

Copyright © 1998 – 2011 EMC Corporation. All rights reserved.

Published December 2011

EMC believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

THE INFORMATION IN THIS PUBLICATION IS PROVIDED "AS IS." EMC CORPORATION MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WITH RESPECT TO THE INFORMATION IN THIS PUBLICATION, AND SPECIFICALLY DISCLAIMS IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Use, copy, and distribution of any EMC software described in this publication requires an applicable software license.

For the most up-to-date regulatory document for your product line, go to the Technical Documentation and Advisories section on EMC Powerlink.

For the most up-to-date listing of EMC product names, see EMC Corporation Trademarks on EMC.com.

All other trademarks used herein are the property of their respective owners.

# **Contents**

## Preface

Chapter 1	Overview	
	Celerra Network Server	12
	Main components	12
	Control Station	
	Access to Control Station	13
	Access to Command Line Interface	13
	Role-based access	14
	Command set conventions	15
	Synopsis	15
	Command prefixes	
	General notes	
Chapter 2	The cel and cs Commands  cel_fs  cs_standby	
Chapter 3	The fs Commands	
	fs_ckpt	28
	fs_dedupefs_dedupe	39
	fs_dhsm	
	fs_group	87
	fs_rdf	
	fs_timefinder	98

## Chapter 4 The nas Commands nas\_automountmap...... 112 nas\_ca\_certificate ...... 114 nas\_pool ...... 245 nas server ...... 309 **Chapter 5** The server Commands

server_cepp	391
server_checkup	
server_cifs	
server_cifssupport	435
server_cpu	449
server_date	451
server_dbms	456
server_devconfig	463
server_df	471
server_dns	473
server_export	
server_extlog	
server_file	
server_ftp	
server_http	508
server_ifconfig	
server_ip	
server_iscsi	
server_kerberos	
server_ldap	
server_log	
server_mount	
server_mountpoint	
server_mpfs	
server_mt	
server_name	
server_netstat	
server_nfs	
server_nis	
server_param	
server_pax	
server_ping	
server_ping6	
server_rip	
server_route	
server_security	
server_setup	
server_snmpd	
server_ssh	
server_standby	
server stats	
server_sysconfig	
server_sysstat	

	server_tftp	
	server_umount	685
	server_uptime	687
	server_user	688
	server_usermapper	
	server_version	
	server_viruschk	696
	server_vtlu	700
Appendix A	Using Celerra Command Scripts	
Appendix A	Using Celerra Command Scripts Celerra scripting guidelines	712
	Celerra scripting guidelines	712
Appendix A  Appendix B		712
	Celerra scripting guidelines	
	Celerra scripting guidelines  GNU General Public License	

## **Preface**

As part of an effort to improve and enhance the performance and capabilities of its product line, EMC periodically releases revisions of its hardware and software. Therefore, some functions described in this document may not be supported by all revisions of the software or hardware currently in use. For the most up-to-date information on product features, refer to your product release notes.

If a product does not function properly or does not function as described in this document, please contact your EMC representative.

#### **Audience**

This guide is part of the Celerra Network Server documentation set, and is intended for use by administrators and users of the EMC Celerra Network Server.

# Related documentation

Other Celerra Network Server system publications are available on the EMC Celerra Documentation on Powerlink.

# Conventions used in this guide

EMC uses the following conventions for notes, cautions, warnings, and danger notices.

 $\textbf{Note:} \ A \ note \ presents \ information \ that \ is \ important, \ but \ not \ hazard-related.$ 



#### **CAUTION**

A caution contains information essential to avoid data loss or damage to the system or equipment. The caution may apply to hardware or software.



### **WARNING**

A warning contains information essential to avoid a hazard that can cause severe personal injury, death, or substantial property damage if you ignore the warning.



#### **DANGER**

A danger notice contains information essential to avoid a hazard that will cause severe personal injury, death, or substantial property damage if you ignore the message.

## Typographical conventions

EMC uses the following type style conventions in this guide:

Bold	<ul> <li>User actions (what the user clicks, presses, or selects)</li> <li>Interface elements (button names, dialog box names)</li> <li>Names of keys, commands, programs, scripts, applications, utilities, processes, notifications, system calls, services, applications, and utilities in text</li> </ul>
italic	<ul><li>Book titles</li><li>New terms in text</li><li>Emphasis in text</li></ul>
Courier	<ul> <li>Prompts</li> <li>System output</li> <li>Filenames</li> <li>Pathnames</li> <li>URLs</li> <li>Syntax when shown in command line or other examples</li> </ul>
Courier, bold	<ul><li> User entry</li><li> Options in command-line syntax</li></ul>
Courier italic	<ul> <li>Arguments in examples of command-line syntax</li> <li>Variables in examples of screen or file output</li> <li>Variables in pathnames</li> </ul>
<>	Angle brackets for parameter values (variables) supplied by user.
[]	Square brackets for optional values.
1	Vertical bar symbol for alternate selections. The bar means or.
•••	Ellipsis for nonessential information omitted from the example.

### Where to get help

EMC support, product, and licensing information can be obtained as follows.

**Product information** — For documentation, release notes, software updates, or for information about EMC products, licensing, and service, go to the EMC Powerlink website (registration required) at:

http://Powerlink.EMC.com

**Technical support** — For technical support, go to EMC WebSupport on Powerlink. To open a case on EMC WebSupport, you must be a WebSupport customer. Information about your site configuration and the circumstances under which the problem occurred is required.

#### Your comments

Your suggestions will help us continue to improve the accuracy, organization, and overall quality of the user publications. Please send your opinion of this guide to:

celerradoc\_comments@EMC.com

Preface	

# **Overview**

This chapter provides a brief description of the commands that can be used to configure and manage the Celerra Network Server. Topics included are:

•	Celerra Network Server	12
•	Control Station	13
•	Role-based access	14
•	Command set conventions	15
	General notes	

## Celerra Network Server

The EMC® Celerra® Network Server is a dedicated file server solution that is easily integrated into existing networks by using standard network interface protocols.

## Main components

The Celerra Network Server hardware platform is the Celerra cabinet that works with a storage system to produce shared network-attached storage. The Celerra cabinet:

- Contains up to 14 Data Movers available as dedicated file servers that provide access to file systems resident on the storage system to a requesting client on the network.
- Includes a Control Station that manages all Data Mover functions both locally and remotely.

## **Control Station**

The Control Station provides utilities for managing, configuring, and monitoring of the Data Movers in the Celerra Network Server.

As the system administrator, you may type commands through the Control Station to perform the following tasks:

- Manage and Configure the database and Data Movers
- Monitor statistics of the Celerra cabinet components

# Access to Control Station

You may use either local or remote access to the Control Station.

Local

Local access to the command line interface is available directly at the Control Station console.

Remote

Remote access to the command line interface by using a secure, encrypted login application allows the use of the Celerra Network Server command set.

### Access to Command Line Interface

A description of how to gain local or remote access to the command line interface for the Celerra Network Server follows.

Local

For local access to the command line interface, at the prompt, log in with your administrative username and password.

#### Remote

For remote access to the command line interface:

- 1. Use a secure, encrypted, remote login application. Type the IP address of the Control Station.
- 2. Log in with your administrative username and password.

## **Role-based access**

The administrative user account you use to access the command line interface is associated with specific privileges, also referred to as roles. A role defines the privileges (operations) a user can perform on a particular Celerra object. The ability to select a predefined role or define a custom role that gives a user certain privileges is supported for users who access Celerra through the CLI, EMC Unisphere<sup>TM</sup>, and the XML API.

The Security Configuration Guide for Celerra provides detailed information about how role-based access is used to determine the commands a particular user can execute. You create and manage user accounts and roles in Unisphere by using Settings > User Management.

## Command set conventions

This manual uses commonly known command set conventions for the Celerra Network Server man pages. Each man page presents the command name at the top of the man page followed by a brief overview of what the command does. The synopsis contains the actual command usage. The description contains a more detailed breakdown of the features of the command, and the options describe what each switch or option does specifically.

The 'See Also' section refers to the technical modules that support the feature, in addition to any other commands that interact with the command. The examples are at the end of the command.

The naming convention for the Data Mover variable in the command line interface is <movername> (default = server\_2 to server\_15).

The commands are prefixed, then appear in alphabetical order.

## **Synopsis**

The synopsis is usage of each command. The synopsis appears in courier typeface, with variables such as movername, filename, and device name enclosed by angle brackets, with the command name appearing in **bold**. The switches and other options also appear in **bold** and, in most cases, are prefixed by a minus sign:

server\_umount {<movername>|ALL}[-perm|-temp]{-all|<fs\_name>|<pathname>}

Required entries

A switch or variable enclosed with curly brackets, or not enclosed at all, indicates a required entry:

{<movername> | ALL}

Optional entries

A switch or variable enclosed with square brackets indicates an

optional entry:

[-perm | -temp]

Formatting

The variable name enclosed by angle brackets indicates the name of a

specified object:

{<movername> | ALL}

**Options** 

An option is prefixed with a minus (-) sign: -perm

If the option is spelled out, for example, -perm, in the command

syntax, you may use just the first letter: -p

Options and names are case-sensitive. If an uppercase letter is specified in the syntax, a lowercase letter is not accepted.

{-all | <fs\_name> | <pathname>}

## **Command prefixes**

Commands are prefixed depending on what they are administering. For example, commands prefixed with:

- **cel**\_ execute to the remotely linked Celerra system.
- **cs**\_ execute to the Control Station.
- **fs**\_ execute to the specified file system.
- nas\_ execute directly to the Control Station database.
- server\_require a movername entry and execute directly to a Data Mover.

# **General notes**

If a command is interrupted by using Ctrl-C, then the following messages or traces on the console are expected:

- nas\_cmd: system execution failed.
- nas\_cmd: PANIC: caught signal #11 (Segmentation fault) --Giving up.

Overview	
	-

# The cel and cs Commands

This chapter lists the Celerra Network Server Command Set provided to manage, configure, and monitor the Data Movers. The commands are prefixed with **cel** or **cs**, and appear alphabetically. The prefix assigned to a command indicates what the command is administering. The **cel** prefix administers to the remote Celerra Network Server, and **cs** administers to the local Control Station. The command line syntax (Synopsis), a description of the options, and an example of usage are provided for each command. Topics included are:

- ♦ cel\_fs
- cs\_standby

## cel\_fs

Manages filesystems residing on a remotely linked Celerra Network Server.

#### **SYNOPSIS**

```
cel_fs <cel_name>
    -list
| -info [-size] {-all | {<fs_name> | id=<fs_id>},...}
| -Extract {-all | {<fs_name> | id=<fs_id>},...}
| -Import {{<fs_name>,...} | -file <filename>}
```

#### **DESCRIPTION**

From the remotely linked Celerra Network Server, cel\_fs imports and enables the definition of a remote filesystem known as the R2 onto the local system known as the R1. The <cel\_name> is the default hostname of the remote Control Station active during installation, or when performing a nas\_rdf -init to set up the Remote Data Facility (RDF) environment. To find the <cel\_name>, go to nas\_cel and execute the -list option. The local Celerra Network Server, by default, has an ID of zero.

**cel\_fs** displays a listing of the filesystems residing on the remote Celerra Network Server including their attributes.

This command is used in FarCopy and NearCopy environments.

#### **OPTIONS**

#### -list

Lists all of the filesystems residing on the remote Celerra Network Server as specified by its <cel\_name>.

**Note:** The ID of the object is an integer and is assigned automatically. The name of a filesystem may be truncated if it is too long for the display. To display the full name, use the **-info** option with a filesystem ID.

```
-info [-size] {-all|{<fs_name>|id=<fs_id>},...}
Displays the attributes and size of the filesystem residing on the
<cel_name>. The <fs_name> can be a filesystem name or a group
name and can be up to 16 characters.
```

```
-Extract {-all|{<fs_name>|id=<fs_id>},...} Extracts all filesystem attributes that can be redirected for import into the Celerra Network Server. The second <fs_name> specifies a filesystem where the attributes can be imported.
```

**-Import** {{<fs\_name>,...}|**-file** <filename>} Imports filesystem attributes obtained from the extraction of a filesystem definition either from the specified <fs\_name> or <filename>.

**SEE ALSO** *Using SRDF/S with Celerra for Disaster Recovery*, nas\_cel, nas\_fs, nas\_rdf, and fs\_group.

**EXAMPLE #1** To display a listing of the filesystems residing on the remote Celerra Network Server **cs30**, type:

				\$ cel_fs	cs30 -list	
id	inuse	type	acl	volume	name	server
1	n	1	0	66	root_fs_1	
2	У	1	0	68	root_fs_2	1
3	У	1	0	70	root_fs_3	2
4	У	1	0	72	root_fs_4	3
5	У	1	0	74	root_fs_5	4
6	n	1	0	76	root_fs_6	
7	n	1	0	78	root_fs_7	
8	n	1	0	80	root_fs_8	
9	n	1	0	82	root_fs_9	
10	n	1	0	84	root_fs_10	
11	n	1	0	86	root_fs_11	
12	n	1	0	88	root_fs_12	
13	n	1	0	90	root_fs_13	
14	n	1	0	92	root_fs_14	
15	n	1	0	94	root_fs_15	
16	У	1	0	96	root_fs_common	4,3,2,1
17	n	5	0	145	root_fs_ufslog	
18	У	1	0	156	ufs1	1

#### Where:

	*******
Value	Definition
id	ID of a filesystem (assigned automatically).
inuse	Whether a filesystem has been registered into the mount table.
type	What type of filesystem. nas_fs provides information.
acl	Access control value for a filesystem. nas_fs provides information.
volume	Name of the volume on which a filesystem resides.
name	Name assigned to a filesystem.
server	Server associated with a filesystem.

**EXAMPLE #2** To display the attributes and the size for **ufs1** on **cs30**, type:

\$ cel\_fs cs30 -info -size ufs1

 $\begin{array}{ll} \text{id} & = 18 \\ \text{name} & = \text{ufs1} \\ \text{acl} & = 0 \end{array}$ 

```
in_use = True
type = uxfs
volume = mtv1
pool
rw_servers= server_2
ro servers=
rw_vdms
ro_vdms =
size = total = 22660 avail = 22659 used = 0 (0%) (sizes in MB)
volume: total = 23012 (sizes in MB)
stor_devs =
002804000190 - 0034,002804000190 - 0035,002804000190 - 0036,002804000190 - 0037
disks = d3, d4, d5, d6
 disk=d3 stor_dev=002804000190-0034 addr=c0t318-15-0
                                                                server=server_2
 disk=d4 stor_dev=002804000190-0035 addr=c0t319-15-0 server=server_2
 disk=d5 stor_dev=002804000190-0036 addr=c0t3110-15-0 server=server_2 disk=d6 stor_dev=002804000190-0037 addr=c0t3111-15-0 server=server_2
```

#### Where:

Value Definition			
Definition			
ID of a filesystem (assigned automatically).			
Name of the imported filesystem.			
Access control value for a filesystem. nas_fs provides information.			
Whether a filesystem has been registered into the mount table.			
What type of filesystem. nas_fs provides information.			
Name of the volume on which a filesystem resides.			
Storage pool for the filesystem.			
Servers with read-write access.			
Servers with read-only access.			
VDM servers with read-write access to a filesystem.			
VDM servers with read-only access to a filesystem.			
Total size, available size, and amount used in MB.			
Volume size that is used.			
ID of the storage system device and the serial number.			
The disks on which a filesystem resides dd indicates a remote disk.			

## **EXAMPLE #3** To extract a filesystem definition for **ufs1** from **cs30**, type:

```
$ cel_fs cs30 -Extract ufs1
0:bofs:18
18:ufs1:0:y:1:164:1::::0:::
164:mtv1:0:y:3:2:18:156:
156:stv1:0:y:2:1:164:64:3,4,5,6:
6:d6:0:y:4:1:156:6:
6:d6:0:y:5753:002804000190:1,2,3,4:6:0037:5:
5:d5:0:y:4:1:156:5:
```

5:d5:0:y:5753:002804000190:1,2,3,4:5:0036:5:

```
4:d4:0:y:4:1:156:4:

4:d4:0:y:5753:002804000190:1,2,3,4:4:0035:5:

3:d3:0:y:4:1:156:3:

3:d3:0:y:5753:002804000190:1,2,3,4:3:0034:5:

0:eof:18
```

**Where:** The output is the extracted filesystem definition.

**EXAMPLE #4** To extract a filesystem definition for **ufs1** from **cs30** into the extract-file, type:

\$ cel\_fs cs30 -Extract ufs1 > extract-file

#### **EXAMPLE #5**

To import a remote filesystem definition for **ufs1** from **cs30** onto the current Celerra Network Server, type:

\$ cel\_fs cs30 -Import ufs2

```
id
          = 18
          = ufs2
name
          = 0
acl
          = False
in_use
          = uxfs
type
         = mtv2
volume
pool
rw_servers=
ro_servers=
rw_vdms
ro vdms
backup_of = 1901 Fri Apr 23 15:25:59 EDT 2004
remainder = 0 MB (0%)
stor_devs =
002804000218-0038,002804000218-0039,002804000218-003A,002804000218-003B
          = dd7, dd8, dd9, dd10
disks
```

EXAMPLE #2 provides an output description of all values except those which are unique to the **-Import** option.

#### Where:

Value	Definition
backup_of	System ID of the remote filesystem.
remainder	Number of MB copied from the filesystem on the remote Celerra Network Server.

#### **EXAMPLE #6**

To import a remote filesystem definition from the file, extract-file, from **cs30** into the current Celerra Network Server, type:

\$ cel\_fs cs30 -Import -file extract-file

```
id = 19
name = ufs1
ac1 = 0
in_use = False
```

#### The cel and cs Commands

```
type = uxfs
volume = mmtv1
pool =
rw_servers=
ro_servers=
rw_vdms =
ro_vdms =
backup_of = 18@1 Fri Apr 23 15:26:25 EDT 2004
remainder = 0 MB (0%)
stor_devs =
002804000218-0034,002804000218-0035,002804000218-0036,002804000218-0037
disks = dd3,dd4,dd5,dd6
```

**EXAMPLE #2** and **EXAMPLE #5** provide a description of command output.

## cs\_standby

Initiates a takeover and failover of a Control Station on a Celerra Network Server with dual Control Stations.

#### **SYNOPSIS**

cs\_standby
{-takeover|-failover}

#### **DESCRIPTION**

The **cs\_standby** command initiates a Control Station takeover and failover. When a Control Station is activated, the name of the primary Control Station is displayed.

**su** to **root** and execute this command from the /nas/sbin or /nasmcd/sbin directory.

**Note:** EMC SRDF<sup>®</sup> is not supported on the secondary Control Station.

### **OPTIONS**

#### -takeover

Executed from the standby Control Station, initiates a reboot of the primary Control Station, then changes the state of the standby to that of the primary. The original primary Control Station now becomes the standby Control Station. The **-takeover** option can only be executed from the standby Control Station.



#### **CAUTION**

When executing a takeover or failover, Data Movers performing functions such as RDF, EMC TimeFinder®/FS, filesystem extends, or quotas may be interrupted.



#### **CAUTION**

If a primary Control Station fails over to a standby Control Station, for remote replication, service continues to run but replication management capabilities are no longer available.

**Note:** After executing a takeover or failover, a few minutes may be needed to stop Linux and other services active on the Control Station.

#### -failover

Executed from the primary Control Station, initiates a reboot of the primary Control Station, then activates the standby to take over the role of the primary Control Station. A **-failover** can only be executed from the primary Control Station.

To display the primary Control Station, type:

#### \$ nas/sbin/getreason

#### **EXAMPLE #1**

To change the state of the standby Control Station to primary, **cd** to the /nasmcd/sbin directory of the standby Control Station, then type:

#### #./cs\_standby -takeover

Taking over as Primary Control Station.....done

If the **takeover** command is executed on the primary Control Station, the following error message appears:

The -takeover option is only valid on a standby Control Station

#### **EXAMPLE #2**

To initiate a failover from the primary Control Station to the standby Control Station, **cd** to the /nas/sbin directory of the primary Control Station, then type:

### #./cs\_standby -failover

The system will reboot, do you wish to continue [yes or no]: y Failing over from Primary Control Station

## The fs Commands

This chapter lists the Celerra Network Server Command Set provided for managing, configuring, and monitoring the specified filesystem. The commands are prefixed with **fs** and appear alphabetically. The command line syntax (Synopsis), a description of the options, and examples of usage are provided for each command. Topics included are:

- ♦ fs\_ckpt
- fs\_dedupe
- fs\_dhsm
- fs\_group
- ♦ fs\_rdf
- fs\_timefinder

## fs\_ckpt

Manages checkpoints using the EMC SnapSure™ functionality.

#### **SYNOPSIS**

```
fs_ckpt {<fs_name>|id=<fs_id>}
-list [-all]
| [-name <name>] -Create [-readonly {y|n}][<volume_name>][-option <options>]
| [-name <name>] -Create [-readonly {y|n}][size=<integer>[T|G|M|%]]
| [pool=<pool>][storage=<system_name>][-option <options>]
| -refresh [-option <options>]
| [-name <name>] -Restore [-Force][-option <options>]
| -modify [%full=<value>][maxsavsize=<integer>[T|G|M]]
```

#### DESCRIPTION

The **fs\_ckpt** command creates a checkpoint of a Production filesystem (PFS), lists associated checkpoints, refreshes a checkpoint to the current time, and restores a PFS back to a specific point in time using a checkpoint. Checkpoints are deleted using nas\_fs.

# What is a checkpoint filesystem?

A PFS is made up of blocks. When a block within a PFS is modified, a copy containing the original contents of that block is saved to a metavolume called the SavVol. Subsequent changes made to the same block in the PFS are not copied into the SavVol. The original blocks from the PFS (in the SavVol) and the unchanged PFS blocks (that remain in the PFS) are read according to a bitmap and blockmap data tracking structure. These blocks combine to provide a complete point-in-time filesystem image which is called a checkpoint.

#### **OPTIONS**

#### -list

Displays all of the associated checkpoints for the specified filesystem. The **-all** option displays system-generated Replication checkpoints in addition to checkpoints created by the user.

```
[-name <name>] -Create
```

Creates, mounts, and optionally assigns a name to the checkpoint of the PFS. The checkpoint must be unmounted prior to unmounting the PFS. Names assigned to a checkpoint cannot be all numeric. If a name is not chosen, one is assigned by default.

```
[-readonly {y|n}]
```

Specifies whether a checkpoint is read-only or not. y (default) sets the checkpoint as read-only; n sets the checkpoint as writeable.

[<volume\_name>]

Specifies an unused metavolume for the checkpoint.

**Note:** A volume can be specified for only the first checkpoint of a PFS since all of the subsequent checkpoints share the same SavVol. The minimum size required for a SavVol is 64 MB. The volume size is 10 GB unless the PFS is less than 10 GB, then the volume is the same size as the filesystem.

[-option <options>]

Specifies the following comma-separated options:

%full=<value>

Specifies a value as the percentage threshold permitted for the SavVol. When that value is reached, a warning is sent to the server\_log and the syslog files. The Control Station acknowledges the event and automatically extends the checkpoint. The SavVol is automatically extended by 10 GB if its default **%full** value is reached. If the **%full** value is set to zero, the option is disabled.

maxsavsize=<integer>[T|G|M]

Limits the final size to which the SavVol can be automatically extended when the high watermark value specified in **%full** has been reached. Automatic extension of the SavVol stops when the size of the SavVol reaches the value specified in **maxsavsize**. The range for **maxsavsize** is 64 MB to 16 TB.

automount=no

Stops the checkpoint from being automatically mounted.

[-name <name>] -Create

Creates, mounts, and optionally assigns a name to the checkpoint of the PFS. The checkpoint must be unmounted prior to unmounting the PFS. Names assigned to a checkpoint cannot be all numeric. If a name is not chosen, one is assigned by default.

[-readonly {y|n}]

Specifies whether a checkpoint is read-only or not. y (default) sets the checkpoint as read-only; n sets the checkpoint as writeable.

[size=<integer>[T|G|M|%]]

Specifies a size for the checkpoint filesystem. Type an integer between 1 and 1024, specify **T** for terabytes, **G** for gigabytes (default) or **M** for megabytes, or type an integer representing the percentage of a filesystem's size, followed by the percent sign.

#### [pool=<pool>]

Specifies the storage pool to be used for the checkpoint. Storage pools can either be user-defined or system-defined. The **nas\_pool -list** command displays a listing of available pool types.

#### [storage=<system\_name>]

Specifies the attached storage system for the checkpoint SavVol to reside.

#### [-option <options>]

Specifies the following comma-separated options:

#### %full=<value>

Specifies a value as the percentage threshold permitted for the SavVol. When that value is reached, a warning is sent to the server\_log and the syslog files. The Control Station acknowledges the event and automatically extends the checkpoint. The SavVol is automatically extended by 10 GB if its default %full value is reached. If the %full value is set to zero, the option is disabled. The default for <value> is 90 and it can be within the range of 10 to 99.

#### automount=no

Stops the checkpoint from being automatically mounted.

#### -refresh

Initiates an immediate update of a checkpoint, thereby allowing the SavVol space to be reused. Refreshing a checkpoint does not add to the number of checkpoints of the PFS.

```
[-option <options>] %full=<value>
```

Specifies a value as the percentage threshold permitted for the metavolume. When that value is reached, a warning is sent to the server\_log and the syslog files. The Control Station acknowledges the event and automatically extends the checkpoint. The SavVol is automatically extended by 10 GB if its default **%full** value is reached. If the **%full** value is set to zero, the option is disabled. The default for <value> is 90.

#### -modify

Modifies one or all of the following options:

**Note:** The **-modify** action works only on the PFS and not on the checkpoint.

```
[%full=<value>]
```

Modifies the value of the percentage threshold permitted for the metavolume.

#### [maxsavsize=<integer>[T|G|M]]

Modifies the final size to which the SavVol can be automatically extended, when the size specified in **%full** is reached.

#### [-name <name>] -Restore

Restores the PFS from the specified checkpoint and optionally assigns a name to the automatically created checkpoint. If a name is not chosen, one is assigned by default.

**Note:** As part of the restore, a new checkpoint is automatically created to capture the latest point-in-time image of the PFS. This is for protection in the event that the restored image is discarded.

#### [-Force]

The **-Force** option must be used when restoring a PFS with enabled.



#### CAUTION

Forcing a restore of a production filesystem with enabled from a checkpoint will delete or overwrite files that were written after this checkpoint was created or refreshed.

[-option <options>]

Specifies the following comma-separated options:

#### %full=<value>

Specifies a value as the percentage threshold permitted for the SavVol. When that value is reached, a warning is sent to the server\_log and the syslog files. The Control Station acknowledges the event and automatically extends the checkpoint. The SavVol is automatically extended by 10 GB if its default %full value is reached. If the %full value is set to zero, the option is disabled. The <value> can be an integer between 10 and 75 (default).

#### automount=no

Stops the checkpoint from being automatically mounted.

#### **SEE ALSO**

*Using SnapSure on Celerra*, nas\_fs, and nas\_pool.

# STORAGE SYSTEM OUTPUT

The number associated with the storage device is dependent on the attached storage system. EMC CLARiiON® storage systems display a prefix of APM before a set of integers, for example,

APM00033900124-0019. For example, EMC Symmetrix® storage systems display as 002804000190-003C. The outputs displayed in the examples use a CLARiiON storage system.

### **EXAMPLE #1** To display the checkpoint for the filesystem fs4, type:

#### \$ fs\_ckpt fs4 -list

id ckpt\_name creation\_time inuse fullmark total\_savvol\_used ckpt\_usage\_on\_savvol 1406 fs4\_ckpt1 05/26/2008-16:22:19-EDT y 90% 51% 0% id wckpt\_name inuse fullmark total\_savvol\_used base ckpt\_usage\_on\_savvol

# **EXAMPLE #2** To display all checkpoints including internal checkpoints for the filesystem fs4, type:

#### \$ fs\_ckpt fs4 -list -all

id ckpt\_name creation\_time inuse fullmark total\_savvol\_used ckpt\_usage\_on\_savvol
1401 root\_rep\_ckpt\_1398\_21625\_1 05/26/2008-16:11:10-EDT y 90% 51% 0%
1402 root\_rep\_ckpt\_1398\_21625\_2 05/26/2008-16:11:22-EDT y 90% 51% 0%
1406 fs4\_ckpt1 05/26/2008-16:22:19-EDT y 90% 51% 0%
id wckpt\_name inuse fullmark total\_savvol\_used base ckpt\_usage\_on\_savvol

## **EXAMPLE #3** To create a checkpoint of **ufs1**, on the volume, **ssmtv1**, type:

#### \$ fs\_ckpt ufs1 -Create ssmtv1

```
operation in progress (not interruptible)...id
                                                     = 22
name
       = ufs1
acl
in_use = True
type
        = uxfs
worm
         = off
         = mtv1
volume
loog
rw_servers= server_2
ro_servers=
rw vdms =
ro_vdms
ckpts = ufs1_ckpt1
stor_devs = APM00043807043-0010, APM00043807043-0014
disks = d7.d9
disk=d7 stor_dev=APM00043807043-0010 addr=c0t110 disk=d7 stor_dev=APM00043807043-0010 addr=c16t110
                                                              server=server 2
            stor_dev=APM00043807043-0010 addr=c16t110
                                                              server=server_2
 disk=d9 stor_dev=APM00043807043-0014 addr=c0t114
                                                             server=server_2
 disk=d9
            stor_dev=APM00043807043-0014 addr=c16t114
                                                              server=server_2
id
         = 24
name
        = ufs1_ckpt1
acl
        = 0
in_use = True
        = ckpt
type
worm
        = off
volume = vol32
pool
member_of =
rw_servers=
ro_servers= server_2
rw_vdms =
```

#### Where:

Value	Definition
id	Automatically assigned ID of a filesystem or the checkpoint.
name	Name assigned to the filesystem or the checkpoint.
acl	Access control value for a filesystem. nas_acl provides information.
in_use	If a filesystem is registered into the mount table of a Data Mover.
type	Type of filesystemlist provides a description of the types.
worm	Whether the feature is enabled.
volume	Volume on which a filesystem resides.
pool	Storage pool for the filesystem.
member_of	Group to which the filesystem belongs.
rw_servers	Servers with read-write access to a filesystem.
ro_servers	Servers with read-only access to a filesystem.
rw_vdms	VDM servers with read-write access to a filesystem.
ro_vdms	VDM servers with read-only access to a filesystem.
ckpts	Associated checkpoints for the filesystem.
checkpt_of	Name of the PFS related to the existing checkpoints.
used	Percentage of SavVol space used by the checkpoints of the PFS.
full(mark)	SavVol usage point which, when reached, sends a warning message to the system log, and auto-extends the SavVol as system space permits.
stor_devs	Storage system devices associated with a filesystem.
disks	Disks on which the metavolume resides.

#### **EXAMPLE #4**

To create a checkpoint of **ufs1** named **ufs1\_ckpt2** with a size of **2 GB** by using the **clar\_r5\_performance** pool, with the specified storage system, with the **%full** set to **95**, type:

\$ fs\_ckpt ufs1 -name ufs1\_ckpt2 -Create size=2G
pool=clar\_r5\_performance storage=APM00043807043 -option
%ful1=95

```
operation in progress (not interruptible)...id = 27
name = ufs1
acl = 0
in_use = True
type = uxfs
```

```
worm = off
volume = mtv1
pool
rw_servers= server_2
ro_servers=
rw vdms =
ro_vdms =
ckpts = ufs1_ckpt1,ufs1_ckpt2
stor_devs = APM00043807043-0010, APM00043807043-0014
disks = d7, d9
disk=d7 stor_dev=APM00043807043-0010 addr=c0t110 disk=d7 stor_dev=APM00043807043-0010 addr=c16t110
                                                           server=server_2
                                                           server=server 2
disk=d9 stor_dev=APM00043807043-0014 addr=c0t114
                                                           server=server_2
disk=d9 stor_dev=APM00043807043-0014 addr=c16t114
                                                           server=server 2
iд
        = 30
        = ufs1_ckpt2
name
        = 0
acl
in_use = True
type = ckpt
worm = off
volume = vp145
pool
member_of =
rw_servers=
ro_servers= server_2
rw_vdms
ro vdms
checkpt_of= ufs1 Wed Nov 10 14:00:20 EST 2004
used = 0%
full(mark) = 95%
stor_devs = APM00043807043-0011, APM00043807043-0017
disks = d12, d15
 disk=d12 stor_dev=APM00043807043-0011 addr=c16t111
                                                           server=server_2
 disk=d12 stor_dev=APM00043807043-0011 addr=c0t111
                                                            server=server_2
 disk=d15 stor_dev=APM00043807043-0017 addr=c16t117
                                                           server=server 2
 disk=d15 stor_dev=APM00043807043-0017 addr=c0t117
                                                            server=server_2
```

EXAMPLE #3 provides a description of command output.

# **EXAMPLE #5** To create a writeable checkpoint of baseline checkpoint **ufs1\_ckpt1**, type:

```
$ fs_ckpt ufs1_ckpt1 -Create -readonly n
operation in progress (not interruptible)...id = 45
name = ufs1_ckpt1
acl = 0
in_use = False
type = ckpt
worm = off
volume = vp145
pool = clar_r5_performance
```

```
member_of =
rw_servers=
ro servers=
rw_vdms
ro_vdms
checkpt_of= ufs1 Tue Nov 6 14:56:43 EST 2007
          = ufs1_ckpt1_writeable1
ckpts
used
          = 38%
full(mark) = 90%
stor_devs =
APM00042000814-0029, APM00042000814-0024, APM00042000814-0021, APM000420
00814-001C
disks
          = d34, d17, d30, d13
id
          = 46
name
          = ufs1_ckpt1_writeable1
acl
          = 0
in use
          = True
type
          = wckpt
worm
          = off
          = vp145
volume
pool
          = clar_r5_performance
member_of =
rw_servers= server_2
ro servers=
rw_vdms
ro_vdms
checkpt_of= ufs1
baseline_ckpt = ufs1_ckpt1 Tue Nov 6 14:56:43 EST 2007
          = 38%
used
full(mark) = 90%
stor_devs =
APM00042000814-0029, APM00042000814-0024, APM00042000814-0021, APM000420
00814-001C
disks
          = d34, d17, d30, d13
 disk=d34
            stor dev=APM00042000814-0029 addr=c16t219
                                                               server=server 2
 disk=d34
            stor_dev=APM00042000814-0029 addr=c32t219
                                                               server=server_2
            stor_dev=APM00042000814-0029 addr=c0t219
 disk=d34
                                                               server=server_2
 disk=d34
            stor dev=APM00042000814-0029 addr=c48t219
                                                               server=server 2
 disk=d17
            stor_dev=APM00042000814-0024 addr=c0t214
                                                               server=server_2
 disk=d17
            stor_dev=APM00042000814-0024 addr=c48t214
                                                               server=server_2
 disk=d17
            stor_dev=APM00042000814-0024 addr=c16t214
                                                               server=server_2
 disk=d17
            stor_dev=APM00042000814-0024 addr=c32t214
                                                               server=server_2
 disk=d30
            stor_dev=APM00042000814-0021 addr=c16t211
                                                               server=server_2
            stor dev=APM00042000814-0021 addr=c32t211
 disk=d30
                                                               server=server 2
 disk=d30
            stor_dev=APM00042000814-0021 addr=c0t211
                                                               server=server_2
 disk=d30
            stor_dev=APM00042000814-0021 addr=c48t211
                                                               server=server_2
 disk=d13
            stor_dev=APM00042000814-001C addr=c0t1112
                                                               server=server_2
 disk=d13
            stor_dev=APM00042000814-001C addr=c48t1112
                                                               server=server_2
 disk=d13
            stor_dev=APM00042000814-001C addr=c16t1112
                                                               server=server_2
 disk=d13
            stor_dev=APM00042000814-001C addr=c32t1112
                                                               server=server_2
```

#### Where:

Value	Definition
baseline_ckpt	Name of the read-only checkpoint from which the writeable checkpoint is created.

### **EXAMPLE #3** provides a description of command output.

### **EXAMPLE #6** To list checkpoints for **ufs1**, type:

operation in progress (not interruptible)...id

#### \$ fs\_ckpt ufs1 -list

id	ckpt_name	creation_time	inuse	full(mark)	used
29	ufs1_ckpt1	11/04/2004-14:54:06-EST	n	95%	0%
30	ufs1_ckpt2	11/10/2004-14:00:20-EST	У	95%	0%

#### Where:

Value	Definition	
id	Automatically assigned ID of a filesystem or checkpoint.	
ckpt_name	Name assigned to the checkpoint.	
creation_time	Date and time the checkpoint was created.	
inuse	If a checkpoint is registered into the mount table of a Data Mover.	
full(mark)	SavVol-usage point which, when reached, sends a warning message to the system log, and auto-extends the SavVol as system space permits.	
used	Percentage of SavVol space used by checkpoints of the PFS.	

## **EXAMPLE #7** To refresh **ufs1\_ckpt2** by using the %full at 85, type:

```
$ fs_ckpt ufs1_ckpt2 -refresh -option %full=85
```

```
name
      = ufs1_ckpt2
acl
        = 0
in_use = True
type = ckpt
worm = off
volume = vp145
pool
member_of =
rw_servers=
ro_servers= server_2
rw vdms
ro_vdms
checkpt_of= ufs1 Wed Nov 10 14:02:59 EST 2004
used
       = 0%
full(mark) = 85%
stor_devs = APM00043807043-0011, APM00043807043-0017
disks = d12, d15
 disk=d12 stor_dev=APM00043807043-0011 addr=c16t111
                                                         server=server_2
 disk=d12 stor_dev=APM00043807043-0011 addr=c0t111
                                                         server=server_2
disk=d15 stor_dev=APM00043807043-0017 addr=c16t117
                                                        server=server 2
 disk=d15 stor_dev=APM00043807043-0017 addr=c0t117
                                                        server=server_2
```

# EXAMPLE #3 provides a description of command output.

# **EXAMPLE #8**

Using **root** command, to restore **ufs1\_ckpt2** and capture the latest point-in-time image of the PFS on **ufs1\_ckpt3**, type:

# \$ /nas/sbin/rootfs\_ckpt ufs1\_ckpt2 -name ufs1\_ckpt3 -Restore

```
operation in progress (not interruptible)...id
name = ufs1_ckpt2
acl
         = 0
in_use = True
        = ckpt
type
        = off
worm
volume = vol45
pool
member_of =
rw_servers=
ro_servers= server_2
rw vdms
ro_vdms
checkpt_of= ufs1 Wed Nov 10 14:02:59 EST 2004
        = 0%
full(mark) = 90%
stor_devs = APM00043807043-0011, APM00043807043-0017
disks = d12, d15
disk=d12 stor_dev=APM00043807043-0011 addr=c16t111
                                                          server=server_2
 disk=d12 stor_dev=APM00043807043-0011 addr=c0t111
                                                          server=server_2
disk=d15 stor_dev=APM00043807043-0017 addr=c16t117
                                                          server=server 2
 disk=d15
           stor_dev=APM00043807043-0017 addr=c0t117
                                                          server=server_2
```

# EXAMPLE #3 provides a description of command output.

### **EXAMPLE #9**

To modify the **%full** value of the SavVol associated with the filesystem **ufs1** and set it to **95**, type:

# \$ fs\_ckpt ufs1 -modify %full=95

```
operation in progress (not interruptible)...id
name = ufs1
         = 0
acl
in_use
        = True
        = uxfs
type
        = off
worm
volume
         = vp145
rw_servers= server_2
ro servers=
rw_vdms =
ro_vdms =
auto_ext = no, virtual_provision=no
ckpts = wipckpt
stor_devs = APM00062400708-0014, APM00062400708-0016
disks = d26, d27
```

```
        disk=d26
        stor_dev=APM00062400708-0014
        addr=c0t114
        server=server_2

        disk=d26
        stor_dev=APM00062400708-0014
        addr=c16t114
        server=server_2

        disk=d27
        stor_dev=APM00062400708-0016
        addr=c0t116
        server=server_2

        disk=d27
        stor_dev=APM00062400708-0016
        addr=c16t116
        server=server_2
```

# **EXAMPLE #10**

To modify the **maxsavsize** value of the SavVol associated with the filesystem **ufs1** and set it to **65 GB**, type:

### \$ fs\_ckpt ufs1 -modify maxsavsize=65G

```
operation in progress (not interruptible)...id
name = ufs1
         = 0
acl
in_use = True
type
        = uxfs
worm = off
volume = vp145
pool
rw_servers= server_2
ro_servers=
rw_vdms
ro_vdms =
auto_ext = no, virtual_provision=no
ckpts = wipckpt
stor_devs = APM00062400708-0014, APM00062400708-0016
disks = d26.d27
 disk=d26 stor_dev=APM00062400708-0014 addr=c0t114 server=server_2
 disk=d26 stor_dev=APM00062400708-0014 addr=c16t114 server=server_2
 disk=d27 stor_dev=APM00062400708-0016 addr=c0t116 server=server_2
 disk=d27 stor_dev=APM00062400708-0016 addr=c16t116 server=server_2
```

### **DIAGNOSTICS**

The **fs\_ckpt** command returns one of the following return codes:

- ◆ 0 Command completed successfully
- ◆ 1 Usage error
- ◆ 2 Invalid object error
- ◆ 3 Unable to acquire lock
- ◆ 4 Permission error
- ◆ 5 Communication error
- ◆ 6 Transaction error
- ◆ 7 Dart error
- ♦ 8 Backend

# fs\_dedupe

Manages filesystem deduplication state.

