetBackup feature.

- Archival Snapshot copy—A Snapshot copy that is saved by SnapVault for NetBackup as a pointin-time copy of the backed up file system. Data ONTAP uses Snapshot copies to create the archival Snapshot copy.
- Client—A storage system whose data you want to back up.
- Master server—A NetBackup server that performs all administrative actions and backup scheduling.
- Media server—A NetBackup server controlled by the master server that manages secondary storage units.
- DSU—A storage device where NetBackup stores files and data. For the SnapVault for NetBackup feature, a disk storage unit (DSU) is a FlexVol volume on the secondary storage system.
- Primary storage system—See client.
- Policy—A set of rules that define backup criteria for a particular client or group of clients.
- Secondary storage system—A storage system or NearStore system to which data is backed up.

- Secondary system qtree—A qtree on a secondary storage system to which data from a primary storage system is backed up.
- Snapshot copy—Backup images that SnapVault creates at intervals on the secondary storage system.

What the NetBackup catalog is

The NetBackup catalog is an internal database that resides on the NetBackup master server.

This database contains a record of all backup activity and information about the media and storage devices. The NetBackup catalog can be used to monitor and view SnapVault qtree archives. The NetBackup catalog does not include a record of individual files contained in each qtree archive.

For more information, see the NetBackup documentation.

How NetBackup transfers data to the NearStore system

The NetBackup master or media server controls the flow of data from the primary system (client) to the secondary system. Client data on the primary system is backed up inside a secondary qtree on the NearStore system.

Backups initiated by NetBackup are based on policies. Policies determine what data is backed up, when and how often the data is backed up, how long the backed up data is retained, and where the backed up data is stored. For more information about backup policies, see the NetBackup documentation.

How to back up file system data using NetBackup

You use SnapVault for NetBackup to back up UNIX, Linux, Solaris, and Windows file and directory data generated by the NetBackup client.

The NetBackup GUI has two check boxes that control the type of backup mode. The first check box, the **Enable Block Sharing** check box, controls whether the tar image that is sent to the NearStore secondary system is aligned on the Data ONTAP block boundary for block sharing. The second check box, the **File System Export** check box, controls whether the unpacked file system is exported to the secondary storage system using NFS or CIFS protocols and whether backups are captured in archival Snapshot copies.

Let us suppose that you are using SnapVault for NetBackup File System Export mode for backing up UNIX, Linux, Solaris, and Windows file and directory data generated by the NetBackup client. Backed up tar images that use the File System Export mode are aligned on the Data ONTAP block boundary for block sharing. The blocks are shared through context-dependant block-sharing during the backup process. The unpacked file system is exported using NFS and CIFS protocols to allow "drag-and-drop" restore functionality, and the backups are captured in archival Snapshot copies.

For File System Export backup, you should select the **Enable Block Sharing** and **File System Export** check boxes.

Related concepts

How to take archival Snapshot copies on page 296 *How backups are stored in archival Snapshot copies* on page 303

How the NearStore secondary system processes the data

It is important to understand how the NearStore secondary system processes the data.

Backup data transfer: Each backup data transfer to the NearStore secondary system consists of two streams: a standard NetBackup tar stream that contains header and file data, and a metadata stream that describes the contents of the tar stream.

The tar data is written to the NearStore secondary system and stored on disk. Keeping the tar stream on disk allows restores to be initiated by NetBackup from the NetBackup Administration console.

Using the metadata stream, the files and directories inside the tar stream are unpacked into the WAFL file system in Data ONTAP. The unpacking process creates files and directories in Data ONTAP as the tar stream is written to the NearStore secondary system. The file data in the unpacked file system does not consume additional blocks because it shares all of the tar stream data blocks.

Context-based deduplication: As Data ONTAP unpacks file data, it performs context-based deduplication. Context-based deduplication eliminates common blocks between the current tar stream and all previous tar streams; therefore, only new or changed data blocks in the new tar stream are written to disk on the NearStore secondary system.

Snapshot copy creation: After all of the data is written to disk and all the data inside the tar stream is unpacked into an active file system, the backup is captured to a Snapshot copy. In this way, backups of the primary file system are captured to unique Snapshot copies that you can browse if you need to restore previous versions of files or directories.

Data ONTAP optimizes the number of Snapshot copies consumed by coalescing backups for different polices into a single Snapshot copy. For example, if three backups for different policies finish before the next update to any of the backups, Data ONTAP coalesces the three backups into one Snapshot copy.

Restoring data: Before Data ONTAP 7.2, you could only restore data from SnapVault for NetBackup volumes by accessing the tar stream using the NetBackup administration console.

Starting with Data ONTAP 7.2, you can access the unpacked file system using either the NFS or the CIFS protocol after a backup finishes. This allows a second method to restore data, using the "dragand-drop" functionality to copy files and directories to the source from the backup.

Next topics

How to take archival Snapshot copies on page 296 *Configuring the default option for new volumes* on page 296

Related concepts

Efficient backup schedule on page 303 *Data restore from the exported file system* on page 313

How to take archival Snapshot copies

Starting with Data ONTAP 7.3.1, there is an option to disable taking archival Snapshot copies at the end of a data transfer.

With archival Snapshot copies disabled, you can still rely on context-based block sharing to deduplicate blocks. However, you cannot perform drag-and-drop restores. If you need the drag-and-drop functionality, you must enable this option for taking archival Snapshot copies.

This option is set to on, by default. You can set this option to off to disable taking archival Snapshot copies. However, after a DSU or volume stops taking archival Snapshot copies, you cannot revert and set this option to on.

Note: The archival Snapshot copies are enabled, by default, on all new volumes. Because archival Snapshot copies cannot be re-enabled on a volume that contains SnapVault for NetBackup backups, you must plan the archival Snapshot copies when planning for the backup solution.

Configuring the default option for new volumes

You can configure the default option for new volumes by using the options command. For the pervolume setting, you can use the vol options command.

Step

1. As required, choose one of the actions from the following table.

If you want to set or display	Then enter the following command
The global default on the system for new volumes	options snapvault.nbu.archival_snap_default {on off}
The option for a specific volume	<pre>vol options volume_name nbu_archival_snap {on off} [-f] -f is used when you do not want a confirmation (yes or no) on running this command. This option is especially useful when running commands from rsh.</pre>
	If the nbu_archival_snap volume option is not configured at the time when the first SnapVault for NetBackup backup starts for that volume, the volume option is at that time set according to the value of the snapvault.nbu.archival_snap_default option.
	Note: You can enable archival Snapshot copies only if no SnapVault for NetBackup backups exist on the volume. When you disable archival Snapshot copies, a warning appears with a yes/no prompt. A warning also appears when you specify the -f option.

Volumes replicated using volume SnapMirror inherit this option value set at the source. Use this option only for volumes that you intend to use as SnapVault for NetBackup volumes.

Upgrading to Data ONTAP 7.3.1: When you upgrade the storage system to Data ONTAP 7.3.1, archival Snapshot copies are enabled on all new volumes, by default. The configuration of the existing SnapVault for NetBackup volumes is automatically updated to reflect that archival Snapshot copies are enabled on these volumes. Therefore, when you upgrade from Data ONTAP 7.3 or a release in the Data ONTAP 7.2 family, the behavior remains the same.

Reverting to an earlier Data ONTAP release: When you revert from Data ONTAP 7.3.1 to Data ONTAP 7.3, all volumes take Snapshot copies irrespective of the setting for archival Snapshot copies in Data ONTAP 7.3.1 for the volume.

Limitations of SnapVault for NetBackup

There are some limitations when backing up data to a NearStore secondary system using NetBackup software.

The limitations of using SnapVault for NetBackup to back up data to a NearStore are given in the following list.

- NetBackup checkpoint restarts are not supported. If a backup fails, the backup is discarded and the next backup operation resends all of the previous data.
- DSU backups cannot be replicated to a tertiary system using synchronous SnapMirror, semisynchronous SnapMirror, qtree SnapMirror, or SnapVault.
- You cannot replicate data you backed up using NetBackup to a tape using NDMP or the native dump command; however, NetBackup Inline Tape Copy or the NetBackup Vault option can be used to create redundant copies of the NetBackup backup images.
- NDMPcopy is not supported to migrate SnapVault for NetBackup volumes.
- Maximum supported volume size is as follows:
 - FAS3020 system—1 TB
 - FAS3040 system—3 TB
 - FAS3050 system—2 TB
 - R200 system—4 TB
 - FAS3070 system—6 TB
 - FAS3140 system—3 TB
 - FAS3160 system—16 TB
 - FAS3170 system—10 TB
 - FAS6030 system—10 TB
 - FAS6040 system—10 TB
 - FAS6070 system—16 TB
 - FAS6080 system—16 TB
- A volume that contains NetBackup client data cannot be shared with other clients, for example, data backed up by another system using SnapVault.

- Only FlexVol volumes are supported; traditional volumes and WORM volumes are not supported.
- There is typically a space overhead of about one percent of the size of the backed up data.
- When backing up a dataset that consists of mostly small files (files less than 10 KB), storage efficiency decreases due to NetBackup to Data ONTAP block alignment overhead.
- When one node handles requests for both nodes (that is, the system is running in takeover mode), the observed throughput to each of the nodes might be affected, because the two nodes are now sharing the resources of a single system.
- SnapVault for NetBackup is not supported with Symantec Veritas NetBackup 6.5. Due to a software problem with NetBackup 6.5, SnapVault for NetBackup does not work with NetBackup 6.5. Symantec has fixed this problem in NetBackup 6.5.1. SnapVault for NetBackup works with NetBackup 6.0 MP5 or later, or NetBackup 6.5.1.

SnapVault for NetBackup usage recommendations

There are certain recommendations to consider when backing up NetBackup client data to a NearStore secondary system using NetBackup.

- Do not back up more than 100 NetBackup policies to a single volume.
- Ensure that the security setting for volumes containing Windows backups is either Mixed or NTFS to include ACLs and streams.
- Ensure that SnapVault for NetBackup volumes containing Windows backup data have the create_unicode option enabled.
- Do not use a SnapVault for NetBackup volume as the destination for any other types of data, such as SnapVault or SnapMirror data.

Next topics

Recommendations for saving space on page 298 Recommendations when using multistream backups on page 299 Other recommendations when using SnapVault for NetBackup on page 300

Related concepts

Efficient backup schedule on page 303

Related tasks

Using backup and standby service for Snap Vault on page 283

Recommendations for saving space

You should follow certain recommendations to save space when using SnapVault for NetBackup.

Do not rename volumes on a secondary system. Renaming volumes on the secondary system can create orphaned images on the volume. Starting with Data ONTAP 7.2.4, renaming a volume causes subsequent SnapVault for NetBackup backups directed to that volume to fail.

Orphaned images are backup images on the NearStore system that do not have corresponding NetBackup catalog entries. This can happen for one of the following reasons:

- When an image expires, NetBackup attempts to remove it from the DSU and the catalog. If the delete image operation fails for any reason, the image might be removed from the catalog, but not from the DSU.
- A NetBackup backup can fail, leaving an incomplete image on the DSU. NetBackup generally attempts to clean images for failed backups, but images might be left behind on the DSU.

The following problems can occur as a result of orphaned images:

- Orphaned images cannot be restored or expired.
- Orphaned images might occupy space that is not recoverable using NetBackup, leading to excess storage overhead.

Note: The volume must be destroyed to recover storage space from orphaned images.

Do not rename NetBackup backup policies. SnapVault for NetBackup achieves space savings using context-dependent deduplication technology, a technology that presumes multiple backups of the same context are likely to contain duplicate data. The NetBackup policy name is a parameter of the context of a backup; therefore, if the name is altered, backups based on the new policy name are no longer considered to be in the same context as backups made before the name change occupy more storage space than is necessary because the deduplication process cannot be used on them. The storage space cannot be recovered until all of the images made before the policy name change have expired.

Do not rename source data sets of NetBackup policies. NetBackup policies contain backup selections that are paths pointing to one or more source data sets. The paths are parameters of the context of a backup. As with renaming policy names, if the paths are changed, the context for the data to which they refer also changes. Changes to the context reduce the deduplication opportunities and increase the amount of storage space used.

You might change paths in backup selections of a policy if you rename underlying directories or mountpoints.

Note: Any change to the text of a backup selection alters the context for the underlying data, even if the resulting backup selection points to the same source data set as before the change.

Recommendations when using multistream backups

Multistream backup is a NetBackup feature that allows multiple backups (streams) of a single client to occur simultaneously to one or more DSUs. This feature allows clients with large amounts of data to meet a given backup window which would not be met if the data was backed up serially.

The NearStore system to which these streams are being directed has a limit on the maximum number of simultaneous streams allowed at a time. For example, 6070 has a limit of 128 simultaneous streams.

SnapVault for NetBackup supports multistream backups using the File System Export mode; however, maintaining such multistream backups is difficult.

Attention: You should not implement multistreaming backup configurations because of this difficulty. You should consult with your NetApp representative to determine the optimal number of multistream backups for your system.

NetBackup creates a stream for each data path in the backup selection list for a policy. NetBackup tags each stream by the order of the backup selections in the policy. A backup selection at the top of the list is tagged differently from the same backup selection at the bottom of the list. Shifting location in the backup selection list for a policy is a shift of context. If the backup selection list is reordered, the context is changed for the reordered backup selections. A context change means loss of extra storage requirement for the backups.

You can use the following guidelines to ensure that the ordering of backup selections is maintained to provide a consistent context for context-based deduplication:

- Do not alter the sequence of the backup selections in a policy.
- Do not delete any backup selection from the list if it was part of a previous backup operation.
- Only append new paths to the end of the backup selection list.
- Do not use ALL_LOCAL_DRIVES as a backup selection because the directive maps to all drives on the local host. Any change in the drive letter assignment due to mapping or unmapping of network drives, results in behavior similar to data set rename.
- Do not use wildcards in the data selection list. If NetBackup sends one stream per wildcard expansion and the number of wildcard expansions changes between consecutive backups, this results in a change in backup context and behavior similar to adding or deleting a data set in the middle of the backup selection list.

Related references

Maximum number of concurrent replication operations on page 128

Other recommendations when using SnapVault for NetBackup

You should follow certain recommendations when using SnapVault for NetBackup.

Recommendations for grouping policies: When you assign policies to DSUs, you should group these policies based on similar backup schedules, that is, the frequency of backups and retention periods.

A policy's frequency of backups combined with its retention period determines the number of Snapshot copies that will be consumed. A maximum of 250 Snapshot copies can exist at a time on a NearStore volume, so it is essential that Snapshot copy consumption is carefully managed.

Note: If the archival Snapshot copies option is set to off, this recommendation does not apply.

Recommendation for snap reserve setting: You should set the Snapshot reserve space on each SnapVault for NetBackup volume to 0 to ensure that maximum disk space is available for backups. You can change the Snapshot reserve space using the snap reserve command.

Recommendation for snap sched setting: SnapVault for NetBackup manages all Snapshot copies on SnapVault for NetBackup volumes. If Snapshot copies are also scheduled using the snap sched command, unnecessary Snapshot copies are created and stored. You should turn off Snapshot copy scheduling on each SnapVault for NetBackup volume to avoid creating unnecessary Snapshot copies. You can disable regular Snapshot copy scheduling using the snap sched command (snap sched volume_name 0 0).

Recommendation for using separate DSUs: To avoid possible language and naming issues, you should use separate SnapVault for NetBackup DSUs for Windows NetBackup clients and UNIX NetBackup clients.

Likewise, separate SnapVault for NetBackup DSUs should be used for NetBackup clients configured for different language settings. You can view language settings using the vol lang command.

Note: The language setting for a volume should be an initial setup procedure and should never be changed for volumes containing backups.

Related tasks

Configuring the default option for new volumes on page 296

How to configure SnapVault for NetBackup transfers

To set up SnapVault for NetBackup transfers to a NearStore secondary system, the system configuration must meet some requirements.

- The system must be a NearStore system or a system with the NearStore personality enabled.
- The NearStore system must be running Data ONTAP 7.1 or later. If you want to use "drag-anddrop" techniques to restore files and directories, the NearStore system must be running Data ONTAP 7.2 or later.
- A SnapVault secondary license must be enabled on the NearStore system.
- A SnapVault for NetBackup volume must be a FlexVol volume.
- NetBackup Enterprise 6.0 MP5 or later running on the master server and media server.
- NetBackup client systems must be running NetBackup 5.0 or later.

Configuring a NearStore system as a secondary system

You can configure a NearStore as a secondary system. If the secondary system is not a NearStore system, you should make sure that the NearStore personality license is enabled.

Steps

1. To set up the SnapVault secondary license, enter the following command:

license add sv_ontap_sec_license

2. To add a new backup user to the "Backup Operators" useradmin group list, enter the following command:

useradmin user add backupuser -g "Backup Operators"

3. To generate an NDMP password for the new user, enter the following command:

ndmpd password backupuser

The NDMP password is used to authenticate the NetBackup media server to the NearStore storage system. This password is required for NearStore setup on the NetBackup management station.

Note: It is important to avoid using the root password as an NDMP password because the root password is not encrypted and can compromise the integrity of your storage system. Although the default is NDMP version 4, ensure that the NDMP version is set by running the ndmpd version version_number command where version_number is equal to 4.

For more details about using NDMP, see the *Data ONTAP Data Protection Tape Backup and Recovery Guide*.

4. To enable SnapVault, enter the following command:

options snapvault.enable on

5. Use the options snapvault.access command to specify the names of the NetBackup media servers. Enter the following command:

```
options snapvault.access host=netbackup_server1,netbackup_server2 ...
```

Note: NetBackup client connect requests are received on TCP port 10571.

6. To confirm or set the appropriate global option, enter the following command:

options snapvault.nbu.archival_snap_default {on|off}

Related references

NearStore personality on page 0

Management of data transfers

Managing data transfers using SnapVault for NetBackup includes scheduling backups, file naming, monitoring qtree status, checking the status of a data transfer, and viewing space savings.

Next topics

Scheduling backups on page 303 Efficient backup schedule on page 303 How backups are stored in archival Snapshot copies on page 303 How to manage consumption of Snapshot copies on page 304 How to coordinate schedules to optimize space savings on page 309 Concurrent transfer limits on page 310 File system naming for SnapVault for NetBackup qtree images on page 310 Monitoring qtree status on page 311 Checking the status of a data transfer on page 311 Viewing space savings on page 312

Scheduling backups

Data ONTAP manages all data transfers when responding to requests from the NetBackup interface. However, backup policies are scheduled using the NetBackup Administration Console. Policies are a set of rules that define backup criteria for a particular client or group of clients. For example, NetBackup policies control backup intervals, number of versions, and how long to retain a particular archive. For more information about backup policies, see the NetBackup documentation.

For backups, it is important that each DSU receives backups from policies with similar backup schedules. Directing only similar backup schedules to a DSU will optimize space savings. It is also important to choose a backup schedule that will limit the number of consumed Snapshot copies.

Related concepts

Efficient backup schedule on page 303 *How to manage consumption of Snapshot copies* on page 304

Efficient backup schedule

Efficient backup schedules can help maximize the number of backups retained in the fewest number of archival Snapshot copies. In addition to planning each individual schedule, it is important to carefully synchronize backup schedules directed to the same volume. Coordinating schedules from different clients or policies to the same volume can improve overall space savings.

This information applies only to FSE mode storage systems, because archival Snapshot copies are kept only for FSE mode backups. Ensure that the archival Snapshot copies option is set to on.

Related tasks

Configuring the default option for new volumes on page 296

How backups are stored in archival Snapshot copies

SnapVault for NetBackup uniquely identifies each backup based on the policy, client, and the time at which the backup was taken. Each backup for a specific client-policy pair is uniquely stored in a Snapshot copy referred to as an archival Snapshot copy.

The following information applies only to FSE backups. Each Snapshot copy can store backups of many different client-policy pairs.

Data ONTAP software keeps the minimum number of Snapshot copies required to capture every backup uniquely in a Snapshot copy.

In the following example, multi-streaming is not used.

Example: You have a policy that backs up two clients, client_1 and client_2, in the same secondary volume. After a backup to client_1, Data ONTAP creates a Snapshot copy (for example, Snap_1) to preserve it. A backup to client_2 follows. After the backup to client_2, Data ONTAP creates another Snapshot copy (for example, Snap_2) to store it. The backup to client_2 occurred after the backup to

client_1, therefore Snap_2 contains both backups. Data ONTAP can delete Snap_1 because the backup to client_1 is also preserved in Snap_2.

A subsequent backup to client_1 results in the creation of another Snapshot copy (for example, Snap_3). However, Snap_3 and Snap_2 both need to be preserved because they contain two different backups to the same client (client_1).

How to manage consumption of Snapshot copies

The maximum number of Snapshot copies allowed by Data ONTAP is 255 for a volume; however, the recommended maximum number of Snapshot copies available for backups is 250, because other Data ONTAP services require Snapshot copies. You can use different strategies to optimize consumption of Snapshot copies.

When creating backup policies, you should carefully determine backup schedules, retention periods, and backup frequencies of the client-policy pairs directed to the same destination volume. This maximizes the number of backups stored in a given destination volume without the risk of running out of Snapshot copies.

In the following examples, multi-streaming is not used. The examples show different backup strategies that illustrate Snapshot copy optimization and the effects of backup schedules, retention periods, backup frequencies, and the use of FlexVol volumes.

Next topics

Example 1: One policy for multiple clients, one destination volume on page 304 Example 2: Multiple policies for multiple clients, one destination volume on page 305 Example 3: Two policies, different schedules, multiple clients, one destination volume on page 306 Example 4: Two policies, different schedules, multiple destination FlexVol volumes on page 307

Example 1: One policy for multiple clients, one destination volume

This strategy uses a single NetBackup policy to back up several clients once a day to the same destination volume.

Configuration: Multiple Clients (client1 through client5). Using one policy, you specify that all five clients are backed up once a day to the same destination volume.

Result: At the end of each day, one new Snapshot copy is created that contains a backup of each of the five clients. For example, at the end of the first day, there is one Snapshot copy containing the five backups, one for each client; at the end of the next day, there are two Snapshot copies, the one from the first day and one new Snapshot copy containing another backup to each client; and so on. After five days, 25 unique backups to 5 clients are backed up in 5 Snapshot copies. Visually, this appears like the following table.

Archival Snapshot copy	Contains
snapshot1—day 1	client1, client2, client3, client4, client5

Archival Snapshot copy	Contains
snapshot1—day 2	client1, client2, client3, client4, client5
snapshot1—day 3	client1, client2, client3, client4, client5
snapshot1—day 4	client1, client2, client3, client4, client5
snapshot1—day 5	client1, client2, client3, client4, client5

Maximum number of backups preserved in the volume: Using this strategy, 250 Snapshot copies or 250 backups of each client—one backup of each client in each Snapshot copy—is reached in 250 days.

Expiring backups to remove the oldest Snapshot copies: Because 250 Snapshot copies is the recommended maximum per volume, you should expire (delete) the oldest five backups (one of each client) from NetBackup after the 250th day. This deletes the first Snapshot copy, reducing the total number of Snapshot copies to 249. Now, you can take yet another backup of each of the five clients and save subsequent Snapshot copies. After day 250, you must expire the oldest backup for each client daily to have a Snapshot copy available for future updates to the five clients. The retention period for each client backup therefore cannot exceed 250 days to prevent the risk of exceeding the Snapshot copy limit for the volume.

Example 2: Multiple policies for multiple clients, one destination volume

This strategy uses multiple NetBackup policies to back up several clients once a day to the same destination volume.

Configuration: Five clients are to be backed up. Using multiple NetBackup policies (maximum of five), you specify that all five clients are backed up once a day to the same destination volume. You can schedule the policies to run at any time of the day, but the backups for all five clients are taken within a day.

Result: The results of this strategy are the same as that for Example 1: At the end of each day, one new Snapshot copy is created that contains a backup of each of the five clients. After five days, 25 unique backups to 5 clients are backed up in 5 Snapshot copies. Visually, this appears as the following table:

Archival Snapshot copy	Contains
snapshot1—day 1	client1, client2, client3, client4, client5
snapshot2—day 2	client1, client2, client3, client4, client5
snapshot3—day 3	client1, client2, client3, client4, client5
snapshot4—day 4	client1, client2, client3, client4, client5
snapshot5—day 5	client1, client2, client3, client4, client5

Maximum number of backups preserved in the volume: Using this strategy, 250 Snapshot copies or 250 backups to each client—one backup of each client in each Snapshot copy—is reached in 250 days.

Expiring backups to remove the oldest Snapshot copies: Because 250 Snapshot copies is the recommended maximum, you should expire (delete) the oldest backup to each of the five clients from NetBackup after day 250. This deletes the first Snapshot copy, reducing the total number of Snapshot copies to 249. Now, you can take yet another backup for each of the five clients and save them in a subsequent Snapshot copy. After day 250, you must expire the oldest backup of each of the five clients daily to have a Snapshot copy available for future backups.

The retention period for each client backup cannot exceed 250 days so as not to exceed the maximum number of Snapshot copies allowed per volume.

Example 3: Two policies, different schedules, multiple clients, one destination volume

This strategy uses two NetBackup policies to back up two sets of clients to the same destination volume. One policy schedules backups once a day; the other policy schedules backups twice a day.

Configuration: Five clients are to be backed up. Using one policy (policy_1) you back up client1, client2, and client3 once a day. Using another policy (policy_2), you back up client 4 and client5 twice a day. Both policies specify that all five clients are backed up to the same destination volume.

Result: The first scheduled backup of policy_1 creates backup copies for client1, client2, and client3. Data ONTAP makes a Snapshot copy to preserve the three backups.