### **SYNOPSIS**

```
fs_dedupe {
  -list
  -info {-all|<fs_name>|id=<fs_id>}
  -modify {<fs_name> | id=<fs_id>} [-state
    {off|suspended|on}][-minimum_scan_interval <days>][-minimum_size <KB>]
    [-maximum_size <MB>][-access_time <days>][-modification_time <days>]
    [-case_sensitive {yes no}][-pathname_exclude_list <path_list>]
    [-file_ext_exclude_list <ext_list>][-duplicate_detection_method
    {sha1|byte|off}][-savvol_threshold <percent>][-backup_data_threshold
    <percent>] [-cifs_compression_enabled {yes | no}]
  -clear {<fs_name>|id=<fs_id>}[-minimum_scan_interval][-minimum_size]
    [-maximum_size][-access_time][-modification_time][-case_sensitive]
    [-pathname_exclude_list] [-file_ext_exclude_list]
    [-duplicate_detection_method] [-savvol_threshold]
    [-backup_data_threshold][-cifs_compression_enabled]
  -default
     -info {<movername> | -all}
   -set {<movername>|-all}[-minimum_scan_interval <days>]
       [-minimum_size<KB>][-maximum_size <MB>][-access_time
      <days>] [-modification_time <days>] [-case_sensitive
       {yes | no}][-file_ext_exclude_list <ext_list>]
       [-duplicate_detection_method {sha1|byte|off}][-savvol_threshold
      <percent>] [-cpu_usage_low_watermark <percent> ]
       [-cpu_usage_high_watermark <percent>][-backup_data_threshold
      <percent>] [-cifs_compression_enabled {yes | no}]
     -clear {<movername> | -all}
       [-minimum scan interval][-minimum size][-maximum size][-access time]
       [-modification_time][-case_sensitive][-file_ext_exclude_list]
       [-duplicate_detection_method] [-savvol_threshold]
       [-cpu usage low watermark] [-cpu usage high watermark]
       [-backup_data_threshold <percent>][-cifs_compression_enabled]
 }
```

### **DESCRIPTION**

**fs\_dedupe** allows the Celerra administrator to enable, suspend, and undo all deduplication processing on a filesystem or a Data Mover. The Data Mover settings are the global settings that can be used for both the Data Mover and the filesystem. If a user sets a value for a specific filesystem, then that value overrides the Data Mover global value. If a user clears a value set for a specific filesystem, then that value is reset to the Data Mover global value.

# **OPTIONS**

#### -list

Lists all deduplication-enabled filesystems on the Celerra.

# -info {-all|<fs\_name>|id=<fs\_id>}

Lists the existing filesystems and provides information on the state of deduplication processing.

### -a11

Lists all filesystems and provides detailed information on the state of deduplication processing.

<fs name>

Lists the filesystem information for the specified filesystem name.

id=<fs\_id>

Lists the filesystem information for the specified identifier.

The filesystem state and status information displayed includes:

- If the **state** is **off** and the status is **not reduplicating**:
  - ID
  - Name
  - Deduplication state
- If the **state** is **off** and the status is **reduplicating**:
  - ID
  - Name
  - Deduplication state
  - Progress information (the percentage of files scanned)
- ◆ If the **state** of the filesystem is **on** or **suspended**, and the status is **Idle** or **Scanning**:
  - ID
  - Name
  - Reduplication state
  - Status
  - The percentage of files scanned
  - Last system scan time
  - Number of files scanned
  - Number of files deduplicated
  - The percentage of files deduplicated
  - Filesystem capacity
  - Logical data size

- Percentage of filesystem usage
- Space saved (in MB and percent)

# -modify {<fs\_name>|id=<fs\_id>} [-state {off| suspended|on}]

Modifies the deduplication state of the filesystem for each specified filesystem identifier or filesystem name. The state can be set to **off**, **on**, or **suspended**.

### [-minimum\_scan\_interval <days>]

Defines the minimum number of days between completing one scan of a filesystem and before scanning the same filesystem again. The values range from 1 to 365 and the default value is 7 days.

### [-minimum\_size <KB>]

Defines the file size in KB that limits deduplication. File sizes equal to this value or smaller will not be deduplicated. Setting this value to zero disables it. This value should not be set lower than 24 KB. The values range from 0 to 1000 and the default value is 24 KB.

# [-maximum\_size <MB>]

Defines the file size in MB of the largest file to be processed for deduplication. Files larger than this size in MB will not be deduplicated. Setting this value too high may affect system write performance as the first write operation reduplicates the file in its entirety. Setting this value to zero disables it. The values range from 0 to 8388608 and the default value is 8388608 MB.

### [-access\_time <days>]

Defines the minimum required file age in days based on read access time. Files that have been read within the specified number of days will not be deduplicated. This setting does not apply to files with an File Retension Level (FLR) locked state. Setting this value to zero disables it. The values range from 0 to 365 and the default value is 15 days.

### [-modification\_time <days>]

Defines the minimum required file age in days based on modification time. Files updated within the specified number of days will not be deduplicated. Setting this value to zero disables it. The values range from 0 to 365 and the default value is 15 days.

### [-case\_sensitive {yes|no}]

Defines whether case-sensitive (for NPS environments) or case-insensitive (for CIFS environments) string comparisons will be used during scans. By default, case-insensitive comparisons will be done to be consistent for CIFS environments. The default value is zero (false).

## [-pathname\_exclude\_list <path\_list>]

This is a filesystem setting only (no global setting). It is empty by default.

Defines a semicolon-delimited list of relative pathnames, in UTF-8 format, to be excluded from deduplication. Any directory below a specified pathname will be excluded from deduplication. You can specify a maximum of 10 pathnames and each one can be up to 1024 bytes. The default value is ' ' (empty).

## [-file\_ext\_exclude\_list <ext\_list>]

Specifies a colon-delimited list of filename extensions to be excluded from deduplication. Each extension must include the leading dot. The default value is ' ' (empty).

# [-duplicate\_detection\_method {sha1|byte|off}]

- 0 (off) This means that duplicate data detection is disabled. With this setting, every deduplicated file is considered unique and the only space savings made are accomplished with compression.
- 1 (sha1) The SHA-1 hash is used to detect duplicate data. It is faster than a byte comparison. This is the default method.
- 2 (byte) This will use a byte-by-byte comparison to detect duplicate data. This adds considerable overhead especially for large files.

### [-savvol\_threshold <percent>]

Represents the percentage of the configured save volume (SavVol) auto extension threshold that can be used during deduplication. Once the specified amount of SavVol is used, deduplication stops on this filesystem. By default, this value is 90 percent and the SavVol auto extension is also 90 percent. This option applies when the SavVol is 81 percent full (90 \* 90). Setting this value to zero disables it. The values range from 0 to 100.

### [-backup\_data\_threshold <percent>]

Indicates the full percentage that a deduplicated file has to be below in order to trigger space-reduced backups for Network Data Management Protocol (NDMP). For example, when set to 90, any deduplicated file whose physical size (compressed file plus changed blocks) is greater than 90 percent of the logical size of the file will have the entire file data backed up without attempting to back it up in a space-reduced format. Setting this value to zero disables it. The values range from 0 to 200 and the default value is 90 percent.

### [-cifs\_compression\_enabled {yes|no}]

This option controls whether CIFS compression is allowed. The default is yes, enable CIFS compression is allowed. When set to yes and the deduplication state of the filesystem is either on or suspended, then CIFS compression is enabled. If the deduplication state is either off or in the process of being turned off, then CIFS compression is not allowed, regardless of whether this option is set to yes.

# -clear {<fs\_name>|id=<fs\_id>}

Sets the filesystem setting back to the Data Mover setting, which is the default setting.

### [-minimum\_scan\_interval]

Defines the minimum number of days between completing one scan of a filesystem and before scanning the same filesystem again. The values range from 1 to 365 and the default value is 7 days.

### [-minimum size]

Defines the file size in KB that limits deduplication. File sizes equal to this value or smaller will not be deduplicated. File sizes greater than this value will be candidates for deduplication. Setting this value to zero disables it. This value should not be set lower than 24 KB. The values range from 0 to 1000 and the default value is 24 KB.

### [-maximum size]

Specifies the file size in MB of the largest file to be processed for deduplication. Files larger than this size in MB will not be deduplicated. Setting this value too high may affect system write performance as the first write operation reduplicates the file in its entirety. Setting this value to zero disables it. The values range from 0 to 8388608 and the default value is 8388608 MB.

### [-access\_time]

Defines the minimum required file age in days based on read access time. Files that have been read within the specified number of days will not be deduplicated. This setting does not apply to files with an FLR locked state. Setting this value to zero disables it. The values range from 0 to 365 and the default value is 15 days.

### [-modification\_time]

Defines the minimum required file age in days based on modification time. Files updated within the specified number of days will not be deduplicated. Setting this value to zero disables it. The values range from 0 to 365 and the default value is 15 days.

### [-case\_sensitive]

Defines whether case-sensitive (for NPS environments) or case-insensitive (for CIFS environments) string comparisons will be used during scans. By default, case-insensitive comparisons will be done to be consistent for CIFS environments. The default value is zero (false).

### [-pathname\_exclude\_list]

This is a filesystem setting only (no global setting). It is empty by default.

Specifies a semicolon-delimited list of relative pathnames, in UTF-8 format, to be excluded from deduplication. Any directory below a specified pathname will be excluded from deduplication. You can specify a maximum of 10 pathnames and each one can be up to 1024 bytes. The default value is ' ' (empty).

### [-file\_ext\_exclude\_list]

Specifies a colon-delimited list of filename extensions to be excluded from deduplication. Each extension must include the leading dot. The default value is ' ' (empty).

### [-duplicate\_detection\_method {sha1|byte|off}]

- 0 (off) This means that duplicate data detection is disabled. With this setting, every deduplicated file is considered unique and the only space savings made are accomplished with compression.
- 1 (sha1) The SHA-1 hash is used to detect duplicate data. It is faster than a byte comparison. This is the default method.
- 2 (byte) This will use a byte-by-byte comparison to detect duplicate data. This adds considerable overhead especially for large files.

### [-savvol\_threshold]

Represents the percentage of the configured save volume (SavVol) auto extension threshold that can be used during deduplication. After the specified amount of SavVol is used, deduplication stops on this filesystem. By default, this value is 90 percent and the SavVol auto extension is also 90 percent. This option applies when the SavVol is 81 percent full (90 \* 90). Setting this value to zero disables it. The values range from 0 to 100.

### [-backup\_data\_threshold]

Indicates the full percentage that a deduplicated file has to be below in order to trigger space-reduced backups for NDMP. For example, when set to 90, any deduplicated file whose physical size (compressed file plus changed blocks) is greater than 90 percent of the logical size of the file will have the entire file data backed up without attempting to back it up in a space-reduced format. Setting this value to zero disables it. The values range from 0 to 200 and the default value is 90 percent.

### [-cifs\_compression\_enabled]

This option controls whether CIFS compression is allowed. The default is yes, enable CIFS compression. When set to yes and the deduplication state of the filesystem is either on or suspended, then CIFS compression is allowed. If the deduplication state is either off or in the process of being turned off, then CIFS compression is not allowed, regardless of whether this option is set to yes.

```
| -default {-info {<movername>|-all}| -set
{<movername>|-all}
```

Manages the Data Mover settings. The **-set** option determines the Data Mover settings.

### [-minimum\_scan\_interval <days>]

Defines the minimum number of days between completing one scan of a filesystem and before scanning the same filesystem again. The values range from 1 to 365 and the default value is 7 days.

### [-minimum\_size <KB>]

Defines the file size in KB that limits deduplication. File sizes equal to this value or smaller will not be deduplicated. File sizes greater than this value will be candidates for deduplication. Setting this value to zero disables it. This value should not be set lower than 24 KB. The values range from 0 to 1000 and the default value is 24 KB.

### [-maximum\_size <MB>]

Defines the file size in MB of the largest file to be processed for deduplication. Files larger than this size in MB will not be deduplicated. Setting this value too high may affect system write performance as the first write operation reduplicates the file in its entirety. Setting this value to zero disables it. The values range from 0 to 8388608 and the default value is 8388608 MB.

### [-access time <days>]

Specifies the minimum required file age in days based on read access time. Files that have been read within the specified number of days will not be deduplicated. This setting does not apply to files with an FLR locked state. Setting this value to zero disables it. The values range from 0 to 365 and the default value is 15 days.

### [-modification\_time <days>]

Specifies the minimum required file age in days based on modification time. Files updated within the specified number of days will not be deduplicated. Setting this value to zero disables it. The values range from 0 to 365 and the default value is 15 days.

### [-case\_sensitive {yes|no}]

Defines whether case-sensitive (for NPS environments) or case-insensitive (for CIFS environments) string comparisons will be used during scans. By default, case-insensitive comparisons will be done to be consistent for CIFS environments. The default value is zero (false).

### [-file\_ext\_exclude\_list <ext\_list>]

Specifies a colon-delimited list of filename extensions to be excluded from deduplication. Each extension must include the leading dot. The default value is ' ' (empty).

# [-duplicate\_detection\_method {sha1|byte|off}]

- 0 (off) This means that duplicate data detection is disabled. With this setting, every deduplicated file is considered unique and the only space savings made are accomplished with compression.
- 1 (sha1) The SHA-1 hash is used to detect duplicate data. It is faster than a byte comparison. This is the default method.
- 2 (byte) This will use a byte-by-byte comparison to detect duplicate data. This adds considerable overhead especially for large files.

### [-savvol\_threshold <percent>]

Represents the percentage of the configured save volume (SavVol) auto extension threshold that can be used during deduplication. Once the specified amount of SavVol is used, deduplication stops on this filesystem. By default, this value is 90 percent and the SavVol auto extension is also 90 percent. This option applies when the SavVol is 81 percent full (90 \* 90). Setting this value to zero disables it. The values range from 0 to 100.

### [-cpu\_usage\_low\_watermark <percent>]

Defines the average percent of CPU usage that can be used during the deduplication process at which full throttle mode is re-entered. The values range from 0 to 100 and the default value is 40 percent. This is a global setting only.

### [-cpu\_usage\_high\_watermark <percent>]

Defines the average percent of CPU usage that can be used during the deduplication process which should trigger a slow throttle mode. The system starts in full throttle mode. The values range from 0 to 100 and the default value is 75 percent. This is a global setting only.

# [-backup\_data\_threshold <percent>]

Defines the full percentage that a deduplicated file has to be below in order to trigger space-reduced backups for NDMP. For example, when set to 90, any deduplicated file whose physical size (compressed file plus changed blocks) is greater than 90 percent of the logical size of the file will have the entire file data backed up without attempting to back it up in a space-reduced format. Setting this value to zero disables it. The values range from 0 to 200 and the default value is 90 percent.

### [-cifs\_compression\_enabled {yes|no}]

This option controls whether CIFS compression is allowed. The default is yes, enable CIFS compression. When set to yes and the deduplication state of the filesystem is either on or suspended, then CIFS compression is allowed. If the deduplication state is either off or in the process of being turned off, then CIFS compression is not allowed, regardless of whether this option is set to yes.

### -clear {<movername> | -all}

The -clear option sets the global setting back to the default value.

### [-minimum\_scan\_interval]

Defines the minimum number of days between completing one scan of a filesystem and before scanning the same filesystem again. The values range from 1 to 365 and the default value is 7 days.

### [-minimum\_size]

Defines the file size in KB that limits deduplication. File sizes equal to this value or smaller will not be deduplicated. File sizes greater than this value will be candidates for deduplication. Setting this value to zero disables it. This value should not be set lower than 24 KB. The values range from 0 to 1000 and the default value is 24 KB.

### [-maximum size]

Defines the file size in MB of the largest file to be processed for deduplication. Files larger than this size in MB will not be deduplicated. Setting this value too high may affect system write performance as the first write operation reduplicates the file in its entirety. Setting this value to zero disables it. The values range from 0 to 8388608 and the default value is 8388608 MB.

### [-access\_time]

Defines the minimum required file age in days based on read access time. Files that have been read within the specified number of days will not be deduplicated. This setting does not apply to files with an FLR locked state. Setting this value to zero disables it. The values range from 0 to 365 and the default value is 15 days.

### [-modification\_time]

Defines the minimum required file age in days based on modification time. Files updated within the specified number of days will not be deduplicated. Setting this value to zero disables it. The values range from 0 to 365 and the default value is 15 days.

### [-case\_sensitive]

Defines whether case-sensitive (for NPS environments) or case-insensitive (for CIFS environments) string comparisons will be used during scans. By default, case-insensitive comparisons will be done to be consistent for CIFS environments. The default value is zero (false).

### [-file\_ext\_exclude\_list]

Specifies a colon-delimited list of filename extensions to be excluded from deduplication. Each extension must include the leading dot. The default value is ' ' (empty).

### [-duplicate\_detection\_method]

0 (off) — This means that duplicate data detection is disabled. With this setting, every deduplicated file is considered unique and the only space savings made are accomplished with compression.

1 (sha1) — The SHA-1 hash is used to detect duplicate data. It is faster than a byte comparison. This is the default method.

2 (byte) — This will use a byte-by-byte comparison to detect duplicate data. This adds considerable overhead especially for large files.

### [-savvol\_threshold]

Represents the percentage of the configured save volume (SavVol) auto extension threshold that can be used during deduplication. After the specified amount of SavVol is used, deduplication stops on this filesystem. By default, this value is 90 percent and the SavVol auto extension is also 90 percent. This option applies when the SavVol is 81 percent full (90 \* 90). Setting this value to zero disables it. The values range from 0 to 100.

# [-cpu\_usage\_low\_watermark]

Specifies the average percent of CPU usage that can be used during the deduplication process at which full throttle mode is re-entered. The values range from 0 to 100 and the default value is 25 percent. This is a global setting only.

### [-cpu\_usage\_high\_watermark]

Specifies the average percent of CPU usage that can be used during the deduplication process which should trigger a slow throttle mode. The system starts in full throttle mode. The values range from 0 to 100 and the default value is 75 percent. This is a global setting only.

### [-backup\_data\_threshold <percent>]

Specifies the full percentage that a deduplicated file has to be below in order to trigger space-reduced backups for NDMP. For example, when set to 90, any deduplicated file whose physical size (compressed file plus changed blocks) is greater than 90 percent of the logical size of the file will have the entire file data backed up without attempting to back it up in a space-reduced format. Setting this value to zero disables it. The values range from 0 to 200 and the default value is 90 percent.

# [-cifs\_compression\_enabled]

This option controls whether CIFS compression is allowed. The default is yes, enable CIFS compression. When set to yes and the deduplication state of the filesystem is either on or suspended, then CIFS compression is allowed. If the deduplication state is either off or in the process of being turned off, then CIFS compression is not allowed, regardless of whether this option is set to yes.

**SEE ALSO:** nas\_fs

**EXAMPLE #1** To list the filesystems and their deduplication states, type:

### \$ fs\_dedupe -list

id	name	state	status	time_of_last_scan	original_ data_size	usage	space_saved
141	ranap1_ replica	Suspended	Idle	Wed Nov 12 09:04:45 EST 2008	-	0%	0 MB (0%)
	ds850gb_ replica1	On	Idle	Fri Nov 21 10:31:15 EST 2008		84%	341590 MB(39%)
495	cworm	On	Idle	Thu Nov 20 09:14:09 EST 2008	-	0%	0 MB (0%)
33	chrisfs1	On	Idle	Sat Nov 22 10:04:33 EST 2008		18%	424 MB (38%)

## Where:

Value	Definition	
id	Filesystem identifier.	
name	Name of the filesystem.	
state	Deduplication state of the filesystem. The file data is transferred to the storage which performs the deduplication and compression on the data. The states are:	
	On — Deduplication on the filesystem is enabled.	
	Suspended — Deduplication on the filesystem is suspended. Deduplication does not perform any new space reduction but existing space-reduced files remain the same.	
	<ul> <li>Off — Deduplication on the filesystem is disabled. Deduplication does not perform any new space reduction and the data is now reduplicated, which is the process used to restore a file that was deduplicated to its original condition.</li> </ul>	
status	Current state of the deduplication-enabled filesystem. The progress statuses are:	
	Idle — Deduplication process is currently idle.	
	Scanning — Filesystem is being scanned for deduplication. It displays the percentage of scanned files in the filesystem.	
	<ul> <li>Reduplicating — Filesystem files are being reduplicated from the deduplicated files. It displays the percentage of reduplicated files.</li> </ul>	
time_of_last_scan	Time when the filesystem was last scanned.	

Value	Definition
original_data_size	Original size of the filesystem before deduplication.
usage	The current space usage of the filesystem.
space_saved	The filesystem space saved after deduplication.

# **EXAMPLE #2** To list the filesystems and provide detailed reports on the state of the deduplication processing, type:

### \$ fs\_dedupe -info -all

```
Id
                               = 53
                               = svr2fs1
Name
Deduplication
                               = Off
File system parameters:
   Case Sensitive
                               = no
   Duplicate Detection Method = sha1
   Access Time
                               = 15
   Modification Time
                              = 15
   Minimum Size
                              = 24 KB
   Maximum Size
                               = 8388608 MB
   File Extension Exclude List =
                            = 7
   Minimum Scan Interval
   Savevol Threshold
                              = 90
   Backup Data Threshold
                              = 90
   Cifs Compression Enabled
                              = yes
   Pathname Exclude List
Ιd
                               = 2040
Name
                               = server 2 fsltest2
Deduplication
                               = Suspended
As of the last file system scan (Mon Aug 17 11:33:38 EDT 2009):
   Files scanned
                               = 4
   Files deduped
                               = 3 (75% of total files)
                             = 2016 \text{ MB}
   File system capacity
   Original data size
                             = 6 MB (0% of current file system capacity)
   Space saved
                               = 0 MB (0% of original data size)
File system parameters:
   Case Sensitive
                               = no
   Duplicate Detection Method = sha1
   Access Time
                               = 15
   Modification Time
                              = 15
   Minimum Size
                               = 24 KB
   Maximum Size
                               = 8388608 MB
   File Extension Exclude List =
                              = 7
   Minimum Scan Interval
   Savevol Threshold
                               = 90
                             = 90
   Backup Data Threshold
   Cifs Compression Enabled
                              = yes
   Pathname Exclude List
Ιd
                               = 506
```

```
Name
                              = demofs
                              = Off
Deduplication
File system parameters:
  Case Sensitive
                              = no
  Duplicate Detection Method = sha1
  Access Time
                            = 15
  Modification Time
                            = 15
  Minimum Size
                             = 24 KB
  Maximum Size
                            = 8388608 MB
  File Extension Exclude List =
  Minimum Scan Interval = 7
  Savevol Threshold
                            = 90
                            = 90
  Backup Data Threshold
  Cifs Compression Enabled = yes
  Pathname Exclude List
Ιd
                              = 2113
Name
                              = testrdefs
Deduplication
                             = Suspended
As of the last file system scan (Thu Aug 13 14:22:31 EDT 2009):
  Files scanned
                            = 1
  Files deduped
                            = 0 (0% of total files)
  File system capacity
                            = 1008 \text{ MB}
  Original data size
                            = 0 MB (0% of current file system capacity)
  Space saved
                            = 0 MB (0% of original data size)
File system parameters:
  Case Sensitive
                             = no
  Duplicate Detection Method = sha1
  Access Time
                             = 15
  Modification Time
                             = 15
  Minimum Size
                            = 24 KB
  Maximum Size
                            = 8388608 MB
  File Extension Exclude List =
  Minimum Scan Interval = 7
  Savevol Threshold
                            = 90
                           = 90
  Backup Data Threshold
  Cifs Compression Enabled = yes
  Pathname Exclude List
ЪТ
                              = 2093
Name
                              = kfs ckpt1
Deduplication
                              = Off
File system parameters:
  Case Sensitive
  Duplicate Detection Method = sha1
  Access Time
                             = 15
                            = 15
  Modification Time
  Minimum Size
                             = 24 KB
  Maximum Size
                             = 8388608 MB
  File Extension Exclude List =
  Minimum Scan Interval = 7
  Savevol Threshold
                             = 90
```

52

```
Backup Data Threshold
                             = 90
   Cifs Compression Enabled = yes
   Pathname Exclude List
ЬТ
                               = 2095
Name
                               = ranap-test3
Deduplication
                               = On
Status
                               = Idle
As of the last file system scan (Tue Aug 11 17:37:58 EDT 2009):
   Files scanned
                              = 30
   Files deduped
                              = 2 (7% of total files)
   File system capacity
                             = 5041 \text{ MB}
   Original data size
                              = 1109 MB (22% of current file system capacity)
   Space saved
                              = 0 MB (0% of original data size)
File system parameters:
   Case Sensitive
                              = no
   Duplicate Detection Method = sha1
   Access Time
                              = 15
   Modification Time
                              = 15
   Minimum Size
                              = 24 KB
   Maximum Size
                              = 8388608 MB
   File Extension Exclude List =
                          = 7
   Minimum Scan Interval
   Savevol Threshold
                              = 90
   Backup Data Threshold
                             = 90
   Cifs Compression Enabled = yes
   Pathname Exclude List
```

### Where:

Value	Definition
Deduplication	Current deduplication state of the filesystem.
Status	Progress status of the files being scanned.
Name	Name of the filesystem.
Id	Filesystem identifier.
Files scanned	Number of files scanned.
Files deduped	Number of files in the filesystem that has been deduplicated.
Original data size	Proportion of space in use with respect to the filesystem capacity.
File system capacity	Current space usage of the filesystem.
Space saved	Proportion of space saved with respect to the original data size.
Case Sensitive	Method of string comparison: case-sensitive or case-insensitive.
Duplicate Detection Method	Method of duplication detection: 0, sha-1, or byte-by-byte.
Access Time	Minimum required file age in days based on read access time.
Modification Time	Minimum required file age in days based on modification time.
Minimum Size	Minimum file size to be processed for deduplication.
Maximum Size	Maximum file size to be processed for deduplication.
File Extension Exclude List	Lists filename extensions to be excluded from the deduplication.

Minimum Scan Interval	Minimum number of days between completing one scan of a filesystem and before scanning the same filesystem again.	
SavVol Threshold	Percentage of SavVol space that can be used during deduplication.	
Backup Data Threshold	Percentage below which a deduplicated file has to be in order to trigger space-reduced NDMP backups.	
Cifs Compression Enabled	Controls whether CIFS permission is enabled.	
Pathname Exclude List	Lists relative pathnames to be excluded from the deduplication.	

**Note:** If reduplication fails, then the state transitions to the suspended state and a CCMD message will be sent to the server's event log. If reduplication succeeds, then it remains in the off state.

# **EXAMPLE #3** To list the filesystems for a given filesystem name, type:

```
$ fs_dedupe -info server3_fs3
Ιd
                                   = 98
Name
                                   = server3 fs3
Deduplication
                                   = On
Status
                                   = Idle
As of the last file system scan on Tue Sep 23 13:28:01 EDT 2008:
                         = 30 (100%)
= 413590 MB
= 117 MB (0% of current file system capacity)
= 106 MB (90% of original data size)
  Files deduped
  File system capacity
  Original data size
   Space saved
File system parameters:
  Case Sensitive
                                  = yes
  Duplicate Detection Method = sha1
  Access Time
                                  = 30
  Modification Time
                                  = 30
  Minimum Size
                                  = 20
  Maximum Size
                                  = 200
  File Extension Exclude List = .jpg:.db:.pst
  Minimum Scan Interval
                                  = 1
   SavVol Threshold
                                  = 90
   Backup Data Threshold
                                  = 90
   Pathname Exclude List
                                  = root;etc
```

EXAMPLE #2 provides a description of command output.

# EXAMPLE #4

To list the filesystems for a given filesystem name with the deduplication properties set to off, type:

# \$ fs\_dedupe -info server3\_fs3

```
Duplicate Detection Method = sha1
Access Time = 30
Modification Time
                           = 30
                           = 20
Minimum Size
Maximum Size
                           = 200
File Extension Exclude List = .jpg:.db:.pst
Minimum Scan Interval
                          = 1
SavVol Threshold
                           = 90
Backup Data Threshold
                           = 90
                   = root;etc
Pathname Exclude List
```

EXAMPLE #2 provides a description of command output.

# **EXAMPLE #5**

To list the filesystems for a given filesystem name with the deduplication properties set to off, type:

```
$ fs_dedupe -info server3_fs3
ЬТ
                                 = 98
                                 = server3 fs3
Name
Deduplication
                                 = Off
Status
                                 = re-duplicating - 20% complete
Deduplication
                                 = Off
File system parameters:
  Case Sensitive
                                = ves
  Duplicate Detection Method
                                = sha1
  Access Time = 30
  Modification Time
                               = 30
  Minimum Size
                               = 20
  Maximum Size
                                = 200
  File Extension Exclude List = .jpg:.db:.pst
  Minimum Scan Interval
                               = 1
  SavVol Threshold
                                = 90
  Backup Data Threshold
                               = 90
  Cifs Compression Enabled
                               = yes
   Pathname Exclude List
                                = root;etc
```

EXAMPLE #2 provides a description of command output.

# **EXAMPLE #6** To list the duplication properties of a given Data Mover, type:

### \$ fs\_dedupe -default -info server\_2

```
Server parameters:
  Case Sensitive
                                   = yes
  Duplicate Detection Method
                                   = sha1
  Access Time = 30
  Modification Time
                                  = 30
  Minimum Size
                                  = 20
  Maximum Size
                                  = 200
  File Extension Exclude List = .jpg:.db:.pst
Minimum Scan Interval = 1
  SavVol Threshold
                                  = 90
  Backup Data Threshold
                                  = 90
```

## The fs Commands

```
CPU % Usage Low Water Mark = 25
CPU % Usage High Water Mark = 90
Cifs Compression Enabled = yes
```

# Where:

Value	Definition		
Deduplication	Current deduplication state of the filesystem.		
Status	Progress status of the files being scanned.		
Name	Name of the filesystem.		
Id	Filesystem identifier.		
Files scanned	Number of files scanned.		
Files deduped	Number of files in the filesystem that has been deduplicated.		
Original data size	Proportion of space in use with respect to the filesystem capacity.		
File system capacity	Current space usage of the filesystem.		
Space saved	Proportion of space saved with respect to the original data size.		
Case Sensitive	Method of string comparison: case-sensitive or case-insensitive.		
Duplicate Detection Method	Method of duplication detection: 0, sha-1, or byte-by-byte.		
Access Time	Minimum required file age in days based on read access time.		
Modification Time	Minimum required file age in days based on modification time.		
Minimum Size	Minimum file size to be processed for deduplication.		
Maximum Size	Maximum file size to be processed for deduplication.		
File Extension Exclude List	Lists filename extensions to be excluded from the deduplication.		
Minimum Scan Interval	Minimum number of days between completing one scan of a filesystem and before scanning the same filesystem again.		
SavVol Threshold	Percentage of SavVol space that can be used during deduplication.		
Backup Data Threshold	Percentage below which a deduplicated file has to be in order to trigger space-reduced NDMP backups.		
CPU % Usage Low Water Mark	Average percentage of CPU usage which should trigger full throttle mode.		
CPU % Usage High Water Mark	Average percentage of CPU usage which should trigger slow throttle mode.		
Pathname Exclude List	Lists relative pathnames to be excluded from the deduplication.		

# **EXAMPLE #7** To modify the filesystem, type:

\$ fs\_dedupe -modify testrdefs -state on

Done

**EXAMPLE #8** To modify the filesystem settings to the user-specified values, type:

\$ fs\_dedupe -modify testrdefs -maximum\_size 100
-file\_extension\_exclude\_list .jpg:.db:.pst

Done

**EXAMPLE #9** To modify specific Data Mover settings, type: \$ fs\_dedupe -default -set server\_2 -maximum\_size 100 -minimum\_size 20 -duplicate\_detection\_method shall Done **EXAMPLE #10** To reset the filesystem settings to the default settings (which are the Data Mover settings), type: \$ fs\_dedupe -clear testrdefs -maximum\_size -minimum\_size -duplicate\_detection\_method Done **EXAMPLE #11** To reset specific Data Mover settings to the default settings, type: \$ fs\_dedupe -default -clear server\_2 -maximum\_size -minimum\_size -duplicate\_detection\_method Done **EXAMPLE #12** To reset all options for a specific Data Mover to the default settings, type: \$ fs\_dedupe -default -clear server\_2 Done **EXAMPLE #13** To reset all options on all Data Movers to the default settings, type: \$ fs\_dedupe -default -clear -all Done

# fs\_dhsm

Manages the Celerra FileMover filesystem connections.

# **SYNOPSIS**

```
fs dhsm
  -list
  -info [<fs_name>|id=<fs_id>]
 -modify {<fs_name>|id=<fs_id>}[-state enabled]
    [-popup_timeout <sec>][-backup {offline | passthrough}]
    [-log {on|off}][-max log size <mb>][-offline_attr {on|off}]
    [-read_policy_override {none|full|passthrough|partial}]}
  -modify {<fs_name>|id=<fs_id>}[-state disabled]
 -connection {<fs_name>|id=<fs_id>}
    -list
    -info [<cid>]
   -create -type {nfsv3 | nfsv2} -secondary <nfs_server>:/<path>
        [-read policy override {full|passthrough|partial|none}]
        [-useRootCred {true | false}][-proto {UDP | TCP}][-nfsPort <port>]
        [-mntPort <port>][-mntVer {3 | 2 | 1}][-localPort <port>]
   |-create -type cifs -admin [<fqdn>\]<admin_name>
        -secondary \\<fqdn>\<share>[\<path>]
        -local_server <host_name> [-wins <address>][-password <password>]
        [-read policy override {full|passthrough|partial|none}]
   -create -type http -secondary http://<host><url_path>
        [-read policy override {full|passthrough|partial|none}]
        [-httpPort <port>] [-localPort <port>]
        [-user <username> [-password <password>]]
        [-timeout <seconds>][-cgi {y n}]
   -create -type https -secondary https://<host><url_path>
        [-read_policy_override {full|passthrough|partial|none}]
        [-httpsPort <port>][-localPort <port>]
        [-user <username> [-password <password>]]
        [-timeout <seconds>][-cgi {y | n}]
    -delete {-all|<cid>[,<cid>...]} [-recall_policy {check|no|yes}]
    -modify {-all|<cid>[,<cid>...]} [-state {enabled|disabled|recallonly}]
        [-read policy override {full|passthrough|partial|none}]
        [{[-nfs_server <address>] [-localPort <port>]
         [-proto {TCP | UDP}] [-useRootCred {true | false}]}
        [-cifs_server <fqdn>][-local_server <host_name>]
         [-password <password>][-admin [<fqdn>\]<admin_name>]
         [-wins <address>]}
        [{[-http_server <host>][-httpPort <port>][-httpsPort <port>]
         [-localPort <port>][-user <username>]
         [-password <password>] [-timeout <seconds>] }
```

# DESCRIPTION

The **fs\_dhsm** command modifies the properties on filesystems enabled for Celerra FileMover. The **fs\_dhsm** command creates, deletes, and modifies NFS, CIFS, and HTTP connections to remote hosts, lists Celerra FileMover filesystems, and provides information on the connections.

### **OPTIONS**

### -list

Lists all filesystems enabled with the Celerra FileMover.

Displays information for the specified Celerra FileMover filesystems.

Sets Celerra FileMover parameters for the specified filesystem.

Note: When specifying the -modify option on a disabled filesystem, the state is automatically changed to enabled. When specifying the -state disabled option, it is not possible to specify any other parameter to modify.

### [-state enabled]

Enables Celerra FileMover operations on the specified filesystem. The filesystem must be **enabled** to accept other options.

#### [-state disabled]

Disables Celerra FileMover operations on the specified filesystem. New FileMover attributes cannot be specified as part of a disable command, nor can be specified for a filesystem that is in the disabled state. The attributes persist. If the filesystem is enabled after a disable command, then the attributes prior to the disable command take effect.

# [-popup timeout <sec>]

Specifies the Windows popup timeout value in seconds. If a CIFS I/O request cannot be processed within the specified time, then a popup notification of the delay is sent to the CIFS client. The default for <sec> is 0 (zero) which disables Windows popups.

**Note:** It may take up to 10 seconds before the popup is displayed.

# [-backup {offline | passthrough}]

Specifies the nature of CIFS network backups. The **offline** option backs up the stub file only. The **passthrough** (default) option backs up all of the file data by using passthrough read.

### [-log {on|off}]

Enables or disables Celerra FileMover logging. The default log filename is dhsm.log; it resides in the /.etc directory on the FileMover-enabled filesystem.

### [-max\_log\_size <mb>]

Specifies the maximum size of the log file. The current log file, in addition to four old log files, is saved. The minimum log file size is 10 MB.

## [-offline\_attr {on off}]

Specifies whether the Data Mover should set the CIFS offline file attributes on the stub files. The default is **on**.



# **CAUTION**

It is recommended that you do not disable the CIFS offline attributes.

# [-read\_policy\_override {none|full|passthrough| partial}]

Specifies the migration method option used by the Celerra Network Server, in the connection level or filesystem level, to override the migration method specified in the stub file. **none** (default) specifies no override, **full** recalls the whole file to the Celerra Network Server on a read request before the data is returned, **passthrough** retrieves data without recalling the data to the Celerra Network Server, and **partial** recalls only the blocks required to satisfy the client read request.

**Note:** The full migration may take several minutes or hours if the file is very large.

-connection {<fs\_name>|id=<fs\_id>} -list
Lists all connections for the specified filesystem.

-connection {<fs\_name>|id=<fs\_id>} -info [<cid>]

Displays details on all connections for the specified filesystem. If the <cid> is specified, only information for that connection is displayed.

**Note:** A connection ID is automatically created when a connection is established. The connection ID is displayed using the **-list** and is referred to as the <cid> in other commands.

# **NFS CONNECTIONS**

-connection {<fs\_name>|id=<fs\_id>} -create -type
{nfsv3|nfsv2} -secondary <nfs\_server>:/<path>

Creates a connection using the NFS protocol between the specified filesystem and the secondary filesystem. The secondary filesystem stores migrated data. The **-type** option specifies the NFS version that the Data Mover should use when connecting to the secondary server.

Note: Celerra FileMover does not currently support NFSv4 protocol.

The **-secondary** option specifies the location of the remote filesystem.

**Note:** Although an IP address can be specified for an <nfs\_server>, EMC strongly suggests using the hostname of the server, which allows you to take advantage of Domain Name System (DNS) failover capability.

[-read\_policy\_override {full|passthrough|partial |none}]

Specifies the migration method for data recall in response to client read requests. **full** migrates the whole file before it returns the requested blocks. **passthrough** leaves the stub file, but retrieves the requested data from the secondary filesystem. **partial** migrates only the blocks required to satisfy the client read request. **none** (default) defaults to the read method option specified in the stub file.

**Note:** The full migration may take minutes or hours if the file is very large.

[-useRootCred {true | false}]

Specifies the user credentials that the Data Mover uses when requesting data from the secondary Celerra Network Server. When set to true, the Data Mover requests data as the **root** user (UID 0). When set to false (default), the Data Mover requests data as the owner of the file as specified in the stub file.

**Note:** If the **-useRootCred** option is set to **true**, the secondary storage NFS server must grant the Data Mover **root** privilege for NFS traffic.

[-proto {TCP | UDP}]

Specifies the protocol for the Data Movers to use for communication to the secondary <nfs\_server>. TCP is the default.

```
[-nfsPort <port>]
```

Specifies an NFS port on the secondary <nfs\_server>. A default port is discovered automatically.

```
[-mntPort <port>]
```

Specifies a mount port on the secondary <nfs\_server>. A default mount port is discovered automatically.

**Note:** The **-nfsPort** and the **-mntPort** options are used for secondary servers which do not have the Portmapper running. The admin starts the **nfsd** and **mountd** daemons on specific ports to avoid hackers.

```
[-mntVer {1 | 2 | 3}]
```

Specifies the mount version for the NFS connection. If the **-type** is nfsv3, then the **-mntVer** must be 3. If the **-type** is nfsv2, then 1 or 2 can be specified. The default for nfsv2 is 2.

```
[-localPort <port>]
```

Overrides the default port that the Data Mover uses during connection to be compatible with firewalls. The default for UDP is 1020. By default, TCP uses a random port over 1024 to make the connection.

```
-connection {<fs_name>|id=<fs_id>} -modify {-all|
<cid>[,<cid>...]}
```

Changes parameters on an existing NFS Celerra FileMover connection. Either all connections can be removed or just the specified <cid> connection can be removed.

```
[-state {enabled|disabled|recallonly}]
```

Sets the state of Celerra FileMover operations on the specified filesystem. **enabled** (default) allows both the creation of stub files and data migration through reads and writes. If the state is **disabled**, neither stub files nor data migration is possible. Data currently on the Celerra Network Server can be read and written to in the disabled state.