The first scheduled backup of policy_2 creates backup copies for client4 and client5. Data ONTAP makes another Snapshot copy, preserving not only these two backups, but also the three backups of clients 1 through 3 made by the first scheduled backup of policy_1; therefore, Data ONTAP deletes the first Snapshot copy and keeps the second Snapshot copy.

After the second scheduled backup of policy_2, backups of clients client4 and client5 have been made once again and another Snapshot copy made. In this Snapshot copy, client1, client2, and client3 are the same as the backups in the first Snapshot copy, but the backups for client4 and client5 are different; therefore, Data ONTAP keeps this Snapshot copy as well, for a total of two Snapshot copies.

At the end of the day, there are two Snapshot copies that store seven unique backups of the five clients on the same destination volume. The first Snapshot copy contains unique updates to all five clients, while the second Snapshot copy contains three unchanged backups of client1, client2, and client3, and two unique backups of client4 and client5.

After five days, 35 unique backups of the 5 clients are stored in 10 Snapshot copies. Visually, this appears like the following table.

Archival Snapshot copy	Contains
snapshot1—day 1	client1, client2, client3, client4, client5

Archival Snapshot copy	Contains
snapshot2—day 1	client1 (unchanged), client2 (unchanged), client3 (unchanged), client4, client5
snapshot3—day 2	client1, client2, client3, client4, client5
snapshot4—day 2	client1 (unchanged), client2 (unchanged), client3 (unchanged), client4, client5
snapshot5—day 3	client1, client2, client3, client4, client5
snapshot6—day 3	client1(unchanged), client2 (unchanged), client3 (unchanged), client4, client5
snapshot7—day 4	client1, client2, client3, client4, client5
snapshot8—day 4	client1(unchanged), client2 (unchanged), client3 (unchanged), client4, client5
snapshot9—day 5	client1, client2, client3, client4, client5
snapshot10—day 5	client1 (unchanged), client2 (unchanged), client3 (unchanged), client4, client5

Maximum number of backups preserved in the volume: Using this strategy, the 250 Snapshot copy limit is reached in 125 days. Backups of client4 and client5 are present in all of the Snapshot copies, while backups of client1, client2, and client3 are present in half of the Snapshot copies. For simplicity, let us suppose that Snapshot copies do not expire. By the time the 250 Snapshot copy limit is reached, 250 Snapshot copies contain backups of client4 and client5 and 125 Snapshot copies contain backups of client1, client2, and client3.

Expiring backups to remove the oldest Snapshot copies: At the end of the 125th day, before taking any more backups, you should expire (delete) the oldest two backups of the five clients from NetBackup. This deletes the oldest two Snapshot copies, reducing the total number of Snapshot copies to 123. Now, you can take yet another backup of all five clients and save them in a couple of subsequent Snapshot copies. After day 125, you must expire the oldest two backups of all five clients daily to have two Snapshot copies available for future backups to the five clients.

Example 4: Two policies, different schedules, multiple destination FlexVol volumes

This strategy uses two NetBackup policies to back up two sets of clients to two FlexVol volumes. One policy schedules backups once a day; the other policy schedules backups twice a day.

Configuration: Five clients, client1 through client5, are backed up. Using one policy (policy_1), you back up client1, client2, and client3 once a day. Using another policy (policy_2) you back up client 4 and client5 twice a day. The clients in policy_1 are backed up to FlexVol volume volA and the clients in policy_2 are backed up to FlexVol volume volB.

Result: The first scheduled backup of policy_1 creates backup copies for client1, client2, and client3. Data ONTAP makes a Snapshot copy to preserve the three backups. After five days, 15 unique

backups of 3 clients are stored in 5 Snapshot copies on FlexVol volA. Visually, this appears like the following table.

Archival Snapshot copy	Contains
snapshot1—day 1	client1, client2, client3
snapshot2—day 2	client1, client2, client3
snapshot3—day 3	client1, client2, client3
snapshot4—day 4	client1, client2, client3
snapshot5—day 5	client1, client2, client3

Table 11: FlexVol volume volA

The first scheduled backup of policy_2 creates backup copies for client4 and client5. Data ONTAP takes a Snapshot copy to preserve the two backups. After the second set of backups scheduled for policy_2 finishes, backups for the updated client4 and client5 are created. Data ONTAP takes a second Snapshot copy to preserve the updated backups. The earlier Snapshot copy cannot be deleted because they contain unique client updates; therefore, two Snapshot copies per day are kept.

After five days, 20 unique backups of the 2 clients are stored in 10 Snapshot copies on FlexVol volB. Visually, this appears like the following table.

Table 12: FlexVol volume volB

Archival Snapshot copy	Contains
snapshot1—day 1	client4, client5
snapshot2—day 1	client4, client5
snapshot3—day 2	client4, client5
snapshot4—day 2	client4, client5
snapshot5—day 3	client4, client5
snapshot6—day 3	client4, client5
snapshot7—day 4	client4, client5
snapshot8—day 4	client4, client5
snapshot9—day 5	client4, client5
snapshot10—day 5	client4, client5

Maximum number of backups preserved in the volumes: FlexVol volume volA, saving one archival Snapshot copy a day, reaches the maximum Snapshot copy limit after 250 days. FlexVol volume volB, saving two Snapshot copies a day, reaches the limit after 125 days. The advantage of

using separate FlexVol volumes is that you can direct backups with different backup frequencies to the same aggregate and store 250 unique updates before you have to start expiring old backups.

Expiring backups to remove the oldest Snapshot copies: At the end of the 250th day, FlexVol volume volA reaches the recommended maximum number of 250 Snapshot copies. You should expire (delete) the oldest backup of each of the three clients from NetBackup after the 250th day, thereby reducing the number of Snapshot copies to 249 and enabling Data ONTAP to take a Snapshot copy the next day. The maximum retention period for the clients backed up to FlexVol volume volA is 250 days (to prevent running out of Snapshot copies in the volume).

At the end of the 125th day, FlexVol volume volB reaches the recommended maximum number of 250 Snapshot copies. You should expire (delete) the oldest two backups of each of the two clients from NetBackup after the 125th day, thereby reducing the number of Snapshot copies to 248 and enabling Data ONTAP to take two Snapshot copies the next day. The maximum retention period for the clients backed up to FlexVol volume volB is 125 days.

How to coordinate schedules to optimize space savings

When directing backups from multiple policies to a single volume, you should make sure that these backups have similar backup frequency and retention periods. Even though backups expire based on their retention period, the space consumed by these backups in the Snapshot copy might not be released until all the other backups (from different policies) in the same Snapshot copy have expired.

Archival Snapshot copies are taken for the entire volume. A volume can contain backups from multiple policies, which means that backups from different policies can be captured in the same archival Snapshot copy. SnapVault for NetBackup will remove an existing archival Snapshot copy only after all of its captured backups reach the end of their retention periods. The backup with the longest retention period therefore dictates when the archival Snapshot copy will be deleted. The space allocated for the other backups will continue to be consumed until the Snapshot copy is deleted.

Example

Let us consider two policies, PolA and PolB. The retention period for PolA is one year and the retention period for PolB is three months. After a full backup of both these policies, there is one archival Snapshot copy, SnapX, which captures the backup from both of these policies. SnapX is deleted (thereby reclaiming the allocated storage space) only the backups from both PolA and PolB expire. This can result in less than optimal storage savings. In this case, the space allocated for backup for PolB will be retained for one year even though it is expected to be kept for only three months. This occurs because both PolA and PolB share a common archival Snapshot copy, SnapX, which will get deleted only after the expiry of backups from both PolA and PolB. To avoid this issue, you should direct only backups with similar backup frequency and retention to the same volume.

Concurrent transfer limits

Each model has a limit on the number of concurrent transfers for SnapVault for NetBackup.

The maximum number of concurrent transfers for each model is displayed in the following table.

Table 13: Maximum number of concurrent streams with NearStore license

Model	Maximum number of concurrent transfers supported for SnapVault for NetBackup
FAS3020	32
FAS3040	64
FAS3050	64
FAS3070	128
FAS3140	64
FAS3160	128
FAS3170	128
FAS6030	96
FAS6040	96
FAS6070	128
FAS6080	128
R200 (no NearStore license)	128

Related references

Maximum number of concurrent replication operations on page 128

File system naming for SnapVault for NetBackup qtree images

SnapVault for NetBackup qtree images follow the default file system naming convention.

nbu_image_data is a special qtree that contains information specific to the tar images associated with each qtree.

nbu_basename_policyname_suffix is the qtree naming convention where each name is specific to NetBackup client and policy settings.

nbu_online_data is the directory in which data is unpacked. It is a subdirectory of the nbu_basename_policyname_suffix qtree. Data ONTAP 7.2 and later support viewing backed up data in this qtree.

For more information, see the NetBackup documentation.

Monitoring qtree status

To confirm that a destination qtree has been created on the system, you can use the qtree status command to display the SnapVault for NetBackup volume status.

Step

1. Enter the following command:

system> qtree status

Monitoring gtree status system> qtree status Style Oplocks Status Volume Tree ____ -----_____ vo10 unix enabled normal flexsle unix enabled normal sie unix enabled normal mixed enabled normal mixed enabled snapvaulted nbu_ibmx123_C1_F1.CLF_daily_0000 mixed enabled snapvaulted nbu_sunv240_C1_F1.CLF_daily_0000 mixed enabled snapvaulted nbu_ibmx123_C1_HDR.IMG_daily_0000 mixed enabled snapvaulted nbu_sunv240_C1_HDR.IMG_daily_0000 mixed enabled snapvaulted p3 p3 p3 p3 p3 p3 nbu_ibmx200_C1_F1.CLF_daily_0000 mixed enabled snapvaulted p3

Checking the status of a data transfer

You can use the snapvault status command to check the status of a data transfer. You can use this display to view the recent backup activity and confirm transfer status.

Step

1. Enter the following command:

```
system> snapvault status
```

```
Snapvault secondary is ON.
Source Destination
State Lag Status
system1:daily client:/vol/p3/nbu_client_1_daily_0000 Snapvaulted 00:35:32 Idle
system2:daily client:/vol/p3/nbu_client_2_daily_0000 Snapvaulted 00:35:32 Idle
system:daily client:/vol/p3/nbu_client_daily_0001 Snapvaulted 00:35:32 Idle
system:daily client:/vol/p3/nbu_client_daily_0002 Snapvaulted 00:35:32 Idle
system:daily client:/vol/p3/nbu_client_daily_0003 Snapvaulted 00:35:32 Idle
```

Viewing space savings

You can monitor the space savings for each DSU using the snapvault status -b command.

Step

1. To see how much space is saved, enter the following command in the console of the secondary storage system:

snapvault status -b

The amount of time it takes for the snapvault status -b operation to complete depends on how many backups were created on the volumes; therefore, the operation might take a long time to complete.

Viewing space savings

The following is a sample output of the snapvault status -b command:

```
filer1> snapvault status -b
```

To abort the snapvault status -b operation at any time, enter the following command:

Ctrl+C

The result of the snapvault status -b command is a list of volumes that contain NetBackup client data and information about space savings. The following list describes the information displayed by the snapvault status -b command:

- actual—Total disk space used on the volume by all of the tar files sent from the NetBackup application. This value represents the tar image files only and does not include any of the internal metafiles that might be associated with the backups.
- used—Total space used on the volume by the active file system and Snapshot copies. This value is equivalent to the sum of the values returned by the df command for space used by the active file system and the Snapshot copy.
- saved—Total space saved on the volume. This is equivalent to the actual value minus the used value. In the case when the actual value is less than the used value, this field is 0.
- %saved—Percentage of storage saved. In the case where there is no space savings, this field is 0.
- ratio—Ratio of shared block data to unshared block data. When there are no space savings, this field is 1:1.

Data restore from the exported file system

You can perform drag-and-drop restores of FSE backups from a SnapVault for NetBackup volume.

Note: This information is applicable for FSE mode backups only. It is not applicable if the snapvault.nbu.archival_snap_default option is set to off.

You can restore data from FSE backups in one of the following ways:

- Use the NetBackup Administration Console.
- Access the SnapVault for NetBackup volume by mounting the volume (NFS) or creating a share (CIFS), and either copy or drag the data.

For information about restoring data using the NetBackup Administration Console, see the NetBackup documentation.

Where the exported file system is: Data ONTAP 7.2 and later export the NetBackup client file system that allows you to access backed up NetBackup client files. When you use NetBackup, Data ONTAP software unpacks file data in addition to transferring the tar stream to disk. Data ONTAP puts the unpacked data in a subdirectory called nbu_online_data in the backup transfer qtree.

Backed up file systems have the following naming convention:

nbu_basename_policyname_suffix

basename is the NetBackup client name.

policyname is the NetBackup policy name.

suffix is a four-digit number.

Example: nbu_client1.CLF_backup1_0000

client1 is the NetBackup client name and backup1 is the policy name.

Next topics

Accessing a backed-up file system on page 313 Restoring data from a backed up file system on page 314

Accessing a backed-up file system

You can access backed-up file system data using SnapVault for NetBackup.

Steps

1. Mount the SnapVault for NetBackup volume.

Example

mount filer4:/vol/clientvol /t/filer4/clientvol

2. Navigate to the qtree that you want to access.

Example

cd /t/filer4/clientvol/nbu_clientvol.CLF_unixbackup_0000

You can also access Snapshot copies. The Snapshot directory is at the same directory level as the backed up qtrees. SnapVault for NetBackup uses the same Snapshot redirection mechanism used by the SnapVault software. This redirection mechanism does not show data in transition. Rather, the previous export of the backup is shown to maintain consistency. When the data transfer finishes, the new data is shown.

Note: If a transfer fails, you continue to see the previous consistent state of the backup until a new transfer succeeds.

Restoring data from a backed up file system

You can restore directories or files only after you access the exported file system.

Steps

- 1. Copy the directory or file from the exported file system.
- 2. Navigate to the location in which you want to restore data.
- **3.** Paste the directory or file.

Removing SnapVault for NetBackup volumes

You might want to destroy a SnapVault for NetBackup volume when you no longer use it as an archive.

Steps

1. Expire all of the backups to the storage unit configured as this volume using the NetBackup application. For more information, see the NetBackup documentation.

Note: The NetBackup application might show that all the backups have been expired, but the NearStore system might be in the process of destroying data from the volume. You might not be able to destroy the volume until all the file data is completely destroyed from the volume.

2. To take the volume offline, enter the following command:

vol offline volume_name

3. To destroy the volume, enter the following command:

vol destroy volume_name

Next topics

Destroying a volume before backups are expired on page 315 Reusing a volume without destroying it on page 315

Destroying a volume before backups are expired

You might destroy a volume before expiring the backups using the NetBackup application. If this is the case, stale information exists in the NetBackup catalog and on the NearStore system. To remove stale information on the storage unit configured as this volume and on the NearStore system, complete the following steps.

Steps

- **1.** Expire all of the backups to the storage unit configured as this volume using the NetBackup application. For more information, see the NetBackup documentation.
- **2.** Determine what stale information exists in the destroyed volume by using the following command:

system1> snapvault status -c

Example

systeml> snapvault status -c

```
Snapvault secondary is ON.
/vol/p3/nbu_clientname_C1_F1.CLF_daily_0000
source=clientname_C1_F1.CLF:daily
/vol/p3/nbu_clientname_C1_TIR.IMG_daily_0000
source=clientname_C1_TIR.IMG:daily
/vol/p3/nbu_clientname_C1_HDR.IMG:daily
/vol/p3/nbu_clientname_C1_F1.CLF_weekly_0000
source=clientname_C1_F1.CLF:weekly
/vol/p3/nbu_clientname_C1_TIR.IMG_weekly_0000
source=clientname_C1_TIR.IMG:weekly
/vol/p3/nbu_clientname_C1_HDR.IMG_weekly_0000
source=clientname_C1_HDR.IMG:weekly_0000
```

3. For each qtree listed for the volume, enter the following command:

snapvault stop -f qtree_path

Example

system> snapvault stop -f
/vol/p3/nbu clientname C1 F1.CLF daily 0000

Reusing a volume without destroying it

You might have a SnapVault for NetBackup volume that is no longer used as an archive, which you want to reuse for another purpose without destroying and re-creating it.

Note: It is not recommended that you reuse a volume without destroying it. The recommended procedure is to destroy the volume and then re-create the volume.

Steps

1. Expire all of the backups to the storage unit configured as this volume using the NetBackup application. For more information, see the *VERITAS NetBackup System Administrator's Guide*.

Note: The NetBackup application might show that all the backups have been expired, but the NearStore storage system might be in the process of destroying data from the volume. You might not be able to destroy the volume until all the file data is completely destroyed from the volume.

- 2. Delete the NetBackup storage unit configured for this volume using the NetBackup application. For more information, see the *VERITAS NetBackup System Administrator's Guide*.
- 3. Check if there are any qtrees remaining on the volume by entering the following command:

```
qtree status volume_name
```

volume_name is the name of the volume.

4. Remove every qtree shown, other than the .nbu_image_data qtree, by entering the following command for each qtree:

```
snapvault stop -f qtree_path
```

5. Remove the .nbu_image_data qtree by entering the following command:

snapvault stop -f /vol/volume_name/.nbu_image_data

6. Verify if deduplication is enabled for a volume by entering the following command:

sis status /vol/volume_name

7. End any deduplication activity on the volume by entering the following command:

sis off /vol/volume_name

8. Reset the schedule for automatic Snapshot copy creation by entering the following command:

snap sched volume_name

Example

snap sched vol1 0 2 6@8,12,16,20

9. To ensure that space is reserved for Snapshot copies, reset the snap reserve settings for the volume:

snap reserve volume_name [percent_reserve]

percent_reserve is the percentage of space you want to reserve for the Snapshot copies.

Example

snap reserve vol1 20

Related tasks

Removing Snap Vault for NetBackup volumes on page 314

Destroying a volume before backups are expired on page 315

SnapVault for NetBackup with active/active configurations

Starting with Data ONTAP 7.2.1, it was possible to configure NetBackup storage units for volumes residing on systems that were part of an active/active configuration, but there was no failover support for SnapVault for NetBackup. Starting with Data ONTAP 7.3, failover for SnapVault for NetBackup functionality is supported.

To know how a takeover and giveback scenario works in terms of SnapVault for NetBackup, see the following example. Before going through the example, it is recommended that you review the general information about active/active configurations in the *Data ONTAP Active/Active Configuration Guide*.

Takeover/giveback example for SnapVault for NetBackup: The following assumptions have been made:

- A NetBackup disk storage unit, DSU_A, points to a volume residing on SystemA.
- SystemA is part of an active/active configuration, and SystemB is its partner.

What happens if SystemA fails while a backup is in progress to DSU_A: When SystemA fails, all connections from NetBackup to SystemA are terminated. SystemB detects that SystemA is down and initiates a takeover (that is, SystemB takes over SystemA). After the takeover is complete, NetBackup retries all requests that were terminated when SystemA went down. Automated retries of failed backups can be configured in NetBackup. (For more information on how to configure automatic retries, see the NetBackup documentation.)

Note: No configuration change is required in NetBackup in the event of a takeover, because SystemA can still be contacted on the same IP address even while it is taken over by SystemB. The takeover appears to NetBackup as a restart of SystemA.

What happens while SystemA continues to be down and SystemB is in takeover mode: When in takeover mode, SnapVault for NetBackup backup, restore, and all other operations initiated from NetBackup to the taken-over node (SystemA) are handled the same way as they are when not in takeover mode. Everything appears same from a NetBackup perspective, except for the following limitation that applies to SystemA while it is taken over by SystemB.

The number of concurrent connections are shared between SystemA and SystemB while in takeover mode. Performance might decrease for both SystemA and SystemB while in takeover mode, because the services for both SystemA and SystemB are now handled with the resources of a single system.

Backups sent to DSU_A during takeover mode are written to the same set of disks as before SystemA failed. It is only the physical entity controlling these disks that change on takeover. Thus, restores work regardless of whether the backup to be restored is taken prior to or after SystemA's failure.

What happens when SystemA's system comes online: When the system on which SystemA originally resided comes online, SystemB can initiate a giveback to SystemA. For information on how to initiate a giveback, see the *Data ONTAP Active/Active Configuration Guide*. The giveback of SystemA appears similiar to a node restart for NetBackup. Ongoing NetBackup connections are

interrupted. After the giveback operation is complete, you must reinstantiate these connections from NetBackup.

Note: The connections to SystemB are not affected by the giveback operation.

Disaster recovery using volume SnapMirror on NearStore systems

You can use NetBackup and volume SnapMirror to provide a disaster recovery solution for Disk Storage Units (DSUs).

SnapVault for NetBackup with volume SnapMirror provides disaster recovery in a very efficient manner. Deduplicated volume blocks are sent only once over wire from the SnapMirror source to the SnapMirror destination, and deduplication properties from the source are preserved on the destination.

To provide failover support in a disaster recovery (DR) situation using the volume SnapMirror feature outside of the purview of NetBackup, you need to replicate the DSUs from one NearStore system (SystemA) to a second NearStore system (SystemB). In the case of a failure where the volumes on SystemA are unrecoverable (that is, when either the entire SystemA is unavailable or when volumes on SystemA are destroyed and cannot be recovered), you need to manually break the SnapMirror relationship, which enables the replicated volumes on SystemB for read and write access. Only those volumes that are replicated can be recovered. After a few configuration adjustments in NetBackup, all future backups and restores to the volumes that were replicated can now be served by SystemB.

You can subsequently reconfigure NetBackup to direct restores, verifications, deletions and backups back to SystemA. Your disaster recovery plan should include NetBackup catalog backup and restore procedures. For more information regarding catalog backup and restore, see the *VERITAS NetBackup System Administrator's Guide*.

Next topics

Configuring SnapVault for NetBackup with volume SnapMirror on page 318 Actions required in the event of a NearStore system failure on page 319 Returning to the initial setup after the failed system has recovered on page 320 Use cases of the commands used in the failover process on page 321

Configuring SnapVault for NetBackup with volume SnapMirror

To set up a disaster recovery solution for SnapVault for NetBackup using volume SnapMirror, you need to configure volume SnapMirror relationships for the DSUs to be protected on the NearStore systems.

For each DSU that needs protection, the DSU's volume on the NearStore system needs to be configured as the source of a volume SnapMirror relationship. For details on how to configure volume SnapMirror relationships, refer to the volume SnapMirror section of this document. The destination of each volume SnapMirror relationship needs to reside on a different NearStore system than the source and have a name identical to that of the source volume.

Note: Identical names for the source and the destination volumes is not a generic requirement for volume SnapMirror when used outside the context of SnapVault for NetBackup. It is not in any

way enforced when configuring volume SnapMirror relationships. However, for failovers to work in the context of SnapVault for NetBackup, the volume names must be identical on the SnapMirror source and destination.

Steps

- 1. Configure both NearStore systems (SystemA and SystemB) with the required licenses and option settings to allow SnapVault for NetBackup as well as volume SnapMirror.
- 2. For NetBackup, the initial configuration remains the same. Create NetBackup policies for DSUs that specify SystemA as the NearStore system. In addition to the usual configuration for backing up data to volumes on SystemA, make sure to add SystemB to the list of NDMP Hosts in NetBackup.

For more information on how to add a NearStore system as an NDMP Host in NetBackup, see the NetBackup documentation.

3. Configure catalog backups if protecting the catalog is part of your disaster recovery plan.

For configuring catalog backups, see the *NetBackup Administrator's Guide* for the given platform.

- 4. On SystemB, create volumes with names identical to those volumes to be protected on SystemA.
- **5.** On SystemB, configure one SnapMirror relationship for each volume to be protected. Each volume will have a volume on SystemA as the source and a volume with the exact same name on SystemB as the destination.

Note: Configure a SnapMirror schedule with relatively frequent updates to minimize any data mismatches between the source and destination in the case of a disaster.

Related concepts

About moving SnapMirror sources on page 167

Actions required in the event of a NearStore system failure

In the event that SystemA becomes unavailable, the backup administrator must manually break the applicable SnapMirror relationships on SystemA and then make a few configuration adjustments in NetBackup to allow NetBackup to fail over from volumes on SystemA to volumes on SystemB.

Steps

- 1. To enable read and write access to the replicated volumes on SystemB, break the SnapMirror relationships by issuing a snapmirror break command for each of the replicated volumes on SystemA.
- 2. To enable SystemA to restore images that had been backed up to SystemB, add an entry to the / usr/openv/netbackup/bp.conf file on UNIX, or add a multi-string type registry key on

Windows. One entry or registry key is required for each replicated volume and all affected media servers must be updated.

The exact content of the entry or the registry key is illustrated in the disaster recovery example later in this section.

- 3. Because the SnapMirror replication is performed asynchronously, there is a small chance that images might not have been fully replicated before SystemA went down. A new option to the bpstsinfo command-line utility in NetBackup compares the contents of the NetBackup catalog with the list of images reported by the NearStore system. This command compares the catalog entries for images on the failed SystemA volumes and the current SystemA volumes. It pulls a list of images on SystemA and compares that list to the images that SystemA reports that it has. It generates as output any images that are not present both in the catalog and on SystemA as well as the location of that image, whether it is only in the catalog or on media. This informs the administrator of any discrepancy so that appropriate action can be taken.
- 4. Change the NearStore system field in each NetBackup storage unit from SystemA to SystemB.
- 5. If the catalog had been backed up to SystemA and the disaster necessitates the recovery of the catalog, see the information about recovering the catalog in the *NetBackup Administrators Guide* for the platform.

Note: Catalog backups are restricted to NearStore systems that do not have file system exposure enabled.

Related references

Use cases of the commands used in the failover process on page 321

Returning to the initial setup after the failed system has recovered

When SystemA is backed up, you can again adjust NetBackup settings and SnapMirror relationships on the NearStore systems to reinstate the initial configuration on which NetBackup sends backups to volumes on SystemA. These volumes are replicated to SystemB.