If the state is **recallonly**, the policy engine is not allowed to create stub files, but the user is still able to trigger data migration using a read or write request from the secondary filesystem to the Celerra Network Server.

```
[-read_policy_override {full|passthrough|partial
|none}]
```

Specifies the migration method option used by the Celerra Network Server, in the connection level or filesystem level, to override the migration method specified in the stub file. **none** (default) specifies no override, **full** recalls the whole file to the Celerra Network Server on read request before the data is returned, **passthrough** retrieves data without recalling the data to the Celerra Network Server, and **partial** recalls only the blocks required to satisfy the client read request.

**Note:** The full migration may take minutes or hours if the file is very large.

```
[-nfs_server <address>]
```

Specifies the name or IP address of the secondary NFS server.

**Note:** Although an IP address can be specified for the <nfs\_server>, EMC strongly suggests using the hostname of the server, which allows use of the DNS failover capability.

```
[-localPort <port>]
```

Specifies a port to override the default port used by the Data Mover during connection for compatibility with firewalls.

```
[-proto {TCP | UDP}]
```

Specifies the protocol for the Data Mover to use for NFS communications to the secondary <nfs\_server>. TCP is the default.

```
[-useRootCred {true | false}]}
```

Specifies the user credentials that the Data Mover uses when requesting data from the secondary Celerra Network Server. When set to true, the Data Mover requests data as the **root** user (UID 0). When set to false (default), the Data Mover requests data as the owner of the file as specified in the stub file.

**Note:** If the **-useRootCred** option is set to **true**, the secondary storage NFS server must grant the Data Mover **root** privilege for NFS traffic.

**-connection** connection cfs\_name> -delete {-all|<cid> [,<cid>...]}
Removes an existing NFS connection between the filesystem and the secondary filesystem. Either all connections can be removed or just the specified <cid> connection can be removed.

### [-recall\_policy {check | no | yes}]

Specifies the recall policy for any migrated file during the **-delete**. **check** (default) scans the filesystem for stub files that depend on the connection and fails on the first one. **no** deletes the connection without checking for stub files that depend on the connection, and **yes** migrates the files back to the Celerra Network Server before the connection is removed. If **no** is specified and stub files exist, an I/O error appears when the file is read because the connection no longer exists.

# **CIFS CONNECTIONS**

**-connection** {<fs\_name>|id=<fs\_id>} **-create -type cifs** Creates a connection using the CIFS protocol between the specified filesystem and a secondary filesystem. A connection ID is automatically created when a connection is established. The connection ID is seen using the **-list** and is referred to as the <cid> in other commands.

### -admin [<fqdn>\]<admin\_name>

Specifies the <admin\_name> used to make the CIFS connection. If an optional <fqdn> is specified, it must be a fully qualified domain name. The [<fqdn>\]<admin\_name> entry must be enclosed within quotes as shown in EXAMPLE #2. If the <fqdn> is not specified, the -local\_server domain is used.

### -secondary \\<fqdn>\<share>[\<path>]

Specifies the CIFS server, the share, and path for the secondary server for connection. The <fqdn>\<share>[\<path>] entry must be enclosed within quotes. The domain must be fully qualified; an IP address will not work.

#### -local\_server <host\_name>

Specifies the NetBIOS name or computer name of the local CIFS server on the Data Mover.

### [-wins <address>]

Specifies a WINS server to resolve names in a Windows domain.

### [-password <password>]

Allows the user to specify the admin password. The password is not recorded in the command log. If the **-password** option is given but no password is specified, the user is prompted interactively.



### **CAUTION**

When specifying the password with this option, be aware it is unmasked, and visible to other users. The command may also be read from the log of the shell.

# [-read\_policy\_override {full|passthrough|partial |none}]

Specifies the migration method for data recall in response to client read requests. **full** migrates the whole file before it returns the requested blocks. **passthrough** leaves the stub file, but retrieves the requested data from the secondary filesystem. **partial** migrates only the blocks required to satisfy the client read request. **none** (default) defaults to the read method option specified in the stub file.

**Note:** The full migration may take several minutes or hours if the file is very large.

```
-connection {<fs_name>|id=<fs_id>} -modify {-all|
<cid>[,<cid>...]}
```

Changes parameters on an existing NFS Celerra FileMover connection.

```
[-state {enabled|disabled|recallonly}]
```

Sets the state of Celerra FileMover operations on the specified filesystem. **enabled** (default) allows both the creation of stub files and data migration through reads and writes. If the state is **disabled**, neither stub files nor data migration is possible. Data currently on the Celerra Network Server can be read and written to in the disabled state.

If the state is **recallonly**, the policy engine is not allowed to create stub files, but the user is still able to trigger data migration using a read or write request from the secondary filesystem to the Celerra Network Server.

```
[-read_policy_override {full|passthrough|partial |none}]
```

Specifies the migration method option used by the Celerra Network Server, in the connection level or filesystem level, to override the migration method specified in the stub file. **none** (default) specifies no override, **full** recalls the whole file to the Celerra Network Server on read request before the data is returned, **passthrough** retrieves data without recalling the data to the Celerra Network Server, and **partial** recalls only the blocks required to satisfy the client read request.

**Note:** The full migration may take minutes or hours if the file is very large.

```
[-cifs_server <fqdn>]
```

Specifies the fully qualified domain name of the secondary CIFS server.

```
[-local_server <host_name>]
```

Specifies the NetBIOS name or computer name of the local CIFS server on the Data Mover.

```
[-password <password>]
```

Allows the user to specify the admin password. The password is not recorded in the command log. If the **-password** option is given but no password is specified, the user is prompted interactively.

When specifying the password with this option, be aware it is unmasked, and visible to other users. The command may also be read from the log of the shell.

```
[-admin [<fqdn>\]<admin_name>]
```

Specifies the <admin\_name> used to make the CIFS connection. If an optional <fqdn> is specified, it must be a fully qualified domain name. If the <fqdn> is not specified, the -local\_server domain is used.

```
[-wins <address>] }
```

Specifies a WINS server to resolve names in a Windows domain.

**-connection** -connection cfs\_name -delete {-all|<cid> [,<cid>...]}
Removes an existing CIFS connection between the filesystem and the secondary filesystem.

```
[-recall_policy {check|no|yes}]
```

Specifies the recall policy for any migrated file during the **-delete** option. **check** (default) scans the filesystem for stub files that depend on the connection and fails on the first one. **no** deletes the connection without checking for stub files that depend on the connection, and **yes** migrates the files back to the Celerra

Network Server before the connection is removed. If **no** is specified and stub files exist, an I/O error appears when the file is read because the connection no longer exists.

### HTTP CONNECTIONS

```
-connection {<fs_name>|id=<fs_id>} -create -type http
-secondary http://<host><url_path>
```

Creates a connection using the HTTP protocol between the specified primary filesystem and a secondary filesystem. There are two types of HTTP connections: CGI and non-CGI. For CGI connections, the value of the **-secondary** option specifies the hostname of the server running the secondary storage HTTP server and the location of the CGI application that provides access to a storage system. For non-CGI connections, the value for the **-secondary** option specifies the hostname and, optionally, a portion of the hierarchical namespace published by the web server.

**Note:** Although an IP address can be specified for a <host>, EMC strongly suggests using the hostname of the server, which allows the DNS failover capability.

# [-read\_policy\_override {full|passthrough|partial |none}]

Specifies the migration method option used by the Celerra Network Server, in the connection level or filesystem level, to override the migration method specified in the stub file. **none** (default) specifies no override, **full** recalls the whole file to the Celerra Network Server on read request before the data is returned, **passthrough** retrieves data without recalling the data to the Celerra Network Server, and **partial** recalls only the blocks required to satisfy the client read request.

**Note:** The full migration may take several minutes or hours if the file is very large.

### -httpPort <port>

Specifies the remote port number that the Data Mover delivers the HTTP request to. If not specified, the Data Mover issues HTTP requests to port 80 on the secondary storage HTTP server.

### -localPort <port>

Specifies the local port number the Data Mover uses to issue HTTP requests to the web server active on the secondary storage.

The <port> specified should be an integer no less than 1024. If not specified, the Data Mover selects a port to issue the HTTP requests.

**Note:** The two end points of an HTTP connection are specified by the filesystem name and the value specified for the **-secondary** option. If multiple connections are created by using identical end points with different attributes such as **-cgi**, **-user**, **-password**, **-localPort**, **-httpPort**, the connection will fail.

### [-user <username>]

Defines the username the HTTP client uses if digest authentication is required by the secondary storage HTTP server.

### [-password <password>]

Allows the user to specify the admin password. The password is not recorded in the command log. If the **-password** option is given but no password is specified, the user is prompted interactively.

Use the **-password** option when digest authentication is required by the secondary storage HTTP server.

```
[-timeout <seconds>
```

Specifies the timeout value in seconds. By default, the Celerra HTTP client waits 30 seconds for a reply from the HTTP server and then retries the operation once.

```
[-cgi \{y | n\}]
```

Specifies the HTTP connection type: CGI or non-CGI. By default, FileMover assumes that the web server is using CGI connections to access migrated file data by using a CGI application. For non-CGI connections, set the **-cgi** option to **n**; FileMover then assumes the web server has direct access to migrated file content on secondary storage.

```
-connection {<fs_name>|id=<fs_id>} -modify {-all|
<cid>[,<cid>...]}
```

Changes parameters on an existing NFS Celerra FileMover connection.

```
[-state {enabled|disabled|recallonly}]
```

Sets the state of Celerra FileMover operations on the specified filesystem. **enabled** (default) allows both the creation of stub files and data migration through reads and writes. If the state is

**disabled**, neither stub files nor data migration is possible. Data currently on the Celerra Network Server can be read and written to in the disabled state.

If the state is **recallonly**, the policy engine is not allowed to create stub files, but the user is still able to trigger data migration by using a read or write request from the secondary filesystem to the Celerra Network Server.

# [-read\_policy\_override {full|passthrough|partial |none}]

Specifies the migration method option used by the Celerra Network Server, in the connection level or filesystem level, to override the migration method specified in the stub file. none (default) specifies no override, full recalls the whole file to the Celerra Network Server on read request before the data is returned, passthrough retrieves data without recalling the data to the Celerra Network Server, and partial recalls only the blocks required to satisfy the client read request.

**Note:** The full migration may take minutes or hours if the file is very large.

# [-http\_server <host>]

Specifies the hostname of the secondary storage HTTP server.

### -httpPort <port>

Specifies the remote port number that the Data Mover delivers the HTTP request to. If not specified, the Data Mover issues HTTP requests to port 80 on the secondary storage HTTP server.

# -localPort <port>

Specifies the local port number the Data Mover uses to issue HTTP requests to the web server active on the secondary storage. The <port> specified should be an integer no less than 1024. If not specified, the Data Mover selects a port to issue the HTTP requests.

**Note:** If you attempt to create multiple HTTP connections by using identical end points with different attributes such as **-cgi**, **-user**, **-password**, **-localPort**, **-httpPort**, the connection will fail.

### [-user <username>]

An optional attribute used to define the username the HTTP client uses if digest authentication is required by the secondary storage HTTP server.

# [-password <password>]

Allows the user to specify the admin password. The password is not recorded in the command log. If the **-password** option is given but no password is specified, the user is prompted interactively.

### [-timeout <sec>]

Specifies the timeout value in seconds. By default, Celerra's HTTP client waits 30 seconds for a reply from the HTTP server and then retries the operation once before commencing the failover operation.

**-connection** <fs\_name> **-delete** {**-all** | <cid> [, <cid>...]} Removes an existing HTTP connection between the filesystem and the secondary filesystem. Either all connections can be removed or just the specified <cid> connection can be removed.

# [-recall\_policy {check | no | yes}]

Specifies the recall policy for any migrated file during the **-delete** option. The **check** (default) argument scans the filesystem for stub files that depend on the connection and fails on the first one. **no** deletes the connection without checking for stub files that depend on the connection, and **yes** migrates the files back to the Celerra Network Server before the connection is removed. If **no** is specified and stub files exist, an I/O error appears when the file is read because the connection no longer exists.

# HTTPS CONNECTIONS

-connection {<fs\_name>|id=<fs\_id>} -create -type https -secondary https://<host><url\_path>

Creates a connection by using the HTTPS protocol between the specified primary filesystem and a secondary filesystem. There are two types of HTTPS connections: CGI and non-CGI. For CGI connections, the value of the **-secondary** option specifies the hostname of the server running the secondary storage HTTPS server and the location of the CGI application that provides access to a storage system. For non-CGI connections, the value for the **-secondary** option specifies the hostname and, optionally, a portion of the hierarchical namespace published by the web server.

**Note:** Although an IP address can be specified for a <host>, EMC strongly suggests using the hostname of the server, which allows the DNS failover capability.

# [-read\_policy\_override {full|passthrough|partial |none}]

Specifies the migration method option used by the Celerra Network Server, in the connection level or filesystem level, to override the migration method specified in the stub file. none (default) specifies no override, full recalls the whole file to the Celerra Network Server on read request before the data is returned, passthrough retrieves data without recalling the data to the Celerra Network Server, and partial recalls only the blocks required to satisfy the client read request.

**Note:** The full migration may take several minutes or hours if the file is very large.

# [-httpsPort <port>]

Specifies the remote port number that the Data Mover delivers the HTTPS request to. If not specified, the Data Mover issues HTTPS requests to port 443 on the secondary storage HTTPS server.

# [-localPort <port>]

Specifies the local port number the Data Mover uses to issue HTTPS requests to the web server active on the secondary storage. The <port> specified should be an integer no less than 1024. If not specified, the Data Mover selects a port to issue the HTTPS requests.

**Note:** The two end points of an HTTPS connection are specified by the filesystem name and the value specified for the **-secondary** option. If multiple connections are created by using identical end points with different attributes such as **-cgi**, **-user**, **-password**, **-localPort**, **-httpsPort**, the connection will fail.

# [-user <username>]

Defines the username the HTTPS client uses if digest authentication is required by the secondary storage HTTPS server.

### [-password <password>]

Allows the user to specify the admin password. The password is not recorded in the command log. If the **-password** option is given but no password is specified, the user is prompted interactively.

Use the **-password** option when digest authentication is required by the secondary storage HTTPS server.

### [-timeout <seconds>

Specifies the timeout value in seconds. By default, the Celerra HTTPS client waits 30 seconds for a reply from the HTTPS server and then retries the operation once.

# [-cgi {y | n}

Specifies the HTTPS connection type: Common Gateway Interface (CGI) or non-CGI. By default, FileMover assumes that the web server is using CGI connections to access migrated file data by using a CGI application. For non-CGI connections, set the **-cgi** option to **n**; FileMover then assumes the web server has direct access to migrated file content on secondary storage.

```
-connection {<fs_name>|id=<fs_id>} -modify {-all|
<cid>[,<cid>...]}
```

Changes parameters on an existing NFS Celerra FileMover connection.

# [-state {enabled|disabled|recallonly}]

Sets the state of Celerra FileMover operations on the specified filesystem. **enabled** (default) allows both the creation of stub files and data migration through reads and writes. If the state is **disabled**, neither stub files nor data migration is possible. Data currently on the Celerra Network Server can be read and written to in the disabled state.

If the state is **recallonly**, the policy engine is not allowed to create stub files, but the user is still able to trigger data migration by using a read or write request from the secondary filesystem to the Celerra Network Server.

# [-read\_policy\_override {full|passthrough|partial |none}]

Specifies the migration method option used by the Celerra Network Server, in the connection level or filesystem level, to override the migration method specified in the stub file. **none** (default) specifies no override, **full** recalls the whole file to the Celerra Network Server on read request before the data is returned, **passthrough** retrieves data without recalling the data to the Celerra Network Server, and **partial** recalls only the blocks required to satisfy the client read request.

**Note:** The full migration may take minutes or hours if the file is very large.

#### [-http\_server <host>]

Specifies the hostname of the secondary storage HTTPS server.

#### -httpsPort <port>

Specifies the remote port number that the Data Mover delivers the HTTPS request to. If not specified, the Data Mover issues HTTPS requests to port 443 on the secondary storage HTTPS server.

**Note:** Although the **-http\_server** option is used to modify the name of the secondary storage HTTPS server, files that can be converted into a stub by using an HTTPS connection can be brought back online using only HTTPS and not using NFS, CIFS, or even HTTP.

#### -localPort <port>

Specifies the local port number the Data Mover uses to issue HTTPS requests to the web server active on the secondary storage. The <port> specified should be an integer no less than 1024. If not specified, the Data Mover selects a port to issue the HTTPS requests.

**Note:** If you attempt to create multiple HTTPS connections by using identical end points with different attributes such as **-cgi**, **-user**, **-password**, **-localPort**, **-httpsPort**, the connection will fail.

#### [-user <username>]

An optional attribute used to define the username the HTTPS client uses if digest authentication is required by the secondary storage HTTPS server.

#### [-password <password>]

Allows the user to specify the admin password. The password is not recorded in the command log. If the **-password** option is given but no password is specified, the user is prompted interactively.

```
[-timeout <sec>]
```

Specifies the timeout value in seconds. By default, Celerra's HTTPS client waits 30 seconds for a reply from the HTTPS server and then retries the operation once before commencing the failover operation.

**-connection** connection cfs\_name> -delete {-all|<cid> [,<cid>...]}
Removes an existing HTTPS connection between the filesystem and the secondary filesystem. Either all connections can be removed or just the specified <cid> connection can be removed.

```
[-recall_policy {check | no | yes}]
```

Specifies the recall policy for any migrated file during the **-delete**. **check** (default) scans the filesystem for stub files that depend on the connection and fails on the first one. **no** deletes the connection without checking for stub files that depend on the connection, and **yes** migrates the files back to the Celerra Network Server before the connection is removed. If **no** is specified and stub files exist, an I/O error appears when the file is read because the connection no longer exists.

**SEE ALSO** *Using Celerra FileMover*, server\_cifs, server\_http, and server\_nfs.

#### **EXAMPLE #1**

To enable Celerra FileMover on a filesystem, type:

#### \$ fs\_dhsm -modify ufs1 -state enabled

ufs1:

state = enabled
offline attr = on
popup timeout = 0

backup = passthrough

read policy override = none
log file = on
max log size = 10MB

Done

#### Where:

***************************************	
Definition	
Whether Celerra FileMover is enabled or disabled on the filesystem.	
Whether CIFS clients should be notified that a file is migrated.	
Timeout value in seconds, before Windows popup notification is sent to the CIFS client.	
Nature of CIFS network backups.	
Migration method option used to override the read method specified in the stub file.	
Whether FileMover logging is enabled or disabled.	
Maximum size of the log file.	

## **EXAMPLE #2**

To create a CIFS connection for **ufs1** to the secondary filesystem **\winserver2.nasdocs.emc.com\dhsm1** with a specified **administrative account nasdocs.emc.com\Administrator** and local server **dm102-cge0**:

```
$ fs_dhsm -connection ufs1 -create -type cifs -admin
'nasdocs.emc.com\Administrator' -secondary
'\winserver2.nasdocs.emc.com\dhsm1'
-local_server_dm102-cge0
```

```
Enter Password: ******
ufs1:
                  = enabled
state
offline attr
                  = on
popup timeout
                  = 0
backup
                  = passthrough
read policy override = none
log file
                = 10MB
max log size
cid
                  = 0
  type
                     = CIFS
  secondary
                    = \\winserver2.nasdocs.emc.com\dhsm1\
                    = enabled
  read policy override = none
  write policy = full
  local_server
                   = DM102-CGE0.NASDOCS.EMC.COM
  admin
                    = nasdocs.emc.com\Administrator
  wins
```

Done

## Where:

Value	Definition
state	Whether Celerra FileMover is enabled or disabled on the filesystem.
offline attr	Whether CIFS clients should be notified that a file is migrated.
popup timeout	Timeout value, in seconds, before a popup notification is sent to CIFS client.
backup	Nature of CIFS network backups.
read policy override	Migration method option used to override the read method specified in the stub file.
log file	Whether FileMover logging is enabled or disabled.
max log size	Maximum size of the log file.
cid	Connection ID.
type	Type of filesystem. The <b>-list</b> option provides information for a description of the types.
secondary	Hostname or IP address of the remote filesystem.
state	Specifies whether Celerra FileMover is enabled or disabled on the filesystem.
read policy override	Migration method option used to override the read method specified in the stub file.
write policy	Write policy option used to recall data from secondary storage.
local_server	Name of the local CIFS server used to authenticate the CIFS connection.

### **EXAMPLE #3**

To create a CIFS connection for **ufs1** to the secondary filesystem \\winserver2.nasdocs.emc.com\dhsm2 with a specified administrative account nasdocs.emc.com\Administrator, local server dm102-cge0, WINS server, and with the migration method set to full, type:

```
$ fs_dhsm -connection ufs1 -create -type cifs -admin
                    'nasdocs.emc.com\Administrator' -secondary
                    '\\winserver2.nasdocs.emc.com\dhsm1'
                    -local_server dm102-cge0 -wins 172.24.102.25
                    -read policy override full
Enter Password: ******
ufs1:
state
                = enabled
offline attr
                = on
read policy override = none
log file
max log size
                = 10MB
cid
                = 0
  type
                   = \\winserver2.nasdocs.emc.com\dhsm1\
  read policy override = full
  write policy = full
  local server
                  = DM102-CGE0.NASDOCS.EMC.COM
                   = nasdocs.emc.com\Administrator
  admin
                    = 172.24.102.25
  wins
Done
```

EXAMPLE #2 provides a description of command output.

#### **EXAMPLE #4**

To display connection information for **ufs1**, type:

\$ fs\_dhsm -connection ufs1 -info 1

EXAMPLE #2 provides a description of command output.

#### **EXAMPLE #5**

 $log file = on \\ max log size = 10MB$ 

To modify the **read\_policy\_override** setting for connection **0** for **ufs1**, type:

```
$ fs_dhsm -connection ufs1 -modify 0
                       -read policy override passthrough
ufs1:
state
                    = enabled
offline attr
                    = on
popup timeout
                    = 0
backup
                    = passthrough
read policy override = none
log file
                   = on
max log size
                   = 10MB
cid
                    = 0
                      = CIFS
   type
   secondary
                      = \\winserver2.nasdocs.emc.com\dhsm1\
   state
                       = enabled
  read policy override = pass
  write policy
                      = full
   local_server
                      = DM102-CGE0.NASDOCS.EMC.COM
   admin
                      = nasdocs.emc.com\Administrator
   wins
                      = 172.24.102.25
Done
                       EXAMPLE #2 provides a description of command output.
         EXAMPLE #6
                       To modify the Celerra FileMover connection for ufs1, type:
                       $ fs_dhsm -connection ufs1 -modify 0 -nfs_server
                       172.24.102.115 -proto TCP
ufs1:
state
                    = enabled
offline attr
                    = on
popup timeout
                    = 10
backup
                    = offline
read policy override = full
log file
                   = on
max log size
                   = 25MB
cid
                    = 0
                      = NFSV3
   type
   secondary
                       = 172.24.102.115:/export/dhsm1
                       = enabled
  state
  read policy override = full
   write policy
                       = full
   options
                       = useRootCred=true proto=TCP
 cid
                    = 1
                       = CIFS
   type
   secondary
                       = \\winserver2.nasdocs.emc.com\dhsm1\
   state
                       = enabled
   read policy override = none
                 = full
   write policy
   local_server
                      = DM102-CGE0.NASDOCS.EMC.COM
```

= nasdocs.emc.com\Administrator

= 172.24.102.25

admin

wins

```
cid
                     = 2
   type
                        = HTTP
   secondary
                       = http://172.24.102.115/export/dhsm1
   state
                        = enabled
   read policy override = none
   write policy = full
   user
   options
                         = cgi=n
Done
                        EXAMPLE #2 provides a description of command output.
          EXAMPLE #7
                        To create the NFSv3 connection for ufs1 to the secondary filesystem
                        172.24.102.115:/export/dhsm1 with the migration method set to full,
                        the -useRootCred set to true and the protocol set to UDP, type:
                        $ fs_dhsm -connection ufs1 -create -type nfsv3 -secondary
                        172.24.102.115:/export/dhsm1 -read policy override full
                        -useRootCred true
                        -proto UDP
ufs1:
state
                     = enabled
\begin{array}{lll} \text{offline attr} & = & \text{on} \\ \text{popup timeout} & = & 0 \\ \text{backup} & = & \text{passthrough} \end{array}
read policy override = none
log file
            = on
max log size
                    = 10MB
 cid
                     = 0
  type
                       = CIFS
  secondary
state
                       = \\winserver2.nasdocs.emc.com\dhsm1\
   state
                        = enabled
   read policy override = pass
   write policy = full
   local_server
                       = DM102-CGE0.NASDOCS.EMC.COM
   admin
                        = nasdocs.emc.com\Administrator
   wins
                        = 172.24.102.25
                    = 1
 cid
   type
                       = NFSV3
   secondary
                       = 172.24.102.115:/export/dhsm1
                       = enabled
   state
   read policy override = full
   write policy = full
   options
                        = useRootCred=true proto=UDP
```

**EXAMPLE #8** To modify the Celerra FileMover connection for **ufs1**, type:

78

Done

```
$ fs_dhsm -connection ufs1 -modify 1 -proto TCP
ufs1:
                    = enabled
state
offline attr
                    = on
popup timeout
                   = 0
backup
                   = passthrough
read policy override = none
log file
max log size
                  = 10MB
 cid
                   = 0
                      = CIFS
  type
  secondary
                     = \\winserver2.nasdocs.emc.com\dhsm1\
  state
                     = enabled
  read policy override = pass
  write policy = full
  local_server
                     = DM102-CGE0.NASDOCS.EMC.COM
  admin
                      = nasdocs.emc.com\Administrator
                      = 172.24.102.25
  wins
 cid
                   = 1
  type
                      = NFSV3
  secondary
                      = 172.24.102.115:/export/dhsm1
  state
                     = enabled
  read policy override = full
  write policy = full
  options
                     = useRootCred=true proto=TCP
Done
```

# **EXAMPLE #9** To display Celerra FileMover connection information for **ufs1**, type:

#### \$ fs\_dhsm -info ufs1

EXAMPLE #1 provides a description of command output.

# **EXAMPLE #10** To list Celerra FileMover connections, type:

# \$ fs dhsm -connection ufs1 -list

## **EXAMPLE #11** To modify the Celerra FileMover connection for **ufs1**, type:

```
$ fs_dhsm -modify ufs1 -popup_timeout 10 -backup offline
                       -log on -max log size 25 -offline_attr on
                       -read_policy_override full
ufs1:
state
                    = enabled
offline attr
                   = on
                   = 10
popup timeout
                   = offline
backup
read policy override = full
log file
                    = on
max log size
                   = 25MB
cid
                   = 0
                       = CIFS
   type
  secondary
                       = \\winserver2.nasdocs.emc.com\dhsm1\
                      = enabled
  state
  read policy override = pass
  write policy
                      = full
  local_server
                      = DM102-CGE0.NASDOCS.EMC.COM
  admin
                      = nasdocs.emc.com\Administrator
                       = 172.24.102.25
  wins
cid
                   = 1
                       = NFSV3
  type
                       = 172.24.102.115:/export/dhsm1
  secondary
                       = enabled
  state
  read policy override = full
                      = full
  write policy
  options
                       = useRootCred=true proto=TCP
Done
                       EXAMPLE #2 provides a description of command output.
        EXAMPLE #12
                       To modify the state of the Celerra FileMover connection 0 for ufs1,
                       type:
```

\$ fs\_dhsm -connection ufs1 -modify 0 -state disabled

```
ufs1:
                    = enabled
state
offline attr
                   = on
popup timeout
                  = 10
backup
                    = offline
read policy override = full
log file
                  = on
                   = 25MB
max log size
cid
                   = 0
  type
                      = CIFS
  secondary
                      = \\winserver2.nasdocs.emc.com\dhsm1\
                      = disabled
  state
  read policy override = pass
  write policy = full
                     = DM102-CGE0.NASDOCS.EMC.COM
  local_server
  admin
                      = nasdocs.emc.com\Administrator
```

**EXAMPLE #2** provides a description of command output.

# **EXAMPLE #13** To modify the state of the Celerra FileMover connection **1** for **ufs1**, type:

\$ fs\_dhsm -connection ufs1 -modify 1 -state recallonly

```
ufs1:
state
                   = enabled
offline attr
                   = on
popup timeout
                  = 10
                   = offline
backup
read policy override = full
log file
                  = on
max log size
                  = 25MB
cid
                  = 0
                     = CIFS
  type
  secondary
                      = \\winserver2.nasdocs.emc.com\dhsm1\
  state
                      = enabled
  read policy override = pass
  write policy = full
  local_server
                    = DM102-CGE0.NASDOCS.EMC.COM
  admin
                     = nasdocs.emc.com\Administrator
                     = 172.24.102.25
  wins
 cid
                   = 1
  type
                    = NFSV3
  secondary
                     = 172.24.102.115:/export/dhsm1
  state
                     = recallonly
  read policy override = full
  write policy = full
  options
                      = useRootCred=true proto=TCP
```

Done

EXAMPLE #2 provides a description of command output.

#### **EXAMPLE #14**

To delete the Celerra FileMover connections **1** and **2** for **ufs1**, and specify the recall policy for any migrated files during the delete, type:

\$ fs\_dhsm -connection ufs1 -delete 0,1 -recall\_policy
no

ufs1:

```
state
                    = enabled
offline attr
                   = on
popup timeout backup
                    = 10
                   = offline
read policy override = full
log file = on
max log size
                   = 25MB
Done
                       EXAMPLE #2 provides a description of command output.
        EXAMPLE #15
                       To change the state of the Celerra FileMover connection for ufs1 to
                       disabled, type:
                       $ fs_dhsm -modify ufs1 -state disabled
ufs1:
state
                    = disabled
offline attr
                    = on
popup timeout = 00
backup = offline
read policy override = full
log file
                    = on
                   = 25MB
max log size
Done
                       EXAMPLE #1 provides a description of command output.
        EXAMPLE #16
                       To create an HTTP connection for ufs1 to the secondary filesystem
                       /export/dhsm1 on the web server http://172.24.102.115 which has
                       direct access to the storage, type:
                       $ fs dhsm -connection ufs1 -create -type http -secondary
                       http://172.24.102.115/export/dhsm1 -cgi n
ufs1:
state
                    = enabled
offline attr
popup timeout
                   = on
                    = 10
backup
                    = offline
read policy override = full
log file = on
\max \log \text{size} = 25\text{MB}
cid
                    = 2
   type
                       = HTTP
  secondary
                      = http://172.24.102.115/export/dhsm1
                       = enabled
   read policy override = none
```

EXAMPLE #2 provides a description of command output.

#### **EXAMPLE #17**

To create an HTTP connection for **ufs1** to the secondary filesystem using CGI connections to access migrated file data using a CGI application, type:

\$ fs\_dhsm -connection ufs1 -create -type http -secondary
http://www.nasdocs.emc.com/cgi-bin/access.sh

```
ufs1:
                   = enabled
state
offline attr
                   = on
popup timeout
                  = 0
backup
                   = passthrough
read policy override = none
log file
                  = on
max log size
                  = 10MB
cid
                  = 0
  type
  secondary
                   = http://www.nasdocs.emc.com/cgi-bin/access.sh
  state
                     = enabled
  read policy override = none
  write policy
                     = full
  user
  options
Done
```

EXAMPLE #2 provides a description of command output.

#### **EXAMPLE #18**

To create an HTTPS connection for **server2\_fs1** on the web server **https://int16543** with read\_policy\_override set to **full**, type:

\$ fs\_dhsm -connection server2\_fs1 -create -type https
-secondary https://int16543 -read\_policy\_override full
-cgi n

```
server2_fs1:
state
                   = enabled
offline attr
                   = on
popup timeout
                   = 0
backup
                   = passthrough
read policy override = passthrough
log file
                   = on
max log size
                  = 10MB
 cid
                   = 0
  type
                     = HTTPS
  secondary
                     = https://int16543
                     = enabled
  read policy override = full
  write policy = full
  user
  options
                      =
```

EXAMPLE #2 provides a description of command output.

#### **EXAMPLE #19**

To create an HTTPS connection for **ufs1** to the secondary filesystem using CGI connections to access migrated file data using a CGI application, type:

\$ fs\_dhsm -connection ufs1 -create -type https -secondary
https://www.nasdocs.emc.com/cgi-bin/access.sh

```
ufs1:
state = enabled
offline attr = on
popup timeout = 0
backup = passthrough
read policy override = none
log file = on
max log size = 10MB
cid = 0
type = HTTPS
secondary = https://www.nasdocs.emc.com/cgi-bin/access.sh
state = enabled
read policy override = none
write policy = full
user =
options =
Done
```

EXAMPLE #2 provides a description of command output.

#### **EXAMPLE #20**

To create an HTTPS connection on httpsPort 443 for server2\_ufs1 on the web server https://int16543 with read\_policy\_override set to passthrough, type:

\$ fs\_dhsm -connection server2\_fs1 -create -type https -secondary https://int16543 -read\_policy\_override passthrough -httpsPort 443 -cgi n

```
server2_fs1:
state
                   = enabled
offline attr
                  = on
popup timeout backup
                  = 0
                  = passthrough
read policy override = passthrough
log file
                  = on
max log size
                  = 10MB
 cid
                  = 1
  type
                     = HTTPS
  secondary = https://int16543
state = enabled
  read policy override = pass
  write policy = full
  user
  options
```

EXAMPLE #2 provides a description of command output.

#### **EXAMPLE #21**

To create an HTTPS connection on **localPort 80** for **server2\_ufs1** on the web server **https://int16543** with read\_policy\_override set to **passthrough**, type:

\$ fs\_dhsm -connection server2\_fs1 -create -type https -secondary https://int16543 -read\_policy\_override passthrough -localPort 80 -cgi n

```
server2_fs1:
state
                  = enabled
offline attr
                  = on
popup timeout
                  = 0
backup
                 = passthrough
read policy override = passthrough
log file
                  = on
max log size
                 = 10MB
cid
                  = 0
  type
                    = HTTPS
  secondary
                    = https://int16543
                    = enabled
  state
  read policy override = pass
  write policy = full
  user
  options
```

Done

server2\_fs1:

EXAMPLE #2 provides a description of command output.

#### **EXAMPLE #22**

To create an HTTPS connection on httpsPort 443 for server2\_ufs1 on the web server https://int16543 with a specified user dhsm\_user, type:

\$ fs\_dhsm -connection server2\_fs1 -create -type https
-secondary https://int16543 -read\_policy\_override full
-httpsPort 443 -user dhsm\_user -password dhsm\_user -cgi n

```
= enabled
state
offline attr
                   = on
popup timeout
                   = 0
backup
                   = passthrough
read policy override = passthrough
log file
                  = on
max log size
                   = 10MB
cid
                   = 1
                     = HTTPS
  type
  secondary
                     = https://int16543
                   = enabled
  state
  read policy override = full
```

```
write policy
                          = full
                          = dhsm_user
   user
   options
Done
                          EXAMPLE #2 provides a description of command output.
         EXAMPLE #23
                          To modify the read_policy_override setting for connection 1 from
                          server2_fs1, type:
                          $ fs_dhsm -connection server2_fs1 -modify 1
                          -read policy override passthrough
server2_fs1:
state
                      = enabled
offline attr
popup timeout
backup
                      = on
                      = 0
                      = passthrough
read policy override = passthrough
log file = on
\max \log \text{size} = 10\text{MB}
  id
type
secondary
 cid
                      = 1
                        = HTTPS
                       = https://int16543
   state
                         = enabled
   read policy override = pass
   write policy = full
                         = dhsm_user
   user
   options
Done
                          EXAMPLE #2 provides a description of command output.
         EXAMPLE #24
                          To delete the Celerra FileMover connection 0 for ufs1, type:
                          $ fs_dhsm -connection ufs1 -delete 0
ufs1:
state
                      = enabled
offline attr
 \begin{array}{lll} \text{offline attr} & = & \text{on} \\ \text{popup timeout} & = & 0 \\ \text{backup} & = & \text{passthrough} \\ \end{array} 
                       = on
read policy override = none
log file
             = on
max log size
                      = 10MB
Done
                          EXAMPLE #1 provides a description of command output.
```

# fs\_group

Creates a filesystem group from the specified filesystems or a single filesystem.

#### **SYNOPSIS**

```
fs_group
  -list
| -delete <fs_group_name>
| -info {<fs_group_name>|id=<fs_group_id>}
| [-name <name>] -create {<fs_name>,...}
| -xtend <fs_group_name> {<fs_name>,...}
| -shrink <fs_group_name> {<fs_name>,...}
```

#### DESCRIPTION

The **fs\_group** command combines filesystems to be acted upon simultaneously as a single group for TimeFinder/FS.

#### **OPTIONS**

#### -list

Displays a listing of all filesystem groups.

**Note:** The ID of the object is an integer and is assigned automatically. The name of a filesystem may be truncated if it is too long for the display. To display the full name, use the **-info** option with a filesystem ID.

```
-delete <fs_group_name>
```

Deletes the filesystem group configuration. Individual filesystems are not deleted.

```
-info {<fs_group_name>|id=<fs_group_id>}
```

Displays information about a filesystem group, either by name or group ID.

```
[-name <name>] -create {<fs_name>,...}
```

Creates a filesystem group from the specified filesystems. If a name is not specified, one is assigned by default.

```
-xtend <fs_group_name> {<fs_name>,...}
```

Adds the specified filesystems or group to a filesystem group.

```
-shrink <fs_group_name> {<fs_name>,...}
```

Removes the specified filesystems or group from a filesystem group. Individual filesystems are not deleted.

#### **SEE ALSO**

Managing Celerra Volumes and File Systems Manually and UsingTimeFinder/FS, NearCopy, and FarCopy with Celerra, fs\_timefinder, and nas\_fs.

# STORAGE SYSTEM OUTPUT

The number associated with the storage device is dependent on the attached storage system. CLARiiON storage systems display a prefix of APM before a set of integers, for example, APM00033900124-0019. Symmetrix storage storage systems appear as 002804000190-003C.

### **EXAMPLE #1**

To create a filesystem group named, ufsg1, and add ufs1, type:

\$ fs\_group -name ufsg1 -create ufs1

disks = d3, d4, d5, d6

#### Where:

Value	Definition
id	ID of the group that is automatically assigned.
name	Name assigned to the group.
acl	Access control value for the group.
in_use	Whether a filesystem is used by a group.
type	Type of filesystem.
fs_set	Filesystems that are part of the group.
pool	Storage pool given to the filesystem group.
stor_devs	Storage system devices associated with the group.
disks	Disks on which the metavolume resides.

# **EXAMPLE #2** To list all filesystem groups, type:

```
$ fs_group -list
```

id	name	acl	in_use	type	member_of	fs_set
20	ufsq1	0	n	100		18

### Where:

Value	Definition
member_of	Groups which the filesystem group belong to.

## **EXAMPLE #3** To display information for the filesystem group, **ufsg1**, type:

```
$ fs_group -info ufsg1
```

```
= 22
id
name
          = ufsg1
acl
          = 0
          = False
in_use
type
          = group
fs_set
          = ufs1
pool
stor_devs =
000187940268 - 0006,000187940268 - 0007,000187940268 - 0008,000187940268 - 0009
disks
          = d3, d4, d5, d6
```

EXAMPLE #1 provides a description of command output.

## **EXAMPLE #4** To add filesystem, **ufs2**, to the filesystem group, **ufsg1**, type:

```
$ fs_group -xtend ufsg1 ufs2
```

```
id
       = 22
name
       = ufsq1
       = 0
acl
in use
       = False
type
       = group
fs_set
       = ufs1,ufs2
pool
stor_devs =
0268-000A,000187940268-000B,000187940268-000C,000187940268-000D
       = d3, d4, d5, d6, d7, d8, d9, d10
disks
```

EXAMPLE #1 provides a description of command output.

# **EXAMPLE #5** To remove filesystem, **ufs2**, from the filesystem group, **ufsg1**, type:

#### \$ fs\_group -shrink ufsg1 ufs2

```
id
          = 22
name
          = ufsg1
          = 0
acl
         = False
in_use
type
          = group
         = ufs1
fs_set
pool
stor devs =
000187940268 - 0006,000187940268 - 0007,000187940268 - 0008,000187940268 - 0009
          = d3, d4, d5, d6
disks
```

EXAMPLE #1 provides a description of command output.