Before making any changes in NetBackup, volumes on SystemA must be brought up-to-date with the current content of the corresponding volumes on SystemB. After this is complete, configure NetBackup to send backups to SystemA.

Steps

- 1. To sync each SystemA's volumes to the most recent content on the corresponding volume on SystemB, issue the snapmirror resync command for each of the previously replicated volumes, specifying SystemB as the source for the resync.
- 2. After the resync is complete, release the SnapMirror relationship for each of the replicated volumes on SystemA by running the snapmirror release command for each of the replicated volumes on SystemA.
- **3.** On SystemB, for each volume, delete the base Snapshot copies that were originally used by SnapMirror while transferring the data from SystemA to SystemB.

- 4. Confirm that the relationships are intact by issuing the snapmirror update command on each applicable volume on SystemA.
- 5. Break the replicated relationship for each applicable volume on SystemA.
- 6. Resync each volume from SystemB back to SystemA, run the snapmirror resync command for each of the DSU volumes on SystemA.
- 7. Break the SnapMirror relationship for each applicable volume on SystemA, by using the snapmirror break command.
- 8. Resync each of the applicable volumes from SystemA to SystemB.
- 9. Repeat steps 2-4, on SystemB.
- **10.** Update the bp.conf entries or the Windows registry keys that were added in the failover procedure to redirect access of backups made to SystemB to SystemA.
- 11. Change the NearStore server field in each affected NetBackup system from SystemB to SystemA.

Use cases of the commands used in the failover process

The following use cases are based on this representative environment: Two media servers (mediaserver1 and mediaserver2) that back up to volumes on a NearStore system (SystemA). SystemA contains volumes that are replicated, using SnapMirror, to another NearStore system (SystemB). Mediaserver1 is backing up to volume1 on SystemA, and mediaserver2 is backing up to volume2 on SystemA.

Next topics

Use case: disaster recovery failover scenario on page 321 *Use case: disaster recovery resync scenario* on page 324

Use case: disaster recovery failover scenario

SystemA is unavailable, and SystemB is configured as the new DSU. The following use case is based on this representative environment:

- Two media servers, mediaserver1 and mediaserver2, that back up to volumes on a NearStore system (that is, SystemA).
- SystemA contains volumes that are replicated, using SnapMirror, to another NearStore system (that is, SystemB).

Steps

1. To enable SystemB, issue snapmirror break on the two volumes on SystemB:

SystemB> snapmirror break volume1 SystemB> snapmirror break volume2

On media server	The following is the content of the bp.conf entry (on UNIX)/registry key (on Windows)
mediaserver1	NEARSTORE_FAILOVER_SERVER = SystemA:/vol/volume1 SystemB:/vol/volume1
mediaserver2	NEARSTORE_FAILOVER_SERVER=SystemA:/vol/volume2 SystemB:/vol/volume2

2. Add a bp.conf entry for each volume that was being failed over on each UNIX media server.

Mediaserver1

• If mediaserver1 is a UNIX server, add the following NEARSTORE_FAILOVER_SERVER entry to the /usr/openv/netbackup/bp.conf file:

NEARSTORE_FAILOVER_SERVER = SystemA:/vol/volume1 SystemB:/vol/volume1

• If mediaserver1 is a Windows server, add a registry key at this location: KEY_LOCAL_MACHINE\SOFTWARE\VERITAS\NetBackup\ConcurrentVersion\Config The key should be named NEARSTORE_FAILOVER_SERVER and be of type multi-string value. The value should be:

SystemA:/vol/volume1 SystemB:/vol/volume1

For exact location of the registry key for Windows, see the NetBackup Administrators Guide.

Mediaserver2

• If mediaserver2 is a UNIX server, add the following NEARSTORE_FAILOVER_SERVER entry to the /usr/openv/netbackup/bp.conf file:

NEARSTORE_FAILOVER_SERVER=SystemA:/vol/volume2_SystemB:/vol/volume2

• If mediaserver2 is a Windows server, add a registry key at this location: KEY_LOCAL_MACHINE\SOFTWARE\VERITAS\NetBackup\ConcurrentVersion\Config The key should be named NEARSTORE_FAILOVER_SERVER and be of type multi-string value.

The value should be:

SystemA:/vol/volume2 SystemB:/vol/volume2

The NearStore hostname must match the listing in the **Storage Device > NDMP Hosts** section in NetBackup. For example, if you add SystemA as an NDMP Host to the NetBackup Storage Device by the name *SystemA* and add SystemB by the name *SystemB.domain.com*, the entry will read:

SystemA:/vol/volume2 SystemB.domain.com:/vol/volume2

3. To display differences between the contents on the catalog, and contents on the DSU on each media server, issue the bpstsinfo -comparedbandstu command. The following is the command for mediaserver1:

/usr/openv/netbackup/bin/admincmd/bpstsinfo -comparedbandstu oldservervolume SystemA:/vol/volume1 -servername SystemB -serverprefix ontap: -lsuname /vol/volume1

The following is the command for mediaserver2:

```
/usr/openv/netbackup/bin/admincmd/bpstsinfo -comparedbandstu -
oldservervolume SystemA:/vol/volume2 -servername SystemB -serverprefix
ontap: -lsuname /vol/volume2
```

If no discrepancy exists between the images in the catalog and images on the NearStore volume, no images are listed in the output. The output will contain only debug logging:

```
STS: STH_ESDEBUG: comparedbandstu: libsts openp() 06/09/12 10:48:37:
opening module /usr/openv/lib/libstspibasicdiskMT.so
STS: STH_ESDEBUG: comparedbandstu: libsts openp() 06/09/12 10:48:38:
opening module /usr/openv/lib/libstspinearstoreMT.so
Nearstore disk subtype
```

If a mismatch exists—for example, if one image did not get replicated to SystemB before SystemA went down—the output contains additional information:

```
STS: STH_ESDEBUG: comparedbandstu: libsts openp() 06/09/12 10:39:37:
opening module /usr/openv/lib/libstspibasicdiskMT.so
Nearstore disk subtype
ONLY IN CATALOG
imagename:mediaserver1_1157990060 policy:powerpoint_backup copy_num:1
frag_num:0 is_header:TRUE resume_num:0
ONLY IN CATALOG
imagename: mediaserver1_1157990060 policy: powerpoint_backup copy_num:1
frag_num:1 is header:FALSE resume_num:0
```

Use the NetBackup GUI or run the bpexpdate command to remove the entries for this image from the NetBackup catalog.

Note: Two images are listed, but they represent only one backup. There are separate files for the header and the actual backup data. If this backup is enabled for True Image Restore, three images are printed. The key is to look at the imagename field. Both images have the same image name, which indicates that these images are part of the same backup.

The following is the bpexpdate command to remove the entry from the catalog:

```
/usr/openv/netbackup/bin/admincmd/bpexpdate -backupid
mediaserver1_1157990060 -d 0
```

- **4.** If the original volumes on SystemA are never expected to recover (that is, no SnapMirror resync to back up the volume), perform the following steps to manually delete the Snapshot copies used by the SnapMirror transfer on volumes, volume1 and volume2.
 - **a.** Issue snapmirror status -1 on the destination volume on SystemB:

```
SystemB> snapmirror status -1 vsmtest1
```

An output similiar to the following sample output will appear:

```
SystemB> snapmirror status -l volume1
SnapMirror is on.
Source: SystemA:volume1
Destination: SystemB:volume1
Status: Idle
Progress: -
```

```
State:Broken-offLag:00:05:28Mirror Timestamp:Fri Sep 29 10:39:28 PDT 2006Base Snapshot:SystemB(0101178726)_volume1.2Current Transfer Type:-Current Transfer Error:-Contents:ReplicaLast Transfer Type:ResyncLast Transfer Size:72 KBLast Transfer Duration:00:00:03Last Transfer From:SystemA:volume1
```

b. Delete the base Snapshot copy associated with the volume:

SystemB> snap delete volume1 SystemB(0101178726)_volume1.2

- c. Repeat this procedure for volume2.
- 5. Remove any SnapMirror schedules that were added in the snapmirror.conf file for these volumes.
- **6.** Reconfigure the storage units originally configured to the volumes on SystemA, to point to the storage units on SystemB using the following command:

```
/usr/openv/netbackup/bin/admincmd/bpsturep -label storage unit label -nh SystemB
```

Use case: disaster recovery resync scenario

SystemA is taken down for a period of time (possibly for maintenance), and SystemB is configured as the new DSU.

The administrator performs the steps outlined in the disaster recovery failover scenario, to enable SystemB as the new temporary DSU. At some later time, SystemA is back online and you want to use it again as the DSU. All data backed up to SystemB while SystemA was unavailable must now be transferred back to SystemA before re-enabling SystemA as the DSU. The following use case is based on this representative environment:

- Two media servers, (that is, mediaserver1 and mediaserver2), that back up to volumes on a NearStore system (that is, SystemA).
- SystemA contains volumes that are replicated using SnapMirror to another NearStore system (that is, SystemB).

To transfer back the data from SystemB to SystemA, and use it as the DSU for NetBackup, complete the following steps:

Steps

1. Resync the volumes from SystemB to SystemA.

```
SystemA> snapmirror resync -S SystemB:volume1 volume1
SystemA> snapmirror resync -S SystemB:volume2 volume2
```
2. Run the snapmirror release commands for both volumes on SystemA, as SystemA is no longer the replica source.

```
SystemA> snapmirror release volume1 SystemB:volume1
SystemA> snapmirror release volume2 SystemB:volume2
```

- **3.** Delete the base Snapshot copies on SystemB (that were originally used by SnapMirror to transfer data from SystemA to SystemB) because these Snapshot copies are not required.
- **4.** To confirm that relationships are intact, run the following command for both volumes of SystemA:

```
SystemA> snapmirror update -S SystemB:volume1 volume1
SystemA> snapmirror update -S SystemB:volume2 volume2
```

5. Break the replicated relationship on SystemA.

SystemA> snapmirror break volume1 SystemA> snapmirror break volume2

6. Resync the volumes from SystemA to SystemB.

SystemB> snapmirror resync -S SystemA:volume1 volume1

SystemB> snapmirror resync -S SystemA:volume2 volume2

This operation might display a message similar to the following, and fail the resync operation:

```
The resync base snapshot will be: SystemA(0101179629)_volume1.4
These older snapshots have already been deleted from the source and
will be deleted from the destination:
    SystemB(0101178726)_volume1.20699
    SystemB(0101178726)_volume1.20698
Are you sure you want to resync the volume? y
Wed Sep 27 12:30:24 PDT [SnapMirror.dst.resync.success:notice]:
SnapMirror resync of volume1 to SystemA:volume1 successful.
Transfer started.
Monitor progress with 'snapmirror status' or the SnapMirror log.
SystemB> Wed Sep 27 12:30:26 PDT [SnapMirror.dst.snapDelErr:error]:
Snapshot SystemB(0101178726)_volume1.20699 in destination volume
volume1 is in use, cannot delete.
Wed Sep 27 12:30:27 PDT [SnapMirror.dst.err:error]: SnapMirror
destination transfer from SystemA:volume1 to volume1 :
SnapMirror transfer failed to complete.
```

If such a case occurs, complete the following steps to resolve this issue:

a. The SnapMirror resync operation would have replicated the volume. Therefore, issue a snapmirror break command to break the replica.

SystemB> snapmirror break volume1

b. Manually delete the appropriate Snapshot copy.

SystemB> snap delete volume1 SystemB(0101178726)_volume1.20699

c. Retry the resync operation.

You many need to repeat these steps a few times till no problems occur with the resync operation because of Snapshot copies.

- 7. Repeat steps 2-4, on SystemB.
- 8. Remove the NEARSTORE_FAILOVER_SERVER entries from the bp.conf file (or registry keys on Windows) that were added during the failover setup (remove what was added in step 2 of the Disaster Recovery Failover use case). After removing those entries, redirect the path from SystemB to SystemA. This new path ensures proper access to backups that were made on SystemB when SystemA was unavailable. The final configuration for each media server is as follows:

On media server	The following is the content of the bp.conf entry (on UNIX)/registry key (on Windows)	
mediaserver1	NEARSTORE_FAILOVER_SERVER = SystemB:/vol/volume1 SystemA:/vol/volume1	
mediaserver2	NEARSTORE_FAILOVER_SERVER = SystemB:/vol/volume2 SystemA:/vol/volume2	

Mediaserver1

• If mediaserver1 is a UNIX server, remove the bp.conf entry that was added in step 2 of the Disaster Recovery Failover use case, and then add the following new NEARSTORE_FAILOVER_SERVER entry to the /usr/openv/netbackup/bp.conf file:

NEARSTORE_FAILOVER_SERVER = SystemB:/vol/volume1 SystemA:/vol/volume1

• If mediaserver1 is a Windows server, remove the registry key that is added in step 2 of the Disaster Recovery Failover use case, and then add a registry key at this location: KEY_LOCAL_MACHINE\SOFTWARE\VERITAS\NetBackup\ConcurrentVersion\Config.
It should be named NEARSTORE_FAILOVER_SERVER and be of type multi-string value.
The value should be:

SystemB:/vol/volume1 SystemA:/vol/volume1

For exact location of the registry key for Windows, see the NetBackup Administrators Guide.

Mediaserver2

• If mediaserver2 is a UNIX server, remove the bp.conf entry added in step 2 of the Disaster Recovery Failover use case, and then add the following new NEARSTORE_FAILOVER_SERVER entry to the /usr/openv/netbackup/bp.conf file:

NEARSTORE_FAILOVER_SERVER = SystemB:/vol/volume2 SystemA:/vol/volume2

• If mediaserver2 is a Windows server, remove the registry key that was added in step 2 of the Disaster Recovery Failover use case, and then add a registry key: The new key should be located in: KEY_LOCAL_MACHINE\SOFTWARE\VERITAS\NetBackup \ConcurrentVersion\Config. It should be named NEARSTORE_FAILOVER_SERVER and be of type multi-string value. The value should be:

SystemB:/vol/volume2 SystemA:/vol/volume2

9. Reconfigure the storage units that were originally configured to the volumes on SystemB, to point to the storage units on SystemA, by using the following command:

/usr/openv/netbackup/bin/admincmd/bpsturep -label storage unit label -nh SystemA

How to access backups on the replica of a SnapVault for NetBackup volume

Volume SnapMirror replication backs up an entire SnapVault for NetBackup volume to another system as a read-only volume. You can access this backed-up read-only volume using NetBackup. This capability is useful if you want to restore backups from the read-only volume to a different NetBackup master and media server combination or if you want to transfer the restored backups to NetBackup managed tape media as a long-term archive.

For more information about configurations, limitations, and accessing the backed-up SnapVault for NetBackup volume using NetBackup, see KB39312, *Restoring SnapMirror mirrored volumes to NetBackup servers using NetBackup*, available on the NOW site.

Related information

https://now.netapp.com/Knowledgebase/solutionarea.asp?id=kb39312

Troubleshooting for SnapVault for NetBackup

To troubleshoot, see the error messages displayed on the secondary storage system console. You can also see one of two log files, the /etc/log/nbu_snapvault log file or the /etc/log/ snapmirror log file.

Note: In takeover mode, all actions performed on behalf of the local node are written to the local node's /etc/log/xxx files. The actions that are performed on behalf of the partner node are written to the partner node's/etc/log/xxx files.

Next topics

SnapVault for NetBackup error messages on page 327 *Troubleshooting tools* on page 331

SnapVault for NetBackup error messages

The following error messages are associated with SnapVault for NetBackup. A corrective action is described, if one exists.

Next topics

Giveback cancel error on page 328 *Giveback veto error* on page 328 *Write failure to the protocol log file* on page 328 *Qtree deletion error* on page 328 Rectifying UTF translation error on page 329 Volume language warning on page 329 Platform mismatch error on page 331 Known issues associated with the exported file system on page 331 What to do if you run out of Snapshot copies on page 331

Giveback cancel error

You can resolve the cancel error "SnapVault for NetBackup transfer to the taken-over node is in progress; canceling giveback."

Corrective action—When issuing a giveback command, ensure that no SnapVault for NetBackup backup operations involving the taken-over node are in progress.

Alternatively, use the giveback -f command, then restart all interrupted backups after the giveback command.

Giveback veto error

You can resolve the giveback error "Cluster monitor: nbu: giveback cancelled due to active state."

Corrective action—When issuing a giveback command, ensure that no SnapVault for NetBackup backup operations involving the taken-over node are in progress.

Alternatively, use the giveback -f command, then restart all interrupted backups.

Write failure to the protocol log file

You can resolve the error "An error was encountered writing to the Veritas protocol log file."

Corrective action—Ensure that the root volume is not full.

Qtree deletion error

Deletion of a SnapVault qtree, created using NetBackup, failed on a DSU.

Corrective action—Create enough free space on the volume and manually delete the qtree.

To create space and delete the qtree, complete the following steps.

1. On the NearStore system console, use the df command to check if the volume in which the qtree resides is full.

2. If the volume is full, either free space or add more disks to the volume.

Rectifying UTF translation error

You can resolve the error "A file or directory name in a backup was either longer than 255 unicode characters or the name is not a valid UTF-8 byte sequence."

Corrective action—On the NetBackup client, rename the file or directory and use valid UTF-8 characters.

Steps

1. On the NearStore system console, use the snapvault status command to obtain the base and policy names for the backup. The base name should indicate the name of the NetBackup client.

Example

snapvault status /vol/volname/qtreename

Note: The base name should indicate the name of the NetBackup client.

- 2. On the NetBackup client, find the file or directory using the full path to the file or directory listed in the error message.
- 3. As required, choose one of the actions from the following table.

If	Then
If the NetBackup client is not running a Microsoft operating system	Rename the file or directory to something that uses 255 bytes or less.
If the NetBackup client is running a Microsoft operating system	If the NetBackup client is running a Microsoft operating system, rename the file directory, or alternate data stream to something no longer than 255 Unicode characters, where each Unicode character translates to three bytes or fewer in the UTF-8 character set. (Names that use ASCII characters satisfy this requirement.)

Volume language warning

You can resolve the error "A file or directory name containing characters that SnapVault interprets as UTF-8 characters, was backed up to a volume whose volume language setting does not specify UTF-8 as the NFS character set."

Corrective action—Back up to a volume whose volume language setting specifies UTF-8 as the NFS character set. This can be accomplished in one of two ways.

Next topics

Rectifying volume language warning on page 330 *Resolving volume language warnings* on page 330

Rectifying volume language warning

You can rectify the volume language warning even if the volume does not contain only NetBackup DSU qtrees.

Steps

1. On the NearStore system console, use the snapvault status command to obtain the base and policy names for the backup.

```
snapvault status /vol/vol_name/qtree_name
```

The base name should indicate the name of the NetBackup client.

- 2. Find or create a volume on the NearStore system that uses UTF-8 characters as its NFS character set.
- **3.** Using the NetBackup GUI or CLI, direct future operations for this base and policy name to the volume that uses UTF-8 characters as its NFS character set.
- 4. Perform a full backup to the volume identified in Step 2.

Resolving volume language warnings

You can resolve volume language warnings if the volume contains only NetBackup DSU qtrees.

Steps

1. Use the vol lang command to determine the volume language type.

Example

vol lang vol1
Volume language is C (POSIX)

2. Use the vol lang command to change the volume language by appending UTF-8.

Example

vol lang vol1 C.UTF-8

- 3. Reboot the NearStore system for the new language setting to take effect.
- 4. Perform a full backup for each secondary qtree on a DSU that uses SnapVault for NetBackup.

Platform mismatch error

You can resolve the error "A NetBackup media server attempted to establish a connection to the DSU on an unsupported storage system."

Corrective action—Ensure that the secondary storage system is either a NearStore system or a storage system with thenearstore_option license enabled. Ensure that all other necessary licenses are enabled.

Known issues associated with the exported file system

The following are known issues with the implementation of exported file systems in SnapVault for NetBackup.

- NetBackup sends certain files, backed up using a NetBackup 4.5 client, as opaque files. Opaque files are files that are not in the exported file system, but are present in the tar image.
- UNIX special files, such as pipes and fifos, are not in the exported file system, but are present in the tar image.
- Windows sparse files and encrypted files show zero size in the exported file system.

What to do if you run out of Snapshot copies

If you run out of Snapshot copies, do not delete archival Snapshot copies manually. Instead, you should use NetBackup to delete these Snapshot copies.

Running out of Snapshot copies is a common problem and one you can avoid using the following guidelines:

- Put data whose policies have different backup schedules in separate volumes. For example, if one policy has a weekly full schedule and another policy has a monthly full schedule, send backed-up data to separate volumes.
- Put data whose policies have different retention periods in separate volumes. For example, if one policy has a retention period of two months and another policy has a retention period of six months, send backed-up data to separate volumes.
- Turn off the archival Snapshot copy option.

Related concepts

How to manage consumption of Snapshot copies on page 304

Troubleshooting tools

There are two troubleshooting tools to resolve issues in a NetBackup environment. One tool to identify unnecessary images in a NetBackup environment and the other tool to aggregate backup monitoring information.

Tool that identifies unnecessary images in a NetBackup environment

Orphaned and stale images are unnecessary images that can prematurely fill a DSU, which might cause subsequent backups to the DSU to fail. You can use this tool to find orphaned images and stale

images in a specified DSU. Based on the results of running this tool, you can manually remove orphaned images and stale images from the DSU.

Note: Stale images are backup images that exist on the DSU and NetBackup catalog after exceeding their retention periods. You should remove stale images, because these images unnecessarily use storage.

Tool that aggregates backup monitoring information

You must know the following information to adequately size DSUs and ensure smooth implementation of the SnapVault for NetBackup feature:

- The change rate for incremental backups
- The amount of space savings between two subsequent full backups
- The space savings for incremental backups
- The variance in size for full backups
- The space savings at a client level or a policy level

You can use the backup monitoring tool to aggregate this information and help you with the sizing and implementing activities. For more information about these tools, contact technical support.

Data replication using volume copy

You can use the vol copy set of commands to copy all data from one volume to another, either on the same system or on a different system.

For using the vol copy command, the source and destination volumes must be of the same type: traditional volumes or FlexVol volumes.

The vol family of commands manages volumes. A volume is a logical unit of storage, containing a file system image and associated administrative options such as Snapshot copy schedules. The disk space that a volume occupies (as well as the characteristics of the RAID protection it receives) is provided by an aggregate (see na_aggr(1)).

You can initiate a volume copy with the vol copy start command, which enables you to copy data from one volume to another volume, either on the same or on a different storage system. The result is a restricted volume containing the same data as the source volume at the time you initiated the copy operation.

Next topics

Benefits of using volume copy on page 333 When to copy volumes on page 334 IPv6 support with volume copy on page 335 Prerequisites before copying a volume on page 335 Copying volumes using the vol copy command on page 339 Checking the status of a volume copy operation on page 343 Displaying the current speed for copying a volume on page 345 Controlling a volume copy operation speed on page 345 Aborting a volume copy operation on page 346

Benefits of using volume copy

Although you can copy data from a system using client programs such as cpio or use the Data ONTAP dump and restore commands, the vol copy command set offers several benefits.

- When a vol copy command reads and writes data, Data ONTAP does not traverse directories on the system. Data is copied block for block directly from the disks, which means that Data ONTAP can finish the copying faster than it could with other methods.
- Using a vol copy command, Data ONTAP preserves the Snapshot copy data of the source volume. If, in the future, users might need to use Snapshot copies that were taken before data was copied from one volume to another, you can use a vol copy command for migrating data. For example, if users accidentally delete files and need to recover them, they can do so from the preserved data.

• You do not need any additional licenses to use the vol copy command.

Maximum number of concurrent volume copy operations: Volume copy has the same limit on concurrent replication operations as volume SnapMirror.

Related references

Maximum number of concurrent replication operations on page 128

When to copy volumes

You might find copying volumes useful under certain situations.

The following table describes some situations where you might find copying volumes useful.

Situation	Reasons for copying one volume to another	
You want to migrate data from one storage system to another.	The destination storage system has more storage or is a model that supports newer technology.	
You want to move a volume from one set of disks to another on the same storage system.	 You want to Split a volume Expand storage Examples: You can copy the vol0 volume to the vol1 volume and then delete duplicate files and directories in these volumes so that the original contents of vol0 are split into two volumes. You have six 9-GB disks for the vol0 volume and four 18-GB spare disks. You can migrate vol0 to the four 18-GB disks and replace all the 9-GB disks with larger capacity disks. 	
You want to copy data from one storage system to another regularly to ensure high data availability.	 After you copy the data, clients can switch to the destination storage system in the following scenarios: When you shut down the storage system for software or hardware upgrades, or when the storage system is not available for reasons such as natural disasters, you can put the destination volume online to continue file service. If a network client process accidentally deletes a large number of files on the storage system, clients can continue to have access to the files on the destination storage system while you are restoring the files to the source system. Note: This scenario is also a good application for SnapMirror. 	

IPv6 support with volume copy

When copying volumes using the vol copy command, you can use IPv6 addresses to specify source and destination systems. However, there are some differences between the specification of IPv6 and IPv4 addresses.

Note: Before using IPv6 functionality for a system, ensure that the ip.v6.enable option is set to on.

The usage is shown in the following example.

Use of IPv6 functionality with the vol copy start command

```
vol copy start [-p {inet | inet6}] [src_system:]src_vol
[dst_system:]dst_vol
```

- The -p option enables to specify an IPv4 or IPv6 connection mode, by using one of the following two options.
 - inet specifies the use of an IPv4 connection.
 - inet6 specifies the use of an IPv6 connection.
- src_system is the name of the source system, and src_vol is the name of the source
 volume.
- dst_system is the name of the destination system, and dst_vol is the name of the destination volume.