# **EXAMPLE #6**

To delete filesystem group, ufsg1, type:

\$ fs\_group -delete ufsg1

EXAMPLE #1 provides a description of command output.

# fs rdf

Manages the Remote Data Facility (RDF) functionality for a filesystem residing on RDF drives.

#### **SYNOPSIS**

```
fs_rdf {<fs_name>|id=<fs_id>}
  -Mirror {on|off|refresh}
| -Restore [-Force]
| -info
```

**Note:** RDF is supported only on a Celerra Network Server attached to a Symmetrix.

#### DESCRIPTION

The **fs\_rdf** command turns mirroring on and off for an RDF filesystem and displays information about RDF relationships.

#### **OPTIONS**

```
-Mirror {on|off|refresh}
```

The **on** option resumes the link between the RDF drives of a filesystem thereby enabling mirroring for the RDF filesystem. The **off** option halts mirroring between the filesystems, and the **refresh** option does an immediate mirror **on** then **off** which refreshes the filesystem image.

## -Restore [-Force]

Restores a filesystem from the R2 side (remote) when remote TimeFinder/FS FarCopy is used. The **-Restore** can only be executed on the R1 side. The **-Force** option must be used when restoring a filesystem with enabled.

#### -info

Displays information about RDF relationships.

#### SEE ALSO

Using SRDF/S with Celerra for Disaster Recovery, Using TimeFinder/FS, NearCopy, and FarCopy with Celerra, and Using File-Level Retention on Celerra.

#### **EXAMPLE #1**

To turn on mirroring for **ufs1\_snap1** from the R1 Control Station, type:

```
$ fs_rdf ufs1_snap1 -Mirror on
```

```
volume = v168
pool
rw_servers=
ro_servers=
rw_vdms =
ro vdms
backup_of = ufs1 Fri Apr 23 16:29:23 EDT 2004
stor_devs =
disks
         = rootd33, rootd34, rootd35, rootd36
         RDF Information:
remote_symid
                          = 002804000218
remote_sym_devname
ra_group_number
                          = 2
dev_rdf_type
dev_ra_status
dev_link_status
                          = R1
                          = READY
                          = READY
rdf_mode
                          = SYNCHRONOUS
rdf_pair_state
                          = SYNCINPROG
rdf_domino
                          = DISABLED
adaptive_copy
adaptive_copy_skew
num_r1_invalid_tracks
num_r2_invalid_tracks
                          = DISABLED
                          = 65535
                          = 0
                         = 736440
dev_rdf_state
                          = READY
remote_dev_rdf_state = WRITE_DISABLED
rdf_status
                          = 0
link_domino
                          = DISABLED
prevent_auto_link_recovery = DISABLED
link_config
suspend_state
                          = NA
consistency_state
                          = DISABLED
adaptive_copy_wp_state = NA
prevent_ra_online_upon_pwron = ENABLED
```

#### Where:

Value	Definition
id	ID of a filesystem that is assigned automatically.
name	Name assigned to a filesystem.
acl	Access control value for a filesystem.
in_use	Whether a filesystem is registered into the mount table.
type	Type of filesystem. nas_fs provides a description of the types.
volume	Volume on which a filesystem resides.
pool	Storage pool for the filesystem.
rw_servers	Servers with read-write access to a filesystem.
ro_servers	Servers with read-only access to a filesystem.
rw_vdms	VDM servers with read-write access to a filesystem.

ro_vdms	VDM servers with read-only access to a filesystem.	
backup_of	The remote RDF filesystem.	
stor_devs	The storage system devices associated with a filesystem.	
disks	The disks on which the metavolume resides.	
remote_symid	The serial number of the storage system containing the target volume.	
remote_sym_devname	The storage system device name of the remote device in an RDF pair.	
ra_group_number	The RA group number (1-n).	
dev_rdf_type	The type of RDF device. Possible values are: R1 and R2.	
dev_ra_status	RA status. Possible values are: READY, NOT_READY, WRITE_DISABLED, STATUS_NA, STATUS_MIXED.	
dev_link_status	Link status. Possible values are: READY, NOT_READY, WRITE_DISABLED, NA, MIXED.	
rdf_mode	The RDF mode. Possible values are: SYNCHRONOUS, SEMI_SYNCHRONOUS, ADAPTIVE_COPY, MIXED.	
rdf_pair_state	The composite state of the RDF pair. Possible values are: INVALID, SYNCINPROG, SYNCHRONIZED, SPLIT, SUSPENDED, FAILED_OVER, PARTITIONED, R1_UPDATED, R1_UPDINPROG, MIXED.	
rdf_domino	The RDF device domino. Possible values are: ENABLED, DISABLED, MIXED.	
adaptive_copy	Possible values are: DISABLED, WP_MODE, DISK_MODE, MIXED.	
adaptive_copy_skew	Number of invalid tracks when in Adaptive copy mode.	
num_r1_invalid_tracks	Number of invalid tracks on the source (R1) device.	
num_r2_invalid_tracks	Number of invalid tracks on the target (R2) device.	
dev_rdf_state	Specifies the composite RDF state of the RDF device. Possible values are: READY, NOT_READY, WRITE_DISABLED, NA, MIXED.	
remote_dev_rdf_state	Specifies the composite RDF state of the remote RDF device. Possible values are: READY, NOT_READY, WRITE_DISABLED, NA, MIXED.	
rdf_status	Specifies the RDF status of the device. Possible values are:	
	READY, NOT_READY, WRITE_DISABLED, NA, MIXED.	
link_domino	RDF link domino. Possible values are: ENABLED, DISABLED.	
<pre>prevent_auto_link_reco very</pre>	When enabled, prevents the automatic resumption of data copy across the RDF links as soon as the links have recovered. Possible values are: ENABLED, DISABLED.	
link_config	Possible values are: CONFIG_ESCON, CONFIG_T3.	
suspend_state	Specifies the status of R1 devices in a consistency group. Possible states are: NA, OFFLINE, OFFLINE_PEND, ONLINE_MIXED.	
consistency_state	Specifies state of an R1 device related to consistency groups. Possible states are: ENABLED, DISABLED.	
adaptive_copy_wp_state	Specifies state of the adaptive copy mode. Possible states are: NA, OFFLINE, OFFLINE_PEND, ONLINE_MIXED.	
prevent_ra_online_upon _pwron	Specifies the state of the RA director coming online after power on. Possible states are: ENABLED, DISABLED.	

# **EXAMPLE #2** To display RDF-related information for **ufs1\_snap1** from the R2 Control Station, type:

```
$ fs_rdf ufs1_snap1 -info
id
         = 20
         = ufs1_snap1
name
acl
         = 0
in_use = False
type = uxfs
volume = v168
pool
rw servers=
ro_servers=
rw_vdms
ro_vdms
backup_of = ufs1 Fri Apr 23 16:29:23 EDT 2004
disks
          = rootd33, rootd34, rootd35, rootd36
           RDF Information:
remote_symid
                               = 002804000218
remote_sym_devname
ra_group_number
                              = 2
dev_rdf_type
dev_ra_status
                             = R1
                             = READY
dev_link_status
rdf_mode
                             = READY
rdf_mode
rdf_pair_state = SYNCINE
rdf_domino = DISABLE
adaptive_copy = DISABLE
adaptive_copy_skew = 65535
num_r1_invalid_tracks = 0
num_r2_invalid_tracks = 696030
= READY
                             = SYNCHRONOUS
                             = SYNCINPROG
                             = DISABLED
                             = DISABLED
dev_rdf_state = READY
remote_dev_rdf_state = WRITE_DISABLED
rdf_status
                             = 0
link_domino
                             = DISABLED
prevent_auto_link_recovery = DISABLED
link_config
                             =
suspend_state
                             = NA
consistency_state
                             = DISABLED
adaptive_copy_wp_state = NA
prevent_ra_online_upon_pwron = ENABLED
```

**EXAMPLE** #1 provides a description of command output.

# **EXAMPLE #3** To turn the mirroring off for **ufs1\_snap1** on the R1 Control Station, type:

```
$ fs_rdf ufs1_snap1 -Mirror off
remainder(MB) = 20548..17200..13110..8992..4870..746 0
   = 20
        = ufs1_snap1
name
         = 0
acl
in_use = False
type
        = uxfs
        = v168
volume
pool
rw_servers=
ro_servers=
rw_vdms
ro_vdms
backup_of = ufs1 Fri Apr 23 16:29:23 EDT 2004
stor devs =
002804000190 - 0052,002804000190 - 0053,002804000190 - 0054,002804000190 - 0055
disks
         = rootd33, rootd34, rootd35, rootd36
         RDF Information:
                            = 002804000218
remote_symid
remote_sym_devname
ra_group_number
                           = 2
dev_rdf_type
                           = R1
dev_ra_status
                            = READY
dev_link_status
                           = NOT_READY
rdf_mode
                           = SYNCHRONOUS
rdf_pair_state
                            = SUSPENDED
rdf_domino
                           = DISABLED
adaptive_copy
                           = DISABLED
adaptive_copy_skew
                            = 65535
num_r1_invalid_tracks
                           = 0
num_r2_invalid_tracks
                           = 0
dev_rdf_state
                            = READY
remote_dev_rdf_state
                          = WRITE_DISABLED
rdf_status
                           = 0
link domino
                            = DISABLED
prevent_auto_link_recovery = DISABLED
link_config
suspend_state
                            = OFFLINE
consistency_state
                           = DISABLED
adaptive_copy_wp_state
                          = NA
prevent_ra_online_upon_pwron = ENABLED
```

EXAMPLE #1 provides a description of command output.

**EXAMPLE #4** 

type: \$ fs\_rdf ufs1\_snap1 -Mirror refresh remainder(MB) = 1 0id = 20 name = ufs1\_snap1 acl = 0 in\_use = False = uxfs type volume = v168pool rw\_servers= ro\_servers= rw\_vdms = ro\_vdms backup\_of = ufs1 Fri Apr 23 16:29:23 EDT 2004 stor\_devs = 002804000190 - 0052, 002804000190 - 0053, 002804000190 - 0054, 002804000190 - 0055disks = rootd33,rootd34,rootd35,rootd36 RDF Information: = 002804000218remote\_symid remote\_sym\_devname = 2ra\_group\_number dev\_rdf\_type
dev\_ra\_status = R1= READY dev\_link\_status = NOT\_READY rdf mode = SYNCHRONOUS rdf\_pair\_state
rdf\_domino
adaptive\_copy = SUSPENDED = DISABLED adaptive\_copy adaptive\_copy\_skew num\_r1\_invalid\_tracks num\_r2\_invalid\_tracks = DISABLED = 65535 = 0 = 0 dev\_rdf\_state = READY remote\_dev\_rdf\_state = WRITE\_DISABLED rdf status = 0link\_domino = DISABLED prevent\_auto\_link\_recovery = DISABLED suspend\_state = OFFLINE consistency\_state = DISABLED adaptive\_copy\_wp\_state = NA

EXAMPLE #1 provides a description of command output.

To perform a mirror refresh for **ufs1\_snap1** on the R1 Control Station,

prevent\_ra\_online\_upon\_pwron = ENABLED

# **EXAMPLE #5** To restore the filesystem **ufs1\_snap1** from the R1 Control Station, type:

```
$ /nas/sbin/rootfs_rdf ufs1_snap1 -Restore
remainder(MB) = 1 0
id
         = 20
        = ufs1_snap1
name
acl
        = 0
in_use = False
       = uxfs
type
volume = v168
pool
rw_servers=
ro_servers=
rw_vdms
ro_vdms
backup_of = ufs1 Fri Apr 23 16:29:23 EDT 2004
stor_devs =
002804000190 - 0052,002804000190 - 0053,002804000190 - 0054,002804000190 - 0055
disks
         = rootd33, rootd34, rootd35, rootd36
         RDF Information:
remote_symid
                           = 002804000218
remote_sym_devname
                           = 2
ra_group_number
dev_rdf_type
                           = R1
dev_ra_status
                          = READY
dev_link_status
                          = READY
rdf mode
                           = SYNCHRONOUS
rdf_pair_state
                          = SYNCHRONIZED
rdf_domino
                          = DISABLED
adaptive_copy
                          = DISABLED
                        = 65535
adaptive_copy_skew
num_r1_invalid_tracks
                          = 0
num_r2_invalid_tracks
                         = 0
dev_rdf_state
                          = READY
remote_dev_rdf_state
                         = WRITE_DISABLED
rdf status
                          = 0
link_domino
                          = DISABLED
prevent_auto_link_recovery = DISABLED
link_config
suspend_state
                          = NA
consistency_state
                          = DISABLED
adaptive_copy_wp_state
prevent_ra_online_upon_pwron = ENABLED
```

EXAMPLE #1 provides a description of command output.

# fs\_timefinder

Manages the TimeFinder/FS functionality for the specified filesystem or filesystem group.

#### **SYNOPSIS**

```
fs_timefinder {<fs_name>|id=<fs_id>}
    -Mirror {on|off|refresh [-Force]}
| [-name <name>] -Snapshot [-volume <volume_name>][-option <options>]
| -Restore [-Force][-option <options>]
```

**Note:** TimeFinder/FS is only supported on a Celerra Network Server attached to a Symmetrix.

#### **DESCRIPTION**

The **fs\_timefinder** command creates a copy of a filesystem or filesystem group that can be placed into a mirrored mode with its original filesystem. The Symmetrix must already have Business Continuance Volumes (BCV) configured to the same size as the volumes on the Celerra Network Server. Once the copy of the filesystem has been made, it can be mounted on any Data Mover.

### **OPTIONS**

```
-Mirror {on off refresh [-Force]}
```

on places the unmounted filesystem copy, created using the
 -Snapshot option, into mirrored mode with its original filesystem.
 The filesystem copy is frozen and remains unavailable to users until mirrored mode is turned off.

The **refresh** option initiates an immediate **-Mirror on** then **off** for the unmounted filesystem copy, thereby refreshing the filesystem copy.

The filesystem copy should not be mounted read-write when placed into mirrored mode or when refreshed. If the filesystem copy is mounted read-write, the **-Force** option can be used to force a refresh if the metavolume is an STD type. The **-Force** option requires **root** command and must be executed by using <code>/nas/sbin/rootfs\_timefinder</code>.



#### **CAUTION**

Performing a mirror refresh may be time consuming, relative to the amount of data that has changed in the filesystem.

```
[-name <name>] -Snapshot
```

Creates a copy of a filesystem and assigns an optional name to the

filesystem copy. If a name is not specified, one is assigned by default. If no options are provided, a name and metavolume are automatically assigned. Use nas\_fs to delete the copy of the filesystem.



#### **CAUTION**

Creating a copy by using -Snapshot may be time consuming, relative to the size of a filesystem.

[-volume <volume\_name>]

Assigns a metavolume to a filesystem copy. The metavolume must be created by using the **nas\_volume -Clone** command prior to executing this option. The metavolume must be a BCV type and have the same characteristics as the metavolume of the original filesystem.

[-option <options>]

Specifies the following comma-separated options:

mirror=on

Leaves the filesystem copy in mirrored mode.

disktype=<disktype>

For systems with both local and R1BCVs, specifies the type of volume to use when creating a snapshot. In a TimeFinder/FS FarCopy configuration, use disktype=R1BCV for creating a snapshot of the PFS on the local Celerra. For creating a snapshot of an imported FarCopy snapshot on the remote Celerra, use disktype=STD.

By default, the system uses the first available R1BCV or BCV, or R1STD or STD device.

Use the **disktype**= option to designate which to use if there are R1 devices in your configuration.

#### -Restore

Restores a filesystem to its original location by using the unmounted filesystem copy created with the **-Snapshot** option. The original filesystem must not have any associated SnapSure checkpoints.



### **CAUTION**

Restoring a filesystem may be time consuming, relative to the amount of data that has changed in the filesystem.

#### [-Force]

Forces a restore of a filesystem copy that is mounted on the metavolume as read-only, or if the volume is an STD type.

[-option <options>]

Specifies the following comma-separated options:

#### mirror=on

Places the filesystem copy in mirrored mode.

SEE ALSO Using Time Finder/FS, Near Copy, and Far Copy with Celerra, fs\_ckpt, fs\_group, and nas\_fs.

### **EXAMPLE #1** To create a TimeFinder/FS copy of the PFS, type:

### \$ fs\_timefinder ufs1 -Snapshot

```
operation in progress (not interruptible)...
remainder(MB) =
43688..37205..31142..24933..18649..12608..7115..4991..4129..3281..2457..1653..81
operation in progress (not interruptible)...id
                                                  = 18
name = ufs1
         = 0
acl
in_use
         = True
type = uxfs
worm
        = off
volume
         = mtv1
loog
         =
rw_servers= server_2
ro_servers=
rw_vdms =
ro_vdms =
backups = ufs1_snap1
stor_devs =
000187940268-0006,000187940268-0007,000187940268-0008,000187940268-0009
disks = d3, d4, d5, d6
 disk=d3 stor_dev=000187940268-0006
                                           addr=c0t110-48-0
                                                               server=server_2
 disk=d3 stor_dev=000187940268-0006 addr=c16t110-33-0 server=server_2
disk=d4 stor_dev=000187940268-0007 addr=c0t1l1-48-0 server=server_2 disk=d4 stor_dev=000187940268-0007 addr=c16t1l1-33-0 server=server_2
 disk=d5 stor_dev=000187940268-0008 addr=c0t112-48-0 server=server_2
disk=d5 stor_dev=000187940268-0008 addr=c16t112-33-0 server=server_2 disk=d6 stor_dev=000187940268-0009 addr=c0t113-48-0 server=server_2
 disk=d6 stor_dev=000187940268-0009 addr=c16t113-33-0 server=server_2
id
          = 19
        = ufs1_snap1
name
acl
         = 0
in_use = False
type = uxfs
worm
        = off
```

```
volume = v456
pool
rw_servers=
ro_servers=
rw_vdms
ro vdms
backup_of = ufs1 Thu Oct 28 14:13:30 EDT 2004
stor_devs =
000187940268 - 0180, 000187940268 - 0181, 000187940268 - 0182, 000187940268 - 0183
          = rootd378, rootd379, rootd380, rootd381
```

#### Where:

backups = ufs1\_snap1

Value	Definition
name	Name assigned to the filesystem.
acl	Access control value for a filesystem. nas_acl provides information.
in_use	If a filesystem is registered into the mount table of a Data Mover.
type	Type of filesystem. <b>-list</b> provides a description of the types.
worm	Whether is enabled.
volume	Volume on which the filesystem resides.
pool	Storage pool for the filesystem.
rw_servers	Servers with read-write access to a filesystem.
ro_servers	Servers with read-only access to a filesystem.
rw_vdms	VDM servers with read-write access to a filesystem.
ro_vdms	VDM servers with read-only access to a filesystem.
backups	Name of associated backups.
backup_of	Filesystem that the filesystem copy is made from.
stor_devs	Storage system devices associated with a filesystem. The storage device output is the result of the Symmetrix hardware storage system.
disks	Disks on which the metavolume resides.

# **EXAMPLE #2**

To create a TimeFinder/FS copy of the PFS, ufs1, and leave a filesystem copy in mirrored mode, type:

```
$ fs_timefinder ufs1 -Snapshot -option mirror=on
operation in progress (not interruptible)...id = 18
name = ufs1
        = 0
acl
in_use = True
type
       = uxfs
worm
       = off
volume = mtv1
pool
       =
rw_servers= server_2
ro_servers=
rw_vdms =
ro_vdms =
```

```
stor_devs =
000187940268 - 0006, 000187940268 - 0007, 000187940268 - 0008, 000187940268 - 0009
        = d3, d4, d5, d6
 disk=d3
            stor_dev=000187940268-0006
                                        addr=c0t110-48-0
                                                            server=server_2
 disk=d3
           stor_dev=000187940268-0006
                                        addr=c16t110-33-0
                                                           server=server_2
 disk=d4
           stor dev=000187940268-0007
                                        addr=c0t1l1-48-0
                                                           server=server 2
 disk=d4
           stor_dev=000187940268-0007
                                        addr=c16t111-33-0
                                                           server=server 2
 disk=d5
           stor_dev=000187940268-0008
                                       addr=c0t112-48-0
                                                          server=server_2
 disk=d5
           stor dev=000187940268-0008 addr=c16t112-33-0 server=server 2
 disk=d6
           stor_dev=000187940268-0009 addr=c0t113-48-0
                                                           server=server_2
 disk=d6
           stor_dev=000187940268-0009
                                      addr=c16t113-33-0 server=server_2
id
         = 19
name
         = ufs1_snap1
acl
         = 0
in_use = False
         = mirrorfs
type
         = off
worm
volume
       = v456
loog
rw_servers=
ro_servers=
rw_vdms
ro vdms
backup_of = ufs1 Thu Oct 28 14:19:03 EDT 2004
remainder = 0 MB (0%)
stor devs =
000187940268-0180,000187940268-0181,000187940268-0182,000187940268-0183
disks
         = rootd378, rootd379, rootd380, rootd381
```

# **EXAMPLE #3** To turn mirroring off for a filesystem copy, **ufs1\_snap1**, type:

\$ fs\_timefinder ufs1\_snap1 -Mirror off

```
operation in progress (not interruptible)...
remainder(MB) = 0
operation in progress (not interruptible)...id
                                                    = 18
name
         = ufs1
acl
         = 0
in_use = True
         = uxfs
type
        = off
WOrm
volume = mtv1
loog
rw_servers= server_2
ro_servers=
rw_vdms
ro_vdms
         = ufs1_snap1
backups
stor_devs =
000187940268-0006,000187940268-0007,000187940268-0008,000187940268-0009
```

```
disks
          = d3, d4, d5, d6
 disk=d3
            stor_dev=000187940268-0006
                                          addr=c0t110-48-0
                                                               server=server_2
 disk=d3
            stor_dev=000187940268-0006
                                          addr=c16t110-33-0
                                                               server=server_2
 disk=d4
            stor_dev=000187940268-0007
                                          addr=c0t111-48-0
                                                               server=server_2
 disk=d4
            stor_dev=000187940268-0007
                                          addr=c16t111-33-0
                                                               server=server_2
 disk=d5
            stor dev=000187940268-0008
                                          addr=c0t112-48-0
                                                               server=server 2
 disk=d5
            stor_dev=000187940268-0008
                                          addr=c16t112-33-0
                                                               server=server 2
                                          addr=c0t113-48-0
 disk=d6
            stor_dev=000187940268-0009
                                                               server=server_2
 disk=d6
            stor dev=000187940268-0009
                                          addr=c16t113-33-0
                                                               server=server 2
id
          = 19
          = ufs1 snap1
name
acl
          = 0
in_use
          = False
          = uxfs
type
          = off
mrow
          = v456
volume
loog
rw_servers=
ro_servers=
rw_vdms
ro_vdms
backup_of = ufs1 Thu Oct 28 14:21:50 EDT 2004
stor devs =
000187940268-0180,000187940268-0181,000187940268-0182,000187940268-0183
disks
          = rootd378, rootd379, rootd380, rootd381
```

# **EXAMPLE #4** To turn mirroring on for a filesystem copy, **ufs1\_snap1**, type:

#### \$ fs\_timefinder ufs1\_snap1 -Mirror on operation in progress (not interruptible)...id = ufs1 name = 0acl in use = True = uxfs type = off WO rm volume = mtv1pool = rw\_servers= server\_2 ro\_servers= rw\_vdms ro\_vdms backups = ufs1\_snap1 stor\_devs = 000187940268-0006,000187940268-0007,000187940268-0008,000187940268-0009 disks = d3, d4, d5, d6disk=d3 stor\_dev=000187940268-0006 addr=c0t110-48-0 server=server\_2 disk=d3 stor\_dev=000187940268-0006 addr=c16t110-33-0 server=server\_2 disk=d4 stor\_dev=000187940268-0007 addr=c0t111-48-0 server=server\_2 disk=d4 stor\_dev=000187940268-0007 addr=c16t111-33-0 server=server 2

```
disk=d5
            stor dev=000187940268-0008
                                          addr=c0t112-48-0
                                                               server=server 2
 disk=d5
            stor_dev=000187940268-0008
                                          addr=c16t112-33-0
                                                              server=server_2
 disk=d6
            stor_dev=000187940268-0009
                                          addr=c0t113-48-0
                                                              server=server_2
 disk=d6
            stor_dev=000187940268-0009
                                          addr=c16t113-33-0
                                                              server=server_2
id
          = 19
          = ufs1_snap1
name
acl
          = 0
in use
         = False
         = mirrorfs
type
          = off
worm
         = v456
volume
loog
rw_servers=
ro_servers=
rw_vdms
ro_vdms
backup_of = ufs1 Thu Oct 28 14:21:50 EDT 2004
remainder = 0 MB (0%)
stor_devs =
000187940268 - 0180, 000187940268 - 0181, 000187940268 - 0182, 000187940268 - 0183
          = rootd378, rootd379, rootd380, rootd381
```

# **EXAMPLE #5** To perform a mirror refresh on **ufs1\_snap1**, type:

# \$ fs\_timefinder ufs1\_snap1 -Mirror refresh operation in progress (not interruptible)... remainder(MB) = 4991..4129..3281..2457..1653..815..0

```
operation in progress (not interruptible)...id
                                             = 18
        = ufs1
name
acl
in_use = True
type
        = uxfs
        = off
worm
        = mtv1
volume
pool
rw servers= server 2
ro_servers=
rw_vdms
ro_vdms
backups
       = ufs1_snap1
stor_devs =
disks
     = d3, d4, d5, d6
disk=d3
          stor_dev=000187940268-0006
                                   addr=c0t110-48-0
                                                     server=server_2
disk=d3
          stor_dev=000187940268-0006
                                   addr=c16t110-33-0
                                                     server=server_2
disk=d4
          stor_dev=000187940268-0007
                                   addr=c0t111-48-0
                                                     server=server_2
disk=d4
          stor_dev=000187940268-0007
                                  addr=c16t111-33-0
                                                     server=server_2
disk=d5
          stor_dev=000187940268-0008
                                  addr=c0t112-48-0
                                                     server=server_2
disk=d5
          stor_dev=000187940268-0008
                                   addr=c16t112-33-0
                                                     server=server_2
```

```
disk=d6
            stor_dev=000187940268-0009 addr=c0t113-48-0
                                                              server=server_2
 disk=d6
            stor_dev=000187940268-0009
                                         addr=c16t113-33-0
                                                              server=server_2
id
          = 19
name
         = ufs1_snap1
acl
         = 0
in_use
        = False
          = uxfs
type
          = off
worm
        = v456
volume
pool
rw servers=
ro_servers=
rw_vdms
ro_vdms
backup_of = ufs1 Thu Oct 28 14:25:21 EDT 2004
000187940268 - 0180, 000187940268 - 0181, 000187940268 - 0182, 000187940268 - 0183
disks
          = rootd378,rootd379,rootd380,rootd381
```

# **EXAMPLE #6** To restore the filesystem copy, **ufs1\_snap1**, to its original location, type:

```
$ /nas/sbin/rootfs timefinder ufs1 snap1 -Restore -Force
operation in progress (not interruptible)...
remainder(MB) = 0
operation in progress (not interruptible)...id = 19
        = ufs1_snap1
          = 0
acl
in_use
         = False
type
         = uxfs
          = off
worm
        = v456
volume
pool
rw_servers=
ro_servers=
rw_vdms
ro_vdms
backup_of = ufs1 Thu Oct 28 14:25:21 EDT 2004
stor_devs =
000187940268 - 0180,000187940268 - 0181,000187940268 - 0182,000187940268 - 0183
disks
          = rootd378, rootd379, rootd380, rootd381
id
          = 18
          = ufs1
name
acl
          = 0
         = True
in use
          = uxfs
type
          = off
worm
volume = mtv1
```

#### The fs Commands

```
pool
rw_servers= server_2
ro_servers=
rw_vdms =
ro_vdms
backups = ufs1_snap1
stor_devs =
000187940268 - 0006, 000187940268 - 0007, 000187940268 - 0008, 000187940268 - 0009
disks
      = d3, d4, d5, d6
 disk=d3
           stor_dev=000187940268-0006
                                       addr=c0t110-48-0
                                                          server=server_2
 disk=d3
           stor_dev=000187940268-0006
                                       addr=c16t110-33-0 server=server_2
 disk=d4
           stor dev=000187940268-0007
                                       addr=c0t1l1-48-0 server=server 2
disk=d4 stor_dev=000187940268-0007
                                       addr=c16t1l1-33-0 server=server_2
 disk=d5 stor_dev=000187940268-0008
                                       addr=c0t112-48-0
                                                         server=server_2
 disk=d5 stor_dev=000187940268-0008
                                      addr=c16t112-33-0 server=server_2
 disk=d6 stor_dev=000187940268-0009 addr=c0t113-48-0
                                                          server=server_2
 disk=d6 stor_dev=000187940268-0009
                                      addr=c16t113-33-0
                                                          server=server_2
```

EXAMPLE #1 provides a description of command output.

# The nas Commands

This chapter lists the Celerra Network Server Command Set provided for managing, configuring, and monitoring of NAS database. The commands are prefixed with **nas** and appear alphabetically. The command line syntax (Synopsis), a description of the options, and an example of usage are provided for each command. Topics included are:

- nas acl
- nas\_automountmap
- nas ca certificate
- nas cel
- nas\_checkup
- nas\_ckpt\_schedule
- nas\_connecthome
- nas\_config
- nas\_copy
- nas\_cs
- nas\_dbtable
- nas\_devicegroup
- nas disk
- nas\_diskmark
- nas emailuser
- nas event
- nas fs
- nas\_fsck
- nas\_halt
- nas\_inventory
- nas license
- nas\_logviewer

## The nas Commands

- nas\_message
- nas\_mview
- nas\_pool
- nas\_quotas
- nas\_rdf
- nas\_replicate
- nas\_server
- nas\_slice
- nas\_stats
- nas\_storage
- nas\_task
- nas\_version
- nas\_volume

# nas\_acl

Manages the access control level table.

# **SYNOPSIS**

```
nas_acl
   -list
| -info {-user|-group|-owner} <numerical_id>
| -delete {-user|-group} <numerical_id>
| [-name <name>] -create {-user|-group} <numerical_id> level=<acl_level>
| -modify {-user|-group} <numerical_id>
| [num_id=<numerical_id>] [,level=<acl_level>]}
```

### DESCRIPTION

nas\_acl creates, lists, and displays information for access control level entries within the table, and deletes the specified group or entries.

The access control level table is created and recognized in the NAS database and contains assigned levels for users and groups. A user must be defined in the /etc/passwd file prior to being assigned an entry in the table. Creating an access control level entry defines the access level allowed for the user or group once a value has been established for an object.

**Note: root** privileges are required to create, modify, or delete the access control level table. The **root** user is permitted access to all objects.

#### **OPTIONS**

-list

Lists the access control level table.

```
-info {-user|-group|-owner} <numerical_id>
Displays information for the user, group, or index entry of the owner
as specified by the <numerical_id>.
```

```
-delete {-user|-group} <numerical_id>
Deletes the entry for the specified user or group from the access
control level table.
```

-create {-user|-group} <numerical\_id> level= <acl\_level>
Creates an access control level entry for the specified user or group.
The <numerical\_id> can be a user ID (UID) or group ID (GID).

**Note:** Before executing this command, the user or group must exist in the Control Station in the /etc/passwd file or the /etc/group file.

The <acl\_level> is a single-digit (between numbers 2 and 9) input representing available access control levels. Levels 2, 3, and 4 which are established by default are:

- 2 admin Is the most privileged level and includes privileges allowed from the operator and observer levels.
- 3 operator Includes privileges from the observer level.
- 4 observer The least privileged.

Levels **5–9** are available for configuration.

```
[-name <name>]
```

The name is case-sensitive and indicates a name by which the entry is referred.

Once a value has been set, the level assigned the user or group is checked in the ACL table and the level of access to the object is determined.

```
-modify {-user|-group} <numerical_id>
{[num_id=<numerical_id>] [,level=<acl_level>] }
Modifies the <numerical_id> and level for an access control level
entry.
```

#### SEE ALSO

Controlling Access to Celerra System Objects, nas\_fs, nas\_volume, nas\_server, and nas\_storage.

## **EXAMPLE #1**

Before creating access control level entries, **su** to **root**. To create entries in the access control level table, type:

```
# nas_acl -name user1 -create -user 211 level=3

done

# nas_acl -name user2 -create -user 212 level=2

done

# nas_acl -name user3 -create -user 213 level=4

done

# nas_acl -name user4 -create -user 214 level=2

done

# nas_acl -name user5 -create -user 215 level=3

done

# nas_acl -name user6 -create -user 216 level=4

done
```

# **EXAMPLE #2** To display the access control level table, type:

# \$ nas\_acl -list

index	type	level	num_id	name
1	user	admin	201	nasadmin
2	user	operator	211	user1
3	user	admin	212	user2
4	user	observer	213	user3
5	user	admin	214	user4
6	user	operator	215	user5
7	user	observer	216	user6

## Where:

Value	Definition
index	Access control level table index entry number.
type	User or group for the entry.
level	Level of access permitted.
num_id	Numerical ID for identifying the entry.
name	Name given to the entry.

# **EXAMPLE #3** To display information for an access control level entry, type:

\$ nas\_acl -info -user 211

# Where:

Value	Definition
id	Index entry.
name	Name given for the entry.
level	Level of access permitted.
user_id	Also known as the num_id.

**EXAMPLE #4** To modify an access control level entry, type:

# nas\_acl -modify -user 211 level=7

done

**EXAMPLE #5** To delete an access control level entry, type:

# nas\_acl -delete -user 211

done

# nas\_automountmap

Manages the automount map file.

## **SYNOPSIS**

#### nas\_automountmap

```
-list_conflict <infile> [-out <outfile>]
| -create [-in <infile>] [-out <outfile>]
```

#### DESCRIPTION

**nas\_automountmap** creates and displays an automount map containing all permanently exported filesystems used by the **automount** daemon.

### **OPTIONS**

```
-list conflict <infile>
```

Prints a list of the mount points that are used more than once.

```
[-out <outfile>]
```

Prints a conflicting list and saves it to an <outfile>.

#### -create

Creates an automount map and prints it to the screen only.

```
[-in <infile>] [-out <outfile>]
```

Merges an automount map with an existing map <infile> and outputs it to an <outfile>.

```
[-out <outfile>]
```

Creates an automount map and outputs it to an <outfile>.

# **EXAMPLE #1** To create an automount map, type:

#### \$ nas\_automountmap -create

```
ufs1 -rw,intr,nosuid 127.0.0.1,10.172.128.47,128.221.253.2,128.221.252.2:/ufs1 ufs2 -rw,intr,nosuid 127.0.0.1,10.172.128.47,128.221.253.2,128.221.252.2:/ufs2
```

# **EXAMPLE #2** To create an automount map and save it to a file, type:

```
$ nas_automountmap -create -out automountmap
```

\$ more automountmap

```
ufs1 -rw,intr,nosuid 127.0.0.1,10.172.128.47,128.221.253.2,128.221.252.2:/ufs1 ufs2 -rw,intr,nosuid 127.0.0.1,10.172.128.47,128.221.253.2,128.221.252.2:/ufs2
```

# **EXAMPLE #3** To print a conflicting list, type:

# \$ nas\_automountmap -list\_conflict automountmap

```
Conflicting lists:

ufs1 -rw,intr,suid 172.16.21.202:/ufs1

ufs1_172.16.21.203 -rw,intr,suid 172.16.21.203:/ufs1
```

# **EXAMPLE #4** To merge an automount map file with an existing map file, type:

# nas\_ca\_certificate

Manages the Control Station as a Certificate Authority (CA) for Celerra's Public Key Infrastructure (PKI).

### **SYNOPSIS**

nas\_ca\_certificate
 -display
 | -generate

## **DESCRIPTION**

nas\_ca\_certificate generates a public/private key set and a CA certificate for the Control Station. When the Control Station is serving as a CA, it must have a private key with which to sign the certificates it generates for the Data Mover. The Control Station CA certificate contains the corresponding public key, which is used by clients to verify the signature on a certificate received from the Data Mover.

nas\_ca\_certificate also displays the text of the CA certificate so you can copy it and distribute it to network clients. In order for a network client to validate a certificate sent by a Data Mover that has been signed by the Control Station, the client needs the Control Station CA certificate (specifically the public key from the CA certificate) to verify the signature of the Data Mover's certificate.

The initial Control Station public/private key set and CA certificate are generated automatically during a Celerra software 5.6 install or upgrade. A new Control Station public/private key set and CA certificate is not required unless the CA key set is compromised or the CA certificate expires. The Control Station CA certificate is valid for 5 years.

You must be **root** to execute the **-generate** option from the /nas/sbin directory.

Once a Control Station CA certificate is generated, you must perform several additional tasks to ensure that the new certificate is integrated into Celerra's PKI framework. The *Celerra Security Configuration Guide* and the Unisphere online help for the PKI interface explain these tasks.

# OPTIONS

### -display

Displays the Control Station CA certificate. The certificate text is displayed on the terminal screen. Alternatively, you can redirect it to a file.

#### -generate

Generates a new CA public/private key set and certificate for the Control Station. This certificate is valid for 5 years from the date it is generated.

**SEE ALSO** server\_certificate.

**EXAMPLE #1** To generate a new Control Station CA certificate, type:

# /nas/sbin/nas\_ca\_certificate -generate

New keys and certificate were successfully generated.