If the IP addresses of the source or destination systems are specified, the -p option is not required. For example, if IPv6 addresses are specified, the connection mode is IPv6.

If host names are used instead of IP addresses without using the -p option, first an IPv6 connection is attempted. If an IPv6 connection is not established, an IPv4 connection is attempted.

If host names are used instead of IP addresses with the -p option, the connection is attempted as specified by the -p option.

Prerequisites before copying a volume

You should make sure that systems involved in a vol copy operation meet certain requirements.

Requirements

• The source and destination volumes must be of the same type: either both traditional or both FlexVol volumes.

- The capacity of the destination volume must be greater than or equal to the capacity of the source volume.
- The source and destination storage systems must have a trust relationship with each other.
- The destination volume must exist, and must not be the root volume.
- The source volume must be online and the destination volume must be restricted.
- Remote Shell access must be enabled.
- The destination volume must not contain data that you want to preserve.

The rest of this section provides more detailed information about verifying whether the source and destination volumes meet these requirements:

You should take care not to overwrite data that you need: If the destination volume is not a new volume, ensure that it does not contain data that you might need in the future. After Data ONTAP starts copying the source volume, it overwrites the entire destination volume. All data in the active file system and in the Snapshot copies of the destination volume is lost after Data ONTAP starts copying the data.

Where volume copies can reside: The source and destination volumes of the copy can reside on the same or on different storage systems.

Recommendation for copying a volume: When a system copies data between two volumes on separate systems, it floods the network between the two systems with packets. Users of the systems involved in a volume copy operation might notice a degradation in response time during the copy. A private network for copying between the source and destination systems helps circumvent network-related performance problems when copying to a different systems.

Maximum number of simultaneous volume copies: Volume copy has the same limit of simultaneous copies that SnapMirror replications have.

Next topics

Verifying the size of each volume on page 336 Verifying the relationship between systems on page 337 Verifying and changing the status of source and destination volumes on page 337 Enabling remote access on page 338

Verifying the size of each volume

To see whether the data in one volume can be copied or replicated to another volume, you need to compare the file system size of the two volumes.

Steps

1. On the source storage system, enter the following command:

vol status -b volume_name

volume_name is the name of the source volume.

Result: Data ONTAP displays the block size of the volume (in bytes), the RAID volume size, and the Write Anywhere File Layout (WAFL) file system size. If no volume name is given, information for all volumes is displayed.

- 2. On the destination storage system, repeat Step 1, replacing *volume_name* with the name of the destination volume.
- **3.** Compare the file system (FS) numbers. If the file system size of the destination is the same as or larger than the file system size of the source, you can use the vol copy command (or SnapMirror) to transfer data from the source to the destination.

Example

```
vol status -bVolumeBlock Size (bytes) Vol Size (blocks) FS Size (blocks)-----------sourcevol409643467524346752destvol409643467524346752
```

Verifying the relationship between systems

You can verify the relationship between systems. If the source and destination volumes in a vol copy operation reside on two different systems, these systems must have a trust relationship with each other.

Steps

1. By using FilerView or mounting the system with NFS, enter the destination system host name in the /etc/hosts.equiv file of the source system, if it is not present already.

The /etc/hosts.equiv file contains a list of host names, each of which is on a separate line. The presence of a host name in this file indicates that the system allows that host to perform remote operations.

2. Repeat Step 1 on the destination system, entering the source system host name in the /etc/ hosts.equiv file, if it is not present already.

Verifying and changing the status of source and destination volumes

You can verify whether the source volume is online, and that the destination volume exists and is restricted. You can also change the status of a volume when necessary.

Before you begin

The destination volume for a vol copy operation cannot be the root volume. This is because the destination volume must be offline when Data ONTAP executes the vol copy command, and a root volume must always be online.

Steps

1. To verify that the destination volume exists and is restricted, enter the following command on the destination storage system:

vol status dest_volume

dest_volume is the name of the volume whose status you want to check.

If you do not provide a volume name, the command displays the status of all volumes in the storage system.

If the volume does not exist, Data ONTAP returns an error. For information about how to create a volume, see the *Data ONTAP System Administration Guide*.

- 2. To verify that the source volume is online, repeat Step 1 on the source storage system, replacing dest_volume with the name of the source volume.
- **3.** If you need to change the status of a volume because of the results of Step 1, enter the following command on the destination storage system:

vol restrict dest_volume

dest_volume is the name of the destination volume.

4. If you need to change the status of a volume because of the results of Step 2, enter the following command on the source storage system:

vol online source_volume

source_volume is the name of the source volume.

5. If you needed to perform Step 3 or Step 4, you might want to perform Step 1 or Step 2 again to verify the changes that you made.

Example

```
systemA> vol status
Volume State Status Options
vol0 online normal root
vol1 online normal raidsize=14
vol2 online restricted
volextra offline
```

Enabling remote access

To perform a volume copy from one volume to another volume on the same storage system, Remote Shell services must be enabled or the volume copy fails.

Step

1. To enable Remote Shell services, enter the following command:

options rsh.enable on

Copying volumes using the vol copy command

You can use the vol copy start command to generate volume copy operations, which produce screen messages that show the progress of the operations.

Each vol copy start command generates two volume copy operations, each of which is assigned a number:

- One operation is for reading data from the source volume. Screen messages displayed by a vol copy command refer to this operation as the volcopy dump operation.
- One operation is for writing data to the destination volume. Screen messages displayed by a vol copy command refer to this operation as the volcopy restore operation.

When to use the volume copy operation number: You need the volume copy operation number if you want to stop a volume copy operation or change the volume copy operation speed. To find the vol copy operation number, you can use the vol copy status command.

Next topics

Number of vol copy operations supported on page 339 Copying Snapshot copies with the vol copy start command on page 340 Copying one volume to another volume using the vol copy command on page 341 Using volume copy to copy LUNs on page 343

Related tasks

Checking the status of a volume copy operation on page 343

Number of vol copy operations supported

Whether Data ONTAP can execute a vol copy start command depends on how many volume copy operations are already in progress on the storage systems specified in the vol copy start command.

To copy volumes locally, you can enter the following two vol copy start commands on a storage system, which supports four simultaneous copy operations:

```
vol copy start vol0 vol1
vol copy start vol2 vol3
```

When these commands are in progress, if you enter additional vol copy start commands, they will fail, because four volume copy operations are already running on the system. Two of

the operations are for reading the vol0 and vol2 volumes, and two of the operations are for writing the vol1 and vol3 volumes.

Suppose you enter the following three vol copy start commands on a storage system named systemA to copy volumes to another storage system named systemB:

vol copy start vol0 systemB:vol0 vol copy start vol1 systemB:vol1 vol copy start vol2 systemB:vol2

When these commands are in progress, systemA runs three volume copy operations to read the volumes, and systemB runs three volume copy operations to write the volumes.

An additional vol copy start command to copy *between* systemA and systemB will succeed because the command adds one more volume copy operation to each storage system.

However, if you enter an additional vol copy start command to copy volumes *locally* on either systemA or systemB, it will fail. This is because the additional command creates two volume copy operations, one for reading and one for writing, on the storage system that performs the local copying.

Copying Snapshot copies with the vol copy start command

The following table describes the Snapshot copies that will be copied from the source volume and the resulting Snapshot copies on the destination volume, depending on the option you use with the vol copy start command.

Option	Snapshot copies to copy from the source volume	Snapshot copies in the Snapshot file system of the destination volume
None	No Snapshot copies are copied. Only the Snapshot copy taken after you enter the vol copy start command, are copied.	A Snapshot copy named snapshot_for_volcopy. <i>n</i> is created, where <i>n</i> is a number starting at 0 and incrementing by one whole number with each vol copy operation is created.
-S	All Snapshot copies in the Snapshot file system of the source volume, and the Snapshot copy taken after you enter the vol copy start command, are copied.	All Snapshot copies in the source volume, and snapshot_for_volcopy. <i>n</i> , where <i>n</i> is a number starting at 0 and incrementing by one whole number with each vol copy operation, are created.

Option	Snapshot copies to copy from the source volume	Snapshot copies in the Snapshot file system of the destination volume
-s followed by the name of the Snapshot copy	The specified Snapshot copy will be copied.	The specified Snapshot copy is created.

Note: The vol copy start -S command does not copy any Snapshot copies that are created while the copying is in progress. For example, if the copying lasts from 11:45 p.m. to 1:00 a.m. the next day and Data ONTAP creates a Snapshot copy named nightly.0 at midnight, Data ONTAP does not copy the nightly.0 Snapshot copy.

Copying one volume to another volume using the vol copy command

You can copy one volume to another volume, by using the vol copy command.

Step

1. Enter the following command on either the source or destination system:

```
vol copy start [-S | -s snapshot_name] source_volume dest_volume
```

The -s and -s arguments specify the Snapshot copies to copy.

source_volume and dest_volume are the names of the source and destination volumes. If a volume is on a different storage system, precede the volume name with the system name and a colon. For examples illustrating how to specify volume names, see "Examples of the vol copy start command" in the following table.

Note: If the copying takes place between two storage systems, you can enter the vol copy start command on either the source or destination storage system. However, you cannot enter the command on a third storage system that does not contain the source or destination volume.

Examples of the vol copy start command		
The following table shows several examples of the vol copy start command.		
If you want to	Use	
Copy all Snapshot copies from the vol0 volume to the vol1 volume on the same storage system	vol copy start -S vol0 vol1	
Copy a nightly Snapshot copy from the vol0 volume to the vol1 volume on the same storage system	vol copy start -s nightly.1 vol0 vol1	

If you want to	Use
Create a Snapshot copy in the vol0 volume to be copied to the vol1 volume on the same storage system	vol copy start vol0 vol1
Copy all Snapshot copies from the vol0 volume to the vol1 volume on a different storage system named systemA	vol copy start -S vol0 systemA:vol1

Error messages generated by vol copy start commands

If your storage system does not meet the requirements for copying a volume, the vol copy start command generates one or more error messages.

The following table explains the possible error messages and their meanings.

Error message	Meaning
Permission denied. VOLCOPY: Could not connect to system systemB	The source system does not have permission to copy to the destination storage system.
	Action: Ensure that the storage systems have a trust relationship with each other.
VOLCOPY: volcopy restore: volume is	The destination volume is online.
onitine, aborting	Action: Take the destination volume offline.
VOLCOPY: volcopy restore: volume is too small, aborting	The destination volume is smaller than the source volume.
	Action : Add more disk space to the destination volume or choose another destination volume of sufficient capacity.
write: setting up STDERR broken pipe	A local volume copy tried to start, but Remote Shell access is not enabled on the storage system.
	Action: Enable Remote Shell access on the storage system so that it can receive rsh commands.

Using volume copy to copy LUNs

You can use the vol copy command to copy LUNs; however, this requires that applications accessing the LUNs are quiesced and offline prior to the copy operation.

Before you begin

You must save contents of host file system buffers to disk before running vol copy commands on the storage system.

Note: The term *LUNs* in this context refer to the LUNs that Data ONTAP serves to clients, not to the array LUNs used for storage on a storage array.

About this task

The vol copy command enables you to copy data from one WAFL volume to another, either within the same storage system or to a different storage system. The result of the vol copy command is a restricted volume containing the same data that was on the source storage system at the time you initiate the copy operation.

Step

1. To copy a volume containing a LUN to the same or different storage system, enter the following command:

vol copy start -S source:source_volume dest:dest_volume

-S copies all Snapshot copies in the source volume to the destination volume. If the source volume has Snapshot copy-backed LUNs, you must use the -S option to ensure that the Snapshot copies are copied to the destination volume.

If the copying takes place between two storage systems, you can enter the vol copy start command on either the source or destination storage system. You cannot, however, enter the command on a third storage system that does not contain the source or destination volume.

Example

vol copy start -S /vol/vol0 filerB:/vol/vol1

Checking the status of a volume copy operation

You can use the vol copy status command to check the status of volume copy operations.

About this task

This command displays the status for a specified volume copy operation. If you do not specify the operation number, the command displays the status of all volume copy operations in progress. In the

command output, the operations are differentiated from one another with unique volume copy operation numbers.

Restrictions: Remember the following restrictions when checking volume copy status:

- If you start a volume copy operation from the system console, you can enter the vol copy status command only through the rsh command when the copy operation is in progress. This is because you do not have access to the system prompt on the console when Data ONTAP is copying the volume.
- If data is being copied between two storage systems, you can enter the vol copy status command through a remote shell connection to either system. The operation numbers displayed on the source system and the destination system are different because the reading and the writing are considered two different operations.

Step

1. Enter the following command:

vol copy status [operation_number]

operation_number is the specific volume copy operation.

Omit *operation_number* to display the status of all current volume copy operations. The operations are numbered from 0 through 3.

Sample status message from the vol copy start command

The following example shows a vol copy start command that copies the vol0 volume to the vol1 volume on the same storage system. When the operation is in progress, it displays the volume copy operation status.

```
systemA>vol copy start -S vol0 vol1
Copy Volume: vol0 on machine 127.0.0.1 to Volume: vol1
Reading the dump stream
VOLCOPY: Starting on volume 1.
This dump contains 257 blocks
10:04 pm : volcopy restore 1 : begun.
10:04 pm : volcopy restore 1 : 5 % done. Estimate 3 minutes
remaining.
.
.
.
10:04 pm : volcopy restore 1 : 95% done. Estimate 1 minutes
remaining.
```

The following example shows a vol copy status command using rsh.

Before the prompt is displayed again, you can use the vol copy status command on a trusted host of the storage system, as shown in the following example:

rsh systemA vol copy status
10:04 pm : volcopy dump 0 : 99 % done. Estimate 1 minutes remaining.
10:04 pm : volcopy restore 1 : 99 % done. Estimate 1 minutes

```
remaining.
No operation 2 in progress.
No operation 3 in progress.
```

In the previous examples, volume copy operation 0, shown as volcopy dump 0 in the display, is for reading the data from the vol0 volume; volume copy operation 1, shown as volcopy restore 1 in the display, is for writing the data to the vol1 volume.

Displaying the current speed for copying a volume

You can display the speed for copying a volume when you want to determine the current setting, and to verify the speed before changing the setting. This procedure enables you to verify the default speed for all volume copy operations.

Step

1. To display the speed for copying a volume, enter the following command:

options vol.copy.throttle

Result: The value 10 (full speed) through 1 (one-tenth full speed) to be used by all volume copy operations is displayed. The default value is 10.

Controlling a volume copy operation speed

You can control the speed of a volume copy operation before you start the volume copy operation and during a volume copy operation.

About this task

The speed for reading data from the source volume and the speed for writing data to the destination volume can be different. The slower of the two values determines the time required for Data ONTAP to finish copying the data. You can change the speed of a volume copy operation when you suspect it might cause performance problems on your storage system.

Note: Changing the vol.copy.throttle option changes the default speed for all volume copy operations to follow.

Step

1. To control volume copy operation speed, choose one of the actions from the following table.

If you want to control the speed of the volume copy operation	Then
Before starting the copy	Enter the following command:
operations	options vol.copy.throttle value
	value is the specific speed you want.
During the copy operation	Enter the following command through a Remote Shell:
	vol copy throttle [operation_number] value
	<i>operation_number</i> is the specific volume copy operation whose speed you want to adjust.
	If you do not specify an operation number, the command applies to all volume copy operations that are in progress.
	value is the specific speed you want.

Example

The following example illustrates changing the speed of all volume copy operations in progress to one-tenth of full speed through a Remote Shell:

rsh systemA vol copy throttle 1
volcopy operation 0: Throttle adjusted from 100% to 10%.
volcopy operation 1: Throttle adjusted from 100% to 10%.

Aborting a volume copy operation

If data is being copied between two storage systems, you can stop copying by executing the vol copy abort command on either storage system. If you start the volume copying operation from the system console, you can enter the vol copy abort command only through the rsh command. This is because you do not have access to the system prompt on the console during the copying.

Before you begin

To abort a specific volume copy operation, you need to specify the operation number. You can obtain the operation number from the vol copy status output.

Step

1. To abort a volume copy operation, enter the following command:

```
vol copy abort [all | operation_number]
```

operation_number is the specific volume copy operation to be aborted. Specify all to abort all operations.

Attention: An incomplete volume copy operation leaves unusable data in the destination volume.

Data mirroring using SyncMirror

You can use SyncMirror to mirror aggregates, and thus provide increased data resiliency. SyncMirror removes single points of failure in connecting to disks or array LUNs.

Note: SyncMirror is supported for V-Series systems also. Information about SyncMirror in this chapter applies to both FAS systems and V-Series systems, unless specified otherwise.

Next topics

What SyncMirror is on page 349 Advantages of using SyncMirror on page 350 What mirrored aggregates are on page 350 Requirements for using SyncMirror with disks on page 351 How SyncMirror works with third-party storage on page 351 Enabling the SyncMirror license on page 361 Considerations for using mirrored aggregates on page 363 Addition of disks or array LUNs to a mirrored aggregate on page 370 The states of a plex on page 374 Splitting a mirrored aggregate on page 375 Rejoining split aggregates on page 376 Removing a plex from a mirrored aggregate on page 377 Comparing plexes of a mirrored aggregate on page 378

What SyncMirror is

SyncMirror is an optional feature of Data ONTAP. It is used to mirror aggregates.

SyncMirror provides for synchronous mirroring of data, implemented at the RAID level. You can use SyncMirror to create aggregates that consist of two copies of the same WAFL file system. The two copies, known as plexes, are simultaneously updated. Therefore, the copies are always identical.

Note:

- The two plexes are directly connected to the same system.
- SyncMirror can be used to mirror aggregates and traditional volumes. (A traditional volume is essentially an aggregate with a single volume that spans the entire aggregate.)
- SyncMirror cannot be used to mirror FlexVol volumes. However, FlexVol volumes can be mirrored as part of an aggregate.
- SyncMirror is different from synchronous SnapMirror.

For more information about aggregates and volumes, see the *Data ONTAP Storage Management Guide*.

Advantages of using SyncMirror

A SyncMirror aggregate has two plexes. This setup provides a high level of data availability because the two plexes are physically separated.

For a system using disks, the two plexes are on different shelves connected to the system with separate cables and adapters. Each plex has its own collection of spare disks. For a system using third-party storage, the plexes are on separate sets of array LUNs, either on one storage array or on separate storage arrays.

Note: You cannot set up SyncMirror with disks in one plex and array LUNs in the other plex.

Physical separation of the plexes protects against data loss if one of the shelves or the storage array becomes unavailable. The unaffected plex continues to serve data while you fix the cause of the failure. Once fixed, the two plexes can be resynchronized.

Another advantage of mirrored plexes is faster rebuild time.

In contrast, if an aggregate using SnapMirror for replication becomes unavailable, you can use one of the following options to access the data on the SnapMirror destination (secondary).

- The SnapMirror destination cannot automatically take over the file serving functions. However, you can manually set the SnapMirror destination to allow read-write access to the data.
- You can restore the data from the SnapMirror destination to the primary (source) storage system.

An aggregate mirrored using SyncMirror requires twice as much storage as an unmirrored aggregate. Each of the two plexes requires a independent set of disks or array LUNs. For example, you need 2,880 GB of disk space to mirror a 1,440-GB aggregate—1,440 GB for each plex of the mirrored aggregate.

What mirrored aggregates are

A mirrored aggregate is a single WAFL storage file system with two physically separated and synchronously up-to-date copies on disks or array LUNs. These copies are called plexes. Data ONTAP typically names the first plex plex0 and the second plex plex1.

Each plex is a physical copy of the same WAFL file system, and consists of one or more RAID groups. As SyncMirror duplicates complete WAFL file systems, you cannot use the SyncMirror feature with a FlexVol volume—only aggregates (including all contained FlexVol volumes) are supported.

Related references

Considerations for using mirrored aggregates on page 363

Requirements for using SyncMirror with disks

For using SyncMirror to mirror aggregates, you need a SyncMirror license, systems that support the SyncMirror feature, and an appropriate configuration of disk shelves.

The following are prerequisites for using SyncMirror.

• You must enable the SyncMirror license to receive important product information before enabling the SyncMirror feature.

The SyncMirror license is available for free.

• The systems, or the active/active configuration, should support the SyncMirror feature.

Note: FAS200 series/GF270c systems do not support SyncMirror configurations.

• You must connect disk shelves to the storage system in a configuration that supports mirrored aggregates.

For more information about configurations that support mirrored aggregates, see the *Data ONTAP Active/Active Configuration Guide*.

For information about the disk shelves and other hardware supported with different storage systems, see the *System Configuration Guide*.

How SyncMirror works with third-party storage

For both aggregates composed of native disks and aggregates composed of array LUNs, SyncMirror creates two physically-separated copies of an aggregate.

These copies of the aggregate, called plexes, are simultaneously updated; therefore, the two copies of the data are always identical. Data continues to be served if one copy becomes unavailable.

For third-party storage, the physical separation of the plexes protects against data loss if the following occurs:

• An array LUN fails.

For example, a LUN failure can occur because of a double disk failure on the storage array.

- A storage array becomes unavailable.
- In a MetroCluster configuration, an entire site fails. An entire site could fail because of a disaster or prolonged power failure. If this situation occurs, the site administrator enters a command to enable the surviving node to take over the functions of the partner. Data is accessed on the plex of the surviving node.

For third-party storage, each plex must be on a separate set of array LUNs. The plexes can be in two physically separate locations on the same storage array, or each of the two plexes can be on a different storage array. In a MetroCluster configuration with third-party storage, each plex must be

on a separate set of LUNs on different storage arrays. (V-Series systems on which native disk shelves are installed cannot be deployed in a MetroCluster configuration.)

Data ONTAP needs to know whether a plex is local to the system on which the aggregate is configured or in a remote location. Local in the context of third-party storage means on the storage array connected to the V-Series system on which the aggregate is configured. The SyncMirror *pool* to which an array LUN is assigned provides the information that Data ONTAP needs to determine whether the plex is local or remote.

The following illustration shows the relationships of plexes and pools to an aggregate. One plex is associated with pool0 and one plex is associated with pool1. The number 0 is typically associated with the local pool and the number 1 is typically associated with the remote pool. The remote plex is the mirror of the aggregate.



Next topics

Implications of storage type when mirroring with V-Series systems on page 352 Requirements for setting up SyncMirror with array LUNs on page 353 SyncMirror pool assignment planning for third-party storage on page 357 Example of SyncMirror pool assignments for array LUNs on page 358 Common errors when setting up SyncMirror pools with array LUNs on page 360 Troubleshooting errors with SyncMirror pool assignment for array LUNs on page 360

Implications of storage type when mirroring with V-Series systems

For V-Series systems, you can mirror data only between the same types of storage.

You can mirror aggregates on a V-Series system as follows:

- If your V-Series system has disk shelves, you can mirror an aggregate with disks between two different disk shelves. The rules for setting up mirroring with disks are the same for FAS systems and V-Series systems.
- If your V-Series system is using third-party storage, you can mirror a RAID0 aggregate between two different sets of array LUNs that are in separate locations on the same storage array or on two storage arrays.

You must follow the requirements for setting up SyncMirror third-party storage, some of which are different from setting up SyncMirror with disks. After SyncMirror is set up properly, Data

ONTAP can switch to using the mirrored storage without interruption in service if one or more LUNs on a storage array become unavailable.

• You cannot mirror an aggregate between a native disk shelf on a V-Series system and a storage array.

Related concepts

Requirements for setting up SyncMirror with array LUNs on page 353 Example of SyncMirror pool assignments for array LUNs on page 358

Requirements for setting up SyncMirror with array LUNs

To set up SyncMirror with array LUNs, you must fulfill standard requirements for any SyncMirror deployment plus a number of requirements that are unique to setting up SyncMirror with third-party storage.

For any SyncMirror deployment, the requirements are as follows:

- Using SyncMirror requires twice as much storage as an unmirrored deployment because you must have enough storage for both copies of the aggregate.
- You must install a SyncMirror license.

Note: If the SyncMirror license is not installed when you assign an array LUN to your system, Data ONTAP specifies pool0 for the LUN. After you install the SyncMirror license, you would have to unassign the LUNs that you want to use in the second plex (the remote location). You must then reassign each of those LUNs to the V-Series system, this time specifying a pool parameter of pool1. If the LUNs in the remote location were added to an aggregate before the SyncMirror license was installed, you have to destroy the aggregates that the LUNs were added to, unassign the LUNs, reassign them to the system in pool1, and re-create the aggregates.

The following table summarizes the additional requirements and restrictions specific to setting up SyncMirror with third-party storage.

For	The requirements and restrictions are
A V-Series system that uses both native disk shelves and third-party storage	You cannot mirror data between a native disk shelf on a V-Series system and a storage array.

For	The requirements and restrictions are
Storage arrays in your SyncMirror deployment	You can use one or two storage arrays in your SyncMirror deployment.
	If you are using two storage arrays for mirroring, the requirements are as follows:
	• Both storage arrays must be from the same vendor and from the same model family.
	• You must have two sets of LUNs—one set for the aggregate on the local storage array and another set of LUNs at the remote storage array for the mirror of the aggregate (the other plex of the aggregate).
	If you are using only one storage array for mirroring, the requirements are as follows:
	 The two sets of LUNs must be physically separated on the storage array. Each LUN must be from a different disk group (RAID group).

For	The requirements and restrictions are
Number and size of array LUNs needed	You must have the same number and size of array LUNs available in each set of array LUNs that you are going to use for the two plexes of the aggregate.
	For example, assume that you want have a 40 GB aggregate that is composed of four 10-GB LUNs, and you want to mirror it. You must have four 10-GB LUNs available in the local location and four 10-GB LUNs in the remote location to be able to mirror the aggregate.
	If the LUNs are not the same size, the following occurs:
	 If the pool0 (remote) array LUN is larger: If a LUN in the remote location is larger than the LUN in the local location, the mirror is created. However, space is wasted and cannot be reused. For example, if the array LUN in pool0 is 20 GB and the array LUN in pool0 is 20 GB, the mirror will be 10 GB (the pool0 LUN size.) The remaining 10 GB of space in the pool0 LUN is wasted and cannot be reused. If the pool0 LUN is larger: If the local LUN is larger than the remote LUN, Data ONTAP does not allow creation of the mirror. For example, if the pool0 (local) array LUN is 20 GB and the pool0 array LUN is 10 GB, mirroring fails.
Assignment of ownership of V-Series LUNs	You must assign all array LUNs that will be used for the aggregate and its mirror to the same V- Series system. This V-Series system will own the aggregate.
Checksum type of an array LUN	You must assign the same checksum type (either block or zoned) to all LUNs in both sets of LUNs that will be used for the plexes of the aggregate.

For	The requirements and restrictions are
SyncMirror pool assignment	You want the data mirrored exactly on the two storage arrays so that if one plex becomes unavailable, all data can continue to be served.
	Ensure that you plan how the array LUNs need to be assigned to pools before you start configuring Data ONTAP for SyncMirror. For array LUNs, you must explicitly assign each array LUN to the local pool or the remote pool. To group the LUNs correctly, you must know which array LUNs are located on which storage array. Data ONTAP cannot determine this for you.
Pathing	Before you create your aggregate and mirror it, ensure that there are two paths to a LUN.
	For more information about checking paths to array LUNs, see the <i>V-Series Installation Requirements and Reference Guide.</i>
MetroCluster	SyncMirror is required in MetroCluster configurations. In a MetroCluster configuration with V-Series systems, each plex must be on a separate set of LUNs on different storage arrays. For performance reasons, in a MetroCluster configuration you want the read operations to be served from the local pool to avoid read operations over long distances. See the V-Series MetroCluster Guide for more information about a V-Series system in a MetroCluster environment.
	Note: V-Series systems on which native disk shelves are installed cannot be deployed in a MetroCluster configuration.

Related concepts

Implications of storage type when mirroring with V-Series systems on page 352 SyncMirror pool assignment planning for third-party storage on page 357 Example of SyncMirror pool assignments for array LUNs on page 358 Common errors when setting up SyncMirror pools with array LUNs on page 360 Troubleshooting errors with SyncMirror pool assignment for array LUNs on page 360 Rules for adding array LUNs to a mirrored aggregate on page 371

SyncMirror pool assignment planning for third-party storage

To set up SyncMirror with third-party storage, you must provide Data ONTAP information about which array LUNs are local and which array LUNs are remote.

For native disks, Data ONTAP automatically assigns a disk to the local pool or remote pool, as appropriate, or you can assign a disk to a pool. However, Data ONTAP cannot detect whether an array LUN is located on the local storage array (the local pool) or on the remote storage array (the remote pool). You must explicitly provide this information to Data ONTAP.

You want the data mirrored exactly on the two storage arrays so that if one plex becomes unavailable, all data can continue to be served. Your goal is to group the LUNs belonging to the two storage arrays or two locations on the same storage array into two SyncMirror pools—one the local pool and the other the remote pool. Then, when you later create a mirrored aggregate, the LUNs for the same plex are derived from the same pool.

To group the LUNs, you must identify the appropriate SyncMirror pool for each array LUN you are using to create the two plexes of the aggregate. To specify the correct pool for each array LUN, you must know which array LUNs are located on which storage array. Data ONTAP cannot determine this for you.

Ensure that each LUN set has the same number of LUNs and that the LUNs in each set are the same size.

If you are using one storage array, ensure that each LUN is from a different disk group (RAID group) on the storage array.

Physical location of storage (assuming two storage arrays)	Pool to which the array LUNs need to be assigned	Command setting
Array LUN is on the storage array that is connected to the V-Series system (the local storage array). The aggregate is created on this V- Series system. (You can think of the V-Series system as the owner of the aggregate.)	Local pool (pool0)	disk assign -p 0
Array LUN is on the storage array whose LUNs are to be used to mirror the array LUNs in the aggregate. (This is the remote storage array.)	Remote pool (pool1)	disk assign -p 1

Note: You use the -p parameter of the disk assign command to specify the SyncMirror pool assignment.

Related concepts

Requirements for setting up SyncMirror with array LUNs on page 353

Example of SyncMirror pool assignments for array LUNs on page 358

Example of SyncMirror pool assignments for array LUNs

A SyncMirror deployment with third-party storage requires careful planning so that you assign array LUNs on the local storage array to the local pool and array LUNs on the remote storage array to the remote pool.

The following illustration shows the configuration for this example. The storage arrays are same model, which is required to set up SyncMirror with third-party storage.



For simplicity, this example discusses mirroring of just one aggregate on V-Series system vs1 (oneway mirroring). However, you can set up two-way mirroring; that is, you can mirror an aggregate owned by vs1 and also mirror an aggregate owned by vs2.

Assume that you want to create a 20 GB aggregate on V-Series system vs1 using two 10 GB LUNs, and you want to use SyncMirror to be able to mirror that aggregate. The total amount of storage that is required to set up mirroring is 40 GB, 20 GB from each storage array. The aggregate will be created from the LUNs on USP-A (because USP-A is the storage that is local to vs1). To be able to mirror the aggregate, two 10 GB LUNs must be available on USP-B.

Availability of array LUNs

Assume that the storage array administrator made the following 10 GB LUNs available to Data ONTAP and that you are going to use these LUNs for mirroring.

- From USP-A: LUN1 and LUN2
- From USP-B: LUN20 and LUN21

Assignment of LUNs to a V-Series system

System vs1 will be the owner of the aggregate. Therefore, you must assign both sets of LUNs that will be used for the plexes of the aggregate to vs1. You should assign the pool parameter when you initially assign each array LUN to vs1. If you assign array LUNs to pools later, you must unassign the LUNs in the remote location and assign them to vs1 again, this time with the remote LUNs assigned to pool1.

Pool (-p) parameter settings for this example are as follows:

- The array LUNs on storage array USP-A (the local storage array) must be assigned to the local pool—pool0
- The array LUNs on storage array USP-B (the remote storage array) must be assigned to the remote pool—pool1

The commands are as follows (assuming the checksum type is block):

- disk assign LUN1 LUN2 -p0 -c block
- disk assign LUN20 LUN21 -p1 -c block

System vs1 now owns two LUNs on each storage array. The pools are specified correctly so that when the mirrored aggregate is created, LUN20 and LUN21 can be the mirror of LUN1 and LUN2. The checksum type is the same for all the array LUNs that will be in the two plexes of the aggregate.

Note: Be sure to check that there are two paths to each LUN before creating the aggregate and mirroring it.

Creation of the aggregate and mirroring it

You can create the aggregate and mirror it all in the same command or create a mirrored aggregate later by adding a plex to an existing aggregate. The command formats are as follows:

То	The command format to use is
Create an aggregate and mirror it in the same step	aggr create aggrname -m -d disk- list_plex0 -d disklist_plex1 Where disk-list is the array LUNs to add to the plex
Add a plex to an existing aggregate	aggr aggrname mirror -d disk- list_plex1 Where disk-list is the array LUNs to add to the plex

The following command creates the aggregate for this example and mirrors it. Plex 0 of the aggregate named vs1aggregate is created with LUN 1 and LUN2, and the other plex of the aggregate is created with LUN20 and LUN21:

aggr create vslaggregate -m -d LUN1 LUN2 -d LUN20 LUN21

Note: When you create a mirrored aggregate, be sure that you specify the same number and size of array LUNs for each plex. Also be sure that the LUNs you need for plex 1 exist on the storage array and have been assigned to pool1 (the remote pool).

Related concepts

Requirements for setting up SyncMirror with array LUNs on page 353 Implications of storage type when mirroring with V-Series systems on page 352 SyncMirror pool assignment planning for third-party storage on page 357

Common errors when setting up SyncMirror pools with array LUNs

Your SyncMirror setup for array LUNs will not be successful if your local and remote pool assignments do not match the actual location of the array LUNs.

Error	Result
You assign some array LUNs from the local storage array to the remote pool, or you assign some array LUNs from the remote storage array to the local pool.	You cannot create the mirror for the aggregate. The mirror creation process does not allow mixed pools within a plex.
You reverse the pool settings for each set of array LUNs. That is, you assign all the LUNs on the local storage array that you want to use for mirroring the aggregate to the remote pool (-p1) and assign the set of LUNs on the remote storage array to the local pool (-p0)	Data ONTAP allows you to create the mirrored aggregate. If one storage array becomes unavailable, the wrong side of the plex is reported as unavailable. The data is still on the storage array that is available.
You plan to use two storage arrays for SyncMirror but you mistakenly create a mirrored aggregate with both pools from the same storage array.	Data is lost if the storage array fails.

The follow table shows the result of common errors in array LUN SyncMirror pool assignment.

Related concepts

Requirements for setting up SyncMirror with array LUNs on page 353 Troubleshooting errors with SyncMirror pool assignment for array LUNs on page 360 Rules for adding array LUNs to a mirrored aggregate on page 371

Troubleshooting errors with SyncMirror pool assignment for array LUNs

To troubleshoot SyncMirror pool assignment problems with array LUNs, you need to look at backend setup and Data ONTAP configuration. You need to determine whether the pool assignment in Data ONTAP matches the actual location of the LUNs.

If the plexes are on two different storage arrays, you need to know which of the two storage arrays a specific array LUN is located on. If you are using just one storage array, you need to make sure that
your pool assignments in Data ONTAP match the sets of LUNs that you determined were local and remote.

You must know how the storage array is cabled to the switch to be able to determine which array LUNs are from the local storage array and which array LUNs are from the remote storage array. You can use a combination of the following methods to obtain information about where the LUNs are located:

- Look at switch zoning
- Look at the output of the Data ONTAP storage show disk -p command

You can use the following Data ONTAP commands to check the pool assignments of specific array LUNs:

- aggr status -r
- sysconfig -r

If you find problems, correct array LUN pool assignment errors as needed. In the Data ONTAP disk assign -p command, -p0 represents the local pool and -p1 represents the remote pool.

Related concepts

Common errors when setting up SyncMirror pools with array LUNs on page 360 *Requirements for setting up SyncMirror with array LUNs* on page 353

Enabling the SyncMirror license

Before using SyncMirror, you need to enable the appropriate license. The license code name for SyncMirror is known as syncmirror_local.

Steps

1. Enter the following command:

license add license_code

license_code is the syncmirror_local license code.

2. Reboot the system.

Disabling the SyncMirror license

You cannot disable the SyncMirror license if mirrored aggregates exist and are online. Before disabling the license, you must take one of the plexes offline for each mirrored aggregate and destroy it.

Steps

1. For each mirrored aggregate, decide which plex you want to take offline.

Note: Every mirrored aggregate must have one plex taken offline and destroyed before you can disable the mirroring license.

2. Take offline each of the FlexVol volumes in the plexes you decided to take offline by entering the following command:

vol offline vol-name

vol-name is the name of the FlexVol volume.

Note: Only one FlexVol volume can be taken offline at a time.

3. Remove the FlexVol volumes from the plexes you decided to take offline by entering the following command:

vol destroy vol-name

vol-name is the name of the FlexVol volume.

4. Take each of the plexes offline by entering the following command:

aggr offline plex-name

plex-name is the name of one of the mirrored plexes.

Note: Only one plex at a time can be taken offline.

5. Destroy the plex you took offline by entering the following command:

aggr destroy plex-name

plex-name is the name of one of the mirrored plexes.

6. Enter the following command:

license delete syncmirror_local

Considerations for using mirrored aggregates

If you want to use mirrored aggregates, you can either create a new aggregate with two mirrored plexes, or add a plex to an existing aggregate.

Note: A mirrored aggregate can have only two plexes.

The rules for the selection of disks or array LUNs, for using as mirrored aggregates, are as follows:

- Disks or array LUNs selected for each plex must be in different pools.
- The same number of disks or array LUNs must be in both the plexes.
- Disks are selected first on the basis of equivalent bytes per sector (bps) size, then on the basis of the size of the disk.
- If there is no equivalent-sized disk, Data ONTAP uses a larger-capacity disk, and limits the size to make it identically sized.
- Data ONTAP names the plexes of the mirrored aggregate.

Note: When creating an aggregate, Data ONTAP selects disks from the plex which has the most available disks. You can override this selection policy by specifying the disks to use.

Next topics

How disks are assigned to plexes on page 363 *Viewing plexes and spare pools* on page 364 *Creating a mirrored aggregate* on page 366 *Converting an aggregate to a mirrored aggregate* on page 368

Related concepts

What mirrored aggregates are on page 350

How disks are assigned to plexes

You need to understand how Data ONTAP assigns disks to plexes in order to configure your disk shelves and host adapters.

When a mirrored aggregate is created, Data ONTAP uses spare disks from a collection of disks, to create two disk pools, pool0 and pool1.

When assigning a disk to a pool, Data ONTAP determines the shelf for the disk and ensures that the disks in pool0 are from different shelves than the disks in pool1. So, before enabling SyncMirror, you should ensure that the disks are installed in at least two shelves and the shelves are connected to the system with separate cables and adapters. Disk pools must be physically separate to ensure high availability of the mirrored aggregate.

Disks from pool0 are used to create plex0 while disks from pool1 are used to create plex1.

Plexes local to the host node in an active/active configuration must be connected to the disk pool named pool0. pool0 consists of the storage attached to host adapters in slots 3 through 7.

Note: Pool rules for MetroCluster configurations that use switches are different.

For more information about storage system slot assignments, see the System Configuration Guide.

Viewing plexes and spare pools

You can view the plexes and spare disks or array LUNs. It is useful to view the assignments when you are adding disks or array LUNs to an aggregate, or when you need to identify the pool used by each plex.

Step

1. Enter one of the following commands.

```
sysconfig -r
```

or

```
aggr status -r
```

or

```
vol status -r
```

Example

In this example, the aggr status -r command is used to view the disks in plexes and spare disks in disk pools.

system1> aggr status -r Aggregate vol0 (online, raid4) (block checksums) Plex /vol0/plex0 (online, normal, active, pool1) RAID group /vol0/plex0/rg0 (normal) RAID Disk Device HA SHELF BAY CHAN Pool Type RPM Used (MB/blks) Ph) -------- ---- ----parity9a.169a10FC:A1FCAL1000034000/6963200034data9a.179a11FC:A1FCAL10000600/122880076data9a.209a14FC:A1FCAL1000034000/6963200034 34 Aggregate GreG (online, raid4) (block checksums) Plex /GreG/plex0 (online, normal, active, pool1) RAID group /GreG/plex0/rg0 (normal) RAID Disk Device HA SHELF BAY CHAN Pool Type RPM Used (MB/blks) Ph) _____ ___ ----- ----___ parity9a.189a12FC:A1FCAL 10000 34000/6963200034data9a.199a13FC:A1FCAL 10000 34000/6963200034 Pooll spare disks

 RAID Disk
 Device
 HA
 SHELF
 BAY
 CHAN
 Pool
 Type
 RPM
 Used (MB/blks)
 Ph)

 Spare disks for block or zoned checksum traditional volumes or aggregates
 spare
 9a.24
 9a
 1
 8
 FC:A
 1
 FCAL 10000
 34000/69632000
 34

spare	9a.29	9a	1	13	FC:A	1	FCAL	10000	34000/69632000	34
Pool0 spare dis	ks (empt	y)								
Partner disks										
RAID Disk	Device	HA	SHELF	BAY	CHAN	Pool	Туре	RPM	Used (MB/blks)	Ph)
partner partner partner partner partner partner	9b.25 9b.16 9b.17 9b.21 9b.18 9b.22	9b 9b 9b 9b 9b 9b	1 1 1 1 1	9 0 1 5 2 6	FC:B FC:B FC:B FC:B FC:B FC:B	1 1 1 1 1	FCAL FCAL FCAL FCAL FCAL FCAL	10000 10000 10000 10000 10000 10000	0/0 0/0 0/0 0/0 0/0 0/0	34 34 34 34 34 34
1									-, -	

tpubs-cf1>

Example

In this example, the vol status -r command is used to view the disks in plexes and spare disks in disk pool.

system1> vol status -r Volume vol0 (online, raid4) (block checksums) Plex /vol0/plex0 (online, normal, active, pool1) RAID group /vol0/plex0/rg0 (normal) RAID Disk Device HA SHELF BAY CHAN Pool Type RPM Used (MB/blks) Ph) _ _ _ parity 9a.16 9a 1 0 FC:A 1 FCAL 10000 34000/69632000 34 data9a.179a1FC:A1FCAL 10000600/1228800data9a.209a14FC:A1FCAL 1000034000/69632000 76 34 Aggregate aggrz (online, raid4) (block checksums) Plex /aggrz/plex0 (online, normal, active, pool1) RAID group /aggrz/plex0/rg0 (normal) RAID Disk Device HA SHELF BAY CHAN Pool Type RPM Used (MB/blks) Ph) ----- ---- ----- ---- ---- ---- ----_ _ _ parity9a.259a19FC:A1FCAL 1000034000/69632000data9a.269a110FC:A1FCAL 1000034000/69632000data9a.279a111FC:A1FCAL 1000034000/69632000data9a.289a112FC:A1FCAL 1000034000/69632000 34 34 34 34 Pooll spare disks RAID Disk Device HA SHELF BAY CHAN Pool Type RPM Used (MB/blks) Ph) _____ _____ ____ _ _ _ _ Spare disks for block or zoned checksum traditional volumes or aggregates
 Spare
 9a.24
 9a
 1
 8
 FC:A
 1
 FCAL
 10000
 34000/69632000

 spare
 9a.29
 9a
 1
 13
 FC:A
 1
 FCAL
 10000
 34000/69632000
 34 34 Pool0 spare disks (empty) Partner disks RAID Disk Device HA SHELF BAY CHAN Pool Type RPM Used (MB/blks) Ph) _____ _____ ____ _ _ _

 partner
 9b.25
 9b
 1
 9
 FC:B
 1
 FCAL 10000
 0/0

 partner
 9b.16
 9b
 1
 0
 FC:B
 1
 FCAL 10000
 0/0

 partner
 9b.16
 9b
 1
 0
 FC:B
 1
 FCAL 10000
 0/0

 partner
 9b.17
 9b
 1
 1
 FC:B
 1
 FCAL 10000
 0/0

 partner
 9b.21
 9b
 1
 5
 FC:B
 1
 FCAL 10000
 0/0

 partner
 9b.18
 9b
 1
 2
 FC:B
 1
 FCAL 10000
 0/0

 partner
 9b.22
 9b
 1
 6
 FC:B
 1
 FCAL 10000
 0/0

 partner
 9b.22
 9b
 1
 6
 FC:B
 1
 FCAL 10000
 0/0

 34 34 34 34 34 34 partner 9b.23 9b 1 7 FC:B 1 FCAL 10000 0/0 34

Related tasks

Converting an aggregate to a mirrored aggregate on page 368 *Creating a mirrored aggregate* on page 366

Creating a mirrored aggregate

When creating an aggregate, you can specify the aggregate to use SyncMirror. This ensures that the aggregate is a mirrored one from the start. You have several options for how to specify the disks or array LUNs when you create a mirrored aggregate.

About this task

There are three methods of creating a mirrored aggregate:

- Ascertain the disks or array LUNs available, and specify which ones to use.
- Allow Data ONTAP to automatically use the disks or array LUNs that are available.
- Preview the disks or array LUNs that Data ONTAP has selected. Then, you can either use the same selection or modify the selection.

Steps

1. You can use the aggr create command to create a mirrored aggregate. As required, choose one of the actions from the following table.

If	Then
You know the disks or array LUNs to be used for the mirrored aggregate, and want to specify the disks	Enter the following command:
	aggr create aggr_name -m -d disk-list -d disk- list Ensure that you choose the correct number and size of disks for each set of disks.
	Note: Both the -d options must be used, one for each plex. If you specify only one disk set, the creation of a mirrored aggregate fails.
	This step completes the creation of a mirrored aggregate.
You want Data ONTAP to	Go to the next step.
select the disks for the mirrored aggregate	Note: Allowing Data ONTAP to select the disks is the easiest method of creating a mirrored aggregate.

aggr_name is the name of the mirrored aggregate.

-m specifies the creation of a mirrored aggregate.

disk-list consists of the disk IDs of two or more available disks. You should separate multiple disks with the space character.

Example

The following command creates a mirrored aggregate named aggrA with disks 6.1 and 6.2 on one plex, and disks 8.1 and 8.2 on the other plex.

aggr create aggrA -m -d 6.1 6.2 -d 8.1 8.2

2. As required, choose one of the actions from the following table.

If you want	Then enter the following command
Data ONTAP to automatically specify the disks for the mirrored aggregate	aggr create aggr_name -m ndisks[@disk-size] This step completes the creation of a mirrored aggregate.
To preview the disks selected by Data ONTAP for creating the mirrored aggregate	aggr create aggr_name -n -m ndisks[@disk-size] Note: The -n option instructs Data ONTAP not to create the aggregate, but to display the disks selected automatically for creating the mirrored aggregate.
	The system displays an aggr create command indicating the disks to be used for the mirrored aggregate.
	Go to the next step.

ndisks is the number of disks to use, which must be a minimum of four and an even number. Even numbers are required because the disks are equally divided between the two plexes.

disk-size specifies the disk capacity, in gigabytes. There must be enough disks of this size available.

Note: If you want to specify the disk size, first determine the size of spare disks in the disk pools by using the vol status -r command.

Example

The following command creates a mirrored aggregate named aggrB, with two plexes, each plex containing three disks.

```
aggr create aggrB -m 6
```

3. As required, choose one of the actions from the following table.

If you want to	Then
Use the selection provided in the preview	Enter the aggr create command, displayed by the system in the previous step.
Change one or more of the disks provided in the preview	Enter the aggr create command, displayed by the system in the previous step. Substitute one or more disks with other disks that you want to specify.

Example

The following command displays the aggr create command, with a preview of the disk selection for creating a four-disk mirrored aggregate named aggrC.

aggr create aggrC -n -m 4

The system returns the following command:

aggr create aggrC -m -d 5.1 5.3 -d 8.3 8.4

Use the preceding command to create a mirrored aggregate named aggrC, with two disks in each plex.

For more information about aggregates, see the Data ONTAP Storage Management Guide.

Related tasks

Viewing plexes and spare pools on page 364

Related references

Considerations for using mirrored aggregates on page 363

Converting an aggregate to a mirrored aggregate

You can convert an aggregate to a mirrored aggregate by adding a plex to the aggregate. Thus, you can use SyncMirror to mirror a previously unmirrored aggregate. You have several options for how to specify the disks or array LUNs when you convert to a mirrored aggregate.

Before you begin

A mirrored aggregate can only have two plexes. You can check the details of an aggregate by using the aggr status -r command.

About this task

If the aggregate that you want to mirror uses disks or array LUNs of different capacities, Data ONTAP can select disks or array LUNs that match the smallest capacity from a different pool. If there are not enough disks or array LUNs of that capacity in the pool, Data ONTAP selects higher-capacity disks or array LUNs and downsizes them.

The three methods of converting an aggregate to a mirrored aggregate are as follows:

- Ascertain the disks or array LUNs available, and specify which ones to use.
- Allow Data ONTAP to automatically use the disks or array LUNs that are available.
- Preview the disks or array LUNs that Data ONTAP has selected. Then, you can either use the same selection or modify the selection.

Steps

1. Use the aggr mirror command to add a plex to an aggregate. As required, choose one of the actions from the following table.

If	Then
You know the disks or array LUNs to be used for the plex addition and want to specify the disks or array LUNs	Enter the following command:
	aggr mirror aggr_name -d disk-list Ensure that you choose the correct number and size of disks or array LUNs in the list.
	Note: These disks or array LUNs should be from a different pool than that already being used by the aggregate.
	This step adds a plex to the aggregate, making it a mirrored aggregate.
You want Data ONTAP to select	Go to the next step.
the disks or array LUNs for the plex addition	Note: Allowing Data ONTAP to select the disks or array LUNs is the easiest method of adding a plex to an aggregate.

aggr_name is the name of the mirrored aggregate.

disk-list is the list of IDs of two or more available disks or array LUNs. Separate the entries with the space character.

Example

The following command adds a plex to the aggregate aggrD with disks 7.1 and 7.2, and makes aggrD a mirrored aggregate.

```
aggr mirror aggrD -d 7.1 7.2
```

2. As required, choose one of the actions from the following table.

If you want	Then enter the following command
Data ONTAP to automatically specify the disks or array LUNs for the plex addition	aggr mirror aggr_name This step adds a plex to the aggregate, making it a mirrored aggregate.
To preview the disks or array LUNs selected by Data ONTAP for the plex addition	aggr mirror aggr_name -n Note: The -n option instructs Data ONTAP not to add the plex, but to display the disks or array LUNs selected automatically, for the plex addition.
	The system displays an aggr mirror command indicating the disks or array LUNs to be used for the plex addition.
	Go to the next step.

Example

The following command adds a plex to the aggregate aggrE, making aggrE a mirrored aggregate.

aggr mirror aggrE

3. As required, choose one of the actions from the following table.

If you want to	Then
Use the selection provided in the preview	Enter the aggr mirror command displayed by the system in the previous step.
Change one or more of the disks or array LUNs provided in the preview	Enter the aggr mirror command displayed by the system in the previous step. Substitute any disks or array LUNs with what you want to specify.