**EXAMPLE #2** To display the Control Station's CA certificate, type:

# /nas/sbin/nas\_ca\_certificate -display

35:b1:e9:a8:aa:a8:4a:72:8a:b8:c4:bf:04:70:12:

**Note:** Clients need only the certificate text enclosed by BEGIN CERTIFICATE and END CERTIFICATE although most clients can handle the entire output.

```
Certificate:
   Data:
        Version: 3 (0x2)
        Serial Number: 3 (0x3)
        Signature Algorithm: shalWithRSAEncryption
        Issuer: O=Celerra Certificate Authority, CN=eng173100
        Validity
            Not Before: Mar 23 21:07:40 2007 GMT
            Not After: Mar 21 21:07:40 2012 GMT
        Subject: O=Celerra Certificate Authority, CN=eng173100
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
            RSA Public Key: (2048 bit)
                Modulus (2048 bit):
                    00:da:b2:37:86:05:a3:73:d5:9a:04:ba:db:05:97:
                    d2:12:fe:1a:79:06:19:eb:c7:2c:c2:51:93:7f:7a:
                    93:59:37:63:1e:53:b3:8d:d2:7f:f0:e3:49:42:22:
                    f4:26:9b:b4:e4:a6:40:6d:8d:e7:ea:07:8e:ca:b7:
                    7e:88:71:9d:11:27:5a:e3:57:16:03:a7:ee:19:25:
                    07:d9:42:17:b4:eb:e6:97:61:13:54:62:03:ec:93:
                    b7:e6:f1:7f:21:f0:71:2d:c4:8a:8f:20:d1:ab:5a:
                    6a:6c:f1:f6:2f:26:8c:39:32:93:93:67:bb:03:a7:
                    22:29:00:11:e0:a1:12:4b:02:79:fb:0f:fc:54:90:
                    30:65:cd:ea:e6:84:cc:91:fe:21:9c:c1:91:f3:17:
                    1e:44:7b:6f:23:e9:17:63:88:92:ea:80:a5:ca:38:
                    9a:b3:f8:08:cb:32:16:56:8b:c4:f7:54:ef:75:db:
                    36:7e:cf:ef:75:44:11:69:bf:7c:06:97:d1:87:ff:
                    5f:22:b5:ad:c3:94:a5:f8:a7:69:21:60:5a:04:5e:
                    00:15:04:77:47:03:ec:c5:7a:a2:bf:32:0e:4d:d8:
                    dc:44:fa:26:39:16:84:a7:1f:11:ef:a3:37:39:a6:
```

```
b3:31
                Exponent: 65537 (0x10001)
X509v3 extensions:
            X509v3 Subject Key Identifier:
            35:06:F2:FE:CC:21:4B:92:DA:74:C9:47:CE:BB:37:21:5E:04:E2:E6
            X509v3 Authority Key Identifier:
kevid:35:06:F2:FE:CC:21:4B:92:DA:74:C9:47:CE:BB:37:21:5E:04:E2:E6
            DirName:/O=Celerra Certificate Authority/CN=eng173100
            serial:00
X509v3 Basic Constraints:
            CA: TRUE
            X509v3 Subject Alternative Name:
            DNS:eng173100
Signature Algorithm: shalWithRSAEncryption
        09:c3:13:26:16:be:44:56:82:5d:0e:63:07:19:28:f3:6a:c4:
        f3:bf:93:25:85:c3:55:48:4e:07:84:1d:ea:18:cf:8b:b8:2d:
        54:13:25:2f:c9:75:c1:28:39:88:91:04:df:47:2c:c0:8f:a4:
        ba:a6:cd:aa:59:8a:33:7d:55:29:aa:23:59:ab:be:1d:57:f6:
        20:e7:2b:68:98:f2:5d:ed:58:31:d5:62:85:5d:6a:3f:6d:2b:
        2d:f3:41:be:97:3f:cf:05:8b:7e:f5:d7:e8:7c:66:b2:ea:ed:
        58:d4:f0:1c:91:d8:80:af:3c:ff:14:b6:e7:51:73:bb:64:84:
        26:95:67:c6:60:32:67:c1:f7:66:f4:79:b5:5d:32:33:3c:00:
        8c:75:7d:02:06:d3:1a:4e:18:0b:86:78:24:37:18:20:31:61:
        59:dd:78:1f:88:f8:38:a0:f4:25:2e:c8:85:4f:ce:8a:88:f4:
        4f:12:7e:ee:84:52:b4:91:fe:ff:07:6c:32:ca:41:d0:a6:c0:
        9d:8f:cc:e8:74:ee:ab:f3:a5:b9:ad:bb:d7:79:67:89:34:52:
        b4:6b:39:db:83:27:43:84:c3:c3:ca:cd:b2:0c:1d:f5:20:de:
        7a:dc:f0:1f:fc:70:5b:71:bf:e3:14:31:4c:7e:eb:b5:11:9c:
        96:bf:fe:6f
----BEGIN CERTIFICATE----
MIIDoDCCAoiqAwIBAqIBAzANBqkqhkiG9w0BAQUFADA8MSYwJAYDVQQKEx1DZWx1
cnJhIEN1cnRpZmljYXRlIEF1dGhvcml0eTESMBAGA1UEAxMJZW5nMTczMTAwMB4X
DTA3MDMyMzIxMDc0MFoXDTEyMDMyMTIxMDc0MFowPDEmMCOGA1UEChMdO2VsZXJy
YSBDZXJ0aWZpY2F0ZSBBdXRob3JpdHkxEjAQBqNVBAMTCWVuZzE3MzEwMDCCASIw
DQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBANqyN4YFo3PVmgS62wWX0hL+GnkG
GevHLMJRk396k1k3Yx5Ts43Sf/DjSUIi9CabtOSmOG2N5+oHjsq3fohxnREnWuNX
FgOn7hklB91CF7Tr5pdhE1RiA+yTt+bxfyHwcS3Eio8g0ataamzx9i8mjDkyk5Nn
uwOnIikAEeChEksCefsP/FSQMGXN6uaEzJH+IZzBkfMXHkR7byPpF20IkuqApco4
mrP4CMsyFlaLxPdU73XbNn7P73VEEWm/fAaX0Yf/XyK1rcOUpfinaSFqWqReABUE
d0cD7MV6or8yDk3Y3ET6JjkWhKcfEe+jNzmmNbHpqKqoSnKKuMS/BHASszECAwEA
AaOBrDCBqTAdBqNVHQ4EFqQUNQby/swhS5LadM1Hzrs3IV4E4uYwZAYDVR0jBF0w
W4AUNQby/swhS5LadMlHzrs3IV4E4uahQKQ+MDwxJjAkBgNVBAoTHUNlbGVycmEg
Q2VydG1maWNhdGUqQXV0aG9yaXR5MRIwEAYDVQQDEwl1bmcxNzMxMDCCAQAwDAYD
VROTBAUwAwEB/zAUBgNVHREEDTALgg11bmcxNzMxMDAwDQYJKoZIhvcNAQEFBQAD
ggEBAAnDEyYWvkRWgl0OYwcZKPNqxPO/kyWFw1VITgeEHeoYz4u4LVQTJS/JdcEo
OYiRBN9HLMCPpLqmzapZijN9VSmqI1mrvh1X9iDnK2iY813tWDHVYoVdaj9tKy3z
Qb6XP88Fi3711+h8ZrLq7VjU8ByR2ICvPP8UtudRc7tkhCaVZ8ZqMmfB92b0ebVd
MjM8AIx1fQIG0xpOGAuGeCQ3GCAxYVndeB+I+Dig9CUuyIVPzoqI9E8Sfu6EUrSR
/v8HbDLKQdCmwJ2PzOh07qvzpbmtu9d5Z4k0UrRrOduDJ00Ew8PKzbIMHfUq3nrc
```

8B/8cFtxv+MUMUx+67URnJa//m8=
----END CERTIFICATE----

# nas\_cel

Performs management of remotely linked Celerra Network Servers or a linked pair of Data Movers.

#### **SYNOPSIS**

```
nas_cel
   -list
| -delete {<cel_name>|id=<cel_id>} [-Force]
| -info {<cel_name>|id=<cel_id>}
| -update {<cel_name>|id=<cel_id>}
| -modify {<cel_name>|id=<cel_id>}
| {[-passphrase <passphrase>][-name <new_name>][-ip <ipaddr>]}
| -create <cel_name> -ip <ipaddr> -passphrase <passphrase>
| -interconnect <interconnect_options>
```

#### DESCRIPTION

**nas\_cel** manages the linking of the remote Celerra Network Servers to the local Celerra Network Server. **nas\_cel** also creates the trusted relationship between source and destination Celerra Network Server Control Stations in configurations such as Celerra Replicator (V2).

For Celerra Replicator (V2) only, **nas\_cel-interconnect** also builds the connection (interconnect) between a pair of Data Movers.

Linked Celerra Network Servers are acknowledged:

- Automatically during the installation
- When executing the nas\_cel -create
- When performing a nas\_rdf -init to set up the SRDF relationship between two Celerra Network Servers

#### **OPTIONS**

#### -list

Lists all Celerra Network Servers linked to the current Celerra Network Server. The hostname of the Control Station active during installation appears as the <cel\_name>.

**Note:** The ID of the object is an integer and is assigned automatically. The name of the Celerra Network Server might be truncated if it is too long for the display. To view the full name, use the **-info** option with the Celerra Network Server ID.

```
-delete <cel_name>|id=<cel_id>} [-Force]
```

Deletes the relationship of the remote Celerra Network Server, and removes its entry from the NAS database on the local Celerra Network Server.

The **-Force** option applies to SRDF and EMC MirrorView<sup>TM</sup>/S configurations only. If the Celerra to be deleted is part of an SRDF or MirrorView/S configuration, **-delete** must be specified with the **-Force** option; otherwise, an error is generated. You cannot use **-Force** if the specified Celerra is also being used by Celerra Replicator, filesystem copy (for example, with nas\_copy), or TimeFinder/FS NearCopy or FarCopy. If the deletion is necessary, clean up these configurations before performing the forced deletion.

```
-info {<cel_name>|id=<cel_id>}
```

Displays information for the remote Celerra Network Server. To view the <cel\_id> of configured Celerra Network Servers, use -list.

```
-update {<cel_name>|id=<cel_id>}
```

Updates the local Celerra Network Server entry with the local Control Station's hostname and IP address configuration. It also updates the local Data Mover-to-Data Mover authentication setup.

For the remote Celerra Network Server, updates all Data Movers that were down or experiencing errors during the **-create** or **-modify** and restores them to service by using the configuration required for Data Mover authentication.

**Note:** Data Mover authentication is used in iSCSI (Internet Small Computer System Interface) replication as the mechanism enabling two Data Movers (local or remote) to authenticate themselves and perform the requested operations. The **-update** option communicates with each Data Mover and either updates the configuration, or creates the configuration if it is being done for the first time.

```
-modify {<cel_name>|id=<cel_id>}
{[-passphrase <passphrase>][-name <new_name>][-ip
<ipaddr>]}
```

Changes the current passphrase, name, or IP address of the remote Celerra Network Server to the new passphrase, name, or IP address in the local Celerra Network Server database and modifies the remote Data Mover authentication setup by communicating with each Data Mover in the cabinet. The passphrase must have 6 to 15 characters.

**-create** <cel\_name> **-ip** <ipaddr> **-passphrase** <passphrase> Builds the trusted relationship between one Celerra Network Server and another Celerra Network Server in a configuration such as Celerra Replicator V2, SRDF, and MirrorView/S.

The **-create** must be executed twice to ensure communication from both sides, first on the source Celerra (to identify the destination

Celerra) and then on the destination Celerra (to identify the source Celerra). You must assign a name when you create the relationship (for example, a name that identifies the remote Celerra in a local entry). The IP address specified represents the appropriate remote Celerra's primary Control Station (in slot 0); the passphrase specified is used to manage the remote Celerra Network Server. The passphrase must have 6 to 15 characters and be the same between the source and destination Celerras to enable communication.

# INTERCONNECT OPTIONS

Type **nas\_cel -interconnect** to display interconnect options:

```
-interconnect
  { -create <name>
    -source server <movername>
    -destination_system {<cel_name>|id=<cel_id>}
    -destination_server <movername>
    -source interfaces {<name service interface name>|ip=<ipaddr>}
       [, {<name_service_interface_name>|ip=<ipaddr>},...]
    -destination_interfaces {<name_service_interface_name>|
        ip=<ipaddr>}[, {<name_service_interface_name>|ip=<ipaddr>},...]
       [-bandwidth <bandwidthSched>]
    -modify {<name> | id=<interConnectId>}
       {[-source_interfaces {<name_service_interface_name>|ip=<ipaddr>},...]
       -destination interfaces
       {<name_service_interface_name> | ip=<ipaddr>},...]
       [-bandwidth <bandwidthSched>]
       [-name <newName>] }
    -pause {<name>|id=<interConnectId>}
    -resume {<name>|id=<interConnectId>}
    -delete {<name>|id=<interConnectId>}
    -info {<name>|id=<interConnectId>|-all}
    -list [-destination_system {<cel_name>|id=<cel_id>}]
    -validate {<name>|id=<interconnectId>}}
```

An interconnect supports Celerra Replicator (V2) sessions by defining the communications path between a given Data Mover pair located on the same cabinet or different cabinets. The interconnect configures a list of local (source) and peer (destination) interfaces (using IP addresses and interface names), and a bandwidth schedule for all replication sessions using the interconnect. Only one interconnect can be established for a given Data Mover pair in any direction.

**Note**: You must delete all user-defined interconnects configured for a Data Mover before you can rename it. After you rename the Data Mover, you must re-create the source and peer interconnects with the new Data Mover name and then restart any associated replication sessions.

To fully establish an interconnect, nas\_cel -interconnect must be issued twice, once from each side (the local side and its peer side). Both sides of the interconnect must exist before Celerra Replicator (V2) sessions for local or remote replication can use the interconnect. Only the local side of an interconnect on which the source replication object resides is specified when creating the replication session. Loopback interconnects are created and named automatically and can be viewed using nas\_cel -interconnect -list. You cannot create, modify, or delete loopback interconnects.

#### -create <name>

Assigns a name, up to 255 characters, to the appropriate side of the interconnect. The name must be unique for each Data Mover. Make the name meaningful, identifying servers and, for remote replication, Celerra names or sites.

Remote replication naming example:

```
s2CelA_s3CelB or NYs3_LAs4 (local side)
s3CelB_s2CelA or LAs4_NYs3 (peer side)
```

Local replication naming example:

```
s2_s3 (source side on local system)
s3_s2 (peer side on the same system)
```

#### -source\_server <moverName>

Specifies the name of an available local Data Mover to use for the local side of the interconnect.

```
-destination_system {<cel_name>|id=<cel_id>}
Specifies the name or ID of the Celerra Network Server where the
peer Data Mover resides.
```

### -destination\_server <movername>

Specifies the name of an available Data Mover, on the same or different system, to use for the peer side of the interconnect.

# -source\_interfaces

```
{<name_service_interface_name>|ip=<ipaddr>}
[, {<name_service_interface_name>|ip=<ipaddr>},...]
Configures a list of interfaces available for the local side of the
```

interconnect. You can define the list by using IP addresses (IPv4 or IPv6) or name service interface names or a combination of both, but how you specify an interface determines how it must be specified by the replication session later (by name service interface name or IP address).

If you define an interface by using an IP address, make sure that the source interface list uses the same IPv4/IPv6 protocol. An IPv4 interface cannot connect to an IPv6 interface and vice versa. Both sides of the connection must use the same protocol.

For each network protocol type (IPv4/IPv6) specified in the source interface list, at least one interface from the same type must be specified in the destination interfaces list and vice versa. For example, if the source interface list includes one or more IPv6 addresses, the destination interface list must also include at least one IPv6 address.

The name service interface name is a fully qualified name given to a network interface that must resolve to a single IP address (for example, using a DNS server).

**Note**: To prevent potential errors during interface selection (especially after a failover/switchover), it is highly recommended that you specify the same local and peer interface lists when configuring each side of the interconnect.

#### -destination interfaces

{<name\_service\_interface\_name>| ip=<ipaddr>} [, {<name\_service\_interface\_name>| ip= <ipaddr>},...] Configures a list of interfaces available on the peer side of the interconnect. You can define the list by using IP addresses (IPv4 or IPv6) or name service interface names or a combination of both, but how you specify each interface determines how it is specified by the replication session.

If you define an interface using an IP address, make sure that the source interface list uses the same IPv4/IPv6 protocol. An IPv4 interface cannot connect to an IPv6 interface and vice versa. Both sides of the connection must use the same protocol.

For each network protocol type (IPv4/IPv6) specified in the destination interface list, at least one interface from the same type must be specified in the source interfaces list and vice versa. For example, if the source interface list includes one or more IPv6 addresses, the destination interface list must also include at least

one IPv6 address. The name service interface name is a fully qualified name given to a network interface that must resolve to a single IP address (for example, using a DNS server).

```
[-bandwidth <bandwidthSched>]
```

Specifies a schedule to control the interconnect bandwidth used on specific days, or times instead of using all available bandwidth at all times for the interconnect (the default).

**Note**: The bandwidth schedule executes based on Data Mover time, not Control Station time.

The schedule applies to all Celerra Replicator (V2) sessions using the interconnect. Specify a schedule with one or more comma-separated entries, most specific to least specific, as follows:

```
[{Su|Mo|Tu|We|Th|Fr|Sa}][HH:00-HH:00][/Kbps]
,[ <next_entry>],[...]
```

# Example:

MoTuWeThFr07:00-18:00/2000,/8000 means use a limit of 2000 Kb/s from 7 A.M. to 6 P.M. Monday through Friday; otherwise, use a bandwidth limit of 8000 Kb/s.

**-interconnect -modify**{<name>|**id**=<interConnectId>} Modifies one or more of the following characteristics of an existing interconnect, as specified by the name or ID for the appropriate side of the interconnect.

**Note**:You cannot modify the peer side of an interconnect configured on a remote system; you must modify it from that system. Also, you cannot modify an interface in use by a replication session.

```
[-source_interfaces(<name_service_interface_name>|
ip= <ipaddr>),... ]
```

Modifies the list of interfaces (name service interface name or IP addresses or both) available for use on the local side of an interconnect. The new list of interfaces completely replaces the previous list.

**Note**: To avoid problems with interface selection, any changes made to the interface lists should be reflected on both sides of an interconnect.

```
[-destination_interfaces{<name_service_interface_name
>|ip= <ipaddr>},...]
```

Modifies the list of interfaces (name service interface name or IP addresses or both) available for use on the peer side of an interconnect. The new list of interfaces completely replaces the previous list.

```
[-bandwidth <bandwidth>]
```

Modifies the existing bandwidth schedule for the specified interconnect, or creates a schedule if none existed previously. The schedule allocates the interconnect bandwidth for specific days or times or both instead of using all available bandwidth at all times for the interconnect (the default). The schedule applies to all replication sessions using the interconnect. Specify a schedule with one or more comma-separated entries, most specific to least specific, as follows:

```
[{Su|Mo|Tu|We|Th|Fr|Sa}][HH:00-HH:00][/Kbps]
,[ <next_entry>],[...]
```

# Example:

MoTuWeThFr07:00-18:00/2000,/8000 means use a limit of 2000 Kb/s from 7 A.M. to 6 P.M. Monday through Friday; otherwise, use a bandwidth limit of 8000 Kb/s.

```
[-name <newName>]
```

Changes the name of the specified interconnect to a new name.

- -interconnect -pause {<name>|id=<interConnectId>} Halts data transmission over the existing Data Mover interconnect until you resume transmission over the interconnect or delete the interconnect. This affects all replication sessions using the specified interconnect.
- -interconnect -resume {<name>|id=<interConnectId>} Resumes data transmission over the Data Mover interconnect, making the interconnect available for use by replication sessions.
- -interconnect -delete {<name>|id= <interConnectId>}
  Deletes the Data Mover interconnect, thereby making the
  interconnect unavailable for use by any replication sessions. You
  cannot delete an interconnect if it is in use by a replication session.
  You can delete a paused interconnect.
- -interconnect -info  ${<$ name>|id=<interConnectId>|-all} Displays information about the specified interconnect or about all interconnects known to the local system.

```
-interconnect -list [-destination_system <cel_name>
|id=<cel_id>]
```

By default, lists the interconnects available on the local Celerra. Specifying the name or ID of a remote Celerra also lists the interconnects available on that Celerra.

-interconnect -validate {<name>|id= <interconnectId>} Verifies the interconnect, verifying that authentication is configured properly by opening the connection between the Data Mover pair. Validation is done for loopback, local, and remote configuration.

**SEE ALSO** *Using Celerra Replicator (V2)*, nas\_copy, nas\_replicate, and nas\_task.

## **EXAMPLE #1**

To create an entry for the remote Celerra Network Server, type:

\$ nas\_cel -create cs110 -ip 172.24.102.240 -passphrase nasdocs

```
operation in progress (not interruptible)...
```

id = 3
name = cs110
owner = 0
device =

channel = net path = 1

net\_path = 172.24.102.240
celerra\_id = APM000438070430000

passphrase = nasdocs

### Where:

Value	Definition		
id	ID of the remote Celerra Network Server on the local Celerra.		
name	Name assigned in the local view to the remote Celerra Network Server.		
owner	ACL ID assigned automatically.		
device	R2 device mounted by the local Control Station to read the database of the remote Control Station in the SRDF environment. This value is unique to the Symmetrix storage system.		
channel	Pair of devices used in the rdf channel. One is used for writing messages to the remote (wdev), the other to read messages from them. This value is unique to the Symmetrix storage system.		
net_path	IP address of the remote Celerra Network Server.		
celerra_id	Unique Celerra ID number.		
passphrase	Used for authentication with a remote Celerra Network Server.		

# **EXAMPLE #2** For the Celerra Network Server with a CLARiiON storage system, to list all remote Celerra Network Servers, type:

#### \$ nas cel -list

id	name	owner mount_dev	channel	net_path	CMU
0	cs100	0		172.24.102.236	APM000420008180000
3	cs110	0		172.24.102.240	APM000438070430000

For the Celerra Network Server with a Symmetrix storage system, to list all remote Celerra Network Servers, type:

## \$ nas\_cel -list

1	_d	name	owner	mount_dev	channel	net_path	CMU
C	)	cs30	0			172.24.172.152	0028040001900006
1	L	cs40	500	/dev/sdj1	/dev/sdg	172.24.172.151	0028040002180000

#### Where:

Value	Definition
id	ID of the remote Celerra Network Server on the local Celerra.
name	Name assigned in the local view to the remote Celerra Network Server.
owner	ACL ID assigned automatically.
mount_dev	Mounted database from the remote Celerra Network Server in the SRDF environment. This value is unique to the Symmetrix storage system.
channel	RDF channel from where information is read and written. This value is unique to the Symmetrix storage system.
net_path	IP address of the remote Celerra Network Server.
CMU	Celerra Management Unit (unique Celerra ID number).

# **EXAMPLE #3** To display information for the remote Celerra Network Server, **cs110**, type:

## \$ nas\_cel -info cs110

net\_path = 172.24.102.240
celerra\_id = APM000438070430000

passphrase = nasdocs

EXAMPLE #1 provides information for a description of command outputs.

# **EXAMPLE #4** To update the Control Station entry for **cs110**, type:

## \$ nas\_cel -update cs110

```
passphrase = nasdocs
```

EXAMPLE #1 provides information for a description of command outputs.

# **EXAMPLE #5**

To modify the passphrase and name for the remote Control Station cs110, type:

\$ nas\_cel -modify cs110 -passphrase nasdocs\_replication
-name cs110\_target

EXAMPLE #1 provides information for a description of command outputs.

#### **EXAMPLE #6**

To delete the Control Station entry of the remote Celerra Network Server, **cs110\_target**, type:

```
$ nas_cel -delete cs110_target
```

EXAMPLE #1 provides information for a description of command outputs.

# **EXAMPLE #7**

To create an interconnect NYs3\_LAs2 between Data Mover server\_3 and remote Data Mover server\_2, and use a bandwidth limit of 2000 Kb/s from 7 A.M. to 6 P.M. Monday through Friday; otherwise, use a bandwidth limit of 8000 Kb/s, type:

```
$ nas_cel -interconnect -create NYs3_LAs2 -source_server
server_3
-destination_system cs110 -destination_server server_2
-source interfaces
```

# ip=10.6.3.190 -destination\_interfaces ip=10.6.3.173 -bandwidth

MoTuWeThFr07:00-18:00/2000,/8000

```
operation in progress (not interruptible)...
id
                                   = 30003
name
                                   = NYs3 LAs2
                                   = server_3
source_server
source_interfaces
                                   = 10.6.3.190
destination_system
                                   = cs110
destination_server
                                   = server_2
destination_interfaces
                                   = 10.6.3.173
bandwidth schedule
                                   = MoTuWeThFr07:00-18:00/2000,/8000
crc enabled
number of configured replications = 0
number of replications in transfer = 0
status
                                   = The interconnect is OK.
```

### Where:

Value	Definition
id	ID of the interconnect.
name	Name of the interconnect.
source_server	Name of an available local Data Mover to use for the local side of the interconnect.
source_interfaces	IP addresses available for the local side of the interconnect (at least one, or a name service interface name).
destination_system	Control Station names of the Celerra systems available for use in a remote replication session. Local System is the default.
destination_server	Name of an available peer Data Mover to use for the peer side of the interconnect.
destination_interface s	IP addresses available for the peer side of the interconnect (at least one, or a name service interface name). For loopback interconnects, the interface is fixed at 127.0.0.1.
bandwidth schedule	Bandwidth schedule with one or more comma-separated entries, most specific to least specific.
crc enabled	Indicates that the Cyclic Redundancy Check (CRC) method is in use for verifying the integrity of data sent over the interconnect. CRC is automatically enabled and cannot be disabled.
number of configured replications	Number of replication sessions currently configured.
number of replications in transfer	Number of replications are currently in transfer.
status	Status of the interconnect.

# **EXAMPLE #8**

To modify the bandwidth schedule of the interconnect NYs3\_LAs2, type:

```
\  \,  nas_cel -interconnect -modify NYs3_LAs2 -bandwidth MoWeFr07:00-18:00/2000, TuTh07:00-18:00/4000, /8000
```

```
operation in progress (not interruptible)... id = 30003
```

```
= NYs3_LAs2
name
                                    = server_3
source_server
                                    = 10.6.3.190
source_interfaces
destination_system
                                    = cs110
destination_server
                                    = server_2
destination interfaces
                                   = 10.6.3.173
bandwidth schedule
MoWeFr07:00-18:00/2000, TuTh07:00-18:00/4000, /8000
crc enabled
                                    = ves
number of configured replications = 0
number of replications in transfer = 0
                                    = The interconnect is OK.
status
                        EXAMPLE #7 provides a description of the command outputs.
         EXAMPLE #9
                        To list available interconnects, type:
                        $ nas_cel -interconnect -list
id
       name
                          source_server destination_system
                                                                 destination_server
20001 loopback
                           server_2
                                           cs100
                                                                 server_2
30001 loopback
                           server_3
                                           cs100
                                                                 server_3
30003 NYs3_LAs2
                           server_3
                                           cs110
                                                                 server_2
        EXAMPLE #10
                        To pause the interconnect with id=30003, type:
                        $ nas_cel -interconnect -pause id=30003
done
        EXAMPLE #11
                        To resume the interconnect NYs3_LAs2, type:
                        $ nas_cel -interconnect -resume NYs3_LAs2
done
        EXAMPLE #12
                        To validate the interconnect NYs3_LAs2, type:
                        $ nas cel -interconnect -validate NYs3 LAs2
NYs3_LAs2: validating 9 interface pairs: please wait...ok
        EXAMPLE #13
                        To display the detailed information about the interconnect
                        NYs3_LAs2, type:
                        $ nas cel -interconnect -info NYs3 LAs2
id
                                    = 30003
name
                                    = NYs3_LAs2
source server
                                    = server 3
source_interfaces
                                    = 10.6.3.190
destination_system
                                    = cs110
destination_server
                                    = server 2
destination_interfaces
                                    = 10.6.3.173
bandwidth schedule
MoWeFr07:00-18:00/2000, TuTh07:00-18:00/4000, /8000
crc enabled
                                    = yes
number of configured replications = 0
```

EXAMPLE #7 provides a description of the command outputs.

# **EXAMPLE #14** To delete interconnect **NYs3\_LAs2**, type:

## \$ nas\_cel -interconnect -delete NYs3\_LAs2

```
operation in progress (not interruptible)...
id
                                   = 30003
name
                                   = NYs3_LAs2
                                   = server 3
source server
source_interfaces
                                   = 10.6.3.190
destination_system
                                   = cs110
destination_server
                                   = server_2
destination_interfaces
                                   = 10.6.3.173
bandwidth schedule
MoWeFr07:00-18:00/2000, TuTh07:00-18:00/4000, /8000
crc enabled
number of configured replications = 0
number of replications in transfer = 0
status
                                    = The interconnect is OK.
```

EXAMPLE #7 provides a description of the command outputs.

# nas\_checkup

Provides a system health checkup for the Celerra Network Server.

## **SYNOPSIS**

nas\_checkup
[-version|-help|-rerun]

## **DESCRIPTION**

**nas\_checkup** runs scheduled and unscheduled health checks on the Celerra Network Server and reports problems that are found, and the actions needed to fix the problem and acts as a system health monitor.

The scheduled run time for the **nas\_checkup** command is every two weeks by default. If a warning or error is discovered during this time, an alert is posted on the Unisphere Software.

Set up email notification for warnings or errors in the Unisphere Software Notifications page, or modify and load the sample **nas\_checkup** event configuration file.

If a problem is discovered that requires EMC Service Personnel assistance, **nas\_checkup** will notify EMC.

### **OPTIONS**

# No arguments

Runs a series of system health checks on the Celerra and reports the problems that are found and the actions needed to fix the problem.

**Note:** No email, callhome, or Unisphere Software alert is posted when the health check is run unscheduled.

#### -version

Displays the version of health check that is run on the Celerra Network Server.

#### -help

Provides help.

#### -rerun

Reruns the checks that produce error messages in the previous health checkup. It does not rerun the checks that produce warning or information messages. If there are no checks that produce error messages, then the -rerun switch generates a message that there is nothing to rerun.

# CHECKS

**Nas\_checkup** runs a subset of the available checks based on the configuration of your system. The complete list of available checks are:

#### Control Station Checks:

```
Check if minimum free space exists
Check if minimum free space exists ns
Check if enough free space exists
Check if enough free space exists ns
Check if NAS Storage API is installed correctly
Check if NAS Storage APIs match
Check if NBS clients are started
Check if NBS configuration exists
Check if NBS devices are accessible
Check if NBS service is started
Check if standby is up
Check if Symapi data is present
Check if Symapi is synced with Storage System
Check integrity of NASDB
Check if primary is active
Check all callhome files delivered
Check if NAS partitions are mounted
```

#### Data Mover Checks:

```
Check boot files
Check if hardware is supported
Check if primary is active
Check if root filesystem has enough free space
Check if using standard DART image
Check MAC address
Check network connectivity
Check status
```

# Storage System Checks:

```
Check disk emulation type
Check disk high availability access
Check disks read cache enabled
Check disks and storage processors write cache enabled
Check if access logix is enabled
Check if FLARE is committed
Check if FLARE is supported
Check if microcode is supported
Check no disks or storage processors are failed over
Check that no disks or storage processors are faulted
Check that no hot spares are in use
Check that no hot spares are rebuilding
Check control lun size
Check if storage processors are read cache enabled
```

Check Version: Check Command:

Check Log

# **FILES** The files associated with system health checkups are:

/nas/log/nas\_ Contains information about the checks that were run, problems found, and actions needed to fix the problem.

/nas/log/nas\_checkup.<timestam p>.log Produced when a scheduled nas\_checkup is run and contains the same information as the nas\_checkup-run.<timestamp>.log.

/nas/log/syslog Contains the overall results of nas\_checkup.

/nas/site/checkup\_eventlog.cfg Provides a sample nas\_checkup event configuration file. This is the file to be modified to add email addresses and load the file.

# **SEE ALSO** *Configuring Celerra Events and Notifications.*

/nas/log/checkup-run.070611-064115.log

# **EXAMPLE #1** To run a health check on the Celerra Network Server, type:

#### \$ nas\_checkup

/nas/bin/nas\_checkup

5.6.23.1

```
-----Checks------
Control Station: Checking if file system usage is under limit..... Pass
Control Station: Checking if file systems have enough space to upgrade..... Pass
Control Station: Checking if NAS Storage API is installed correctly...... Pass
Control Station: Checking if NBS clients are started..... Pass
Control Station: Checking if NBS configuration exists...... Pass
Control Station: Checking if NBS devices are accessible...... Pass
Control Station: Checking if NBS service is started...... Pass
Control Station: Checking if Symapi data is present...... Pass
Control Station: Checking if Symapi is synced with Storage System..... Pass
Control Station: Checking integrity of NASDB...... Pass
Control Station: Checking all callhome files delivered...... Pass
Control Station: Checking resolv conf...... Pass
Control Station: Checking if NAS partitions are mounted...... Pass
Control Station: Checking nas site eventlog configuration..... Pass
Control Station: Checking nas sys mcd configuration..... Pass
Control Station: Checking nas sys eventlog configuration..... Pass
Control Station: Checking logical volume status..... Pass
Control Station: Checking ups is available..... Fail
          : Checking boot files..... Pass
Data Movers
          : Checking if primary is active..... Pass
Data Movers
Data Movers
          : Checking if root filesystem has enough free space...... Pass
Data Movers
          : Checking if using standard DART image..... Pass
Data Movers
         : Checking network connectivity..... Pass
Data Movers
         : Checking status..... Pass
          : Checking dart release compatibility..... Pass
Data Movers
```

```
Data Movers
           : Checking dart version compatibility..... Pass
           Data Movers
           : Checking unique id...... Pass
Data Movers
Data Movers
           : Checking CIFS file server configuration.....
Data Movers
          : Checking domain controller connectivity and configuration.
          : Checking DNS connectivity and configuration..... N/A
Data Movers
Data Movers
           : Checking connectivity to WINS servers.....
                                                          N/A
          : Checking connectivity to NTP servers...... N/A
Data Movers
Data Movers
          : Checking connectivity to NIS servers..... Pass
          : Checking virus checker server configuration..... N/A
Data Movers
          : Checking if workpart is OK...... Pass
Data Movers
          : Checking if free full dump is available......
Data Movers
Data Movers
           : Checking if each primary data mover has standby..... Fail
Storage System : Checking disk high availability access...... Pass
Storage System : Checking disks read cache enabled...... Pass
Storage System : Checking disks and storage processors write cache enabled. Pass
Storage System : Checking if access logix is enabled...... Pass
Storage System : Checking if FLARE is committed...... Pass
Storage System : Checking if FLARE is supported...... Pass
Storage System : Checking if microcode is supported...... Pass
Storage System : Checking no disks or storage processors are failed over... Pass
Storage System : Checking that no disks or storage processors are faulted.. Pass
Storage System : Checking that no hot spares are in use...... Pass
Storage System: Checking that no hot spares are rebuilding...... Pass
Storage System : Checking minimum control lun size...... Pass
Storage System : Checking maximum control lun size...... Fail
Storage System : Checking system lun configuration...... Pass
Storage System : Checking if storage processors are read cache enabled..... Pass
Storage System : Checking if auto assign are disabled for all luns...... Pass
Storage System : Checking if auto trespass are disabled for all luns...... Pass
```

One or more warnings are shown below. It is recommended that you follow the instructions below to correct the problem then try again.

Data Movers: Check if each primary data mover has standby

Symptom: The following primary Data Movers server\_2, server\_3 does

not have a standby Data Mover configured. It is recommended that each

primary Data Mover have a standby configured for it with automatic

failover policy for high availability.

Storage System: Check maximum control lun size Symptom:

\* The size of control LUN 5 is 32 GB. It is larger than the recommended size of 14 GB. The additional space will be reserved by the system.

\_\_\_\_\_\_

-----Warnings-----

Data Movers: Check if free full dump is available

Symptom: Cannot get workpart structure. Command failed.

\* Command: /nas/sbin/workpart -r

\* Command output: open: Permission denied

\* Command exit code: 2

Action : Contact EMC Customer Service and refer to EMC Knowledgebase

emc146016. Include this log with your support request.

-----

# **EXAMPLE #2** To display help for **nas\_checkup**, type:

\$ nas\_checkup -help

Check Version: 5.6.23.1

Check Command: /nas/bin/nas\_checkup

usage: nas\_checkup

[ -help | -version ]

# **EXAMPLE #3** To display the version of **nas\_checkup** utility, type:

\$ nas\_checkup -version

Check Version: 5.6.23.1

Check Command: /nas/bin/nas\_checkup

#### DIAGNOSTICS

**nas\_checkup** returns one of the following exit statuses:

- ◆ 0 No problems found
- ◆ 1 nas\_checkup posted information
- 2 nas\_checkup discovered a warning
- ◆ 3 nas\_checkup discovered an error
- ◆ 255 Any other error

Examples of errors that could cause a 255 exit status include, but are not limited to:

- If **nas\_checkup** is run when another instance of **nas\_checkup** is running
- If **nas\_checkup** is run by someone other than **root** or the administrator group (generally nasadmin)
- If nas\_checkup is run on the standby Control Station

# nas\_ckpt\_schedule

Manages SnapSure checkpoint scheduling for the Celerra Network Server.

### **SYNOPSIS**

```
nas ckpt schedule
  -list
  -info {-all|<name>|id=<id>}
  -create <name>
    -filesystem {<name>|id=<id>} [-description <description>]
    -recurrence {
    once [-start on <YYYY-MM-DD>] -runtimes <HH:MM>
    [-ckpt_name <ckpt_name>]
    | daily [-every <number_of_days>]
        [-start_on <YYYY-MM-DD>][-end_on <YYYY-MM-DD>]
        -runtimes <HH:MM>[,...]
        {-keep <number_of_ckpts>|-ckpt_names <ckpt_name>[,...]}
    weekly [-every <number_of_weeks>]
        -days_of_week {Mon|Tue|Wed|Thu|Fri|Sat|Sun}[,...]
        [-start_on <YYYY-MM-DD>][-end_on <YYYY-MM-DD>]
        -runtimes <HH:MM>[,...]
        {-keep <number_of_ckpts>|-ckpt_names <ckpt_name>[,...]}
    | monthly [-every <number of months>] -days of month <1-31>[,...]
        [-start_on <YYYY-MM-DD>][-end_on <YYYY-MM-DD>]
        -runtimes <HH:MM>[,...]
        {-keep <number_of_ckpts>|-ckpt_names <ckpt_name>[,...]}}
    [{-cvfsname_prefix <prefix>|-time_based_cvfsname }]
-modify {<name>|id=<id>}
    -modify {<name>|id=<id>}
    [-name <new_name>]
    [{-cvfsname_prefix <prefix>| -time_based_cvfsname}]
    [-description <description>]
    [-recurrence {daily | weekly | monthly}]
    [-every {number_of_days | number_of_weeks | <number_of_months}]
    [-days_of_week {Mon|Tue|Wed|Thu|Fri|Sat|Sun}[,...]]
    [-days_of_month <1-31>[,...]][ -start_on <YYYY-MM-DD>]
    [-end_on <YYYY-MM-DD>][ -runtimes <HH:MM>[,...]]
  -delete {<name>|id=<id>}
  -pause {<name>|id=<id>}
  -resume {<name>|id=<id>}
```

#### DESCRIPTION

nas\_ckpt\_schedule creates and lists the schedules for the SnapSure checkpoints. Schedules can be run once, daily, weekly, or monthly and can be modified, paused, resumed, and deleted.

## **OPTIONS**

#### -list

Lists all checkpoint schedules on the system, the name of the schedule, the next run date, the state, and the description.

```
-info {-all|<name>|id=<id>}
```

Lists detailed information for all schedules or for the specified schedule.

```
-create <name> -filesystem {<name> |id=<id>}
[-description <description>] -recurrence {
```

Creates a checkpoint schedule for the filesystem that is specified by <name> or <id>. The schedule name in -create <name> must be unique. The -description option provides a label for the schedule. The -recurrence option specifes if the checkpoint operation occurs once or daily, weekly, or monthly.

**Note**: It is recommended that a time interval of at least 15 minutes between the creation of two checkpoints on the same production filesystem. *Using SnapSure on Celerra* provides information on checkpoint scheduling.

```
once [-start_on <YYYY-MM-DD>] -runtime <HH:MM>
[-ckpt_name <ckpt_name>]
```

If **once** is specified, the hours and minutes for the snapshot to be run must be specified. A start date and name may be optionally assigned to the checkpoint.

For a one-time checkpoint schedule, only one runtime can be provided. For one-time schedules, the option **-ckpt\_name** can specify a name for the single checkpoint; if omitted, the default naming is used (<schedule\_name>\_<fs\_name>\_<num>) where <num> is a four digit integer beginning with 0001.

```
|daily [-every <number_of_days>] [-start_on
<YYYY-MM-DD>] [-end_on <YYYY-MM-DD>]
-runtimes <HH:MM>[,...]
```

{-keep <number\_of\_ckpts>|-ckpt\_names <ckpt\_name>[,...]} If daily is specified, the checkpoint is taken every day unless -every is specified indicating the number of days between runs. The -start\_on option indicates the day when the checkpoints will start and -end\_on indicates the day when they end.

The **-runtimes** option specifies one or more times to take a checkpoint on each scheduled day. The **-keep** option specifies the maximum number of checkpoints to be kept at any one time (using default checkpoint naming). <number\_of\_ckpts> should

be equal to the number of checkpoint names specified for a schedule. The **-ckpt\_name** option assigns one or more specific names to each checkpoint as it is taken.

```
|weekly [-every <number_of_weeks>] -days_of_week
{Mon|Tue|Wed|Thu|Fri|Sat|Sun} [,...][-start_on
<YYYY-MM-DD>][-end_on <YYYY-MM-DD>]-runtimes
<HH:MM>[,...]
```

{-keep <number\_of\_ckpts>|-ckpt\_names <ckpt\_name>[,...]} If weekly is specified, the checkpoint is taken every week unless the -every is specified indicating the number of weeks between runs. The -days\_of\_week option specifies one or more days during the week on which to run the schedule. The -start\_on option indicates the day when the checkpoints will start and -end\_on indicates the day when they end.

The **-runtimes** option specifies one or more times to take a checkpoint on each scheduled day. The **-keep** option specifies the maximum number of checkpoints to be kept at any one time (using default checkpoint naming). The **-ckpt\_name** option assigns one or more specific names to each checkpoint as it is taken.

```
|monthly [-every <number_of_months>] -days_of_month <1-31>[,...][-start_on <YYYY-MM-DD>][-end_on <YYYY-MM-DD>]-runtimes <HH:MM>[,...] {-keep <number_of_ckpts> |-ckpt_names <ckpt_name>[,...]}} If monthly is specified, the checkpoint is taken every month unless the -every is specified indicating the number of months between runs. The -days_of_month option specifies one or more days during the month on which to run the schedule. <days> is specified as an integer 1 through 31. The -start_on option indicates the day when the checkpoints will start and -end_on indicates the day when they end.
```

The **-runtimes** option specifies one or more times to take a checkpoint on each scheduled day. The **-keep** option specifies either the maximum number of checkpoints to be kept at any one time (using default checkpoint naming) or using the **-ckpt\_name** option, one or more specific names to assign each checkpoint as it is taken.

**Note:** The schedule that is set takes effect immediately unless **-start\_on** is specified. Daily, weekly, and monthly schedules run indefinitely unless **-end\_on** is included.

The **-cvfsname\_prefix** option specifies the customized prefix of a CVFS name. This prefix along with the cvfsname\_delimiter and the cvfs\_starting\_index make up the CVFS name. The **-time\_based\_cvfsname** option specifies the CVFS name based on the creation time of the CVFS. It is the default method for generating CVFS names and will be used if the prefix is not specified.