Example

The following command displays the aggr mirror command with a preview of the disk selection for adding a plex to an aggregate named aggrF.

aggr mirror aggrF -n

The system returns the following command:

aggr mirror aggrF -d 8.1 8.2

Use the preceding command to add a plex to the aggregate aggrF to make it a mirrored aggregate.

For more information about aggregates, see the Data ONTAP Storage Management Guide.

Related tasks

Viewing plexes and spare pools on page 364

Related references

Considerations for using mirrored aggregates on page 363

Addition of disks or array LUNs to a mirrored aggregate

You can add disks or array LUNs to a mirrored aggregate by using one of the following methods.

- Allow Data ONTAP to select the disks or array LUNs.
- Select the disks or array LUNs manually.
- Preview the disks or array LUNs Data ONTAP has selected. You can use the same selection or modify the selection.

Next topics

Rules for adding disks to a mirrored aggregate on page 371 Rules for adding array LUNs to a mirrored aggregate on page 371 Adding disks to a mirrored aggregate, where Data ONTAP selects the disks on page 371 Adding disks or array LUNs to a mirrored aggregate, where the user selects the disks on page 372 Adding disks to a mirrored aggregate, where the user selects the disks with assistance from Data ONTAP on page 373

Rules for adding disks to a mirrored aggregate

When you add disks to a mirrored aggregate or, you need to follow certain rules.

Disk requirements:

- The number of disks must be even, and the disks must be equally divided between the two plexes.
- The disks for each plex must come from different disk pools.
- The disks that you add must have equivalent bytes per sector (bps) sizes.

When adding new disks to a RAID group, the utilization of the new disks depends on the RAID level used. If the storage capacity of the new disks is more than the disks already in the RAID group, the larger-capacity disks might be downsized to suit the RAID group.

- RAID-DP: Larger-capacity disks are downsized to size of parity disks.
- RAID-4: Larger-capacity disks can replace the parity disks.

Rules for adding array LUNs to a mirrored aggregate

When you add array LUNs to a mirrored aggregate, you need to ensure that the number and size of the array LUNs in the two plexes remains identical.

Keep the following rules in mind when adding array LUNs to a mirrored aggregate:

- You must add an even number of array LUNs to the mirrored aggregate.
- You must divide the array LUNs you add equally between the two plexes.
- The array LUNs for each plex must come from a different LUN set. Do not mix the LUNs from the two LUN sets in the same plex.
- All array LUNs in the mirrored aggregate must be the same checksum type.

Related concepts

Requirements for setting up SyncMirror with array LUNs on page 353 Common errors when setting up SyncMirror pools with array LUNs on page 360

Adding disks to a mirrored aggregate, where Data ONTAP selects the disks

When you add disks to a mirrored aggregate, you can allow Data ONTAP to select the disks.

Step

1. Enter the following command:

aggr add aggrname ndisks[@disk-size]

aggrname is the name of the aggregate to which you are adding disks or array LUNs.

ndisks is the number of disks or array LUNs to use. This number must be an even number, because half these disks or array LUNs must be in each pool.

disk-size specifies the disk or array LUN capacity, in gigabytes.

Note: If you use the disk-size option, you should first determine the size of spare disks or array LUNs in the pools using the aggr status -r or vol status -r command.

Adding disks or array LUNs to a mirrored aggregate, where the user selects the disks

If you select the new disks or array LUNs to add to the mirrored aggregate, you must ensure that you add the correct number of disks or array LUNs of the correct sizes. The disks or array LUNs must have the same checksum compatibility, and disks or array LUNs for each plex must be in different pools.

Steps

1. Enter the following command to list available spare disks or array LUNs.

aggr status -r

- 2. Choose the spare disks or array LUNs you want to add.
- **3.** Enter the following command:

```
aggr add aggrname -d disk-list -d disk-list
```

aggrname is the name of the aggregate to which you are adding disks or array LUNs.

disk-list consists of disk or array LUNs IDs of one or more available disks; separate multiple disks or array LUNs with a space character. Both -d options must be used, one for each plex. Data ONTAP automatically assigns the disks or array LUNs to the appropriate plex.

Note: Specifying only one disk or array LUN set using the -d option will fail.

In the following example, the aggr add command adds disk 3.1 to one plex and disk 8.1 to the other plex of the aggrD mirrored aggregate.

```
aggr add aggrD -d 6.1 -d 8.1
```

Related references

Rules for adding disks to a mirrored aggregate on page 371

Adding disks to a mirrored aggregate, where the user selects the disks with assistance from Data ONTAP

You can preview what disks Data ONTAP would select if it were to add new disks to the aggregate. Then, you can add the disks Data ONTAP selected or substitute other disks.

Steps

1. Enter the following command:

aggr add aggrname -n ndisks[@disksize]

aggrname is the name of the mirrored aggregate to which you are adding disks.

ndisks is the number of disks to use. Adding new disks will fail if the number of disks is an odd number.

disk-size specifies the disk capacity, in gigabytes. Half these disks must be in each disk pool.

Note: If you use the disk-size option, first determine the size of spare disks in the disk pools using the vol status -r command.

Data ONTAP returns an aggr add command that specifies the disks it would add to each plex of the mirrored aggregate.

2. As required, choose one of the actions from the following table.

If you want to	Then
Use the disks specified by Data ONTAP	Enter the aggr add command specified.
Substitute other disks for one or more disks specified by Data ONTAP	Enter the aggr add command and substitute other disks for one or more of the disks specified.

In the following example, the aggr add command provides a preview of the disks that Data ONTAP would use when adding two new 144-GB disks to the aggrA mirrored aggregate.

```
aggr add aggrA -n 2@144
```

Data ONTAP returns the following command:

aggr add aggrA -d 6.4 -d 8.6

The states of a plex

A plex can either be in an online state or in an offlline state. In the online state, the plex is available for read or write access and the contents of the plex are current. In an offline state, the plex is not accessible for read or write.

An online plex can be in the following states.

- Active—The plex is available for use.
- Adding disks or array LUNs—Data ONTAP is adding disks or array LUNs to the RAID group or groups of the plex.
- Empty—The plex is part of an aggregate that is being created and Data ONTAP needs to zero out one or more of the disks or array LUNs targeted to the aggregate before adding the disks to the plex.
- Failed—One or more of the RAID groups in the plex failed.
- Inactive—The plex is not available for use.
- Normal—All RAID groups in the plex are functional.
- Out-of-date—The plex contents are out of date and the other plex of the aggregate has failed.
- Resyncing—The plex contents are being resynchronized with the contents of the other plex of the aggregate.

Next topics

Viewing the status of plexes on page 374 *Changing the state of a plex* on page 375

Viewing the status of plexes

At any point, you might want to view the status of a given plex.

Before you begin

To view the status of a plex, the plex must be online.

Step

1. To view the status of plexes, enter one of the following commands.

```
sysconfig -r
or
aggr status -r
or
vol status -r
```

Changing the state of a plex

You can change the state of a plex in a mirrored aggregate, from online to offline, and vice-versa.

About this task

Data ONTAP specifies the state of a plex as resyncing when synchronizing the two plexes of a mirrored aggregate. Also, when you create a mirrored aggregate by adding a plex to an unmirrored aggregate, Data ONTAP puts the added plex in a resyncing state.

Data ONTAP allows you to change the state of a plex from offline to online, and from online to offline.

Step

1. To change the state of a plex, enter the following command:

```
aggr { online | offline } plexname
```

Splitting a mirrored aggregate

Splitting a mirrored aggregate removes the relationship between its two plexes and creates two independent unmirrored aggregates. After splitting, both the aggregates come online.

Before you begin

Ensure that both plexes of the mirrored aggregate you are splitting are online and operational.

About this task

You might split a mirrored aggregate for one of the following reasons.

- You want to stop mirroring an aggregate.
- You want to move a mirrored aggregate to another location.
- You want to modify the mirrored aggregate, and test the modification before applying it. You can apply and test the modifications on the split-off copy of the plex, then apply those changes to the untouched original plex.

Before splitting, a mirrored aggregate or traditional volume has two plexes, plex0 and plex1. After splitting, the new unmirrored aggregate with the new name has one plex, plex0. The new unmirrored aggregate with the original name also has one plex, either plex0 or plex1.

The plex name for an unmirrored aggregate is unimportant because the aggregate has only one plex. If you use SyncMirror to mirror one of these unmirrored aggregates, the resulting plex names will always be plex0 and plex1.

Note: You do not need to stop applications that are using the aggregate, before splitting a mirrored aggregate.

Step

1. Enter the following command:

aggr split aggrname/plexname new_aggr

aggrname is the name of the mirrored aggregate.

plexname is the name of one of the plexes in the mirrored aggregate.

new_aggr is the name of the new aggregate that would be created.

Example

The following command splits plex0 from mirrored aggregate aggr0 and name the new aggregate aggrNew.

aggr split aggr0/plex0 aggrNew

After splitting, there are two unmirrored aggregates, aggr0 and aggrNew.

Related tasks

Converting an aggregate to a mirrored aggregate on page 368 *Rejoining split aggregates* on page 376 *Removing a plex from a mirrored aggregate* on page 377 *Changing the state of a plex* on page 375

Rejoining split aggregates

You can rejoin split aggregates. You might want to do this if you have set up an active/active configuration in a MetroCluster configuration and a disaster breaks the active/active configuration.

Before you begin

Attention: When you rejoin split aggregates, Data ONTAP mirrors the data from one aggregate to the other and destroys data that existed on that aggregate before the rejoin.

About this task

There are additional considerations when planning to rejoin split aggregates that previously used MetroCluster to mirror SnapLock volumes.

You can use MetroCluster to mirror SnapLock volumes from one site to another. With proper configuration, the SnapLock volumes retain their characteristics at the mirror site. In case of a failure at the primary site, and if necessary, you can use the cf forcetakeover -d command to break the mirror relationship and to bring the mirror site online. Once the failure at the primary site is resolved,

the MetroCluster mirror relationship can be reestablished. The mirrors can be resynchronized before resuming normal operation.

Attention: The primary node might have data that was not mirrored before using the cf forcetakeover -d command. For example, the data might have been written to the primary node while the link between the sites was inoperative. In such a case, you should back up the SnapLock volumes in the aggregate on the primary site, before resynchronizing the two mirror aggregates. This step of creating an additional backup for the SnapLock volumes is required to ensure the availability of all data.

For more information about backing up data in SnapLock volumes using SnapMirror, see the *Data* ONTAP Archive and Compliance Management Guide.

For more information about MetroCluster deployment, see the *Data ONTAP Active/Active Configuration Guide*.

Steps

- 1. Determine the aggregate whose data you want to keep and the aggregate whose data you want to be overwritten.
- 2. If the aggregate whose data is to be overwritten is online, take it offline by entering the following command.

aggr offline aggrname

aggrname is the name of the aggregate.

Note: An error message appears if the aggregate is already offline.

3. Re-create the mirrored aggregate by entering the following command.

aggr mirror aggrname1 -v aggrname2

aggrname1 is the name of the aggregate whose data you want to keep.

aggrname2 is the name of the aggregate whose data you want to be overwritten by aggrname1.

Removing a plex from a mirrored aggregate

You can remove a plex from a mirrored aggregate. You might do this if you want to stop mirroring the aggregate, or if there is a problem with the plex. Removing a plex results in an unmirrored aggregate.

About this task

In case of a failure that causes a plex to fail, you can remove the plex from the mirrored aggregate, fix the problem, and then re-create it. You can also re-create it using a different set of disks or array LUNs, if the problem cannot be fixed.

Steps

1. Take the selected plex offline by entering the following command:

aggr offline plex-name

plex-name is the name of one of the mirrored plexes.

Note: Only one plex at a time can be taken offline.

2. Destroy the plex you took offline by entering the following command:

```
aggr destroy plex-name
```

Result

Removing and destroying a plex from a mirrored aggregate results in an unmirrored aggregate, because the aggregate now has only one plex.

After removing the plex, Data ONTAP converts the disks or array LUNs used by the plex into hot spares.

Comparing plexes of a mirrored aggregate

The plexes of a mirrored aggregate are almost always synchronized. However, you might need to compare the plexes of a mirrored aggregate. You can also choose to correct any differences between the plexes.

Before you begin

The mirrored aggregate must be online before you can compare the plexes.

About this task

Comparing plexes might affect system performance.

When comparing the two plexes of a mirrored aggregate, you can choose one of the following options.

- Data ONTAP compares plexes without correcting differences. This is the default behavior.
- Data ONTAP compares plexes and corrects the differences it finds. To correct differences, you need to specify which plex to correct. The plex is specified as *plexnumber* (0, 1, and so on).

Attention: This process might use advanced Data ONTAP commands. Contact technical support before correcting differences using this option.

Step

1. To compare the two plexes of a mirrored aggregate, choose one of the actions from the following table.

If	Then
You do not want Data ONTAP to correct differences	Enter the following command:
	aggr verify start aggrname -n
	aggrname is the name of the mirrored aggregate whose plexes you are comparing.
You want Data ONTAP to	Enter the following command:
correct differences	aggr verify start aggrname -f plexnumber
	aggrname is the name of the mirrored aggregate whose plexes you are comparing.

Note: If *aggrname* is not specified, Data ONTAP compares the plexes of all mirrored aggregates that are online.

Next topics

Stopping plex comparison on page 379 Suspending plex comparison on page 379 Resuming plex comparison on page 380 Viewing the status of a plex comparison on page 380

Stopping plex comparison

You can stop Data ONTAP from comparing the plexes of a mirrored aggregate. Such a step might be required if the comparison affects system performance.

Step

1. To stop Data ONTAP from comparing plexes, enter the following command:

aggr verify stop aggrname

Note: If *aggrname* is not specified, Data ONTAP stops comparing plexes for all mirrored aggregates.

Suspending plex comparison

You can suspend the comparison of the plexes of a mirrored aggregate, instead of stopping the comparison. The comparison remains suspended until you resume it, stop it, or reboot the system.

Step

1. To suspend a comparison of plexes, enter the following command:

aggr verify suspend aggrname

Note: If *aggrname* is not specified, Data ONTAP suspends the comparison of plexes for all mirrored aggregates.

Resuming plex comparison

After you have suspended a comparison of plexes, you can resume the comparison by using the aggr verify resume command. Otherwise, the comparison remains suspended until you stop it, or reboot the system.

Step

1. To resume a suspended comparison of plexes, enter the following command:

aggr verify resume aggrname

Note: If *aggrname* is not specified, Data ONTAP resumes the comparison of plexes for all mirrored aggregates.

Viewing the status of a plex comparison

You can view the status of a plex comparison operation. The status tells you what percentage of the plex comparison has been completed, and whether plex comparison of a mirrored aggregate is suspended.

Step

1. To view the status of a plex comparison, enter the following command:

aggr verify status aggrname

Note: If *aggrname* is not specified, Data ONTAP displays the status of all mirrored aggregates whose plexes are being compared.

Database protection using NVFAIL

Data ONTAP provides database protection using the nvfail option. The nvfail option enables Data ONTAP to detect nonvolatile RAM (NVRAM) inconsistencies at boot time or while taking over in an active/active configuration.

You use this option to warn database administrators of NVRAM problems that can compromise database validity. If Data ONTAP finds any problems, database instances stop responding or shut down, and Data ONTAP sends error messages to the console to alert you to check the state of the database.

Next topics

How NVFAIL protects database files on page 381 *Enabling database file protection* on page 383 *Adding more database file protection* on page 383 *Making LUNs accessible to the host after an NVRAM failure* on page 384

How NVFAIL protects database files

When booting up or while taking over in a failover configuration, Data ONTAP checks for NVRAM errors. If no errors are detected, start the file service is started normally. However, if NVRAM errors are detected Data ONTAP stops database instances from responding.

If	Then
Data ONTAP detects no NVRAM errors	File service starts normally.
Data ONTAP detects NVRAM errors and you use the optional nvfail_rename file	 Data ONTAP returns a stale file handle (ESTALE) error to NFS clients trying to access the database. This causes the application to stop responding, crash, or shut down. Data ONTAP then sends an error message to the system console and log file. Data ONTAP renames database files specified in the nvfail_rename file by appending .nvfail to the original file names, making those files unavailable to both CIFS and NFS clients.

When you enable the nvfail option, one of the following processes takes place during bootup.

If	Then
Data ONTAP detects NVRAM errors and you do not use the optional nvfail_rename file	1. Data ONTAP returns a stale file handle (ESTALE) error to NFS clients trying to access the database, causing the application to stop responding, crash, or shut down. Data ONTAP then sends an error message to the system console and log file.
	2. No database files are renamed. When the application restarts, files are available to CIFS clients, even if you have not verified that they are valid. For NFS clients, files remain inaccessible as long as the file system is not remounted.
Data ONTAP detects NVRAM errors on a volume that contains LUNs and you use the optional nyfail repare file	 Data ONTAP takes the LUNs offline in the volume that had the NVRAM errors.
Note: The term <i>LUNs</i> in this context refer to the LUNs that Data ONTAP serves to clients not to	2. Data ONTAP stops exporting those LUNs over iSCSI or FCP.
the array LUNs used for storage on a storage array.	3. Data ONTAP sends error messages to the system console and log file stating that Data ONTAP took the LUNs offline or that NFS file handles are stale (useful if the LUN is accessed over NAS protocols).
	4. Data ONTAP renames LUNs specified in the nvfail_rename file by appending .nvfail to the original LUN names.
Data ONTAP detects NVRAM errors on a volume that contains LUNs and you do not use the optional	1. Data ONTAP takes the LUNs offline in the volume that had the NVRAM errors.
	2. Data ONTAP stops exporting those LUNs over iSCSI or FCP.
	3. Data ONTAP sends error messages to the system console and log file stating that Data ONTAP took the LUNs offline or that NFS file handles are stale (useful if the LUN is accessed over NAS protocols).
	4. No database files are renamed. When the application restarts, files are available to CIFS clients, even if you have not verified that they are valid. For NFS clients, files remain inaccessible as long as the file system is not remounted.

Enabling database file protection

You can use the nvfail option of the vol options command to configure Data ONTAP to find NVRAM problems, stop database instances from responding or shut down, and send error messages to the console to alert the database administrator to check the state of the database. The nvfail option detects NVRAM inconsistencies at boot time or while taking over in an active/active configuration failover.

Step

1. Enter the following command:

```
vol options volume_name nvfail [on off]
```

volume_name is the name of the volume.

Use on to enable or off to disable protection. The default setting is off.

Where to look for database file verification instructions

For instructions about examining database file validity, see the documentation for the specific database software.

Adding more database file protection

In addition to protecting database files and LUNs using the nvfail option, you can have Data ONTAP rename database files and LUNs that you want protected. Renaming database files and LUNs prevents the database from restarting automatically and gives you an opportunity to examine the files for inconsistencies before clients can access them.

Before you begin

You should ensure that the nvfail option is enabled.

About this task

Data ONTAP requires a file called /etc/nvfail_rename in which to put the names of the files you want protected.

Steps

- 1. Use an editor to create or modify the nvfail_rename file in the system's /etc directory.
- 2. List the path name and file name of database files you want to protect, one file per line, within the nvfail_rename file. You can list any number of files.

Example

/vol/vol1/home/dbs/oracle-WG73.dbf

3. Save the file.

After you finish

For instructions about examining database file validity, see the documentation for your specific database software.

If your database uses LUNs, review the steps to make the LUNs accessible to the host after an NVRAM failure.

Related tasks

Making LUNs accessible to the host after an NVRAM failure on page 384

Making LUNs accessible to the host after an NVRAM failure

After an NVRAM failure, the host no longer has access to data on the LUNs. You must perform a number of actions before the database has access to the LUNs.

Steps

- 1. Examine the LUNs for any data inconsistencies and resolve them.
- 2. If you renamed LUNs using the /etc/nvfail_rename file, remove the .nvfail extension by renaming the LUNs using the lun move command.
- **3.** Bring the LUNs online.
- 4. Export each LUN manually to the initiator.

Database protection using SnapValidator

The SnapValidator software implements Oracle H.A.R.D. checks to detect and reject invalid Oracle data before the data is written to the storage system. Oracle Hardware Assistant Resilient Data (H.A.R.D.) is a system of checks embedded in Oracle data blocks that enable a storage system to validate write operations to an Oracle database.

Next topics

What SnapValidator checks on page 385 *Preparing a database for SnapValidator* on page 386 *Implementing SnapValidator checks* on page 387 *Disabling SnapValidator* on page 390 *Troubleshooting invalid data error messages* on page 391

What SnapValidator checks

The SnapValidator software implements Oracle H.A.R.D. checks to detect and reject invalid Oracle data before it is written to the storage system.

SnapValidator implements the following Oracle H.A.R.D validations:

- Checks for writes of corrupted datafile blocks. This includes the checksum value and validation of selected fields in the block.
- Checks for writes of corrupted redo log blocks. This includes the checksum value and validation of selected fields in the block.
- Checks for writes of corrupted controlfile blocks. This includes the checksum value and validation of selected fields in the block.
- Verifies that writes of Oracle data are multiples of a valid Oracle blocksize for the target device.

Note: SnapValidator is not based on Snapshot technology.

Supported protocols

SnapValidator checks are supported for the following protocols:

- NFS protocol access to files
- FCP or iSCSI protocol access to LUNs

Preparing a database for SnapValidator

Before you can use SnapValidator to protect the Oracle database, you must prepare the database for SnapValidator checks in a test environment to avoid disruptions to the production environment.

Steps

- 1. Ensure that you are working in the test environment, not the production environment.
- 2. Ensure that the Oracle data files or LUNs are in a single volume.

Note: The term *LUNs* in this context refer to the LUNs that Data ONTAP serves to clients, not to the array LUNs used for storage on a storage array.

- 3. Ensure that you do not put the following types of files in the same volume as the Oracle data:
 - Oracle configuration files
 - Files or LUNs that are not Oracle-owned (for example, scripts or text files)
 - Oracle software

For an existing database, you might have to move configuration files and other non-Oracle data to another virtual volume.

4. If you are using new LUNs for Oracle data and the LUN is accessed by non-Windows hosts, set the LUN Multiprotocol Type (ostype) to the appropriate host operating system (solaris, hpux, aix, linux, netware or vmware). If the LUNs are accessed by Windows hosts, the ostype must be windows or windows_gpt, depending on whether your version of Windows uses the GPT partition method. LUNs in an existing database can be used, regardless of their ostype.

For more information about LUN multiprotocol types, see the *Data ONTAP Block Access Management Guide for iSCSI and FC*.

- 5. Ensure Oracle H.A.R.D. checks are enabled on the host running the Oracle application server (the db_block_checksum value in the init.ora file is set to true).
- 6. Implement SnapValidator checks by licensing SnapValidator and then enabling it.
- 7. Ensure you set SnapValidator to log all errors to the storage system consoles for all invalid operations by entering the following command:

```
vol options volume-name svo_reject_errors off
```

- 8. Test your environment by writing data to the storage system.
- **9.** Set SnapValidator to reject invalid operations and return an error to the host and log that error to the storage system consoles for all invalid operations by entering the following command:

```
vol options volume-name svo_reject_errors on
```

10. Put the database into production.

Related tasks

Licensing SnapValidator on page 387 *Enabling SnapValidator checks on volumes* on page 387

Implementing SnapValidator checks

You can protect a new Oracle database stored on a storage system by implementing Oracle Hardware Assistant Resilient Data (H.A.R.D.) checks to detect and reject invalid Oracle data before it is written to the storage system.

- 1. Licensing Snap Validator on page 387
- 2. Enabling SnapValidator checks on volumes on page 387
- 3. Enabling Snap Validator checks on LUNs on page 388

Licensing SnapValidator

Before you can use SnapValidator, you must install the license on the storage system.

Before you begin

Ensure you have the license code that you received when you purchased the SnapValidator license.

Step

1. Enter the following command:

license add license_code

license_code is the license code you received when you purchased the SnapValidator license.

Enabling SnapValidator checks on volumes

You can enable SnapValidator checks at the volume level.

Before you begin

Ensure that SnapValidator has been licensed on the storage system.

Steps

 On the storage system command line, enable SnapValidator by entering the following command: vol options volume-name svo_enable on

All SnapValidator checks are enabled on the volume, with the exception of checksums.

2. Further configure SnapValidator at the volume level by doing some or all of the following steps:

• If you want to enable data checksumming on the volume, enter the following command:

```
vol options volume-name svo_checksum on
```

• If you want to disable block number checks because the volume contains Oracle Recovery Manager (RMAN) backup data enter the following command:

```
vol options volume-name svo_allow_rman on
```

• If you want to set SnapValidator to return an error log to the host and storage system consoles for all invalid operations. You might want to do this when you are testing SnapValidator before you put your database into production enter the following command:

```
vol options volume-name svo_reject_errors off
```

When you set this option to off, SnapValidator only logs errors but does not reject invalid operations.

• If you want to set SnapValidator to reject all invalid operations and return an error log to the host and storage system consoles enter the following command:

vol options volume-name svo_reject_errors on

If this option is not set to on, then SnapValidator detects invalid operations but only logs them as errors. The following is a SnapValidator error example displayed on the storage system console:

```
Thu May 20 08:57:08 GMT [filer_1: wafl.svo.checkFailed:error]:
SnapValidator: Validation error Bad Block Number:: v:9r2 vol:flextest
inode:98 length:512 Offset: 1298432
```

Enabling SnapValidator checks on LUNs

If you enable SnapValidator on volumes that contain database LUNs, you must also enable SnapValidator checks on the LUNs to ensure that SnapValidator does not check write operations to the disk label or partition areas as if they were Oracle write operations.

About this task

To define the offset, you must first identify the offset on your host and then define that offset to the storage system.