**Note:** The prefix must be a PFS-wide unique string and can contain up to 20 ASCII characters. The prefix must not include intervening spaces, colons (:), or slashes (/).

```
-modify {<name>|id=<id>} [-name <new_name>]
[{-cvfsname_prefix <prefix>| -time_based_cvfsname}]
[-description <description>] [-recurrence
{daily|weekly|monthly}] [-every <number_of_days>|
<number_of_weeks>|<number_of_months>]
[-days_of_week {Mon|Tue|Wed|Thu|Fri|Sat|Sun}[,...]]
[-days_of_month <1-31>[,...]][-start_on <YYYY-MM-DD>]
[-end_on <YYYY-MM-DD>][ -runtimes <HH:MM>[,...]]
Modifies the scheduled checkpoint entry as specified.
```

```
-delete {<name> | id=<id>}
```

Deletes the specified checkpoint schedule. This operation does not delete any checkpoints.

```
-pause {<name>| id=<id>}
```

Pauses the specified checkpoint schedule, including checkpoint creations.

```
-resume {<name>|id=<id>} Resumes a paused checkpoint schedule.
```

**SEE ALSO** *Using SnapSure on Celerra.* 

## **EXAMPLE #1**

To create a checkpoint schedule that creates a checkpoint of the filesystem **ufs1** daily at 8 A.M. and 8 P.M. starting on 11/13/06 with the last run on 12/13/07, and keep 7 checkpoints, type:

\$ nas\_ckpt\_schedule -create ufs1\_ckpt\_sched1 -filesystem
ufs1 -description "Daily
Checkpoint schedule for ufs1" -recurrence daily -every 1
-start\_on 2006-11-13
-end\_on 2007-12-13 -runtimes 8:00,20:00 -keep 7
-cvfsname\_prefix daily

This command returns no output.

# **EXAMPLE #2**

To create a checkpoint schedule that creates a checkpoint of the filesystem **ufs1** weekly on Mondays at 6 P.M., starting on 11/13/06 with the last run on 12/13/07, and name new checkpoints ufs1\_ckpt\_mon1, ufs1\_ckpt\_mon2, ufs1\_ckpt\_mon3, ufs1\_ckpt\_mon4, type:

\$ nas\_ckpt\_schedule -create ufs1\_ckpt\_sched2 -filesystem
ufs1 -description "Weekly Checkpoint schedule for ufs1"
-recurrence weekly -every 1 -days\_of\_week Mon -start\_on
2006-11-13 -end\_on 2007-12-13 -runtimes 18:00 -ckpt\_names
ufs1\_ckpt\_mon1,ufs1\_ckpt\_mon2,ufs1\_ckpt\_mon3,ufs1\_ckpt\_m
on4 -cvfsname\_prefix weekly

This command returns no output.

## **EXAMPLE #3**

To create a checkpoint schedule that creates a checkpoint of the filesystem **ufs1** every other 15th of the month at 7 P.M., and keep 12 checkpoints, type:

\$ nas\_ckpt\_schedule -create ufs1\_ckpt\_sched3 -filesystem
ufs1 -description
"Monthly Checkpoint schedule for ufs1" -recurrence
monthly -every 2 -days\_of\_month
15 -runtimes 19:00 -keep 12 -cvfsname\_prefix monthly

This command returns no output.

#### **EXAMPLE #4**

To create a checkpoint schedule that creates a checkpoint of the filesystem **ufs1** once at 3:09 P.M., type:

\$ nas\_ckpt\_schedule -create ufs1\_ckpt\_sched4 -filesystem
ufs1 -description
"One-time Checkpoint Schedule for ufs1" -recurrence once
-runtimes 15:09

This command returns no output.

# **EXAMPLE #5** To list all checkpoint schedules, type:

```
$ nas_ckpt_schedule -list
id
           = ufs1_ckpt_sched2
name
description = Weekly Checkpoint schedule for ufs1
state = Pending
next run = Mon Nov 13 18:00:00 EST 2006
          = 80
id
name
         = ufs1_ckpt_sched4
description = One-time Checkpoint Schedule for ufs1
state = Pending
next run = Tue Nov 14 15:09:00 EST 2006
id
          = 5
          = ufs1_ckpt_sched1
description = Daily Checkpoint schedule for ufs1
state = Pending
next run = Mon Nov 13 20:00:00 EST 2006
iА
name
          = ufs1_ckpt_sched3
description = Monthly Checkpoint schedule for ufs1
        = Pending
state
next run = Wed Nov 15 19:00:00 EST 2006
```

# **EXAMPLE #6** To modify the recurrence of the checkpoint schedule **ufs1\_ckpt\_sched3** to run every 10th of the month, type:

\$ nas\_ckpt\_schedule -modify ufs1\_ckpt\_sched3 -recurrence
monthly -every 1 -days\_of\_month 10

This command returns no output.

# **EXAMPLE #7** To get detailed information about checkpoint schedule, type:

```
$ nas_ckpt_schedule -info ufs1_ckpt_sched3
id = 7
name = ufs1_ckpt_sched3
description = Monthly Checkpoint schedule for ufs1
CVFS name prefix = monthly
tasks = Checkpoint ckpt_ufs1_ckpt_sched3_001 on filesystem id=25, Checkpoint
ckpt_ufs1_ckpt_sched3_002 on filesystem id=25, Checkpoint
ckpt_ufs1_ckpt_sched3_003 on filesystem id=25, Checkpoint
ckpt_ufs1_ckpt_sched3_004 on filesystem id=25, Checkpoint
ckpt_ufs1_ckpt_sched3_005 on filesystem id=25, Checkpoint
ckpt_ufs1_ckpt_sched3_006 on filesystem id=25, Checkpoint
```

ckpt\_ufs1\_ckpt\_sched3\_007 on filesystem id=25, Checkpoint

```
ckpt_ufs1_ckpt_sched3_008 on filesystem id=25, Checkpoint
ckpt_ufs1_ckpt_sched3_009 on filesystem id=25, Checkpoint
ckpt_ufs1_ckpt_sched3_010 on filesystem id=25, Checkpoint
ckpt_ufs1_ckpt_sched3_011 on filesystem id=25, Checkpoint
ckpt_ufs1_ckpt_sched3_012 on filesystem id=25
next run = Sun Dec 10 19:00:00 EST 2006
state = Pending
recurrence = every 1 months
start on = Mon Nov 13 16:47:51 EST 2006
end on =
at which times = 19:00
on which days of week =
on which days of month = 10
```

# EXAMPLE #8

To pause a checkpoint schedule, type:

\$ nas\_ckpt\_schedule -pause ufs1\_ckpt\_sched1

This command returns no output.

#### **EXAMPLE #9**

To resume a checkpoint schedule, type:

\$ nas\_ckpt\_schedule -resume ufs1\_ckpt\_sched1

This command returns no output.

#### **EXAMPLE #10**

To delete a checkpoint schedule, type:

\$ nas\_ckpt\_schedule -delete ufs1\_ckpt\_sched2

This command returns no output.

# nas\_connecthome

Configures email, FTP, and modem transport mechanisms for transporting Callhome event files to user-configured destinations.

#### **SYNOPSIS**

```
nas connecthome
  -info
  -test {-email_1|-email_2|-ftp_1|-ftp_2|-modem_1|-modem_2}
  -modify [-modem_priority {Disabled | 1 | 2 | 3 } ]
    [-modem_number <phone_number>]
    [-modem_number_2 <phone_number>]
    [-ftp_priority {Disabled | 1 | 2 | 3 }]
    [-ftp_server {<hostname> | <ip_addr>}]
    [-ftp_port <port>]
    [-ftp_user <username>]
    [-ftp_passwd [<passwd>]]
    [-ftp folder <path>]
    [-ftp_ipprotocol {IPV4 | IPV6}]
    [-ftp_mode {active passive}]
    [-ftp_server_2 {<hostname> | <ip_addr>}]
    [-ftp_port_2 <port>]
    [-ftp_user_2 <username>]
    [-ftp passwd 2 [<passwd>]]
    [-ftp_folder_2 <path>]
    [-ftp_ipprotocol_2 {IPV4 | IPV6}]
    [-ftp_mode_2 {active | passive}]
    [-email_priority {Disabled | 1 | 2 | 3 }]
    [-email_from <email_addr>]
    [-email_to {<email_addr>[,<email_addr>]}
    [-email_subject <email_subject>]
    [-email_server {<hostname> | <ip_addr>}]
    [-email ipprotocol {IPV4 | IPV6}]
    [-email_server_2 {<hostname> | <ip_addr>}]
    [-email_ipprotocol_2 {IPV4 | IPV6}]
    [-dial_in_number <phone_number>]
    [-serial_number <serial_number>]
    [-site_id <site_id>]
    [-encryption_enabled {yes|no}]
    [-dial in enabled {yes no}]
-help
```

# **DESCRIPTION**

nas\_connecthome pauses and resumes the ConnectHome service, displays and configures parameters for email, FTP, and modem, which are mechanisms used for transmitting event files.
nas\_connecthome enables a user to configure primary and optional secondary destinations for each transport mechanism.

**nas\_connecthome** also tests connectivity to the destination configured for a transport mechanism.

This command must be executed from /nas/sbin/.

## **OPTIONS**

#### -info

Displays the enabled and disabled configuration parameters for all three transport mechanisms.

-test {-email\_1|-email\_2|-ftp\_1|-ftp\_2|-modem\_1|-modem\_2} Tests connectivity to the destination configured and enabled for the specified transport mechanism.

### -modify

Modifies the following configuration parameters for any or all three transport mechanisms:

# [-modem\_priority {Disabled | 1 | 2 | 3 }]

Enables modem as a Primary, Secondary, or Tertiary transport mechanism. Specifying **Disabled** removes modem as a transport mechanism.

```
[-modem_number <phone_number>]
```

Sets or modifies the primary phone number of the modem.

**Note:** Specifying "" (empty double quotes) disables the use of the existing phone number.

```
[-modem_number_2 <phone_number>]
```

Sets or modifies the secondary phone number of the modem.

**Note:** Specifying "" (empty double quotes) disables the use of the existing phone number for this transport mechanism.

### [-ftp priority {Disabled | 1 | 2 | 3 }]

Enables FTP as a Primary, Secondary, or Tertiary transport mechanism. Specifying **Disabled** removes FTP as a transport mechanism.

```
[-ftp_server {<hostname>|<ip_addr>}]
```

Sets or modifies the hostname or IP address of the primary FTP server and corresponding port. The allowable input is IPv4 address, IPv6 address, or domain name.

## [-ftp\_port <port>]

Sets or modifies the port of the primary FTP server and corresponding port. The valid input is an integer between 1 and 65535. If an empty string "" is provided for this option, the port number is reset to the default value 21.

#### [-ftp\_user <username>]

Sets or modifies the username of the login account on the primary FTP server.

**Note:** Specifying "" (empty double quotes) reverts to the default value of **onalert**.

## [-ftp\_passwd [<passwd>]]

Sets or modifies the password of the login account on the primary FTP server.

**Note:** Specifying "" (empty double quotes) reverts to the default value of **EMCCONNECT**.

## [-ftp\_folder <path>]

Sets or modifies the path to the folder on the primary FTP server where the event files have to be deposited.

**Note:** Specifying "" (empty double quotes) reverts to the default value of **incoming**.

# [-ftp\_ipprotocol {IPV4 | IPV6}]

Sets or modifies the transfer mode of the primary FTP transport mechanism. If an IPv4 address is provided to FTP server, the corresponding IP protocol is changed to IPv4 automatically. If an IPv6 address is used, the IP protocol is changed to IPv6. When hostname is specified, no IP protocol change is made.

#### [-ftp\_mode {active | passive}]

Sets or modifies the transfer mode of the primary FTP transport mechanism.

**Note:** Specifying '''' (empty double quotes) reverts to the default value of **active**.

# [-ftp\_server\_2 <hostname>[<ip\_addr>]]

Sets or modifies the hostname or IP address of the secondary FTP server and corresponding port. The allowable input is IPv4 address, IPv6 address, or domain name.

### [-ftp\_port\_2 <port>]

Sets or modifies the port of the secondary FTP server and corresponding port. The valid input is an integer between 1 and 65535. If an empty string "" is provided for this option, the port number is reset to the default value of 21.

### [-ftp\_user\_2 <username>]

Sets or modifies the username of the login account on the secondary FTP server.

**Note:** Specifying "" (empty double quotes) reverts to the default value of **onalert**.

# [-ftp\_passwd\_2 [<passwd>]]

Sets or modifies the password of the login account on the secondary FTP server.

**Note:** Specifying "" (empty double quotes) reverts to the default value of **EMCCONNECT**.

### [-ftp\_folder\_2 <path>]

Sets or modifies the path of the folder on the secondary FTP server where the event files have to be deposited.

**Note:** Specifying "" (empty double quotes) reverts to the default value of **incoming**.

# [-ftp\_ipprotocol\_2 {IPV4 | IPV6}]

Sets or modifies the transfer mode of the secondary FTP transport mechanism.

### [-ftp\_mode\_2 { active | passive}]

Sets or modifies the transfer mode of the secondary FTP transport mechanism.

**Note:** Specifying "" (empty double quotes) reverts to the default value of active.

```
[-email_priority {Disabled | 1 | 2 | 3 }]
```

Enables email as a Primary, Secondary, or Tertiary transport mechanism. Specifying **Disabled** removes email as a transport mechanism.

```
[-email_from <email_addr>]
```

Sets or modifies the sender's email address. The maximum number of characters that can be specified is 63.

**Note**: Specifying "" (empty double qoutes) reverts to the default value of connectemc@emc.com.

```
[-email_to <email_addr>[,<email_addr> ]]
```

Sets or modifies the destination email addresses that receive the event files. Multiple email addresses can be specified with a comma separating each address. The maximum number of characters that can be specified is 255.

**Note:** Specifying "" (empty double quotes) reverts to the default value of emailalert@emc.com.

```
[-email_subject <email_subject>]
```

Sets or modifies the subject of the email message.

**Note:** Specifying "" (empty double quotes) reverts to the default value of **CallHome Alert**.

```
[-email_server {<hostname> | <ip_addr>}]
```

Sets or modifies the primary email server that accepts and routes email messages.

**Note:** Specifying "" (empty double quotes) disables the use of the existing email server for this transport mechanism.

```
[-email_ipprotocol {IPV4 | IPV6}]
```

Sets or modifies the secondary email server that accepts and routes email messages.

```
[-email_server_2 {<hostname> | <ip_addr>}]
```

Sets or modifies the secondary email server that accepts and routes email messages.

**Note:** Specifying "" (empty double quotes) disables the use of the existing email server for this transport mechanism.

# [-email\_ipprotocol\_2 {IPV4 | IPV6}]

Sets or modifies the secondary email server that accepts and routes email messages.

### [-dial\_in\_number <phone\_number>]

Sets or modifies the dial-in phone number of the modem.

**Note:** Specifying '"' (empty double quotes) does not disable the number or restore its default value. The empty string is stored as is.

### [-serial\_number <serial\_number>]

Sets or modifies the Celerra serial number, if it was not automatically detected.

**Note:** Specifying '"' (empty double quotes) does not disable the number or restore its default value. The empty string is stored as is.

### [-site\_id <site\_id>]

Sets or modifies the site ID.

**Note:** Specifying "" (empty double quotes) does not disable the number or restore its default value. The empty string is stored as is.

# [-encryption\_enabled {yes | no}]

Enables or disables the encryption of the CallHome payload during transmission.

**Note:** Specifying "" (empty double quotes) reverts to the default value of **yes**.

### [-dial\_in\_enabled {yes | no}]

Enables or disables dial-in login sessions.

**Note:** Specifying "" (empty double quotes) reverts to the default value of **yes**.

**SEE ALSO** *Configuring Celerra Events and Notifications.* 

# **EXAMPLE #1** To display configuration information, type:

# /nas/sbin/nas\_connecthome -info ConnectHome Configuration: Encryption Enabled = yes Dial In : Enabled = yes Modem phone number Site ID = 9123123123 = MY SITE Site ID Serial number = APM00054703223Email: Priority = 1 Sender Address = admin@yourcompany.com Priority Recipient Address(es) = emailalert@emc.com = CallHome Alert Subject Primary: Email Server = backup.mailhub.company.com Secondary: Email Server FTP: = 2 Priority Primary: FTP Server = 1.2.3.4 FTP Port = 22 FTP Vser Name = onalert
FTP Password = \*\*\*\*\*\*\*\*
FTP Remote Folder = incoming
FTP Transfer Mode = active Secondary: FTP Server FTP Port = 1.2.4.4 FTP Port = 22

FTP User Name = onalert

FTP Password = \*\*\*\*\*\*\*\*

FTP Remote Folder = incoming

FTP Transfer Mode = active Modem : = Disabled Priority Primary : Phone Number BT Tymnet = no Secondary: Phone Number BT Tymnet = no

# **EXAMPLE #2** To test the primary email server, type:

```
# /nas/sbin/nas_connecthome -test -email_1
```

ConnectEMC 2.0.27-bl18 Wed Aug 22 10:24:32 EDT 2007

RSC API Version: 2.0.27-bl18

Copyright (C) EMC Corporation 2003-2007, all rights reserved.

\_\_\_\_\_

Reading configuration file: ConnectEMC.ini.

Run Service begin...

Test succeeded for Primary Email.

# nas\_config

Manages a variety of configuration settings on the Control Station, some of which are security based.

#### **SYNOPSIS**

### DESCRIPTION

nas\_config -IPalias configures different IP addresses to point to the same network device allowing use of IP aliasing to manage the Control Station. This enables communication with the primary Control Station using a single IP address regardless of whether the primary Control Station is running in slot 0 or slot 1.

nas\_config -ssl generates an X.509 digital certificate on the Control Station. Unisphere Software uses the Secure Socket Layer (SSL) protocol to create a secure connection between a user's Web browser and the Control Station's Apache Web server. When a Celerra system is initially installed, a generic certificate is generated. After configuring the Control Station's network configuration (hostname, DNS domain name, or IP address) and before using the Unisphere Software, a new certificate should be generated.

**nas\_config -sessiontimeout** sets a system-wide value that automatically times out a Control Station shell session after a specified period of inactivity.

nas\_config -password supports a password quality policy by requiring that passwords chosen by Celerra users adhere to certain rules.

You must be **root** to execute this command from the /nas/sbin directory.

### **OPTIONS**

#### -IPalias -list

Lists IP aliases configured on the Celerra Network Server.

-IPalias -create [-name <device\_name>] <numeric\_id> Creates an IP alias for the Control Station.

<device\_name> is the name for a specified device:

- If you specify a device name that device must have an IP address.
- If you do not specify a device name, the system uses the external network interface.

<numeric\_id> is a user-defined number, and can be an integer between 0 and 255. The system allows up to 256 aliases for any device.

-delete [-name <device\_name>] <numeric\_id>
Deletes an IP alias for the Control Station.

#### -ss1

Installs a SSL certificate on the Control Station and restarts the HTTP server.

```
-sessiontimeout [<number_in_minutes>| off]
```

Displays the current session timeout value in minutes.

<number\_in\_minutes> sets the number of minutes a Control Station shell session can be inactive before it is timed out. Possible values are 5 to 240 minutes. The default value is 60 minutes. Session timeout is enabled by default. To disable session timeout, type **off** or 0 to indicate zero minutes.

The **-sessiontimeout** option enables the native timeout properties of the underlying shells on the Control Station. The relevant shell man page provides a description of how the mechanism works.

# -password

Prompts for specific password policy definitions. The current value for each policy definition is shown in brackets.

```
[-min <6..15>]
```

Defines the minimum length of the new password. The default length is eight characters. The length has to be a value between 6 and 15 characters.

```
[-retries <max_allowed>]
```

Defines the number of attempts a user can make to define an acceptable new password before the command fails. The default value is three attempts.

```
[-newchars <min_num>]
```

Defines the minimum number of characters that must be in the new password that were not included in the old password. The default value is three characters.

```
[-digits <min_num>]
```

Defines the minimum number of digits that must be included in the new password. The default value is one digit.

```
[-spechars <min_num>]
```

Defines the minimum number of special characters (such as ! @ # \$ % & ^ and \*) that must be included in the new password. The default value is 0.

```
[-lcase <min_num>]
```

Defines the minimum number of lowercase characters that must be included in the new password. The default value is 0.

```
[-ucase <min_num>]
```

Defines the minimum number of uppercase characters that must be included in the new password. The default value is 0.

#### -password -default

Resets the password policy definitions to their default values.

**SEE ALSO** *Celerra Security Configuration Guide.* 

**EXAMPLE #1** To create an IP alias for the Control Station, type:

```
# /nas/sbin/nas_config -IPalias -create 0
```

Do you want slot\_0 IP address <1.2.3.4> as your alias [yes or no]: no Please enter an IP address to use as an alias: 1.2.3.6
Do you want slot\_0 IP address <1.2.3.4> as your alias [yes or no]: yes Please enter a new IP address for slot\_0: 1.2.3.6

**EXAMPLE #2** To view the IP alias that you created, type:

# /nas/sbin/nas\_config -IPalias -list

alias IPaddress state eth2:0 1.2.3.6 UP

**EXAMPLE #3** To delete an IP alias, type:

```
# /nas/sbin/nas_config -IPalias -delete 0
```

All current sessions using alias eth2:0 will terminate Do you want to continue [yes or no]: yes done

**EXAMPLE #4** To generate and install a certificate for the Apache Web server on the Control Station, type:

### # /nas/sbin/nas\_config -ssl

Installing a new SSL certificate requires restarting the Apache web server. Do you want to proceed? [y/n]: y
New SSL certificate has been generated and installed successfully.

**EXAMPLE #5** To change the session timeout value from the default value of 60

minutes to 100 minutes, type:

# /nas/sbin/nas\_config -sessiontimeout 100

done

**EXAMPLE #6** To disable session timeout, type:

# /nas/sbin/nas config -sessiontimeout 0

done

or

# /nas/sbin/nas\_config -sessiontimeout off

done

**EXAMPLE #7** To set specific password policy definitions, type:

# /nas/sbin/nas\_config -password

Minimum length for a new password (Between 6 and 15): [8] Number of attempts to allow before failing: [3]

Number of new characters (not in the the old password): [3]

Number of digits that must be in the new password: [1]

Number of special characters that must be in a new password: [0]

Number of lower case characters that must be in password: [0]

Number of upper case characters that must be in password: [0]

**EXAMPLE #8** To set the minimum length of a new password to 10 characters, type:

# /nas/sbin/nas\_config -password -min 10

**EXAMPLE #9** To reset the current password policy definitions to their default values, type:

# /nas/sbin/nas\_config -password -default

# nas\_copy

Creates a replication session for a one-time copy of a filesystem. This command is available with Celerra Replicator V2 only.

### **SYNOPSIS**

```
nas_copy
   -name <sessionName>
-source
        {-fs {<name>|id=<fsId>}|-ckpt {<ckptName>|id=<ckptId>}}
-destination
        {-fs {id=<dstFsId>|<existing_dstFsName>}
        |-pool {id=<dstStoragePoolId>}|<dstStoragePool>}}
[-from_base {<ckpt_name>|id=<ckptId>}]
-interconnect {<name>|id=<interConnectId>}
[-source_interface {<nameServiceInterfaceName>|ip=<ipaddr>}]
[-destination_interface {<nameServiceInterfaceName>|ip=<ipaddr>}]
[-overwrite_destination]
[-refresh]
[-full_copy]
[-background]
```

### DESCRIPTION

**nas\_copy** from the Control Station on the source side, performs a one-time copy of a source read-only filesystem or a checkpoint filesystem.

**Note**: Depending on the size of the data in the source, this command may take some time to complete. Once a copy session begins, you can monitor it or interrupt it if necessary using the <a href="mailto:nas\_task">nas\_task</a> command. You can list all replication sessions, including copy sessions, using the <a href="mailto:nas\_replicate">nas\_replicate</a> -list command.

# **OPTIONS**

Creates a Celerra Replicator V2 session that performs a one-time copy of a source read-only filesystem or a checkpoint filesystem.

The session name assigned must be unique for the Data Mover pair as defined by the interconnect. The naming convention <source\_fs\_or\_ckpt\_name>\_replica<#> is used if a read-only filesystem or checkpoint at the destination already has the same

name as the source. An integer between 1 and 4 is assigned according to how many replicas of that filesystem or checkpoint already exist.

The **-source** specifies the name or ID of an existing read-only filesystem or checkpoint filesystem as the source for this copy session. This is to be used as a common base for the initial transfer. The checkpoint is identified by checkpoint name or checkpoint filesystem ID. This option is intended to accommodate upgrade situations to Celerra Replicator V2.

The **-destination** specifies either an existing destination filesystem or the storage needed to create the destination filesystem automatically as part of the copy operation. An existing destination filesystem must be read-only and the same size as the source. Specifying a storage pool or ID creates the destination filesystem automatically, as read-only, using the same name and size as the source filesystem.

[-from\_base {ckpt\_name>|id=<ckptId>}] Specifies an existing source filesystem checkpoint to be used as a common base for the initial data transfer. The checkpoint is identified by the checkpoint name or ID.

The **-interconnect** specifies the local (source) side of an established Data Mover interconnect to use for this copy session. Use the **nas\_cel -interconnect -list** command on the source Celerra to list the interconnects available to Celerra Replicator V2 sessions.

```
[-source_interface {<nameServiceInterfaceName> | ip=
<ipaddr>}]
```

Instructs the copy session to use a specific local interface defined for the interconnect on the source Celerra instead of selecting the local interface supporting the lowest number of sessions (the default). If this local interface was defined for the interconnect using a name service interface name, specify the name service interface name; if it was defined using an IP address, specify the IP address. The **source\_interfaces** field of the output from the **nas\_cel-interconnect-info** command shows how the source interface was defined. This option does not apply to a loopback interconnect, which always uses 127.0.0.1.

### [-destination\_interface

{<nameServiceInterfaceName>|ip=<ipaddr>}]

Instructs the copy session to use a specific interface defined for the interconnect on the destination Celerra instead of selecting the peer interface supporting the lowest number of sessions (the default). If this peer interface was defined for the interconnect using a name service interface name, specify the name service interface name; if it was defined using an IP address, specify the IP address. The **destination\_interfaces** field of the output from the **nas\_cel -interconnect -info** command shows how the peer interface was defined. This option does not apply to a loopback interconnect, which always uses 127.0.0.1.

### [-overwrite\_destination]

For an existing destination, discards any changes made to the destination object and restores it from the established common base (differential copy). If this option is not specified and an existing destination object contains different content than the established common base, an error is returned.

## [-refresh {<name>|id=<session\_id>}

Updates a destination checkpoint that has the same name as the copied checkpoint. This option does not refresh the source object; it refreshes only the destination for a existing checkpoint. If you specify this option and no checkpoint exists with the same name, the command returns an error.

### [-full\_copy]

For an existing destination object, if a common base checkpoint exists, performs a full copy of the source checkpoint to the destination, instead of a differential copy. If this option is not specified and an existing destination object has different content than the established common base, an error is returned.

#### [-background]

Executes the command in asynchronous mode. Use the nas\_task command to check the status of the command.

# **SEE ALSO** nas\_cel, nas\_replicate, nas\_task.

### **EXAMPLE #1**

To create a one-time copy of a checkpoint filesystem with session name ufs1\_replica1 with the source checkpoint ufs\_ckpt1 and destination pool clar\_r5\_performance on the interconnect NYs3\_LAs2, source interface 10.6.3.190, and destination interface 10.6.3.173, type:

```
$ nas_copy -name ufs1_replica1 -source -ckpt ufs1_ckpt1
-destination -pool clar_r5_performance -interconnect
NYs3_LAs2 -source_interface 10.6.3.190
-destination_interface 10.6.3.173
```

OK

# **EXAMPLE #2**

To create a one-time copy of a read-only filesystem for the session **ufs1\_replica1** with source filesystem **ufs1** and overwrite an existing destination filesystem **ufs1** on the interconnect **NYs3\_LAs2**, source interface **10.6.3.190**, and destination interface **10.6.3.173**, type:

\$ nas\_copy -name ufs1\_replica1 -source -fs ufs1
-destination -fs ufs1 -interconnect NYs3\_LAs2
-source\_interface 10.6.3.190 -destination\_interface
10.6.3.173 -overwrite destination

OK

# **EXAMPLE #3**

To initiate a differential copy of ufs\_ckpt2 to the ufs1\_destination filesystem using ufs1\_ckpt1 as the common base, using the -from\_base option, type:

\$ nas\_copy -name ufs1\_replica1 -source -ckpt -ufs1\_ckpt2
-destination -fs ufs1\_destination -from\_base ufs1\_ckpt1
-interconnect NYs3 LAs2

OK



### CAUTION

Using the -from\_base option overrides any common base that may exist. Make sure that the specified checkpoint represents the correct state of the destination filesystem.

### **EXAMPLE #4**

To refresh the destination of the replication session **ufs1\_replica1** for the source checkpoint **ufs1\_ckpt1** and destination filesystem **ufs1** on the interconnect **NYs3\_LAs2**, type:

\$ nas\_copy -name ufs1\_replica1 -source -ckpt ufs1\_ckpt1
-destination -fs ufs1 -interconnect NYs3\_LAs2 -refresh

OK

### **EXAMPLE #5**

To perform a full copy of the source checkpoint to the destination for the replication session **ufs1\_replica1** with the source filesystem **ufs1** and destination filesystem **ufs1** on the interconnect **NYs3\_LAs2**, type:

\$ nas\_copy -name ufs1\_replica1 -source -fs ufs1
-destination -fs ufs1 -interconnect NYs3\_LAs2
-overwrite destination -full\_copy -background

Info 26843676673: In Progress: Operation is still running. Check task id 4177 on the Task Status screen for results.

# nas cs

Manages the configuration properties of the Control Station.

### **SYNOPSIS**

```
nas cs
     -info [-timezones]
   -set [-hostname <hostname>]
          [-ip4address <ipv4_address>]
          [-ip4netmask <ipv4_netmask>]
          [-ip4gateway <ipv4_gateway>]]
          [-ip6address <ipv6_address[/prefix_length]>]
          [-ip6gateway <ipv6_gateway>]
          [-dns domain <dns domain name>]
          [-search_domains <domain_name>[,...]]
          [-dns_servers <ip_addr>[,...]]
          [-session_idle_timeout <minutes>]
          [-session_monitor_timeout <days>]
          [-time <yyyymmddhhmm [ss]>]
          [-timezone <time_zone_str>]
          [-ntp_servers <ip_addr>[,...]]
     -clear [-ip4gateway]
          [-ip6address]
          [-ip6gateway]
          [-dns]
          [-search_domains]
          [-session_monitor_timeout]
          [-session_idle_timeout]
          [-ntp servers]
     -reboot
```

### **DESCRIPTION**

nas\_cs sets, clears, and lists the Control Station configuration. nas\_cs
can be used to reboot the Control Station.

### **OPTIONS**

### -info [-timezones]

Displays the Control Station configuration. The **-timezones** option displays all supported time zones on the Control Station.

```
-set [-hostname <hostname>]
```

Sets the user configurable paratmeters of the Control Station configuration. Sets the hostname of the primary Control Station. To specify a hostname, the maximum number of characters is 64, excluding white spaces and dot characters.

```
[-ip4address {<ipv4_address>]
```

Sets the IPv4 network address of the primary Control Station. The IPv4 address must be a valid address.

### [-ip4netmask <ipv4\_netmask>]

Sets the subnet mask for a valid IPv4 network address on the primary Control Station.

### [-ip4gateway <ipv4\_gateway>]

Sets the IPv4 network address of the gateway machine for the primary Control Station on the network. The IPv4 address must be a valid address.

# [-ip6address <ipv6\_addr[/prefix\_length]>]

Sets the IPv6 network address of the primary Control Station. The IPv6 address must be a valid address. The /prefix\_length option sets the integer value, between 8 and 128, for the prefix length of the IPv6 address of the primary Control Station.

# [-ip6gateway <ipv6\_gateway>]

Sets the IPv6 network address of the gateway machine for the primary Control Station on the network. The IPv6 address must be a valid address.

### [-dns\_domain <dns\_domain\_name>]

Sets the Domain Name System of which the primary Control Station is a member. It can accept valid domain names.

# [-search\_domains <domain\_name>[,...]]

Sets the order in which DNS domains are searched to resolve a comma separated list of valid domain names.

### [-dns\_servers <ip\_addr>[,...]]

Sets the IP addresses of the DNS servers of the domain. It is a comma separated list of valid IPv4 or IPv6 addresses and can have a maximum of three DNS addresses.

### [-session\_idle\_timeout <minutes>]

Sets the timeout period in minutes for an inactive administrator session to become invalid.

### [-session\_monitor\_timeout <days>]

Sets the limit for the number of days until when a valid login is allowed to run queries on the primary Control Station. Any active management session requires a login on the primary Control Station.

### [-time <yyyymmddhhmm [ss]>]

Sets the current system date and time in the format <yyyymmddhhmm [ss]>].

### [-timezone <time\_zone\_str>]

Sets a valid time zone value on the primary Control Station.

### [-ntp\_servers <ip\_addr>[,...]]

Sets the IP addresses of the NTP server used by the primary Control Station. It is a comma separated list of valid IPv4 or IPv6 addresses and can have a maximum of four NTP addresses.

### -clear [-ip4gateway]

Clears the user configurable paratmeters of the Control Station configuration. Clears the IPv4 network address of the gateway machine for the primary Control Station on the network.

### [-ip6address]

Clears the IPv6 network address of the primary Control Station.

### [-ip6gateway]

Clears the IPv6 network address of the gateway machine for the primary Control Station on the network.

### [-dns]

Clears the Domain Name System of which the primary Control Station is a member.

#### [-search domains]

Clears the order in which DNS domains are searched to resolve a list of valid domain names.

### [-session\_idle\_timeout]

Clears the timeout period in minutes for an inactive administrator session to become invalid.

### [-session\_monitor\_timeout]

Clears the limit for the number of days until when a valid login is allowed to run queries on the primary Control Station. Any active management session requires a login on the primary Control Station.

### [-ntp\_servers]

Clears the IP addresses of the NTP server used by the primary Control Station.

#### -reboot

Reboots the primary Control Station.

# **EXAMPLE #1** To display the configuration properties of the primary Control Station, type:

### \$ nas\_cs -info

```
Host name
                        = eng24416
Version
                        = 6.0
Location
                        = system:NS40G:HK1908075100410000|controlStation::0
Status
Standby location
                       = system: NS40G: HK1908075100410000 | controlStation::1
Stand by status
                        = Ok
IPv4 address
                       = 172.24.250.26
IPv4 gateway
                       = 172.24.250.10
                       = 255.255.255.0
IPv4 netmask
IPv6 address
                      = 2002:ac18:af02:f4:20e:cff:fe6e:d524/64
IPv6 gateway
                      = 2002:ac18:af02:f4:20e:cff:fe6e:d527
DNS Domain
                       = eng.lss.emc.com
DNS Domain search order = eng.lss.emc.com,rtp.lab.emc.com
DNS servers
                       = 2002:ac18:af02:f4:20e:cff:fe6e:d526
Session idle timeout = 10 Minutes
Session moniotor timeout = 10 Days
Current Time
                      = Thu Nov 6 07:54:52 EST 2008
NTP Servers
                        = 2002:ac18:af02:f4:20e:cff:fe6e:d529
```

# **EXAMPLE #2** To set the hostname, IPv4 network address, subnet mask, and IPv4 gateway for the primary Control Station, type:

```
$ nas_cs -set -hostname eng24416 -ip4address
172.24.244.16 -ip4netmask 255.255.255.0 -ip4gateway
128.221.252.0
```

OK

# **EXAMPLE #3** To set the IPv6 address and the IPv6 gateway for the primary Control Station, type:

```
$ nas_cs -set -ip6address
2002:ac18:af02:f4:20e:cff:fe6e:d524/64 -ip6gateway
2002:ac18:af02:f4:20e:cff:fe6e:d527
```

OK

# **EXAMPLE #4** To set the DNS domain, search domains, and DNS servers for the primary Control Station, type:

```
$ nas_cs -set -dns_domain eng.lss.emc.com -search_domain
lss.emc.com,rtp.lab.emc.com -dns_servers
172.24.175.172,172.24.175.173
```

OK

	EXAMPLE #5	To set the session monitor timeout and session idle timeout for the primary Control Station, type:
OK		<pre>\$ nas_cs -set -session_monitor_timeout 2 -session_idle_timeout 30</pre>
	EXAMPLE #6	To set the date, time, timezone, and NTP servers for the primary Control Station, type:
OK		<pre>\$ nas_cs -set -time 200811070205 -timezone America/New_York -ntp_server 128.221.252.0</pre>
	EXAMPLE #7	To clear the IPv4 gateway for the primary Control Station, type:
OK		<pre>\$ nas_cs -clear -ip4gateway</pre>
	EXAMPLE #8	To clear the IPv6 network address and IPv6 gateway for the primary Control Station, type:
OK		<pre>\$ nas_cs -clear -ip6address -ip6gateway</pre>
	EXAMPLE #9	To clear the DNS domain and DNS server configuration for the primary Control Station, type:
OK		<pre>\$ nas_cs -clear -dns</pre>
	EXAMPLE #10	To clear the domain search configuration for the primary Control Station, type:
OK		<pre>\$ nas_cs -clear -search_domains</pre>
	EXAMPLE #11	To clear the NTP server configuration for the primary Control Station, type:
OK		<pre>\$ nas_cs -clear -ntp_servers</pre>
	EXAMPLE #12	To reboot the primary Control Station, type:
		\$ nas_cs -reboot

# nas\_dbtable

Displays the table records of the Control Station.

# **SYNOPSIS**

nas dbtable

To execute the command against a database that is on the Data Mover area:

```
-info -mover <movername> -db <dbname>
-query <tablename> -mover <movername> -db <dbname>
-filter { (<fieldname> <operator> <value> [{-and|-or} <fieldname>{<|<=|>|>=| .CONTAIN.}; <value>]...] ]
-list -mover <movername>
```

### **DESCRIPTION**

Displays the table records of the specified Data Mover. It also filters the records on a particular field, and lists those records by using primary or secondary key values.

To execute the command against a database that is on the Control Station area:

```
-info -cs_path <cs_pathname> -db <dbname>
-query <tablename> -cs_path <cs_pathname> -db <dbname>
-filter {(<fieldname> <operator> <value> [{-and|-or} <fieldname>{<|<=|>|>=| .CONTAIN.};<value>]...] ]
-list -cs path <cs_pathname>
```

### DESCRIPTION

Displays the table records of the Control Station. It also filters the records on a particular field, and lists those records by using primary or secondary key values.

The database located in the Data Mover can be read directly. The backup of the database is read on the Control Station. If the database is inconsistent, the nas\_dbtable command allows to manually verify the backup of the database before restoring it.

The Data Mover table uses the standard XML interface of the administration commands. Application can structure each table data and keys as a set of fields. Each field has a unique name, type, and size.

The table structure is stored in the db.<br/>
sase name> file. It is backed up and restored with the database. The DBMS reader uses this description of the table structure to read and display the records from the backup database.

### **DATA MOVER OPTIONS**

-info -mover <movername> -db <dbname>

Displays the schema of a table or the list of fields and keys. It also displays the number of records of the table so that the user can know if it is reasonable to dump all the table.

-query <tablename> -mover <movername> -db <dbname>
Displays the records of a table. Selects the records to display on the value of some fields or secondary keys.

```
-filter { (<fieldname><operator><value> [{-and|-or} <fieldname>{<|<=|>|>=|=|.CONTAIN.};<value>]...] ]
Filters the records of a particular field, and lists the records using primary or secondary key values. The default with multiple filters is the -and option. Only the = operator is supported in the first implementation.
```

**Note:** The keys are used when the **-and** option is used. Multiple fields with the **-or** option parses the table, and applies filter on each record.

The <fieldname> argument is the name of a secondary key or field. If the secondary key is declared as a sequence of fields, it is used by specifying either the value of its fields or value. If the secondary key is not declared in the schema, then rename the key and its value as filter.

The <value> argument is the value of the field encoded in character.

# CONTROL STATION OPTIONS

-info -cs\_path <cs\_pathname> -db <dbname>

Displays the schema of a table or the list of fields and keys. It also displays the number of records of the table so that the user can know if it is reasonable to dump all the table.

-query <tablename> -cs\_path <cs\_pathname> -db <dbname>
Displays the records of the table. Selects the records to display on the value of some fields or secondary keys.

```
-filter { (<fieldname><operator><value> [{-and|-or} <fieldname>{<|<=|>|>=|=|.CONTAIN.};<value>]...] ]
Filters the records of a particular field, and lists the records using primary or secondary key values. The default with multiple filters is the -and option. Only the = operator is supported in the first implementation.
```

**Note:** Keys are used when the **-filter** option contains all components of the key, and the **-and** option is used. With the **-or** option, it is necessary to parse all the records.

The <fieldname> argument is the name of a secondary key or field. If the secondary key is declared as a sequence of fields, it is used by specifying either the value of its fields or the secondary key value. If the secondary key is not declared in the schema, rename the key and its value as filter.

The <value> argument is the value of the field encoded in character.

### -list -cs\_path <cs\_pathname>

Displays the list of databases and tables within a particular directory of the Control Station area.

# **SEE ALSO** server\_dbms

### EXAMPLE #1

To display the **Secmap** schema of the Data Mover, type:

\$ nas\_dbtable -info -mover <movername> -db Secmap

```
Database identification
```

Base Name = Secmap
Table Name = Mapping

Primary Key Schema

sid = SID

Secondary Key Components

xid = xidType, fxid

Data Schema

origin = Enumeration

Unknown : 0 Secmap : 16 Localgroup : 32 Etc : 48 Nis : 64

```
AD
                                 : 80
                      Usrmap
                               : 96
                      Ldap
                                : 112
                                : 128
                      Ntx
xidType
                    = Enumeration
                      unknown name: -2
                      unknown_sid : -1
                      unknown_type : 0
                      user
                                  : 1
                                 : 2
                      group
fxid
                    = Unsigned Integer size : 4
cdate
                   = Date
gid
                    = Unsigned Integer size : 4
                    = String, length container size : 2
name
```

# **EXAMPLE #2** To filter the records of the Secmap schema, type:

# \$ nas\_dbtable -query Mapping -mover <movername> -db Secmap -filter fxid=10011

sid = S-1-5-15-2b3be507-6bc5c62-3f32a78a-8cc

 $\begin{array}{lll} \text{origin} & = \text{Nis} \\ \text{xidType} & = \text{user} \\ \text{fxid} & = 10011 \end{array}$ 

cdate = Fri Sep 11 17:39:09 2009

gid = 107

name =  $DVT2KA\MaxUsers0000011$ 

Record count = 1

Last key = 105000000000051500000007e53b2b625cbc068aa7323fcc080000

# nas\_devicegroup

Manages an established MirrorView/Synchronous (MirrorView/S) consistency group, also known as a device group.

### **SYNOPSIS**

```
nas_devicegroup
-list
| -info {<name>|id=<id>|-all} [-sync [yes|no]]
| -acl <acl_value> {<name>|id=<id>}
| -suspend {<name>|id=<id>}
| -resume {<name>|id=<id>}
```

### **DESCRIPTION**

nas\_devicegroup lists the device group information for a MirrorView/S configuration, gets detailed information about a consistency group, specifies an access control level value for the group, suspends MirrorView/S operations, or resumes operations of the device group.

A MirrorView/S with a Celerra Network Server configuration involves source and destination Celerra Network Servers attached to CLARiiON CX Series storage systems. MirrorView/S performs synchronous mirroring of source storage logical units (LUNs) representing production images, where the mirrored LUNs are part of a MirrorView/S consistency group.

On the source Celerra, you must be **root** to issue the **-acl**, **-suspend**, and **-resume** options.

**nas\_devicegroup** must be run from a Control Station in slot 0; it will report an error if run from a Control Station in slot 1.

### **OPTIONS**

### -list

Displays a list of available configured MirrorView/S device groups.

```
-info {<name>|id=<id>|-all} [-sync [yes|no]]
Displays detailed information about the MirrorView/S configuration for a specific device group or for all groups.
```

```
[-sync [yes no]]
```

The **-sync** option first synchronizes the Control Station's view with the CLARiiON storage system before displaying configuration information. The default is **yes**.

```
-acl <acl_value> {<name>|id=<id>}
```

Sets an access control level value that defines the owner of the storage system, and the level of access allowed for users and groups defined in the access control level table. The nas\_acl command provides more information).



### **CAUTION**

The access control level value for the group should not be changed from the default setting. A change in access control level value can prevent MirrorView/S from functioning properly.

```
-suspend {<name>|id=<id>}
```

Temporarily halts mirroring from the source to the destination, thereby suspending the link. Changes can still be made to the source LUNs, but are not applied to the destination LUNs until operations are resumed.

```
-resume {<name> | id=<id>}
```

Resumes device group operations and restarts mirroring, synchronizing the destination LUNs with the source LUNs.

### SEE ALSO

*Using MirrorView/Synchronous with Celerra for Disaster Recovery,* nas\_acl, and nas\_logviewer.

# STORAGE SYSTEM OUTPUT

The number associated with the storage device is dependent on the attached storage system of the system; for MirrorView/S, some CLARiiON storage systems display a prefix of APM before a set of integers, for example, APM00033900124-0019. The CLARiiON storage system supports the following system-defined AVM storage pools for MirrorView/S only: cm\_r1, cm\_r5\_performance, cm\_r5\_economy, cmata\_archive, and cmata\_r3.

### **EXAMPLE #1**

To list the configured MirrorView/S device groups that are available, type:

```
$ nas_devicegroup -list
```

```
ID name owner storage ID acl type 2 mviewgroup 500 APM00053001549 0 MVIEW
```

### **EXAMPLE #2**

To display detailed information for a MirrorView/S device group, type:

### \$ nas\_devicegroup -info mviewgroup

recovery policy = Automatic

number of mirrors = 16 mode = SYNC owner = 500

mirrored disks =

# Where:

Value	Definition
Sync with CLARiiON storage system	Indicates that a sync with the CLARiiON storage was performed to retrieve the most recent information. This does not appear if you specify <b>-info -sync no</b> .
name	Name of the device group.
description	Brief description of device group.
uid	UID assigned, based on the system.
state	State of the device group (for example, Consistent, Synchronized, Out-of-Sync, Synchronizing, Scrambled, Empty, Incomplete, or Local Only).
role	Whether the current system is the Primary (source) or Secondary (destination).
condition	Whether the group is functioning (Active), Inactive, Admin Fractured (suspended), Waiting on Sync, System Fractured (which indicates link down), or Unknown.
recovery policy	Type of recovery policy (Automatic is the default and recommended value for group during storage system configuration; if Manual is set, use <b>-resume</b> after a link down failure).
number of mirrors	Number of mirrors in group.
mode	MirrorView mode (always SYNC in this release).
owner	User whom the object is assigned to, and is indicated by the index number in the access control level table. nas_acl provides information.
mirrored disks	Comma-separated list of disks that are mirrored.
local clarid	APM number of local CLARiiON storage array.
remote clarid	APM number of remote CLARiiON storage array.
mirror direction	On primary system, local to remote (on primary system); on destination system, local from remote.

### **EXAMPLE #3**

To display detailed information about a MirrorView/S device group without synchronizing the Control Station's view with the CLARiiON, type:

### \$ nas\_devicegroup -info id=2 -sync no

name = mviewgroup

description =

uid = 50:6:1:60:B0:60:27:20:0:0:0:0:0:0:0

### The nas Commands

number of mirrors = 16 mode = SYNC owner = 500

mirrored disks

# **EXAMPLE #4** To halt operation of the specified device group, as **root** user, type:

### \$ nas\_devicegroup -suspend mviewgroup

Sync with CLARiiON backend ..... done
STARTING an MV 'SUSPEND' operation.
Device group: mviewgroup ...... done
The MV 'SUSPEND' operation SUCCEEDED.
done

# **EXAMPLE #5** To resume operations of the specified device group, as **root** user, type:

### \$ nas\_devicegroup -resume mviewgroup

Sync with CLARiiON backend ..... done
STARTING an MV 'RESUME' operation.
Device group: mviewgroup ....... done
The MV 'RESUME' operation SUCCEEDED.
done

# nas disk

Manages the disk table.

# **SYNOPSIS**

```
nas_disk
  -list
| -delete <disk_name> [[-perm] | [-unbind]]
| -info {<disk_name> | id = <disk_id>}
| -rename <old_name> <new_name>
```

# **DESCRIPTION**

**nas\_disk** displays a list of known disks and renames, deletes, or displays information for the specified disk.

# **OPTIONS**

#### -list

Lists the disk table.

**Note:** The ID of the object is an integer and is assigned automatically. The name of the disk might be truncated if it is too long for the display. To display the full name, use the **-info** option with the disk ID.

```
-delete <disk_name> [[ -perm] | [-unbind]
```

Deletes an entry from the disk table. In a CLARiiON storage system, restores the CLARiiON LUN name to its default value.

Unless -perm is specified, the disk is still identified as a Celerra disk and can be discovered and marked again using server\_devconfig. The -perm option removes the entry from the disk table and deletes the diskmark. The disk is then available to be deployed for use by another platform. The -unbind option removes the LUN from the Celerra Storage group (if EMC Access Logix™ is enabled). The -unbind option permanently destroys the LUN and its contents. If this is the last LUN using the RAID group, then the RAID group will be deleted.

```
-info {<disk_name>|id=<disk_id>}
```

Displays information for a specific <disk\_name> or <disk\_id> such as size, type, and ACL.

```
-rename <old_name> <new_name> Renames a disk to <new_name>.
```

**Note:** If a CLARiiON LUN uses the default EMC Unisphere Software name, renames it in the format

Celerra\_<celerra-hostname>\_<lun-id>\_<Celerra-dvol-name>.

### SEE ALSO

Celerra Network Server System Operations and server\_devconfig.

# STORAGE SYSTEM OUTPUT

The number associated with the storage device is dependent on the attached storage system. CLARiiON storage systems display a prefix of APM before a set of integers, for example, APM00033900124-0019. Symmetrix storage systems appear as 002804000190-003C.

# **EXAMPLE #1**

To list the disk table for a Celerra Network Server with a CLARiiON storage system, type:

### \$ nas\_disk -list

			,			
id	inuse	sizeMB	storageID-devID	type	name	servers
1	У	11263	APM00042000818-0000	CLSTD	root_disk	1,2,3,4
2	У	11263	APM00042000818-0001	CLSTD	root_ldisk	1,2,3,4
3	У	2047	APM00042000818-0002	CLSTD	d3	1,2,3,4
4	У	2047	APM00042000818-0003	CLSTD	d4	1,2,3,4
5	У	2047	APM00042000818-0004	CLSTD	d5	1,2,3,4
6	У	2047	APM00042000818-0005	CLSTD	d6	1,2,3,4
7	У	273709	APM00042000818-0012	CLSTD	d7	1,2,3,4
8	n	273709	APM00042000818-0013	CLSTD	d8	1,2,3,4
9	У	273709	APM00042000818-0014	CLSTD	d9	1,2,3,4
10	n	273709	APM00042000818-0015	CLSTD	d10	1,2,3,4
11	n	273709	APM00042000818-0016	CLSTD	d11	1,2,3,4
12	n	273709	APM00042000818-0017	CLSTD	d12	1,2,3,4
13	У	1023	FCNTR074200038-0032	CLEFD	d13	1,2,3,4
14	У	2047	FCNTR074200038-0033	CLEFD	d14	1,2,3,4
15	n	273709	APM00042000818-001A	CLSTD	d15	1,2,3,4
16	У	273709	APM00042000818-001B	CLSTD	d16	1,2,3,4
17	n	273709	APM00042000818-001C	CLSTD	d17	1,2,3,4
18	У	273709	APM00042000818-001D	CLSTD	d18	1,2,3,4
19	n	273709	APM00042000818-001E	CLSTD	d19	1,2,3,4
20	У	273709	APM00042000818-001F	CLSTD	d20	1,2,3,4
21	n	456202	APM00042000818-0020	CLATA	d21	1,2,3,4
22	n	456202	APM00042000818-0021	CLATA	d22	1,2,3,4
23	n	456202	APM00042000818-0022	CLATA	d23	1,2,3,4
24	n	456202	APM00042000818-0023	CLATA	d24	1,2,3,4

### Where:

Value	Definition
id	ID of the disk (assigned automatically).
inuse	Used by any type of volume or filesystem.
sizeMB	Total size of disk.

storageID-devID	ID of the storage system and device associated with the disk.
type	Type of disk contingent on the storage system attached; CLSTD, CLATA, CMSTD, CLEFD, CMEFD, and CMATA are CLARiiON disk types and STD, BCV, R1BCV, R2BCV, R1STD, R2STD, ATA, R1ATA, R2ATA,BCVA, R1BCA, R2BCA, and EFD are Symmetrix disk types.
name	Name of the disk; 'dd' in a disk name indicates a remote disk.
servers	Servers that have access to this disk.

# **EXAMPLE #2** To list the disk table for the Celerra Network Server with a Symmetrix storage system, type:

# \$ nas\_disk -list

id	inuse	sizeMB	storageID-devID	type	name	servers
1	У	11507	000190100530-00FB	STD	root_disk	1,2,3,4,5,6,7,8
2	У	11507	000190100530-00FC	STD	root_ldisk	1,2,3,4,5,6,7,8
3	У	2076	000190100530-00FD	STD	d3	1,2,3,4,5,6,7,8
4	У	2076	000190100530-00FE	STD	d4	1,2,3,4,5,6,7,8
5	У	2076	000190100530-00FF	STD	d5	1,2,3,4,5,6,7,8
6	У	65536	000190100530-04D4	STD	d6	1,2,3,4,5,6,7,8
7	n	28560	000190100530-0102	STD	d7	1,2,3,4,5,6,7,8
8	n	28560	000190100530-0103	STD	d8	1,2,3,4,5,6,7,8
9	n	28560	000190100530-0104	STD	d9	1,2,3,4,5,6,7,8
10	n	28560	000190100530-0105	STD	d10	1,2,3,4,5,6,7,8
11	n	28560	000190100530-0106	STD	d11	1,2,3,4,5,6,7,8
12	n	28560	000190100530-0107	STD	d12	1,2,3,4,5,6,7,8
13	n	28560	000190100530-0108	STD	d13	1,2,3,4,5,6,7,8
14	n	28560	000190100530-0109	STD	d14	1,2,3,4,5,6,7,8
15	n	28560	000190100530-010A	STD	d15	1,2,3,4,5,6,7,8
16	n	28560	000190100530-010B	STD	d16	1,2,3,4,5,6,7,8
17	n	28560	000190100530-010C	STD	d17	1,2,3,4,5,6,7,8
18	n	28560	000190100530-010D	STD	d18	1,2,3,4,5,6,7,8
19	n	28560	000190100530-010E	STD	d19	1,2,3,4,5,6,7,8
20	n	28560	000190100530-010F	STD	d20	1,2,3,4,5,6,7,8
21	n	28560	000190100530-0110	STD	d21	1,2,3,4,5,6,7,8
22	n	28560	000190100530-0111	STD	d22	1,2,3,4,5,6,7,8
23	n	28560	000190100530-0112	STD	d23	1,2,3,4,5,6,7,8
24	n	28560	000190100530-0113	STD	d24	1,2,3,4,5,6,7,8
[	.]					
155	n	28560	000190100530-0196	STD	d155	1,2,3,4,5,6,7,8
156	n	28560	000190100530-0197	STD	d156	1,2,3,4,5,6,7,8
157	n	28560	000190100530-0198	BCV	rootd157	1,2,3,4,5,6,7,8
158	n	28560	000190100530-0199	BCV	rootd158	1,2,3,4,5,6,7,8
159	n	28560	000190100530-019A	BCV	rootd159	1,2,3,4,5,6,7,8
160	n	28560	000190100530-019B	BCV	rootd160	1,2,3,4,5,6,7,8
161	n	28560	000190100530-019C	BCV	rootd161	1,2,3,4,5,6,7,8
162	n	28560	000190100530-019D	BCV	rootd162	1,2,3,4,5,6,7,8
163	n	28560	000190100530-019E	BCV	rootd163	1,2,3,4,5,6,7,8
164	n	28560	000190100530-019F	BCV	rootd164	1,2,3,4,5,6,7,8
165	n	28560	000190100530-01A0	BCV	rootd165	1,2,3,4,5,6,7,8
166	n	28560	000190100530-01A1	BCV	rootd166	1,2,3,4,5,6,7,8
167	n	28560	000190100530-01A2	BCV	rootd167	1,2,3,4,5,6,7,8
168	n	28560	000190100530-01A3	BCV	rootd168	1,2,3,4,5,6,7,8

169	n	28560	000190100530-01A4	BCV	rootd169	1,2,3,4,5,6,7,8
170	n	28560	000190100530-01A5	BCV	rootd170	1,2,3,4,5,6,7,8
171	n	28560	000190100530-01A6	BCV	rootd171	1,2,3,4,5,6,7,8
172	n	28560	000190100530-01A7	BCV	rootd172	1,2,3,4,5,6,7,8
173	n	28560	000190100530-01A8	BCV	rootd173	1,2,3,4,5,6,7,8
174	n	28560	000190100530-01A9	BCV	rootd174	1,2,3,4,5,6,7,8
175	n	28560	000190100530-01AA	BCV	rootd175	1,2,3,4,5,6,7,8
176	n	28560	000190100530-01AB	BCV	rootd176	1,2,3,4,5,6,7,8
177	n	28560	000190100530-01AC	BCV	rootd177	1,2,3,4,5,6,7,8
178	n	28560	000190100530-01AD	BCV	rootd178	1,2,3,4,5,6,7,8
179	n	28560	000190100530-01AE	BCV	rootd179	1,2,3,4,5,6,7,8
180	n	28560	000190100530-01AF	BCV	rootd180	1,2,3,4,5,6,7,8
181	n	28560	000190100530-01B0	BCV	rootd181	1,2,3,4,5,6,7,8
182	n	28560	000190100530-01B1	BCV	rootd182	1,2,3,4,5,6,7,8
183	n	28560	000190100530-01B2	BCV	rootd183	1,2,3,4,5,6,7,8
184	n	28560	000190100530-01B3	BCV	rootd184	1,2,3,4,5,6,7,8
185	n	28560	000190100530-01B4	BCV	rootd185	1,2,3,4,5,6,7,8
186	n	28560	000190100530-01B5	BCV	rootd186	1,2,3,4,5,6,7,8
187	n	11507	000190100530-051D	EFD	d187	1,2,3,4,5,6,7,8
188	n	11507	000190100530-051E	EFD	d188	1,2,3,4,5,6,7,8
189	n	11507	000190100530-051F	EFD	d189	1,2,3,4,5,6,7,8
190	n	11507	000190100530-0520	EFD	d190	1,2,3,4,5,6,7,8
191	n	11507	000190100530-0521	EFD	d191	1,2,3,4,5,6,7,8
192	n	11507	000190100530-0522	EFD	d192	1,2,3,4,5,6,7,8
193	n	11507	000190100530-0523	EFD	d193	1,2,3,4,5,6,7,8
194	n	11507	000190100530-0524	EFD	d194	1,2,3,4,5,6,7,8
195	n	11507	000190100530-0525	EFD	d195	1,2,3,4,5,6,7,8
196	n	11507	000190100530-0526	EFD	d196	1,2,3,4,5,6,7,8
197	n	11507	000190100530-0527	EFD	d197	1,2,3,4,5,6,7,8
198	n	11507	000190100530-0528	EFD	d198	1,2,3,4,5,6,7,8
199	n	11507	000190100530-0529	EFD	d199	1,2,3,4,5,6,7,8
200	n	11507	000190100530-052A	EFD	d200	1,2,3,4,5,6,7,8
201	n	11507	000190100530-052B	EFD	d201	1,2,3,4,5,6,7,8
202	n	11507	000190100530-052C	EFD	d202	1,2,3,4,5,6,7,8
203	n	11507	000190100530-052D	EFD	d203	1,2,3,4,5,6,7,8
204	У	11507	000190100530-052E	EFD	d204	1,2,3,4,5,6,7,8

**Note:** This is a partial listing due to the length of the output.

EXAMPLE #1 provides a description of command outputs.

# **EXAMPLE #3**

To view information for disk **d7** for a Celerra Network Server with a CLARiiON storage system, type:

\$ nas\_disk -info d7

```
id = 7
name = d7
ac1 = 0
in_use = True
size (MB) = 273709
type = CLSTD
```

```
protection= RAID5(4+1)
stor_id = APM00042000818
stor_dev = 0012
volume_name = d7
storage_profiles = engineer_APM00042000818
virtually_provisioned = True
mirrored = False
servers = server_2, server_3, server_4, server_5
   server = server_2
                             addr=c0t112
   server = server_2
                             addr=c32t112
                            addr=c16t112
  server = server_2
                            addr=c48t112
  server = server 2
  server = server_3
                            addr=c0t112
  server = server_3
                            addr=c32t112
                            addr=c16t112
  server = server_3
  server = server_3
                            addr=c48t112
                            addr=c0t112
  server = server_4
                            addr=c32t112
   server = server_4
  server = server_4
                            addr=c16t112
                            addr=c48t112
   server = server_4
   server = server_5
                            addr=c0t112
  server = server_5
                            addr=c32t112
  server = server_5
                            addr=c16t112
                            addr=c48t112
   server = server_5
```

### Where:

Value	Definition
id	ID of the disk (assigned automatically).
name	Name of the disk.
acl	Access control level value of the disk.
in_use	Used by any type of volume or filesystem.
size (MB)	Total size of disk.
type	Type of disk contingent on the storage system attached; CLARiiON disk types are CLSTD, CLATA, CMSTD, CLEFD, CLSAS, CMEFD, and CMATA.
protection	The type of disk protection that has been assigned.
stor_id	ID of the storage system associated with the disk.
stor_dev	ID of the device associated with the disk.
volume_name	Name of the volume residing on the disk.
storage_profiles	The storage profiles to which the disk belongs.
virtually_provisioned	Indicates whether EMC Virtual Provisioning™ is enabled or disabled.
mirrored	Indicates whether the disk is mirrored or not.
servers	Lists the servers that have access to this disk.
addr	Path to storage system (SCSI address).

# **EXAMPLE #4**

To view information for disk **d205** for the Celerra Network Server with a Symmetrix storage system, type:

\$ nas\_disk -info d205

```
id
         = 205
        = d205
name
acl
        = 0
in_use = False
size (MB) = 28560
       = STD
type
protection= TDEV
symm id = 000190100530
symm_dev = 0539
volume_name = d205
storage_profiles = symm_std
virtually_provisioned = True
server_2, server_3, server_4, server_5, server_6, server_7, server_8, server_9
  server = server_2
                            addr=c0t1410
                                              FA=03A
                                                        FAport=0
  server = server_2
                             addr=c16t1410
                                              FA=04A
                                                        FAport=0
                             addr=c0t1410
                                              FA=03A FAport=0
  server = server_3
                                              FA=04A FAport=0
FA=03A FAport=0
  server = server_3
                             addr=c16t1410
                             addr=c0t1410
  server = server_4
                             addr=c16t1410
                                              FA=04A FAport=0
  server = server_4
  server = server 5
                             addr=c0t1410
                                              FA=03A FAport=0
  server = server_5
                             addr=c16t1410
                                              FA=04A FAport=0
  server = server_6
                             addr=c0t1410
                                              FA=03A FAport=0
  server = server_6
                             addr=c16t1410
                                              FA=04A FAport=0
                                              FA=03A FAport=0
                             addr=c0t1410
  server = server_7
                                              FA=04A FAport=0
  server = server_7
                             addr=c16t1410
  server = server_8
                            addr=c0t1410
                                              FA=03A FAport=0
                            addr=c16t1410
                                              FA=04A FAport=0
  server = server_8
  server = server 9
                             addr=c0t1410
                                              FA=03A FAport=0
  server = server_9
                             addr=c16t1410
                                              FA=04A
                                                        FAport=0
```

### Where:

Value	Definition
id	ID of the disk (assigned automatically).
name	Name of the disk.
acl	Access control level value of the disk.
in_use	Used by any type of volume or filesystem.
size (MB)	Total size of disk.
type	Type of disk contingent on the storage system attached; Symmetrix disk types are STD, BCV, R1BCV, R2BCV, R1STD, R2STD, ATA, R1ATA, R2ATA, BCVA, R1BCA, R2BCA, and EFD.
protection	The type of disk protection that has been assigned.
symm_id	ID of the Symmetrix storage system associated with the disk.
symm_dev	ID of the Symmetrix device associated with the disk.

volume_name	Name of the volume residing on the disk.
storage_profiles	The storage profiles to which the disk belongs.
virtually_provisioned	Indicates whether Virtual Provisioning is enabled or disabled.
servers	Lists the servers that have access to this disk.
addr	Path to storage system (SCSI address).

# **EXAMPLE #5**

To rename a disk in the Celerra Network Server with a CLARiiON storage system, type:

\$ nas\_disk -rename d7 disk7

```
= 7
id
         = disk7
name
         = 0
acl
in_use
        = True
size (MB) = 273709
type = CLSTD
protection= RAID5(4+1)
stor_id = APM00042000818
stor_dev = 0012
volume_name = disk7
storage_profiles = clar_r5_performance
virtually_provisioned = False
mirrored = False
        = server_2, server_3, server_4, server_5
servers
   server = server_2
                              addr=c0t112
   server = server 2
                            addr=c32t112
                             addr=c16t112
   server = server_2
                            addr=c48t112
   server = server_2
   server = server 3
                            addr=c0t112
   server = server_3
                             addr=c32t112
                            addr=c16t112
   server = server_3
   server = server 3
                            addr=c48t112
                             addr=c0t112
   server = server_4
   server = server_4
                              addr=c32t112
   server = server_4
                              addr=c16t112
   server = server_4
                             addr=c48t112
   server = server_5
                              addr=c0t112
   server = server_5
                              addr=c32t112
                              addr=c16t112
   server = server_5
                              addr=c48t112
   server = server_5
```

EXAMPLE #3 provides a description of command outputs.

# **EXAMPLE #6**

To delete a disk entry from the disk table for the Celerra Network Server with a CLARiiON storage system, type:

```
$ nas_disk -delete d24
```

 $\begin{array}{lll} \text{id} & = 24 \\ \text{name} & = \text{d24} \\ \text{acl} & = 0 \\ \end{array}$ 

### **The nas Commands**

```
in_use = False
size (MB) = 456202
type = CLATA
protection= RAID5(6+1)
stor_id = APM00042000818
stor_dev = 0023
storage_profiles = clarata_archive
virtually_provisioned = False
mirrored = False
servers = server_2, server_3, server_4, server_5
```

EXAMPLE #3 provides a description of command outputs.

# nas\_diskmark

Queries the storage system and lists the SCSI devices.

# **SYNOPSIS**

```
nas_diskmark
  -mark {-all | <movername>} [-discovery {y | n}] [-monitor {y | n}]
  [-Force {y | n}]
| -list {-all | <movername>}
```

### DESCRIPTION

nas\_diskmark queries the available storage system device and tape device configuration; saves the device configuration into the Data Movers database, and lists SCSI devices.



### **CAUTION**

It is recommended that all Data Movers have the same device configuration. When adding devices to the device table for a single Data Mover only, certain actions such as standby failover are not successful unless the standby Data Mover has the same disk device configuration as the primary Data Mover.

The **-all** option executes the command for all Data Movers.

### **OPTIONS**

```
-mark {-all|<movername>}
```

Queries SCSI devices and saves them into the device table database on the Data Mover.

Modifies CLARiiON LUN names to the

Celerra\_<celerra-hostname>\_<lun-id>\_<Celerra-dvol-name> format, if the LUNs use the default Unisphere Software name.



### **CAUTION**

The time taken to complete this command may be lengthy, dependent on the number and type of attached devices.

```
[\,\texttt{-discovery}\ \{\textbf{y}\,\big|\,\textbf{n}\}\,]
```

Enables or disables the storage discovery operation.



### **CAUTION**

Disabling the -discovery option should be done only under the direction of an EMC Customer Service Engineer.

```
[-monitor {y | n}]
```

Displays the progress of the query and discovery operations.

```
[-Force {y | n}]
```

Overrides the health check failures and changes the storage configuration.



# **CAUTION**

Use the -Force option only when directed to do so, as high availability can be lost when changing storage configuration.

```
-list {-all | <movername>}
```

Lists the SCSI devices for the specified Data Mover or all Data Movers.

### **EXAMPLE #1**

To query SCSI devices on server\_2 and display the progress of the query operation, type:

```
$ nas_diskmark -mark server_2 -monitor y
```

Discovering storage (may take several minutes)

```
server_2:
chain 0 ......
chain 16 .....
chain 32 .....
chain 48 .....
chain 96 .....
chain 112 .....

Verifying disk reachability
Verifying file system reachability
Verifying local domain
Verifying disk health
Verifying gate keepers
Verifying device group
done
```

# **EXAMPLE #2** To list the SCSI devices for server\_2, type:

```
$ nas_diskmark -list server_2
```

```
tid/lun= 1/0 type= disk sz= 274811 val= 7 info= DGC RAID 5 03244400100044NI
  tid/lun= 1/1 type= disk sz= 274811 val= -5 info= DGC RAID 5 03244500110045NI
  tid/lun= 1/2 type= disk sz= 274811 val= 8 info= DGC RAID 5 03244600120046NI
  tid/lun= 1/3 type= disk sz= 274811 val= -5 info= DGC RAID 5 03244700130047NI
  tid/lun= 1/4 type= disk sz= 274811 val= 9 info= DGC RAID 5 03245600140056NI
  tid/lun= 1/5 type= disk sz= 274811 val= -5 info= DGC RAID 5 03245700150057NI
  tid/lun= 1/6 type= disk sz= 274811 val= 10 info= DGC RAID 5 03245800160058NI
  tid/lun= 1/7 type= disk sz= 274811 val= -5 info= DGC RAID 5 03245900170059NI
  tid/lun= 1/8 type= disk sz= 274811 val= 99 info= DGC RAID 5 03245A0018005ANI
  tid/lun= 1/9 type= disk sz= 274811 val= -5 info= DGC RAID 5 03245B0019005BNI
  tid/lun= 1/10 type= disk sz= 274811 val= 97 info= DGC RAID 5 03245C001A005CNI
  tid/lun= 1/11 type= disk sz= 274811 val= -5 info= DGC RAID 5 03245D001B005DNI
  tid/lun= 1/12 type= disk sz= 274811 val= 13 info= DGC RAID 5 03245E001C005ENI
  tid/lun= 1/13 type= disk sz= 274811 val= -5 info= DGC RAID 5 03245F001D005FNI
  tid/lun= 1/14 type= disk sz= 274811 val= 14 info= DGC RAID 5 032460001E0060NI
  tid/lun= 1/15 type= disk sz= 274811 val= -5 info= DGC RAID 5 032461001F0061NI
server_2 : chain 1 :
no drives on chain
server_2 : chain 2 :
no drives on chain
server_2 : chain 3 :
no drives on chain
server_2 : chain 4 :
no drives on chain
server_2 : chain 5 :
no drives on chain
server 2 : chain 6 :
no drives on chain
server_2 : chain 7 :
no drives on chain
server_2 : chain 8 :
no drives on chain
server_2 : chain 9 :
no drives on chain
server 2 : chain 10 :
no drives on chain
server_2 : chain 11 :
no drives on chain
server_2 : chain 12 :
no drives on chain
server_2 : chain 13 :
no drives on chain
server_2 : chain 14 :
no drives on chain
server_2 : chain 15 :
no drives on chain
```

Note: This is a partial listing due to the length of the output.

# nas emailuser

Manages email notifications for serious system events.

#### **SYNOPSIS**

```
nas_emailuser
   -info
| -test
| -modify
      [-enabled {yes|no}]
      [-to <email_addr> [,...]]
      [-cc <email_addr> [,...]]
      [-email_server <email_server>]
      [-subject_prefix <email_subject>]
      [-from <email_addr>]
| -init
```

#### **DESCRIPTION**

**nas\_emailuser** enables, configures, and tests email notifications for serious system events.

#### **OPTIONS**

#### -info

Displays the configuration for email notifications.

#### -test

Generates a test event that sends a test email notification to the email addresses configured in **-to** and **-cc**. The recipient email address must be configured prior to testing email notification.

**Note**: After the **-test** option is run, all the configured recipients must be asked to confirm whether they received the test email with the correct system identification information.

#### -modify

Modifies one or more of the following configuration parameters:

```
[-enabled {yes no}]
```

Enables email notification if **yes** is specified. The recipient email address must be configured prior to enabling email notification. Disables email notification if **no** is specified.

```
[-to <email_addr> [,...]]
```

Configures one or more recipient email addresses. The email addresses are comma-separated, enclosed in single-quotes, and follow the mailbox@fully\_qualified\_domain\_name format. For example, 'storage\_admin@yourcompany.com', 'backup\_admin@yourcompany.com'.

Refer the following email address format guidelines for configuring email addresses. An email address can contain:

- A maximum of 63 characters; the field can contain a maximum of 255 characters:
  - ASCII characters: a through z, A through Z, 0 through 9,!#
     % & \* + / = ? ^ \_ ` { | ,} ~ are allowed; a period, if it is not the first or last character in the mailbox
  - Alphanumeric strings
- Single quotes, if they are escaped in the format:
  - your\'email@yourcompany.com
  - 'first'\"email@yourcompany.com,second'\"email@yourcompany.com'

```
[-cc <email_addr> [,...]]
```

Configures a list of carbon-copy recipients. The email addresses are comma-separated, enclosed in single-quotes, and follow the mailbox@fully\_qualified\_domain\_name format. For example, 'storage\_admin@yourcompany.com'. For the email address character set and format guidelines, refer the **-to** option.

```
[-email_server <email_server>]
```

Configures the email server that accepts and routes the email notifications. <email\_server> specifies an IP address or the fully qualified domain name, which can have 1 to 63 characters. The IP addresses 0.0.0.0 and 255.255.255.255 are not allowed.

```
[-subject_prefix <email_subject>]
```

Specifies the email subject prefix. The subject prefix for the email notification can be from 1 to 63 characters long, is enclosed in quotes, and should contain printable ASCII characters. You can customize the subject prefix for specific needs like email filtering. The default subject is "Celerra Notification."

```
[-from <email_addr>]
```

Configures the sender's email address. If the sender's email address is not specfied, a default email address of the format root@<hostname> is configured. The email address follows the mailbox@fully\_qualified\_domain\_name format. For example, 'storage\_admin@yourcompany.com'. For the email address character set and format guidelines, refer the -to option.

#### -init

Initializes the default state; displays a status message if the feature has already been initialized. The **-init** option must be used only when directed.

### **SEE ALSO** *Configuring Celerra Events and Notifications.*

#### **EXAMPLE #1**

To configure email notifications used by the Marketing Department using the email server, **mail.server.yourcompany.com**, with sender as administrator, recipients as storage admin and backup admin, while carbon-copying the network admin and IT support, type:

```
$ nas_emailuser -modify -to
`storage_admin@yourcompany.com','backup_admin@yourcompan
y.com' -cc `network_admin@yourcompany.com',
'it_support@yourcompany.com' -email_server
mail.server.yourcompany.com -subject_prefix "Celerra
Notification - Marketing Department" -from
`administrator@yourcompany.com'
```

OK

## **EXAMPLE #2** To display information on email notifications, type:

```
$ nas_emailuser -info
```

```
Service Enabled = Yes
Recipient Address(es) =
```

storage\_admin@yourcompany.com, backup\_admin@yourcompany.com

Carbon copy Address(es) =

network\_admin@yourcompany.com,it\_support@yourcompany.com
Email Server = mail.server.yourcompany.com

Subject Prefix = Celerra Notification - Marketing Department

Sender Address = administrator@yourcompany.com

# **EXAMPLE #3** To test email notifications, type:

\$ nas\_emailuser -test

OK

# **EXAMPLE #4** To disable email notification, type:

\$ nas\_emailuser -modify -enabled no

OK

# nas\_event

Provides a user interface to system-wide events.

#### **SYNOPSIS**

```
nas_event
   -Load {-info|<file_name>}
| -Unload <file_name>
| -list
   -action {-info|{trap|logfile|mail|callhome|exec|udprpc|tcprpc|terminate}
        [-component {<component>|<id>>}][-facility {<facility>
        | <id>>}][-severity {<severity>|<id>>}]]|[-id]}
|-component {-info|{<component>|<id>>}[-facility {<facility>|<id>>}]|[-id]}
|-severity {-info|<severity>|<id>>}[-component {<component>|<id>>}
| [-facility {<facility>|<id>>}]]|[-id]
|-keyword <keyword> [-component {<component>|<id>>}
| [-facility{<facility>|<id>>}][-severity {<severity>|<id>>}]|[-id]
```

#### **DESCRIPTION**

nas\_event provides a user interface to system-wide events within the Celerra Network Server. The Celerra Network Server includes a default event configuration file that contains a mapping of facilities that generate events, and the associated actions triggered by those events.

To list the default configuration files, type:

```
$ nas_event -Load -info
```

Using a text editor, a new event configuration file can be created and loaded into the database to add an event, or change an action.

Celerra Network Server facilities generate events that trigger specific actions. An event consists of:

- An assigned ID for the event and the facility issuing the event
- The high water mark for the severity of the event
- A description of the event
- The system-defined action to take when the event occurs



#### **CAUTION**

The callhome events in the system are carefully reviewed and configured to provide the right level of support. Do not add, delete, or modify lines that specify the callhome action in the event configuration files in the /nas/sys directory. User-defined event configuration files should not use the callhome action.

#### **OPTIONS**

```
-Load {-info|<file_name>}
```

Loads the event configuration file <file\_name> into the system. The **-info** option displays the currently loaded configuration files.

```
-Unload <file name>
```

Unloads the configuration file <file\_name> from the system.



#### **CAUTION**

The /nas/sys/nas\_eventlog.cfg configuration file must not be deleted, as this can lead to data loss. Unloading or modifying configuration files that are loaded by default can affect logging, alerts, notifications, and system operations.

#### -list

The **-list** option displays components and facilities that generate events, and the actions that are taken as a result. You can search for an event, facility, or action by using a keyword. Component, facility, and severity can be specified by either the text name or ID. The output is displayed with parameter names in the form \$(paraname, typeIndicator, fmtStr).

```
-action {-info|{trap|logfile|mail|callhome|exec|udprpc|tcprpc|
```

#### terminate}

With the **-info** option, lists all the possible actions associated with events. If one of the actions **trap**, **logfile**, **mail**, **callhome**, **exec**, **udprpc**, **tcprpc**, or **terminate** is specified, lists the possible events that trigger the specified action. These events are categorized by component and facility:

```
[-component {<component>|<id>}][-facility
{<facility> |<id>}]
```

Lists the possible events in the specified component that trigger the given action. If facility is specified, lists the events in the specified component and facility that trigger the given action.

```
[-severity {<severity>|<id>}]
```

Lists the possible events with the specified severity that trigger the given action.

```
[-id]
```

Lists the output with the MessageID number in addition to BaseID, Severity, and Brief\_Description.

# -component {-info | {<component> | <id>}[-facility{<facility> | <id>}]

With the **-info** option, lists the ids and names of all the components. If the component is specified, lists the ids and names of all the facilities under that component. Specifying facility lists the events that can be generated by the specified facility and component.

#### [-id]

Lists the output with the message ID number in addition to BaseID and Brief\_Description.

```
-severity {-info|<severity>|<id>}]
```

With the **-info** option, lists the severity levels. If severity is specified, lists the events with the specified severity level.

```
[-component {<component>|<id>} [-facility
<facility>|<id>]
```

Lists the events filtered by the given severity and component. If facility is specified lists the events further filtered by the given facility.

#### [-id]

Lists the output with the MessageID number in addition to BaseID, Severity, and Brief\_Description.

**Note:** To receive email notifications sent to multiple recipients, specify the email addresses within the quotes and separate them with a comma.

#### -keyword <keyword>

Lists all events that match the specified keyword.

```
[-component {<component>|<id>}]
[-facility{<facility> |<id>}]
```

Lists events filtered by the specified keyword and component. If facility is specified, lists the events further filtered by the given facility.

```
[-severity {<severity>|<id>}]
```

Lists events filtered by the specified severity.

#### [-id]

Lists the output with the MessageID number in addition to BaseID, Severity, and Brief\_Description.