- 1. Identifying the disk offset for Solaris hosts on page 389
- 2. Identifying the disk offset for other hosts on page 389
- 3. Defining the disk offset on the system on page 390

Identifying the disk offset for Solaris hosts

You can determine the disk offset for Solaris hosts using the prtvtoc command.

Steps

1. On the host, enter the following command:

```
prtvtoc /dev/rdsk/device_name
```

Example

The following output example shows the partition map for disk c3t9d1s2:

```
prtvtoc /dev/rdsk/c3t9d1s2
* /dev/rdsk/c3t9d1s2 partition map
* Dimensions:
*
            512 bytes/sector
*
           384 sectors/track
*
             16 tracks/cylinder
        6144 sectors/cylinder
          5462 cylinders
*
           5460 accessible cylinders
* Flags:
      1: unmountable
   10: read-only

        *
        First
        Sector
        Last

        * Partition
        Tag
        Flags
        Sector
        Count
        Sector
        Mount
        Directory

        0
        0
        00
        0
        6144
        6143

        2
        5
        01
        0
        33546240
        33546239

        6
        0
        00
        6144
        33540096
        33546239
```

The host console displays a partition map for the disk.

2. Obtain the offset value by multiplying the value of the first sector of the partition you are using by the bytes/sector value listed under Dimensions.

In the example shown in Step 1, which is using slice 6, the disk offset is 6144 * 512 = 3145728.

Identifying the disk offset for other hosts

You can determine the disk offset for hosts that are not Solaris hosts using the dd command.

Steps

- 1. Prepare the LUN for storing Oracle data, for example, by setting up raw volumes.
- 2. On the host console, enter the following command:

dd if=/dev/zero of=/dev/path_to_storage bs=4096 count=1 conv=notrunc

path_to_storage is the path to the LUN on the host.

The host writes an Oracle 4K block of zeros to the storage system.

 Check the SnapValidator error message displayed on the storage system console. The error message displays the offset.

Example

The following error message example shows that the disk offset is 1,048,576 bytes.

```
filerA> Thu Mar 10 16:26:01 EST [filerA:wafl.svo.checkFailed:error]:
SnapValidator: Validation error Zero Data:: v:9r2 vol:test inode:
3184174 length:4096 Offset: 1048576
```

Defining the disk offset on the system

You can define the offset on the system by using the lun set svo_offset command. You need to determine the disk offset for the host system.

Steps

- 1. Use the volume manager tools for your host OS to obtain the value of the offset. For detailed information about obtaining the offset, see the vendor-supplied documentation for your volume manager.
- 2. On the system CLI, enter the following command:

```
lun set svo_offset lun_path offset
```

```
offset is specified in bytes, with an optional multiplier suffix: c (1), w (2), b (512), k (1,024), m (k*k), g (k*m), t (m*m).
```

Disabling SnapValidator

You can turn off SnapValidator checks at the volume level or at the LUN level if checking for invalid Oracle data is no longer a priority.

Next topics

```
Disabling SnapValidator on a volume on page 391
Disabling SnapValidator on a LUN on page 391
```

Disabling SnapValidator on a volume

You can disable all SnapValidator checks on a volume. You can also disable or change a specific SnapValidator option for a volume.

Step

1. As required, choose one of the actions from the following table.

If you want to	Then enter the following command
Disable SnapValidator	vol options volume-name svo_enable off
checks on a volume	SnapValidator does not check Oracle write operations to files or LUNs. The settings for each type of check (for example, checksums) are not disabled. If you re-enable SnapValidator, the settings for each type of check are saved.
Disable a specific SnapValidator option	 vol options volume-name option off option is one of the following: svo_checksum—disables data checksums on the volume. svo_allow_rman—allows block number checks on the volume. You disable this option (set it to off) if the volume does not contain RMAN data. svo_reject_errors—detects invalid operations but does not reject them. Invalid operations are only logged as errors.

Disabling SnapValidator on a LUN

You can disable SnapValidator checks on individual LUNs if you do not want to check for invalid Oracle data on selected LUNs, but want to continue checking on other LUNs.

Step

1. On the system command line, enter the following command:

```
lun set lun_path svo_offset disable
```

Troubleshooting invalid data error messages

You should check that you configured SnapValidator correctly before investigating other reasons when SnapValidator checks return error messages.

About this task

Data ONTAP will send invalid data error messages if SnapValidator detects problems in the following areas:

- Checksum Error
- Bad Block Number
- Bad Magic Number
- No Valid Block Size
- Invalid Length for Log Write
- Zero Data
- Ones Data
- Write length is not aligned to a valid block size
- Write offset is not aligned to a valid block size

Steps

- **1.** If you receive a message indicating that SnapValidator detected or rejected invalid data, verify the following:
 - You enabled the SnapValidator checks on the volumes that contain your data files.
 - You set the SnapValidator checks correctly. For example, if you set the svo_allow_rman volume option to on, then make sure that the volume contains Oracle Recovery Manager (RMAN) backup data. If you store RMAN data in a volume that does not have this option set, then you might receive an error message indicating that SnapValidator detected invalid data.
- **2.** If the SnapValidator options on the system are correctly set but you still receive the above errors, check for the following problems:
 - Your host is writing invalid data to the system. Consult your database administrator to check Oracle configuration on the host.
 - You might have problems with network connectivity or configuration. Consult your system administrator to check the network path between your host and system.
- **3.** If you receive a message indicating that a write operation to a LUN failed, verify that you set the correct disk offset on the LUN. Identify the disk offset and reset the offset defined for the LUN by using the procedures you used when you enabled SnapValidator checks on LUNs.

SnapValidator displays two messages similar to the following when write operations to a LUN fail:

• The first message is generated by SnapValidator and indicates that the system detected invalid data. The error message does not show the full path to the LUN. The following is an example error message:

Thu May 20 08:57:08 GMT [3070: wafl.svo.checkFailed:error]: SnapValidator: Validation error Bad Block Number:: v:9r2 vol:dbtest inode:98 length:512 Offset: 1298432

• The second error message is a scsitarget.write error, which shows the full path to the LUN. The following is an example error message that indicates a write to a specific LUN failed: Thu May 20 14:19:00 GMT [3040: scsitarget.write.failure:error]: Write to LUN /vol/dbtest/oracle_lun1 failed (5)

Related tasks

Enabling SnapValidator checks on LUNs on page 388 *Enabling SnapValidator checks on volumes* on page 387

Virus protection for CIFS

Data ONTAP allows virus-scanning PC clients running a compliant antivirus application to scan files before a CIFS client is allowed to open it.

Next topics

How CIFS virus scanning works on page 395 File types scanned by default on page 396 Setting up and starting virus scanning on page 396 Specifying file types to be scanned on page 399 Excluding file types to be scanned on page 401 Using an inclusion list in combination with an exclusion list on page 403 Specifying shares for scanning on page 404 Displaying the scanner list on page 407 Checking vscan information on page 408 Setting and resetting the request timeout for a virus scan on page 409 Allowing file access when the scan cannot be performed on page 410 Controlling vFiler unit usage of host system's virus scanners on page 410 Checking the status of virus-scanning options on page 411 Stopping a virus scanner session on page 411 *Resetting the scanned files cache* on page 412 Enabling virus scan messages to CIFS clients on page 412 Resolving virus scan server connectivity issues on page 412

How CIFS virus scanning works

CIFS virus scanning is carried out on dedicated PC clients running the Data ONTAP-compliant antivirus application of your choice.

When you enable the virus-scanning process through Data ONTAP on the storage system, the virusscanning application tells the storage system to send file scanning requests.

The virus-scanning application watches for requests from the storage system. Whenever the types of files you specify are opened or changed on the storage system, Data ONTAP sends the PC client a request to scan the file.

The Data ONTAP virus-scanning process can scan multiple storage systems from a single PC client if your virus-scanning application performs this function. For more information about whether a specific virus-scanning application can accommodate scanning multiple storage systems, contact the manufacturer of your virus-scanning application.

File types scanned by default

Certain file types are scanned for viruses by default. You can add file types for better protection or remove file types for faster access.

Table 14: Default file types

??_	DL?	IM?	OFT	SMM
ARJ	DOC	INI	OLE	SWF
ASP	DOT	JS?	OV?	SYS
BAT	DRV	LZH	PIF	VBS
BIN	EML	MD?	РОТ	VS?
CAB	EXE	MPP	PP?	VXD
CDR	GMS	MPT	RAR	WBK
CL?	GZ?	MSG	RTF	WPD
СОМ	HLP	MSO	SCR	XL?
CSC	HT?	OCX	SHS	XML

The following table lists the file types scanned by default.

Note: In this table, the ? character is a wildcard that matches any character or no character. For example, C?? matches C, CL, CPP, C++, and so on.

Attention: To scan all files, you can use ???. However, this might severely degrade performance and is not recommended.

You can also configure Data ONTAP so that file extensions not in the default list are scanned, or only a subset of the default file extensions is scanned.

Setting up and starting virus scanning

You can set up one or more virus-scanning clients to ensure that files on your system are virus-free.

About this task

The storage system anti-virus vscan feature requires NTLM or Kerberos authentication; it does not support Network Information Service (NIS) authentication.

The storage system validates any vscan server which connects to the storage system, and it requires the vscan server to connect as a user who is in the storage system's Backup Operators group.
- 1. Setting up PC clients as virus-scanning clients on page 397
- 2. Enabling virus scanning on the system on page 397
- 3. Setting up secondary scanning clients on page 398
- 4. Setting up McAfee scan detection properties for systems on page 399

Related information

Bug ID 139111: now.netapp.com/NOW/cgi-bin/bol?Type=Detail&Display=139111

Setting up PC clients as virus-scanning clients

You must set up virus-scanning software that Data ONTAP supports, and enable the Data ONTAP virus-scan feature before you can scan for viruses.

Steps

- 1. Make sure that the operators of the PC clients that you want to configure as virus-scanning clients are configured as "Backup Operators" or higher on the storage systems on which they will conduct virus scanning.
- 2. Install the Data ONTAP-customized commercial virus scan software on the PCs that you want to configure as virus-scanning clients. Follow the directions that accompany the virus scan software product.
- **3.** After installation and configuration of the virus scan software on the PCs is complete, confirm the success of the installation by listing the IP addresses of the PCs now configured as virus-scanning clients. At the storage system console, enter the following command:

vscan scanners

The system displays a table listing the IP addresses of the active virus-scanning clients for this storage system.

4. Leave the virus-scanning client on and connected to the storage system or storage systems on which it is carrying out its virus scan operations.

Enabling virus scanning on the system

Before you use the virus-scanning software, you must start the virus scan feature of Data ONTAP.

Step

1. Enter the following command:

```
vscan on [-f][on|off]
```

-f forces virus scanning to be enabled even if no virus-scanning clients are available to scan files.

Setting up secondary scanning clients

If you configured more than one virus-scanning client to scan files on a system, you can place the additional clients on standby, not actively scanning for viruses unless the primary virus-scanning client becomes unavailable.

Steps

1. List the virus-scanning clients configured to scan on this system. On the system console enter the following command:

vscan scanners

Example

>vscan scanners
Virus scanners(IP and Name) P/S ...
132.132.59.12 \\XLAB-WTS Pri

2. Specify, by IP address, the PC clients you want to serve as standby virus scanners by entering the following command:

```
vscan scanners secondary_scanners scanner_ip[, scanner_ip...]
```

scanner_ip can be either of the following:

- IP addresses of one or more of the configured virus-scanning clients displayed in Step 1
- IP addresses of PCs not yet configured as a virus-scanning client

Note: If the IP address you entered belongs to a PC not yet configured as a virus-scanning client for this system, you must configure it for this setting to take effect.

Example

vscan scanners secondary_scanners 132.132.59.14

3. Use the vscan scanners command to confirm your configuration.

Example

>vscan scanners Virus scanners(IP and Name) P/S ... 132.132.59.14 \\BORIS-PC Sec ... 132.132.59.12 \\XLAB-WTS Pri Secondary scanners IP address list 132.132.59.14,10.20.30.40

Note: In this example, the address 10.20.30.40 belongs to a PC that is enabled as a standby scanner but is not turned on; therefore it is not listed in the Virus scanners (IP and Name) table, but it is listed in the Secondary scanners IP address list.

Setting up McAfee scan detection properties for systems

The anti-virus scanner is registered with the system, but the scanner scans the system share only when configured properly.

About this task

When setting up McAfee VirusScan for a system with Data ONTAP 7.1 and later, you need to specify the nature of detection.

Steps

- 1. Open a VirusScan console.
- 2. Click Task > On-Access Scan Properties > All Processes > Go to 'Detection' Tab .

The Detection dialog opens.

3. In the Scan Files section select: When reading from disk and On network drives.

Note: You must select both options to scan the system shares.

This information can be found in the VSNA 7.1 help menu.

Open Help > On-Access Scanning > Configuring the on-access scanner > All processes and default processes > Detection Properties.

Specifying file types to be scanned

If the default list of file types to be scanned is not inclusive enough or you plan to use an exclusion list in conjunction with the inclusion list, you can modify the list of file types to be scanned.

Next topics

Displaying file types to be scanned on page 400 Adding file types to be scanned on page 400 Replacing file types to be scanned on page 400 Removing file types to be scanned on page 401 Resetting file types to be scanned on page 401

Displaying file types to be scanned

You can see the list of file types to determine if you need to add to the list, remove from the list, or replace the list with other file types that you want scanned.

Step

1. Enter the following command:

vscan extensions include

The current list of extensions to be scanned is displayed.

Adding file types to be scanned

You can add file types to the extensions list to include file types that you expect to store on the system.

About this task

A default list of file extensions is made available when you enable virus scanning; however, you can specify additional file extensions that are not in the default list.

Step

1. Enter the following command:

vscan extensions include add ext[,ext...]

ext is the extension you want to add.

Example

```
vscan extensions include add txt
```

Note: Up to 255 file extensions can exist in the file extensions list.

Replacing file types to be scanned

You can replace the file types in the exclusion list with a short list if you are also specifying a list of file types to exclude from a scan.

Step

1. Enter the following command:

```
vscan extensions include set ext[,ext...]
```

ext is the extension you want to set.

Removing file types to be scanned

You can remove file types from the extension list if you are also specifying files types to exclude from a scan.

Step

1. Enter the following command:

```
vscan extensions include remove ext[,ext...]
```

ext is the extension you want to remove.

Resetting file types to be scanned

You can reset the list of file types to be scanned if you want to return to the default list of file types to be scanned.

Step

1. Enter the following command:

vscan extensions include reset

The list of file extensions is set to the default list.

Related references

File types scanned by default on page 396

Excluding file types to be scanned

You might prefer to specify what files types to exclude, rather than include, in a virus scan because of the proliferation of new file types (with new file name extensions) that might be stored on the storage system.

Next topics

Displaying file types to exclude from scanning on page 402 Creating a list of file types to exclude from scanning on page 402 Adding file types to exclude from scanning on page 402 Removing file types to exclude from scanning on page 403 Resetting the exclude file types list to empty on page 403

Displaying file types to exclude from scanning

You can see a list of file types excluded from scanning to determine if you need to add to the list or remove from the list.

Step

1. Enter the following command:

vscan extensions exclude

The current list of extensions to be excluded from virus scan is displayed.

Creating a list of file types to exclude from scanning

You can create a list of file types to exclude from virus scanning if you want to use the exclude feature either by itself or in combination with the include feature.

Step

1. Enter the following command:

vscan extensions exclude set ext[,ext...]

ext is the extension or extensions that you want to set for the list of file types excluded from virus scan.

Note: Using the set parameter will replace completely any existing file type extensions in the exclude list with the extensions you specified in this command. If an exclude list already exists and you merely want to add to it, use the vscan extensions exclude add command.

Adding file types to exclude from scanning

You can add file types to exclude from virus scanning if you decide that a file type is safe and does not need virus scanning.

Step

1. Enter the following command:

vscan extensions exclude add ext[,ext...]

ext is the extension you want to add to the list of file types excluded from virus scan.

```
vscan extensions exclude add txt
```

Note: Up to 255 file extensions can exist in the file extensions list.

Removing file types to exclude from scanning

You can remove file types to exclude from virus scanning if you decide that a file type is not safe and requires virus scanning.

Step

1. Enter the following command:

```
vscan extensions exclude remove ext[,ext...]
```

ext is the extension you want to remove from the list of file types excluded from virus scan.

Resetting the exclude file types list to empty

You can reset the exclude file types list if you decide that none of the listed file types are safe or you want to remove the exclusion list.

Step

1. Enter the following command:

```
vscan extensions exclude reset
```

The list of file extensions is set to the default empty value.

Using an inclusion list in combination with an exclusion list

Because of the proliferation of new file types, you can allow any file types that you consider to be safe to go unscanned and at the same time ensure that any new, unfamiliar file type stored on the system does get scanned.

About this task

You can use the inclusion list to specify a general virus scan and use the exclusion list to specify the file types that are excluded from the general virus scan.

Steps

1. At the system console, enter the following command line specifying the extensions of all file types that you want to exclude from virus scan:

vscan extensions exclude set ext[,ext...]

2. Enter the following command line to specify virus scan of all other file types:

vscan extensions include set ???

This command line instructs the virus scan program to scan all file types stored on the system. But the result of both the "exclude" and the "include" command lines together is that all file types are scanned except for the file types whose extensions have been specified in the vscan extensions exclude command line described in Step 1.

Specifying shares for scanning

You can turn off virus scanning for files in a share if the share is used only by trusted users, the files are restricted to read-only mode, or speed of access is more important than safety.

Next topics

Turning virus scanning off for any access on page 404 Turning scanning on for any access on page 405 Turning scanning off for read-only access on page 405 Turning scanning on for read-only access on page 405 Adding shares with virus scanning turned off on page 406 Adding shares with virus scanning turned off for read-only access on page 406

Turning virus scanning off for any access

You can turn off virus scanning for files in a share if the share is used only by trusted users, or if access speed is more important than safety.

About this task

Virus scanning for a share is turned on by default.

Step

1. Enter the following command:

```
cifs shares -change share_name -novscan
```

share_name is the name of the share for which you want to turn off virus scanning.

The application does not perform a virus scan when clients access this share. The setting is persistent across reboots.

Note: You can set these share attributes for all CIFS user home directories by using cifs.homedir as the share name, as given in the following example.

cifs shares -change cifs.homedir -novscan

Turning scanning on for any access

You can turn on virus scanning for files in a share if the share is used by users other than trusted users or safety is more important than speed of access.

Step

1. Enter the following command:

```
cifs shares -change share_name -vscan
```

share_name is the name of the share for which you want to turn on virus scanning.

The application performs a virus scan when clients open files on this share. The setting is persistent after reboot.

Note: You can set these share attributes for all CIFS user home directories by using cifs.homedir as the share name, for example, cifs shares -change cifs.homedir - vscan

Turning scanning off for read-only access

You can turn off virus scanning for files in a share if the files are restricted to read-only mode or speed of access is more important than safety.

About this task

Virus scanning for a share is turned on by default.

Step

1. Enter the following command:

cifs shares -change share_name -novscanread

share_name is the name of the share for which you want to turn off virus scanning.

The application does not perform a virus scan when clients open files on this share for read access. The setting is persistent after reboot.

Turning scanning on for read-only access

You can turn on virus scanning for files in a share if safety is more important than speed of access.

Step

1. Enter the following command:

cifs shares -change share_name -vscanread

share_name is the name of the share for which you want to turn on virus scanning.

The application performs a virus scan when clients open files on this share for read access. The setting is persistent after reboot.

Adding shares with virus scanning turned off

You can add a share with virus scanning turned off if the share will be used only by trusted users, the files are restricted to read-only mode, or speed of access is more important than safety.

About this task

For backup purposes, you can create two shares on the same directory: one share with scanning disabled and a share-level Access Control List (ACL) that allows access only to a backup account; the other share available to normal users and with scanning enabled. Backup can be performed on the share with no virus scanning and improved performance, while normal users continue to access the data through the regular share and get virus protection.

Note: Virus scanning for a share is turned on by default.

Step

1. Enter the following command:

cifs shares -add share_name /path -novscan

share_name is the name of the share with virus scanning turned off that you want to create.

path specifies where you want the share created.

Data ONTAP creates a share with virus scanning turned off.

Adding shares with virus scanning turned off for read-only access

You can add a share with virus scanning turned off if the share will be used only by trusted users, the files are restricted to read-only mode, or speed of access is more important than safety.

About this task

For backup purposes, you can create two shares on the same directory: one share with scanning disabled and a share-level Access Control List (ACL) that allows access only to a backup account; the other share available to normal users and with scanning enabled. Backup can be performed on the share with no virus scanning and improved performance, while normal users continue to access the data through the regular share and get virus protection.

Note: Virus scanning for a share is turned on by default.

Step

1. Enter the following command:

cifs shares -add share_name /path -novscanread

share_name is the name of the share with virus scanning turned off that you want to create.

path specifies where you want the share created.

Data ONTAP creates a share with virus scanning turned off for read-only access.

Displaying the scanner list

You can determine if scanner clients are receiving requests and scanning files by looking at the scanner list.

Step

1. Enter the following command:

```
vscan scanners
```

Example

The system outputs the names of and information about the virus-scanning clients in a display like the one shown in the following paragraph.

Virus scanners(IP and Name) Connect time Reqs Fails (dd:hh:mm) 10.61.155.118 \\WIN2K-NAN 00:02:23 2 0 10.60.129.152 \\WIN2K-RTB 00:00:01 0 0

The *Connect time* field displays the length of time the scanner has been connected to the system.

The *Reqs* field displays the total number of requests sent by the system to that scanner.

The Fails field displays the number of requests that failed.

Primary virus scanner not listed

Sometimes, after configuring a primary virus scanner, it no longer services client requests and does not appear in the list when you run vscan scanners on the system CLI. However, the previous

configuration can be observed in old AutoSupport messages. To resolve this problem, perform the following steps.

Steps

- 1. Ping the virus scanner to determine whether network connectivity exists between the storage system and the virus scanner. If connectivity exists, proceed to Step 2.
- 2. Determine whether the virus scanning software is properly installed and configured. If it is not functioning properly, see the virus scanner documentation or contact the manufacturer's technical support staff to identify and correct any problems. If there are no problems, proceed to Step 3.
- **3.** Ensure that the virus scanning software service is started. If it is not, restart the virus scanning application on the virus scan server.

Note: Even if if the service appears to be started, it might be frozen. Restarting the service causes it to re-register with the storage system. The primary scanner is not configured on the secondary storage system side. Secondary virus scanners are configured on the secondary storage system only by using the vscan scanner secondary_scanners command.

Checking vscan information

You can quickly determine whether the virus scanner is on, how well the virus scanner is working, and what files extensions are scanned and not scanned.

Step

1. Enter the following command:

vscan

The system displays the on/off status of vscan, information about the virus-scanning clients, the list of extensions to scan, and the number of files scanned and number of scan failures in a display like the one shown in the following paragraph.

Example

RTF,SCR,SHS,SMM,SYS,VBS,VS?,VXD,WBK,WPD,XL?,XML List of extensions not to scan: Number of files scanned: 28 Number of scan failures: 0

Note: The number of scan requests and failures shown in the table at the beginning of the output represents this connection session. The second set of numbers at the end of the output reflects the total scans and failures since vscan was turned on.

Setting and resetting the request timeout for a virus scan

You can change how long Data ONTAP should wait for a virus scan to finish before requesting the status of the scan. You might need to do this when you have a slow network or your virus scanner is slow, and you want to avoid the scan request from timing out prematurely.

About this task

The request is repeated as often as necessary until the scan is complete or the host gives up. By default, the virus scan timeout is 10 seconds.

Step

1. You can either set a new scan request timeout value or reset a value that you previously set back to the default value.

If you want to	Then enter the following command
Set a new scan request timeout value.	vscan options timeout set value
	<i>value</i> is a setting from 1 to 45 seconds. The recommended setting is between 8 and 12 seconds.
Reset the scan request timeout value back to the default value of 10 seconds	vscan options timeout reset

Allowing file access when the scan cannot be performed

You can specify that files can be accessed if a scanner is not available or if scan requests time out.

About this task

The default setting is that file access is denied if a successful scan cannot be performed, that is, the option is set to on. When this option is set to off, access is allowed even if a scan cannot be performed.

Step

1. Enter the following command:

```
vscan options mandatory_scan [on|off]
```

Controlling vFiler unit usage of host system's virus scanners

Depending on your security concerns, you can either have each vFiler unit register with the virus scanner or allow the vFiler units to use the host system's virus scanners.

About this task

By default, vFiler units can scan files using the virus scanners that are connected to the host system. For more information about vFiler units, see the *Data ONTAP MultiStore Management Guide*.

If you have several vFiler units owned by separate departments, and you have security concerns, then you might want to have the virus scanner register with each vFiler unit. For more information, see the documentation for the virus scan software.

You can have the virus scanner register with only the physical system. In this way, only the system authenticates the virus scanner.

Step

1. You can register a virus scanner in one of two ways. As required, choose one of the actions from the following table.

If you want to	Then from the vFiler context, enter the following command
Specify a configuration for each vFiler unit	vscan options use_host_scanners off

If you want to	Then from the vFiler context, enter the following command
Specify a configuration that applies to all vFiler units in the system	vscan options use_host_scanners on

Note: The vscan options command is not supported on the default vFiler unit.

Checking the status of virus-scanning options

You can quickly determine whether the virus-scanning options you want to use are set to the values you want.

Step

1. Enter the following command:

vscan options

The storage system outputs the state of the virus-scanning options in a display like the following:

vscan options mandatory_scan on vscan options timeout: 12 sec vscan options use_host_scanners on

Note: The use_host_scanners option applies only to vFiler units and is displayed only if the vscan options command was run on a vFiler unit.

Stopping a virus scanner session

You can stop a scanner session if you have to terminate CIFS on a storage system or if you upgrade your antivirus program to a new version.

Step

1. Enter the following command:

vscan scanners stop scanner_IP

scanner_IP is the IP address of the virus scanner you want to stop.

Resetting the scanned files cache

If you have a new virus definitions file, you might want to clear the cache and rescan the files that were scanned using an old virus definitions file.

About this task

Data ONTAP caches information about previously scanned files to avoid rescanning those files.

Step

1. Enter the following command:

vscan reset

Enabling virus scan messages to CIFS clients

You can send explicit virus-warning messages to inform CIFS clients, if the virus scanner detects a threat within a file that a CIFS client is trying to access.

About this task

If this feature is not enabled, CIFS clients attempting to access virus-infected files that have been detected by virus scanning will simply receive a general file unavailable message.

Step

1. Enter the following command:

vscan options *client_msgbox* {on|off}

on enables the display of virus warning messages.

off disables the display of virus warning messages.

Resolving virus scan server connectivity issues

If the virus scan server disconnects often and displays an error message, you can try to resolve the issue by using certain checks.

Symptoms: Any one of the following symptoms is observed:

- The virus scan server was unable to scan a file on the storage system.
- The virus scan server disconnects and reconnects after a brief period of time.
- The virus scan server disconnects and reconnects often with port 139 error.

Cause of this problem:

- NetBIOS over TCP was not enabled on the server side.
- The server service was stopped.

Solution:

- **1.** Enable NetBIOS over TCP on the server. For more information on this procedure, review this Microsoft TechNet article, NetBIOS over TCP/IP.
- 2. Make the server service automatic.
- 3. Start the service.

Related information

http://technet.microsoft.com/en-us/library/bb727013.aspx

<u>Glossary</u>

The following terms are used for Data ONTAP features:

ACL	Access control list. A list that contains the users' or groups' access rights to each share.
active file system	A file system excluding its Snapshot copies.
aggregate	A manageable unit of RAID-protected storage, consisting of one or two plexes, that can contain one traditional volume or multiple FlexVol volumes.
blocking factor	The number of tape blocks that are transferred in each write operation.
CIFS	Common Internet File System. A protocol for networking PCs.
console	A terminal that is attached to a storage system's serial port and is used to monitor and manage storage system operation.
DNS	Domain Naming Service. An Internet service for finding IP addresses.
dump path	A path that specifies one volume, qtree, or subtree to back up.
file mark	Data on a tape that signals a boundary of a tape file.
НТТР	HyperText Transfer Protocol. An Internet Web transfer protocol.
increment chain	A series of incremental backups of the same path.
inode	A data structure containing information about files on a storage system and in a UNIX file system.
local tape device	A program-based functionality associated with a tape drive that is directly attached to a storage system that is performing a tape operation.
MD5	Message Digest 5. A checksum algorithm described in RFC (Request For Comments) 1321, a proposal for an Internet standard.
mirror	A volume that contains a read-only copy of data in the active file system of another volume, usually on another storage system.
NDMP	Network Data Management Protocol. A protocol that allows storage systems running Data ONTAP to communicate with backup applications, and provides capabilities for controlling the robotics of multiple tape backup devices.
NFS	Network File System. A protocol for networking UNIX-based computers.
NIS	Network Information Service, formerly called Yellow Pages. An administrative database for networks.

NVFAIL	Software that warns you of compromised database validity and automatically renames the database so that it does not restart automatically.
NVRAM	Nonvolatile RAM in the storage system, used for logging incoming write data and NFS requests. Improves system performance and prevents loss of data in case of a storage system or power failure.
Open Systems platform	A system, such as a server running Solaris, HP-UX, or Windows, whose data can be backed up to a SnapVault secondary storage system.
Open Systems SnapVault agent	A software module that can be installed on a system and enables that system to back up its data to a SnapVault secondary storage system.
plex	A physical copy of a file system. An unmirrored volume has one plex; a mirrored volume has two identical plexes.
primary storage system	A system whose data is to be backed up by SnapVault.
quota	A limit placed on a file system that restricts disk space usage by files with a given User ID (UID) or group ID (GID).
qtree	A special subdirectory of the root of a volume that acts as a virtual subvolume with special attributes.
RAID	Redundant Array of Independent Disks. A technique that protects against disk failure by computing parity information based on the contents of all the disks in the array. Storage systems running Data ONTAP software use RAID Level 4 or RAID DP, which stores all parity information on a single disk. Storage systems running Data ONTAP software use RAID 0 with third-party storage.
Remote Shell	A program that enables a user on one system to execute a program on another system. Remote Shell connections are usually not interactive.
remote tape device	A program-based functionality associated with a tape drive that is not directly attached to a storage system that is performing a tape operation.
root volume	The volume that contains information that controls the entire storage system, usually in the /etc/rc file.
secondary storage system	A storage system to which data is backed up by SnapVault.
share	A directory or directory structure on the storage system that has been made available to network users and can be mapped to a drive letter on a CIFS client.
SnapMirror	Software that performs automated file system replication of a volume onto the same or a separate disk or storage system.

SnapRestore	Software that restores an entire volume to the state recorded in a previously taken Snapshot copy.
Snapshot copy	An online, read-only copy of the entire file system that protects against accidental deletions or modifications of files without duplicating file contents. Snapshot copies enable users to restore files and enable administrators to back up the system to tape while it is in use.
Snapshot reserve	The portion of a volume's disk space that is reserved for Snapshot copies.
source storage system	The storage system from which you are replicating data.
subtree	A directory in a volume or qtree.
tape block	1,024 bytes of data.
tape device	A specific functionality of a physical tape drive that you create by specifying information in the tape device name when you install a tape drive or tape stacker.
tape file	Data on a tape delimited by file marks.
tape library	Hardware that can access tape cartridges randomly.
tape medium changer	Hardware that can load and unload tape cartridges.
tape stacker	Hardware that can access tape cartridges from a stack.
volume	A file system.
volume copy	A way of copying both data in the active file system and data in Snapshot copies from one volume to another.

Index

/etc/hosts.equiv file 337 /etc/nvfail_rename file configuration files 383

A

aborting SnapVault transfers 275 access to remote data using SnapMirror 83 aggregate mirrored, add array LUNs 371 mirrored, converting 368 mirrored, creating 366 splitting, mirrored 375 asynchronous mirroring 81

B

backup and standby service 282, 283 backups online disadvantages 31 online, advantages 30 online, methods 31 using SnapMirror 83 bootup with nvfail enabled 381

С

cifs virus scan 404 cifs shares command 405, 406 CIFS, virus protection for 395 commands aggr split 375 aggr verify resume 380 aggr verify status 380 cifs shares 404 df (finds Snapshot copy disk consumption) 54 dump 297 hostname 137 rsh (with vol copy abort) 346 See also SnapRestore commands 75 snap autodelete, off 66 snap autodelete, reset 66

snap autodelete, show 65 snap create 50 snap delete 68 snap list 187 snap sched 46, 92, 252 SnapMirror 100 snapmirror abort 180 snapmirror break 173 snapmirror destinations 113 snapmirror initialize 122 snapmirror migrate 171 snapmirror off 145 snapmirror quiesce 173, 178, 179 snapmirror release 181, 285 snapmirror resume 179 snapmirror resync 189, 285 snapmirror retrieve 202 snapmirror status 154 snapmirror store 201 snapmirror throttle 165 snapvault modify 264, 265 snapvault release 285 snapvault snap 286 snapvault snap create 268 snapvault snap preservations 289 snapvault snap preserve 289 snapvault snap sched 245, 247, 265 snapvault snap unsched 251 snapvault start 241 snapvault update 285 vol copy 333 vol copy throttle 345 vol options, guarantee 124 compression SnapMirror, viewing ratio 153 compression feature disabling for new Open Systems SnapVault relationship 281 disabling globally 280 enabling for new Open Systems SnapVault relationship 279 enabling globally 279 for Open Systems SnapVault 278 concurrent replication operations maximum, active/active configuration 132 configuration files

420 | Data ONTAP 7.3 Data Protection Online Backup and Recovery Guide

snapmirror.conf 135, 144 context-based deduplication 295 copying one volume to another 341 copying volumes. See volume copy 335 create_unicode option 298

D

data loss disaster 32 database file protection renaming files 383 databases 32 deleting Sanpshot copies automatically 64 disabling the compression feature 281 disabling the compression feature globally 280 disaster recovery using SnapMirror 83 disk geometry 90 displaying SnapVault Snapshot copies 258 dump command 297

Е

enabling Snapshot copies 252 enabling the compression feature 279 enabling the compression feature globally 279 estimating initial backup time using SnapVault 236 example Snapshot copies, restricting access 41 examples vol copy throttle command (controls volume copy speed) 345 volume copy start command 341

F

file hosts 105 SnapMirror log 105 snapmirror.allow 105 snapmirror.conf 105 file access times of Snapshot copies defined 50 from CIFS clients 52 from NFS clients 51 file extensions excluding from a scan 402

for vscan, excluding 402 replacing extensions to exclude from a scan 402 resetting to vscan default exclude list 403 adding to vscan list 400 excluding from a scan 402 for vscan, viewing 400 removing from a vscan exclude list 403 removing from vscan 401 replacing in vscan list 400 resetting to vscan default list 401 viewing extensions to exclude from a scan 402 vscan 404 file folding defined 61 disabling 62 File System Export (FSE) mode backup 295 File System Export backup mode 294 file versions of Snapshot copies finding all 50 files nvfail rename 383 protecting with nvfail option 383 restoring from Snapshot copies 44 FlexVol volumes resynchronize 193

H

hardware SnapMirror over Fibre Channel, requirements 208 HP-UX backing up with SnapVault 228 See also open systems platform 228

I

IPv6 SnapMirror 107 vol copy 335

L

license command 283 licenses for SnapVault 238 limitations of SnapVault for NetBackup 297 load balancing using SnapMirror 83 LUN Snapshot copy, relationship 43 LUN (logical unit number) data protection of volumes containing LUNs 35 LUN clones description 262 modes of transfer 262 transfer using qtree SnapMirror 197 LUN clones transfer non-optimized mode 263 optimized mode 263 LUN clones transfer in optimized mode 263 lun commands lun set 390 LUNs (array) adding to mirrored aggregate 371 planning pools 357-359 requirements 353 troubleshooting 360 with SyncMirror 351-353, 357-360

Μ

manually updating a qtree on the secondary 266 maximum concurrent replication operations, active/active configuration 132 concurrent replications 128 concurrent targets, SnapVault 236 media server 294 migration traditional volume to FlexVol volume 100 mirrored aggregate add array LUNs 371 converting 368 creating 366 splitting 375 mirroring asynchronous 81 synchronous 81 multiple paths convert to 150 failover mode 149 implement 150 multi-plexing mode 149

N

NDMPcopy 297 ndmpd on option (turns NDMP service on) 239 NearStore personality license 131 NetBackup catalog 294 NetBackup master server 294 NFS clients accessing Snapshot copies from 39 determining Snapshot copy file access time from 51 non-optimized mode 262 NVFAIL feature 32 nvfail option bootup process 381 renaming files for database protection 383 using nvfail_rename file 383 nvfail rename file 383

0

online data recovery 29 open systems agent definition 226 open systems platform definition 226 non-filer systems supported for SnapVault backup 228 operation numbers using with vol copy abort 345 optimized mode 262 options cifs.snapshot_file_folding.enable 62 ndmpd 239 replication 103 replication.throttle.enable 164 replication.throttle.incoming.max kbs 164 replication.throttle.outgoing.max_kbs 164 snapmirror 103 snapmirror.access 132 snapvault.access 239 snapvault.enable 238 snapvault.enable off 278 snapvault.enable on (turns SnapVault on) 301 snapvault.preservesnap on 249 vol.copy.throttle (controls volume copy speed) 345 options command snapvault.access 239 Oracle H.A.R.D. with SnapValidator SnapValidator:described 385

P

permissions for Snapshot copies 39 planning primary and secondary qtree locations 233 SnapVault backup schedule 234 planning SnapVault backups 233 plexes 349 policy-based automated data protection Protection Manager 35 primary storage system Data ONTAP and open systems platforms 228 definition 226 primary system qtree definition 226 private networks for volume copy 335 Protection Manager 35

Q

qtree replication guidelines 122 replication quotas 122 qtree replication using SnapMirror 97

R

re-creating the SnapVault relationship 284 recommendations when using multistream backups SnapVault for NetBackup 291, 293, 296-300 replication concurrent operations, maximum 128 requirements SnapMirror over Fibre Channel, hardware 208 resize volume SnapMirror source-destination pair 174 restart transfer 189 restoring Snapshot copies Shadow Copy Client tools 44 retaining SnapVault Snapshot copies 249 retry transfer 189 returning storage system to the original configuration 286 reusing a volume 315 reverting volumes with SnapRestore 75 root volumes, reverting with SnapRestore 74 rsh command using with vol copy abort 346 using with vol copy status 344

S

SAN (data protection in the SAN environment) 35

schedules for default Snapshot copies 44 for user-defined Snapshot copies 48 strategies for Snapshot copies 47 secondary storage system definition 226 role in a basic SnapVault deployment 228 secondary system qtree definition 226 security setting 298 setting the access option 239 shares adding with virus scanning turned off 406 enabling or disabling vscan 405 enabling or disabling vscan for read-only shares 405 virus scan 404 snap list -o command (lists origins of gtree Snapshot copies) 261 snap list -q command (lists qtree Snapshot copies) 259 snap list command determining which Snapshot copies to delete 67 snap list output calculation 56 snap restore -t file command (reverts file from Snapshot copy) 77 SnapDrive for Windows 262, 264 **SnapMirror** abort transfer 180 ACLs, replicating 207 basic setup 108 block transfer 179 cascading 110 change update schedule 144 changing transfer rates 165 check data transfer status 183 check gtree initialization 127 check volume initialization 126 CIFS access 92 commands 100 compression 151 compression ratio, viewing 153 compression statistics, view 151 concurrent replications, maximum 128 considerations 88 considerations for use with tape 166 convert destination to writable 172 convert single-path to multi-path 150 converting asynchronous replication to synchronous 178 deployment 96

destination space guarantee 124 display updates on destination 187 enable 87 files 105 firewall usage 110 FlexClone considerations 199 format of log files 184 implement multiple paths 150 initialize 122 initialize destination 121 initialize destination for non-gtree data 125 initialize destination using tape 122 initialize gtree 126 initialize volume 125 IPv6 107 license 87 list destinations 113 log file examples 184 management through Protection Manager 198 manual update 146, 147 migrate data 171 move gtree source 169 move volume source 167 multiple paths, failover mode 149 multiple paths, multi-plexing mode 149 options 103 over Fibre Channel 207 points of caution 90 prerequisites 88 protection of SnapVault secondaries 204 gtree destination volume clone 200 gtree replication 82, 97 gtree replication guidelines 122 gtree replication quotas 122 quiesce destination 178 quiesce, command 179 quota restrictions 172 recommendations 91 release partners from relationship 181 replication from SnapVault secondary to tape 200 replication of volume to tape 165 resize source-destination volume pair 174 restart transfer 189 restrictions 89 restructure cascade 114 resume transfer after quiesce 179 resync 192 resynchronize relationship 190 retry transfer 189 S family Edition 206

semi-sync, mode 85 snapmirror.access option 132 snapmirror.allow file 133 Snapshot copy management 92 Snapshot copy naming 186 source and destination ports 110 specifying destinations 132, 133 specifying schedule 135 status messages 154 supported cascade configurations 111 supported configurations 97 supported three-hop cascade configurations 111 sync, mode 85 synchronous 84 synchronous modes 85 synchronous replication 86 synchronous replication to asynchronous mode 86 synchronous, considerations 93 synchronous, estimating aggregate size 95 synchronous, guidelines for growing aggregates 86 TCP window size 162 troubleshooting 223 turn off scheduled update 146 turn off update 145 turn on logging 184 use with dump command 203 use with MultiStore 198 use with SnapDrive 198 uses 83 volume replication 82, 97 volume, deduplication 91 when to use complete transfer 121 snapmirror initialize failure 127 interruption 127 SnapMirror over Fibre Channel configure 212 hardware requirements 208 requirements 211 supported functionality 212 supported switches 208 topology 209, 210 traffic zones 210 troubleshooting 219 snapmirror release command 285 snapmirror resync command 285 snapmirror.allow file sample 133 snapmirror.conf file distribution 135

editing 136 limit on entries 135 sample entry 137 syntax 137 SnapRestore avoiding reversion of configuration files 74 effect on Snapshot copies used 73 how it works 71 license requirement for 71 prerequisites for 73 reverting root volumes using 74 using with NFS 75 what it does not revert 71 when to use 72 SnapRestore commands snap restore -t file (reverts file from Snapshot copy) 77 snap restore -t vol (reverts volume from Snapshot copy) 78 Snapshot copies accessing from CIFS clients 41 from NFS clients 39 changing schedules (snap sched) 48 CIFS 40 creating manually 50 default schedule 44 defined 37 deleting manually 68 deleting when locked 69 determining which to delete 57 df command (displays disk usage) 54 directories on NFS clients 39 directory structure on NFS clients 39 disk consumption by 50, 68 how snap list results are calculated 55 SAN environment, using 42 using with SnapRestore 71 Snapshot copy access, restricting 41 finding from CIFS client 52 LUN, relationship 43 newest common 90 preserving 288 schedule arguments 47 schedule, disabling 49 schedules, types 45 Snapvault, aborting creation 276 Snapshot copy basename

definition 226 Snapshot copy commands create Snapshot copies manually 50 snap delete 68 snap list -o (lists origins of gtree Snapshot copies) 261 snap list -q (lists gtree Snapshot copies) 259 snap list (shows disk space used by Snapshot copies) 55 snap list (shows disk space used by Snapshot) 58 snap rename (renames Snapshot copies) 70 snap reserve (sets percent of disk space for) 61 snap sched 46, 48 **SnapValidator** disabling checks on LUNs 391 disabling checks on volumes 391 disk offset for other hosts 389, 390 disk offset for Solaris 389 enabling checks on LUNs 388 enabling checks on volumes 387 error messges 391 licensing 387 preparing database files 386 **SnapVault** ACLs, replicating 207 advantages 227 backup for gtrees ending 276 backup relationship, starting a 240 basic configuration for storage systems running Data ONTAP secondary storage 301 basic deployment 228 changing configuration 265 concurrent replications, maximum 128 data migration using SnapMirror 204 destination volume clone 261 maximum concurrent targets 236 maximum number of Snapshot copies per volume 234 secondary to SnapMirror deployment 230 Snapshot copy creation, aborting 276 use with MultiStore 290, 291 snapvault abort 275 SnapVault baseline transfer definition 226 SnapVault commands snap sched -x command (configures secondary Snapshot copy schedule) 247 snap sched command (configures primary Snapshot copy schedule) 245

snapvault release (releases unneeded Snapshot copies) 271, 277 snapvault restore (restores data from secondary to primary) 271 snapvault snap create (manually updates existing Snapshot copy) 268, 269 snapvault snap unsched command (unconfigures SnapVault) 251 snapvault start (initializes SnapVault backup relationship) 241, 271 snapvault status (checks data replication status) 253 snapvault stop (ends backup for qtrees) 276 snapvault update (updates SnapVault secondary gtree) 267 SnapVault for NetBackup benefits 293 default option for new volumes 296 description 291 limitations 297 NetBackup server master server 293 media server 293 policy definition 293 primary storage system defined 293 recommedations for saving space 298 recommendations for grouping policies 300 recommendations for snap reserve setting 300 recommendations for snap sched setting 300 recommendations for using separate DSUs 300 secondary storage system definition 293 secondary system gtree definition 293 terminology 293 usage recommendations 298 SnapVault incremental transfer definition 226 snapvault modify command 265 SnapVault primary and secondary on the same storage system 234 snapvault release command 285 SnapVault secondary storage system protection 282 snapvault snap command 252 snapvault snap sched -x command (configures secondary Snapshot copy schedule) 247 snapvault snap sched command 265 SnapVault Snapshot copies

retaining 249 SnapVault Snapshot copy definition 226 snapvault start command initializes SnapVault backup relationship 241, 271 snapvault status command 254 snapvault status command (displays information about snapvault Snapshot copies) 253 snapvault update 285 snapvault.enable off option 278 snapvault.enable on option (turns SnapVault on) 301 Solaris backing up with SnapVault 228 Solaris hosts with SnapValidator 389 speed for copying a volume 345 synchronous SnapMirror 84 SnapMirror, considerations 93 synchronous mirroring 81 SyncMirror adding array LUNs to mirrored aggregate 371 advantages 350 aggregate 350 description 349 license, add 361 license, remove for aggregates 362 mirrored aggregate, converting 368 mirrored aggregate, creating 366 mirrored aggregates, create 363 planning pools 357-359 plex, view 364 requirements 351, 353 spare pool, view 364 troubleshooting 360 with third-party storage 351-353, 357-360

Т

tape backup of a SnapVault secondary 229 TCP window size SnapMirror 162 tools to use to protect against data loss disasters 33 tries option 252 troubleshooting accidental deletion of SnapMirror Snapshot copies 224 change of SnapMirror destination volume name 223 SnapMirror issues 223

426 | Data ONTAP 7.3 Data Protection Online Backup and Recovery Guide

SnapMirror over Fibre Channel 219

U

Unpreserving a Snapshot copy 289 using SnapMirror to replicate SnapVault data 283 using SnapVault to protect a volume SnapMirror destination 287

V

vFiler units controlling use of host filer virus scanners 410 virus protection for CIFS 395 virus scan cifs 404 file extensions 404 shares 404 shares, turn off 404 virus scanning adding extensions of files to exclude from a scan 402 adding extensions of files to scan 400 adding shares with virus scanning turned off 406 checking status of vscan options 411 controlling vfiler use of host virus scanners 410 designating secondary virus-scanning clients 398 disables scanning on read-only shares 405 displaying extensions of files to exclude from scan 402 enables scanning on read-only shares 405 enabling and disabling 397 enabling and disabling mandatory scanning 410 excluding extensions of files to scan 402 extension exclude list compared to extension include list 403 identifying clients that are scanning 407 removing extensions of files to exclude from a scan 403 removing extensions of files to scan 401 replacing extensions of files to exclude from a scan 402 replacing extensions of files to scan 400 resetting extensions of files to exclude from a scan to default 403 resetting extensions of files to scan to default list 401 resetting scanned files cache 412 resetting the virus scan request timeout 409 setting the virus scan request timeout 409

stopping a scanner session 411 viewing extensions of files to scan 402 vol options nosnap 49 vol options commands vol options nvfail off (disables protection) 383 vol options nvfail on (enables protection) 383 vol status -b command 336 volcopy dump operation (backs up to tape) 339 volcopy restore operation (restores backups from tape) 339 volume copy arguments for copying Snapshot copies 340 benefits 333 bringing volumes online (vol online) 338 checking status (vol copy status) 343 copying one volume to another (vol copy start) 341 copying volumes (vol copy start) 339 defined 333 IPv6 335 operation numbers using with vol copy abort 346 requirements for 335 restore operation (writes data to destination) 339 sample status message from filer 343 sample status message from storage system 344 stopping (vol copy abort) 346 taking volumes offline (vol offline) 338 to same or different filers 335 volume copy commands examples of arguments to 341 options rsh.enable on (enables Remote Shell) 338 options vol.copy.throttle (changes copying speed) 345 vol copy abort (halts the copy operation) 346 vol copy arguments for copying Snapshot copies 340 vol copy start (copies one volume to another) 341 vol copy status (checks status) 343 vol offline (takes volume offline) 338 vol online (brings volume online) 338 vol status -b (finding size of volume) 336 vol status (shows whether online or offline) 338 volume copy operation numbers using with vol copy abort 346 using with vol.copy.throttle 345 volume data backup 242 volume replication using SnapMirror 97 Volume Shadow Copy Services 264

volume size, verifying for snapmirror 336 volumes changing the speed of copying 345 reverting from Snapshot copies with SnapRestore 75 reverting with SnapRestore 75 vscan authentication 396 file extensions 404 vscan commands vscan extensions (shows extensions of files to scan) virus scanning viewing extensions of files to scan 400 cifs shares -add (adds CIFS shares with virus scanning turned off 406 cifs shares -change enabling or disabling scanning for read-only shares 405 cifs shares -change enabling or disabling virus scanning for a share 405 vscan extensions add (adds extensions of files to scan) 400 vscan extensions exclude (excludes extensions of files to scan) 402 vscan extensions exclude add (adds extensions of files to exclude from a scan) 402 vscan extensions exclude remove (removes extensions of files to exclude from a scan) 403 vscan extensions exclude reset (resets extensions of files to exclude from a scan to default list) 403

vscan extensions exclude set (replaces extensions of files to exclude from a scan) 402 vscan extensions remove (removes extensions of files to scan) 401 vscan extensions reset (resets extensions of files to scan to default list) 401 vscan extensions set (replaces extensions of files to scan) 400 vscan off (disables virus scanning) 397 vscan on (enables virus scanning) 397 vscan reset (resets scanned files cache) 412 vscan scanners (identifies clients that are scanning) 407 vscan scanners secondary scanners (designates backup virus scanners) 398 vscan scanners stop (stops scanner session) 411 vscan options use_host_scanners 410 vscan options mandatory_scan (sets mandatory virus scanning) 410 vscan options timeout reset (resets the virus scan request timeout) 409 vscan options timeout set (sets the virus scan request timeout) 409

W

Windows 2000 backing up with SnapVault 228