**SEE ALSO** *Configuring Celerra Events and Notifications.* 

1:

2:

action mail

trap exec

logfile

BaseID

50

51

52

Ιd

1

2

5

6

```
EXAMPLE #1
                         After using a text editor to create an event configuration file, to load
                         the new configuration file into the NAS database, type:
                         $ nas_event -Load /nas/site/new_eventlog.cfg
EventLog : will load /nas/site/new_eventlog.cfg...done
          EXAMPLE #2
                         To verify that the configuration file was loaded, type:
                         $ nas_event -Load -info
Loaded config. files:
     /nas/sys/nas_eventlog.cfg
     /nas/http/webui/etc/web_client_eventlog.cfg
     /nas/site/new_eventlog.cfg
          EXAMPLE #3
                         To list actions, type:
                         $ nas_event -list -action -info
terminate
callhome
          EXAMPLE #4
                         To list the events that trigger the mail action, type:
                         $ nas_event -list -action mail
CS_PLATFORM(6)
|--> EventLog(130)
        Severity
                            Brief_Description
           EMERGENCY (0)
                            ${text,8,%s}
           ALERT(1)
                            ${text, 8, %s}
           CRITICAL(2)
                            ${text, 8, %s}
          EXAMPLE #5
                         To list the components, type:
                         $ nas_event -list -component -info
       Component
       DART
       CS_CORE
       XML_API
       CS_PLATFORM
```

#### **EXAMPLE #6** To list the facilities under the component **DART**, type:

\$ nas\_event -list -component DART

```
DART(1)
->Id
           Facility
   24
           ADMIN
   26
           CAM
   27
           CFS
   36
           DRIVERS
```

```
40
       FSTOOLS
43
       ΙP
45
       KERNEL
51
       NDMP
52
       NFS
54
       SECURITY
56
       SMB
58
       STORAGE
64
       UFS
68
       LOCK
70
       SVFS
72
       XLT
73
       NETLIB
75
       MGFS
77
       VRPL
81
       VC
83
       RCPD
84
       VMCAST
86
       CHAMII
93
       USRMAP
101
       ACLUPD
102
       FCP
108
       REP
111
       DPSVC
115
       SECMAP
117
       WINS
118
       DNS
122
       DBMS
144
       PERFSTATS
146
       CEPP
```

# **EXAMPLE #7** To list the events generated by **DART** in the facility with the ID **146**, type:

### \$ nas\_event -list -component DART -facility 146

DART(1)	
> CEPP(146)	
BaseID Severity	Brief_Description
1 NOTICE(5)	CEPP server \${ipaddr,8,%s} of pool \${pool,8,%s} is
	\${status,8,%s}. Vendor \${vendor,8,%s}, ntStatus
	$0x$ \${ntstatus,2,%x}.
2 ERROR(3)	Error on CEPP server \${ipaddr,8,%s} of pool
	\${pool,8,%s}: \${status,8,%s}. Vendor \${vendor,8,%s},
	<pre>ntStatus 0x\${ntstatus,2,%x}.</pre>
3 NOTICE(5)	The CEPP facility is started.
4 NOTICE(5)	The CEPP facility is stopped.

#### **EXAMPLE #8**

To list events with severity 4 generated by component **CS\_CORE** and facility **DBMS**, and to display the MessageID in the output, type:

\$ nas\_event -list -severity 4 -component CS\_CORE
-facility DBMS -id

```
CS_CORE(2)
|--> DBMS(122)
MessageID
```

MessageID BaseID 86444212226 2

BaseID Brief\_Description
2 Db: Compact\${compact\_option,8,%s}: \${db\_name,8,%s}:

Failed: \${db\_status,8,%s}.

86444212227 3 Db Env: \${db\_env,8,%s}: Log Remove: Failed:

\${db\_status,8,%s}.

#### **EXAMPLE #9**

To list events filtered by the keyword **freeblocks**, type:

\$ nas\_event -list -keyword freeblocks

#### DART(1)

|--> DBMS(122)

BaseID Severity 2 CRITICAL(2)

Brief\_Description
Only \${freeblocks

Only \${freeblocks,3,%llu} free blocks in the root file system (fsid \${fsid,2,%u}) of the VDM

\${vdm,8,%s}.

3 ALERT(1)

The root file system (fsid \${fsid,2,%u}) of the VDM \${vdm,8,%s} is full. There are only

\${freeblocks,3,%llu} free blocks.

#### **EXAMPLE #10**

To list events with the keyword **data** generated in **DART** with the severity level **6**, type:

\$ nas\_event -list -keyword data -component DART -severity

#### DART(1)

|--> USRMAP(93)

Dabcid	DCVCIICY
1	INFO(6)
4	INFO(6)
8	INFO(6)

Brief\_Description

The Usermapper database has been created. The Usermapper database has been destroyed. The migration of the Usermapper database to the

Celerra Network Server version 5.6 format has started.

ml- - II--

9 INFO(6)

The Usermapper database has been successfully migrated.

DART(1)

|--> SECMAP(115)

BaseID Severity
1 INFO(6)

Brief\_Description

The migration of the secmap database to the Celerra Network Server version 5.6 format has started.

The secmap database has been successfully migrated.

## **EXAMPLE #11** To unload the event configuration file, type:

\$ nas\_event -Unload /nas/site/new\_eventlog.cfg

EventLog: will unload /nas/site/new\_eventlog.cfg... done

INFO(6)

**EXAMPLE #12** To receive email notifications that are sent to multiple recipients, add the following line to your /nas/sys/eventlog.cfg file:

disposition severity=0-3, mail "nasadmin@nasdocs.emc.com, helpdesk@nasdocs.emc.com"

# nas fs

Manages local filesystems for the Celerra Network Server.

#### **SYNOPSIS**

```
nas fs
  -list [-all]
 -delete <fs_name> [-option <options>][-Force]
 -info [-size] {-all|<fs_name>|id=<fs_id>} [-option <options>]
  -rename <old name> <new name> [-Force]
 -size <fs_name>
 -acl <acl_value> <fs_name>
  -translate <fs_name> -access_policy start
    -to {MIXED} -from {NT | NATIVE | UNIX | SECURE}
 -translate <fs_name> -access_policy status
 -xtend <fs_name> {<volume_name>|size=<integer>[T|G|M|%][poo1=<pool>]
    [storage=<system_name>]} [-option <options>]
 -modify <fs_name> -auto_extend {no | yes [-vp {no | yes}]}
    [-hwm <50-99>%][-max_size <integer>[T|G|M]]
  -Type <type> <fs_name> -Force
 [-name <name>][-type <type>] -create <volume_name>
    [samesize=<fs_name>[:cel=<cel_name>]][worm={enterprise|compliance|off}]
    [-option <options>]
| [-name <name>][-type <type>] -create {size=<integer>[T|G|M]
    samesize=<fs_name>[:cel=<cel_name>] }
    pool=<pool> [storage=<system_name>] [worm={enterprise | compliance | off}]
    [-auto_extend {no | yes} [-vp {no | yes}]
    [-hwm <50-99>%][-max_size <integer>[T|G|M]]]
    [-option <options>]
[-name <name>] -type nmfs -create
```

#### **DESCRIPTION**

nas\_fs creates, deletes, extends, and lists filesystems. nas\_fs displays the attributes of a filesystem, translates the access policy, enables, Automatic File System Extension and Virtual Provisioning capabilities, and manages access control level values.

#### OPTIONS

```
-list [-all]
```

Displays a list of filesystems and their attributes such as the name, ID, usage, type, access control level setting, the residing volume, and the server. The **-all** option displays all filesystems including system-generated internal filesystems. For example, Replicator V2 internal checkpoints.

**Note:** The ID is an integer and is assigned automatically, but not always sequentially, depending on ID availability. The name of a filesystem might be truncated if it is more than 19 characters. To display the full filesystem name, use the **-info** option with a filesystem ID.

Filesystem types are:

1=uxfs (default)

5=rawfs (unformatted filesystem)

**6**=mirrorfs (mirrored filesystem)

7=ckpt (checkpoint)

8=mgfs (migration filesystem)

**100**=group filesystem

102=nmfs (nested mount filesystem)

**Note:** Filesystem types **uxfs**, **mgfs**, **nmfs**, and **rawfs** are created by using **nas\_fs**. Other filesystem types are created either automatically or with their specific commands.

#### -delete <fs\_name>

Deletes the filesystem specified by filesystem name or ID. A filesystem cannot be deleted when it is mounted or part of a group.

```
[-option <options>]
```

Specifies the following comma-separated options:

#### volume

Deletes the filesystem's underlying volume structure.

**Note:** If a checkpoint is created with a volume that has been specified by size, the underlying volume is deleted when the checkpoint is deleted. If a filesystem, using a storage pool is deleted, the underlying volume structure is also deleted.

#### [-Force]

Forces the deletion of a filesystem with SnapSure checkpoints known as the PFS, when a task scheduler such as an automated scheduler for SnapSure is running or is enabled.

```
-info [-size] {-all|<fs_name>|id=<fs_id>}
```

Displays the attributes of a single filesystem, or all filesystems, including the configuration of associated disks and replication sessions that are stopped or configured on the filesystem. If a filesystem is mounted, data is reported from the NAS database and

the Data Mover. If a filesystem is unmounted, data is reported from the NAS database only.

The **-size** option also displays the total size of the filesystem and the block count in megabytes.

```
[-option <options>]
```

Specifies the following comma-separated options:

#### mpd

Displays the current directory type and translation status for the specified Multi-Protocol Directory (MPD) filesystem.

```
-rename <old_name> <new_name>
```

Changes the filesystem name from <old\_name> to <new\_name>.

#### [-Force]

Forces the rename of the filesystem with SnapSure checkpoints known as the PFS.

```
-size <fs name>
```

Displays the total size of the filesystem and the block count in megabytes. The total size of a filesystem relates to the mounted or unmounted status of a filesystem.

```
-acl <acl_value> <fs_name>
```

Sets an access control level value that defines the owner of a filesystem, and the level of access allowed for users and groups defined in the access control level table. The nas\_acl command provides more information.

```
-translate <fs_name> -access_policy start -to {MIXED}
```

#### -from {NT | NATIVE | UNIX | SECURE}

Synchronizes the UNIX and Windows permissions on the specified filesystem. Prior to executing the **-translate** option by using **server\_mount**, mount the specified filesystem with the MIXED access-checking policy. The <fs\_name> must be a **uxfs** filesystem type mounted as read/write.

The policy specified in the **-from** option instructs the Celerra Network Server about which operating system (UNIX or Windows) to derive permissions from, when migrating to the **MIXED** or **MIXED\_COMPAT** access-checking policy (set with server\_mount). For example, if you type UNIX in the **-from** option, all ACLs are regenerated from the UNIX mode bits. The policy typed in the **-from** option does not relate to the policy previously used by the filesystem object.

#### -translate <fs\_name> -access\_policy status

Prints the status of the access policy translation for the specified filesystem.

-xtend <fs\_name> <volume\_name>

Adds the specified volume to the mounted filesystem.

### -xtend <fs\_name> size=<integer>[T|G|M|%]

Adds the volume as specified by its desired size to the filesystem or checkpoint. Type an integer within the range of 1 to 1024, then specify **T** for terabytes, **G** for gigabytes (default), **M** for megabytes, or type an integer representing the percentage of a filesystem's size followed by the percent sign. The extended volume added to the filesystem by the system will have a size equal to or greater than the total size specified.



#### **CAUTION**

When executing this command, extends should be performed incrementally by using like volumes to reduce time consumption.

[pool=<pool>]

Applies the specified storage pool rule set to the volume that has been added to the mounted filesystem.

**Note:** The storage pool is a rule set that contains automatically created volumes and defines the type of disk volumes used and how they are aggregated.

[storage=<system\_name>]

Specifies the storage system on which the checkpoint resides. If a storage system is not specified, the default storage system is the one on which the filesystem resides. If the filesystem spans multiple storage systems, the default is to use all the storage systems on which the filesystem resides. Use nas\_storage -list to obtain attached storage system names.

[-option <options>]

Specifies the following comma-separated options:

 $slice=\{y | n\}$ 

Specifies whether the disk volumes used by the filesystem may be shared with other filesystems that use a slice. The **slice=y** option allows the filesystem to share disk volumes with other

filesystems. The **slice=n** option gives the new filesystem exclusive access to the disk volumes it uses, and is relevant when using TimeFinder/FS.

When symm\_std, symm\_std\_rdf\_src, symm\_ata, symm\_ata\_rdf\_src, symm\_ata\_rdf\_tgt, and symm\_std\_rdf\_tgt pools are specified, the default is not to slice the volumes, which is overridden with slice=y. For symm\_efd, the default is slice=y, because TimeFinder/FS is not supported with EFD disk types.

When clar\_r1, clar\_r5\_performance, clar\_r5\_economy, clar\_r6, clarata\_r3, clarata\_r6, clarata\_r10, clarata\_archive, cm\_r1, cm\_r5\_performance, cm\_r5\_economy, cm\_r6, cmata\_r3, cmata\_archive, cmata\_r6, cmata\_r10, clarsas\_archive, clarsas\_r6, clarsas\_r10, clarefd\_r5, clarefd\_r10, cmsas\_archive, cmsas\_r6, cmsas\_r10, and cmefd\_r5 pools are specified, the default is to slice the volumes, which is overridden with slice=n. Use nas\_pool to change the default slice option.

#### src\_only

Extends only the source side for the PFS when replication is running. When extending the PFS while replication is running, by default, both the source and the destination sides are extended.

-modify <fs\_name> -auto\_extend {no|yes [-vp {yes|no}]} [-hwm <50-99>%][-max\_size <integer> [T|G|M]]] For an AVM filesystem, turns Automatic File System Extension and Virtual Provisioning on or off, and sets a high water mark and maximum size for the filesystem. When filesystem extension is turned on, the filesystem is automatically extended up to the maximum size specified when the high water mark is reached. The default for -auto\_extend is no.

Virtual Provisioning reports the maximum filesystem size to the CIFS, NFS, and FTP users, even if the actual size of the filesystem is smaller. If Virtual Provisioning is disabled, the true filesystem size and maximum filesystem sizes are reported to the system administrator. Virtual Provisioning requires that a maximum filesystem size also be set. If a filesystem is created by using a virtual storage pool, the **-vp** option cannot be enabled because Celerra Virtual Provisioning and CLARiiON Virtual Provisioning cannot coexist on a filesystem.

Automatic Filesystem Extension cannot be used for any filesystem that is part of an RDF configuration (for example, filesystems on Data Movers configured with an RDF standby). Do not use the **nas\_fs** 

command with the **-auto\_extend** option for filesystems associated with RDF configurations.

```
[-hwm <50-99>%]
```

Specifies the size threshold that must be reached before the filesystem is automatically extended. Type an integer between the range of 50 and 99 to represent the percentage of filesystem usage. The default is 90.

```
[-max_size <integer> [T|G|M]]
```

Sets the maximum filesystem size to which a filesystem can be extended. Type an integer and specify **T** for terabytes, **G** for gigabytes (default), or **M** for megabytes. If the **-max\_size** option is not specified, then it defaults to the maximum limit of the filesystem size which is 16 terabytes.

```
-Type <type> <fs_name> -Force
```

Changes the filesystem type from the one of <fs\_name> to the new specified <type>.



#### **CAUTION**

The conversion from rawfs to uxfs will fail with "Error 3105: invalid filesystem specified" because a uxfs is not available on the rawfs. However, if the user initially creates a rawfs, and restores an NDMP volume backup on the rawfs, then the conversion from the rawfs to a uxfs will be successful.

# CREATING A FILESYSTEM

Filesystems can be created by using:

- A volume specified by name
- A volume specified by its size and desired storage pool
- An existing local or remote filesystem with the samesize option
- An existing local or remote filesystem with the samesize option and by using space from the available storage pool

[-name <name>] [-type <type>] -create <volume\_name> Creates a filesystem on the specified volume and assigns an optional name to the filesystem. If a name is not specified, one is assigned automatically.

A filesystem name cannot:

- ◆ Begin with a dash (-)
- Be comprised entirely of integers

- Be a single integer
- ◆ Contain the word root or contain a colon (:)

The **-type** option assigns the filesystem type to be **uxfs** (default), **mgfs**, or **rawfs**.

```
[samesize=<fs_name>[:cel=<cel_name>]]
```

Specifies that the new filesystem must be created with the same size as the specified local or remote filesystem. When using the **samesize** option by using the options described below, the **slice**= must be set to **y**.

```
[worm={enterprise | compliance | off}]
```

Enables storage capability on a new filesystem. The option can only be specified when creating a new filesystem; existing filesystems cannot be converted. After a filesystem is enabled, it is persistently marked as such until the time when it is deleted.

Note: The compliance filesystem cannot be deleted if it has protected files.



#### CAUTION

The enterprise version of this feature is intended for self-regulated archiving. The administrator is considered a trusted user and the capability does not protect the archived data from the administrator's actions. If the administrator attempts to delete the filesystem, the filesystem issues a warning message and prompts the administrator to confirm the operation. This version is not intended for high-end compliance applications such as pharmaceuticals, aerospace, or finance.

```
[-option <options>]
```

Specifies the following comma-separated options:

```
nbpi=<number>
```

The number of bytes per inode block. The default is 8192 bytes.

```
mover=<movername>
```

Assigns an optional Data Mover to build a filesystem. If no Data Mover is assigned, the system will automatically pick the first available Data Mover to build the filesystem.

```
slice=\{y | n\}
```

Specifies whether the disk volumes used by the new filesystem may be shared with other filesystems by using a slice. The **slice=y** 

option allows the filesystem to share disk volumes with other filesystems. The **slice=n** option ensures that the new filesystem has exclusive access to the disk volumes it uses, and is relevant when using TimeFinder/FS.

When symm\_std, symm\_std\_rdf\_src, symm\_ata, symm\_ata\_rdf\_src, symm\_ata\_rdf\_tgt, and symm\_std\_rdf\_tgt pools are specified, the default is not to slice the volumes. When slice=y is specified, it overrides the default. For symm\_efd, the default is slice=y, because TimeFinder/FS is not supported with EFD disk types.

When clar\_r1, clar\_r5\_performance, clar\_r5\_economy, clar\_r6, clarata\_r3, clarata\_r6, clarata\_r10, clarata\_archive, cm\_r1, cm\_r5\_performance, cm\_r5\_economy, cm\_r6, cmata\_r3, cmata\_archive, cmata\_r6, cmata\_r10, clarsas\_archive, clarsas\_r6, clarsas\_r10, clarefd\_r5, clarefd\_r10, cmsas\_archive, cmsas\_r6, cmsas\_r10, and cmefd\_r5 pools are specified, the default is to slice the volumes, which is overridden with slice=n. Use nas\_pool to change the default slice option.

id=<desired\_id>

Specifies the ID to be assigned to the new filesystem. If a filesystem already exists with the specified ID, a warning is displayed indicating that the ID is not available, and the new filesystem is assigned the next available ID.

Creates a filesystem on the volume specified by its desired size and storage pool or by using the same size as a specified local or remote filesystem. Also assigns an optional name and filesystem type to a filesystem. If a name is not specified, one is assigned automatically. A filesystem name can be up to 240 characters, but cannot begin with a dash (-), be comprised entirely of integers or be a single integer, contain the word root or contain a colon (:). Available filesystem types are uxfs (default), mgfs, or rawfs.

When using the **samesize** option by using the options described below, the **slice**= should be set to **y**. The new filesystem is created with the same size as the specified filesystem.

The **pool** option specifies a rule set for the new filesystem that contains automatically created volumes and defines the type of disk volumes used and how they are aggregated. Storage pools are system

defined (storage pool description provides more information) or user defined.

#### [worm={enterprise | compliance | off}]

Enables the storage capability on the new filesystem. The capability can only be specified when creating a new filesystem; existing filesystems cannot be converted. After a filesystem is enabled, it is persistently marked as such until the time when it is deleted.



#### **CAUTION**

The enterprise version of this feature is intended for self-regulated archiving. The administrator is considered a trusted user and feature does not protect the archived data from the administrator's actions. If the administrator attempts to delete a filesystem, the filesystem issues a warning message and prompts the administrator to confirm the operation. This version of this feature is not intended for high-end compliance applications such as pharmaceuticals, aerospace, or finance.

[storage=<system\_name>]

Specifies the storage system on which the filesystem resides. Use **nas\_storage -list** to obtain a list of the available system names.

[-auto\_extend {no|yes} [-vp {no|yes}]

For an AVM filesystem, turns Automatic File System Extension and Virtual Provisioning on or off, and sets a high water mark and maximum size for the filesystem. When Automatic File System Extension is turned on, the filesystem is automatically extended up to the maximum size specified when the high water mark is reached. The default for **-auto\_extend** is **no**.

Virtual Provisioning reports the maximum filesystem size to the CIFS, NFS, and FTP users, even if the actual size of the filesystem is smaller. If disabled, the true filesystem size and maximum filesystem sizes are reported to the system administrator. Virtual Provisioning requires that a maximum filesystem size also be set. If a filesystem is created using a virtual storage pool, the **-vp** option cannot be enabled because Celerra Virtual Provisioning and CLARiiON Virtual Provisioning cannot coexist on a filesystem.

Note: SRDF pools are not supported.

#### [-hwm < 50-99>%]

Specifies the size threshold that must be reached before the filesystem is automatically extended. Type an integer between the range of 50 and 99 to represent the percentage of filesystem usage. The default is 90.

#### [-max\_size <integer> [T|G|M]]

Sets the maximum filesystem size to which a filesystem can be extended. Type an integer and specify **T** for terabytes, **G** for gigabytes (default), or **M** for megabytes. If the **-max\_size** option is not specified, then it defaults to the maximum limit of the filesystem size which is 16 terabytes. Maximum size must be set to enable Virtual Provisioning. The maximum size is what is presented to users as the filesystem size through Virtual Provisioning.

#### [-option <options>]

Specifies the following comma-separated options:

#### nbpi=<number>

The number of bytes per inode block. The default is 8192 bytes.

#### mover=<movername>

Assigns an optional Data Mover on which to build a filesystem. If no Data Mover is assigned, the system will automatically pick the first available Data Mover to build the filesystem.

#### $slice=\{y | n\}$

Specifies whether the disk volumes used by the new filesystem may be shared with other filesystems by using a slice. The **slice=y** option allows the filesystem to share disk volumes with other filesystems. The **slice=n** option ensures that the new filesystem has exclusive access to the disk volumes it uses, and is relevant when using TimeFinder/FS.

When symm\_std, symm\_std\_rdf\_src, symm\_ata, symm\_ata\_rdf\_src, symm\_ata\_rdf\_tgt, and symm\_std\_rdf\_tgt pools are specified, the default is not to slice the volumes, which is overridden with slice=y. For symm\_efd, the default is slice=y, because TimeFinder/FS is not supported with EFD disk types.

When clar\_r1, clar\_r5\_performance, clar\_r5\_economy, clar\_r6, clarata\_r3, clarata\_r6, clarata\_r10, clarata\_archive, cm\_r1, cm\_r5\_performance, cm\_r5\_economy, cm\_r6, cmata\_r3, cmata\_archive, cmata\_r6, cmata\_r10, clarsas\_archive, clarsas\_r6, clarsas\_r10, clarefd\_r5, clarefd\_r10, cmsas\_archive, cmsas\_r6,

cmsas\_r10, and cmefd\_r5 pools are specified, the default is to slice the volumes, which is overridden with slice=n. Use nas\_pool to change the default slice option.

#### [-name <name>] -type nmfs -create

Creates a nested mount filesystem (NMFS) that can be used to combine multiple **uxfs** filesystems into a single virtual filesystem. The NMFS can then be mounted and exported as a single share or mount point.

#### **SEE ALSO**

Managing Celerra Volumes and File Systems with Automatic Volume Management, Managing Celerra Volumes and File Systems Manually, Controlling Access to Celerra System Objects, Using File-Level Retention on Celerra, Using TimeFinder/FS, NearCopy, and FarCopy with Celerra, Using Celerra Replicator (V1), fs\_ckpt, fs\_timefinder, nas\_acl, nas\_rdf, nas\_volume, server\_export, server\_mount, fs\_dedupe, and server\_mountpoint.

# STORAGE SYSTEM OUTPUT

The number associated with the storage device is dependent on the attached storage system. CLARiiON storage systems display a prefix of APM before a set of integers, for example, APM00033900124-0019. Symmetrix storage systems appear as 002804000190-003C. The outputs displayed in the examples use a CLARiiON storage system.

Celerra Network Servers with a CLARiiON storage system support the following system-defined storage pools: clar\_r1, clar\_r5\_performance, clar\_r5\_economy, clar\_r6, clarata\_r3, clarata\_r6, clarata\_r10, clarata\_archive, cm\_r1, cm\_r5\_performance, cm\_r5\_economy, cm\_r6, cmata\_r3, cmata\_archive, cmata\_r6, cmata\_r10, clarsas\_archive, clarsas\_r6, clarsas\_r10, clarefd\_r5, clarefd\_r10, cmsas\_archive, cmsas\_r6, cmsas\_r10, and cmefd\_r5.

Celerra Network Servers with a Symmetrix storage system support the following system-defined storage pools: symm\_std\_rdf\_src, symm\_std, symm\_ata, symm\_ata\_rdf\_src, symm\_ata\_rdf\_tgt, symm\_std\_rdf\_tgt, and symm\_efd.

For user-defined storage pools, the difference in output is in the disk type. Disk types when using a Symmetrix are: STD, R1STD, R2STD, BCV, R1BCV, R2BCV, ATA, R1ATA, R2ATA, BCVA, R1BCA, R2BCA, and EFD.

Disk types when using CLARiiON are: CLSTD, CLEFD, and CLATA, and for CLARiiON storage systems involving mirrored disks are: CMEFD, CMSTD, and CMATA.

## **EXAMPLE #1** To create a filesystem named **ufs1** on metavolume **mtv1**, type:

#### \$ nas\_fs -name ufs1 -create mtv1

```
= 37
id
name
         = ufs1
acl
         = 0
in_use = False
type
        = uxfs
worm = compliance
worm_clock = Thu Mar  6 16:26:09 EST 2008
worm Max Retention Date = Fri April 18 12:30:40 EST 2008
        = mtv1
volume
pool
         =
rw_servers=
ro_servers=
rw_vdms =
ro_vdms
auto_ext = no,virtual_provision=no
deduplication = off
stor_devs = APM00042000818-0012, APM00042000818-0014
disks
         = d7, d9
```

#### Where:

Value	Definition				
id	Automatically assigned ID of a filesystem.				
name	Name assigned to a filesystem.				
acl	Access control value assigned to the filesystem.				
in_use	If a filesystem is registered into the mount table of a Data Mover.				
type	Type of filesystemlist provides information for a description of the types.				
worm	Write Once Read Many (WORM) state of filesystem. It states whether the state is disabled or set to either compliance or enterprise.				
volume	Volume on which a filesystem resides.				
pool	Storage pool for the filesystem.				
rw_servers	Servers with read/write access to a filesystem.				
ro_servers	Servers with read-only access to a filesystem.				
rw_vdms	VDM servers with read/write access to a filesystem.				
ro_vdms	VDM servers with read-only access to a filesystem.				
worm_clock	Software clock maintained by the filesystem. The clock functions only when the filesystem is mounted read/write.				
worm Max Retention Date	Time when the protected files expire. The filesystem can be deleted only after this date. The special values returned are:				
	3 — The filesystem is FLR-E with protected files.				
	2 — The filesystem is scanning for max_retention period.				
	1 — The default value (no protected files created).				
	• 0 — Infinite retention period (if the server is up and running).				
auto_ext	Indicates whether auto-extension and Virtual Provisioning are enabled.				

deduplication	Deduplication state of the filesystem. The file data is transferred to the storage which performs the deduplication and compression on the data. The states are:  On — Deduplication on the filesystem is enabled.  Suspended — Deduplication on the filesystem is suspended. Deduplication does not perform any new space reduction but the existing files that were reduced in space remain the same.  Off — Deduplication on the filesystem is disabled. Deduplication does not perform any new space reduction and the data is now reduplicated.
stor_devs	Storage system devices associated with a filesystem.
disks	Disks on which the metavolume resides.

**Note:** The deduplication state is **unavailable** when the filesystem is unmounted.

### **EXAMPLE #2**

To display information about a filesystem by using the filesystem ID, 37, type:

\$ nas\_fs -info id=37

```
id = 37
       = ufs1
name
acl
in_use = False
type = uxfs
worm
        = off
volume = mtv1
pool
rw_servers=
ro_servers=
rw_vdms =
ro_vdms
auto_ext = no, virtual_provision=no
deduplication = off
stor_devs = APM00042000818-0012, APM00042000818-0014
disks = d7, d9
```

**EXAMPLE** #1 provides a description of command outputs.

# **EXAMPLE #3** To display a list of filesystems, type:

				1 ,	•	- I
				<pre>\$ nas_fs</pre>	-list	
id	inuse	type	acl	volume	name	server
1	n	1	0	24	root_fs_1	
2	У	1	0	26	root_fs_2	1
3	У	1	0	28	root_fs_3	2
4	n	1	0	30	root_fs_4	
5	n	1	0	32	root_fs_5	
6	n	1	0	34	root_fs_6	
7	n	1	0	36	root_fs_7	
8	n	1	0	38	root_fs_8	
9	n	1	0	40	root_fs_9	
10	n	1	0	42	root_fs_10	
11	n	1	0	44	root_fs_11	

12	n	1	0	46	root_fs_12	
13	n	1	0	48	root_fs_13	
14	n	1	0	50	root_fs_14	
15	n	1	0	52	root fs 15	
16	У	1	0	54	root_fs_common	2,1
17	y n	5	0	87	root_fs_ufslog	∠,⊥
18	n	5	0	90	root_panic_reserve	
212		1	0	315	v2src1	1
214	y n	1	0	318	v2dst1	_
230		1	0	346	v2srclun1	1
231	У	1	0	349	v2dstlun1	2
342	У	1	0	560	root_fs_vdm_srcvdm1	1
343	У	1	0	563		1
987	У	7	0	1722	root_fs_vdm_srcvdm2	1
988	У	1	0	1725	gstest src1	1
	У		0			1
989	У	5	0	1728	dst1	1
1346	У	1	-	2354	root_fs_vdm_srcvdm1	_
1350	У	1	0	2367	fs1	v9
1354	n	1	0	2374	fs1_replica1	
1361	n	1	0	2385	fs1_replica2	
1365	У	1	0	2392	fs1365	1
1371	У	1	0	2399	root_fs_vdm_v1	1
1372	У	1	0	2401	f1	v40
1376	У	1	0	2406	root_fs_vdm_v1_repl	2
1383	У	1	0	2416	f1_replica1	v41
1387	У	1	0	2423	cworm	1
1388	n	1	0	2425	cworm1	
1389	У	1	0	2427	fs2	2
1390	У	1	0	2429	fs3	2
1397	У	7	0	2432	fs2_ckpt1	2
1398	У	1	0	2439	fs4	2
1399	У	1	0	2441	fs5	2
1406	У	7	0	2444	fs4_ckpt1	2

# **EXAMPLE #4** To list all the filesystems including internal checkpoints, type:

				<pre>\$ nas_fs</pre>	-list -all	
id	inuse	type	acl	volume	name	server
1	n	1	0	24	root_fs_1	
2	У	1	0	26	root_fs_2	1
3	У	1	0	28	root_fs_3	2
4	n	1	0	30	root_fs_4	
5	n	1	0	32	root_fs_5	
6	n	1	0	34	root_fs_6	
7	n	1	0	36	root_fs_7	
8	n	1	0	38	root_fs_8	
9	n	1	0	40	root_fs_9	
10	n	1	0	42	root_fs_10	
11	n	1	0	44	root_fs_11	
12	n	1	0	46	root_fs_12	
13	n	1	0	48	root_fs_13	
14	n	1	0	50	root_fs_14	

15 16 17	n Y n	1 1 5	0 0 0	52 54 87	<pre>root_fs_15 root_fs_common root_fs_ufslog</pre>	2,1
18 212	n Y	5 1	0	90 315	root_panic_reserve v2src1	1
213 214	У n	101 1	0	0 318	<pre>root_avm_fs_group_3 v2dst1</pre>	
230	У	1	0	346	v2srclun1	1
231	У	1	0	349	v2dstlun1	2
342	У	1	0	560	<pre>root_fs_vdm_srcvdm1</pre>	1
343	У	1	0	563	<pre>root_fs_vdm_srcvdm2</pre>	1
986	n	11	0	0	vpfs986	
987	У	7	0	1722	gstest	1
988	У	1	0	1725	src1	1
989	У	5	0	1728	dst1	1
1343	n	11	0	0	vpfs1343	
1344	У	7	0	2351	root_rep_ckpt_342_2	1
1345	У	7	0	2351	root_rep_ckpt_342_2	1
1346	У	1	0	2354	<pre>root_fs_vdm_srcvdm1</pre>	1
1347	n	11	0	0	vpfs1347	
1348	У	7	0	2358	root_rep_ckpt_1346_	1
1349	У	7	0	2358	root_rep_ckpt_1346_	1
1350	У	1	0	2367	fs1	v9
1354	n	1	0	2374	fs1_replica1	
1358	n	11	0	0	vpfs1358	
1359	У	7	0	2383	root_rep_ckpt_1350_	
1360	У	7	0	2383	<pre>root_rep_ckpt_1350_</pre>	v9
1361	n	1	0	2385	fs1_replica2	
1362	n	11	0	0	vpfs1362	
1363	n	7	0	2388	root_rep_ckpt_1361_	
1364	n	7	0	2388	root_rep_ckpt_1361_	1
1365	У	1	0	2392	fs1365	1 v9
1366 1367	У	7 7	0	2383 2383	root_rep_ckpt_1350_	v9 v9
1368	У n	11	0	0	root_rep_ckpt_1350_ vpfs1368	V 9
1369	n	7	0	2395	root_rep_ckpt_1354_	
1370	n	7	0	2395	root_rep_ckpt_1354_	
1371	У	1	0	2399	root_fs_vdm_v1	1
1372	У	1	0	2401	f1	v40
1376	У	1	0	2406	root_fs_vdm_v1_repl	2
1380	n	11	0	0	vpfs1380	_
1381	У	7	0	2414	root_rep_ckpt_1372_	v40
1382	У	7	0	2414	root_rep_ckpt_1372_	v40
1383	У	1	0	2416	f1_replica1	v41
1384	n	11	0	0	vpfs1384	
1385	У	7	0	2419	root_rep_ckpt_1383_	v41
1386	У	7	0	2419	root_rep_ckpt_1383_	
1387	У	1	0	2423	cworm	1
1388	n	1	0	2425	cworm1	
1389	У	1	0	2427	fs2	2
1390	У	1	0	2429	fs3	2
1391	n	11	0	0	vpfs1391	

1392	У	7	0	2432	root_rep_ckpt_1389_	2
1393	У	7	0	2432	root_rep_ckpt_1389_	2
1394	n	11	0	0	vpfs1394	
1395	У	7	0	2435	root_rep_ckpt_1390_	2
1396	У	7	0	2435	root_rep_ckpt_1390_	2
1397	У	7	0	2432	fs2_ckpt1	2
1398	У	1	0	2439	fs4	2
1399	У	1	0	2441	fs5	2
1400	n	11	0	0	vpfs1400	
1401	У	7	0	2444	root_rep_ckpt_1398_	2
1402	У	7	0	2444	root_rep_ckpt_1398_	2
1403	n	11	0	0	vpfs1403	
1404	У	7	0	2447	root_rep_ckpt_1399_	2
1405	У	7	0	2447	root_rep_ckpt_1399_	2
1406	У	7	0	2444	fs4_ckpt1	2

**Note:** NDMP and Replicator internal checkpoints can be identified by specific prefixes in the filename. *Using SnapSure on Celerra* provides more information for internal checkpoints naming formats.

#### **EXAMPLE #5**

To create a uxfs filesystem named **ufs1** on storage system **APM00042000818**, with a size of **100** GB, by using the **clar\_r5\_performance** pool and allowing the filesystem to share disk volumes with other filesystems, type:

\$ nas\_fs -name ufs1 -type uxfs -create size=100G
pool=clar\_r5\_performance storage=APM00042000818 -option
slice=y

```
id
        = 38
        = ufs1
name
acl
        = 0
in_use = False
       = uxfs
type
worm
        = off
volume = v171
pool = clar_r5_performance
member_of = root_avm_fs_group_3
rw_servers=
ro_servers=
rw_vdms =
ro_vdms =
auto_ext = no,virtual_provision=no
deduplication = off
stor_devs = APM00042000818-001F,APM00042000818-001D,APM00042000818-0019,APM00042
000818-0016
disks = d20, d18, d14, d11
```

#### Where:

Value	Definition
member_of	Filesystem group to which the filesystem belongs.

**EXAMPLE** #1 provides a description of command output. The **clar\_r5\_performance** storage pool is available with the CLARiiON storage system.

#### **EXAMPLE #6**

To create a **rawfs** filesystem named **ufs3** with the same size as the filesystem **ufs1** by using the **clar\_r5\_performance** pool and allowing the filesystem to share disk volumes with other filesystems, type:

\$ nas\_fs -name ufs3 -type rawfs -create samesize=ufs1
pool=clar\_r5\_performance storage=APM00042000818 -option
slice=y

ro\_servers=
rw\_vdms =
ro\_vdms =
auto\_ext = no,virtual\_provision=no
deduplication = off

stor\_devs = APM00042000818-001F, APM00042000818-001D, APM00042000818-0019, APM00042

000818-0016

disks = d20, d18, d14, d11

**EXAMPLE #1** and **EXAMPLE #3** provide for a description of command outputs. The **clar\_r5\_performance** storage pool is available with the CLARiiON storage system.

#### **EXAMPLE #7**

To create a **uxfs** filesystem named **ufs4**, with a size of **100** GB, by using the **clar\_r5\_performance** pool, with enabled, **4096** bytes per inode, and **server\_3** for filesystem building, type:

\$ nas\_fs -name ufs4 -create size=100G
pool=clar\_r5\_performance worm=enterprise -option
nbpi=4096,mover=server\_3

```
type = uxfs
       = enterprise
worm
volume = v175
pool = clar_r5_performance
member_of = root_avm_fs_group_3
rw servers=
ro_servers=
rw_vdms
ro vdms
auto_ext = no,virtual_provision=no
deduplication = off
stor devs = APM00042000818-001F, APM00042000818-001D, APM00042000818-0019, APM00042
000818-0016
disks
          = d20, d18, d14, d11
                       To ensure retention of protected files, it can also be set to compliance
                       by typing:
                        $ nas_fs -name ufs4 -create size=100G
                       pool=clar_r5_performance worm=compliance -option
                       nbpi=4096, mover=server_3
id
          = 40
        = ufs4
name
         = 0
acl
in_use = False
        = uxfs
type
worm
        = compliance
volume = v175
       = clar_r5_performance
loog
member_of = root_avm_fs_group_3
rw_servers=
ro_servers=
rw vdms
ro_vdms
auto_ext = no, virtual_provision=no
deduplication = off
stor_devs = APM00042000818-001F, APM00042000818-001D, APM00042000818-0019, APM00042
000818-0016
disks
          = d20, d18, d14, d11
```

**EXAMPLE #1** provides a description of command outputs. The **clar\_r5\_performance** storage pool is available with the CLARiiON storage system.

#### **EXAMPLE #8**

To create a **uxfs** filesystem named **ufs1**, with a size of **10** GB, by using the **clar\_r5\_performance** pool, and an ID of 8000 assigned to **ufs1**, type:

\$ nas\_fs -name ufs1 -type uxfs -create size=10G
pool=clar\_r5\_performance -option slice=y,id=8000

id = 8000

```
name = ufs1
acl = 0
in use = False
type = uxfs
worm = off
volume = v7193
pool = clar_r5_performance
member_of = root_avm_fs_group_3
rw servers=
ro_servers=
rw_vdms =
ro vdms =
auto_ext = no, virtual_provision=no
deduplication = off
stor_{devs} = APM00043401231-0035
disks = d3
```

#### **EXAMPLE #9**

To create a **uxfs** filesystem named **ufs2**, with a size of **10** GB, by using the **clar\_r5\_performance** pool, and an ID of 8000 assigned to **ufs1**, type:

\$ nas\_fs -name ufs2 -type uxfs -create size=10G
pool=clar r5 performance -option slice=y,id=8000

```
id = 12464
name = ufs2
acl = 0
in_use = False
type = uxfs
worm = off
volume = v7195
pool = clar_r5_performance
member_of = root_avm_fs_group_3
rw_servers=
ro servers=
rw_vdms =
ro_vdms =
auto_ext = no, virtual_provision=no
deduplication = off
stor_{devs} = APM00043401231-0034
disks = d5
Warning 17716815881: unavailable id : 8000
```

Note: This output is displayed if the desired ID is not available.

# **EXAMPLE #10** To view the size of **ufs1**, type:

```
$ nas_fs -size ufs1
total = 393 avail = 393 used = 1 ( 0% ) (sizes in MB) ( blockcount = 819200 )
volume: total = 400 (sizes in MB) ( blockcount = 819200 ) avail = 393 used = 7 ( 2% )
```

When a filesystem is mounted, the size info for the volume and a filesystem, as well as the number of blocks that are used are displayed.

#### Where:

Value	Definition
total	Total size of the filesystem.
blockcount	Total number of blocks used.

## **EXAMPLE #11** To rename a filesystem from **ufs1** to **ufs2**, type:

```
$ nas_fs -rename ufs1 ufs2
```

```
id
         = 38
name
         = ufs2
         = 0
acl
in_use
         = True
         = uxfs
type
         = off
worm
volume
         = v171
loog
        = clar_r5_performance
member_of = root_avm_fs_group_3
rw_servers= server_2
ro_servers=
rw_vdms
ro vdms
auto_ext = no, virtual_provision=no
deduplication = off
stor_devs = APM00042000818-001F, APM00042000818-001D, APM00042000818-0019, APM00042
000818-0016
disks
       = d20, d18, d14, d11
 disk=d20
           stor_dev=APM00042000818-001F addr=c0t1115
                                                            server=server_2
 disk=d20 stor_dev=APM00042000818-001F addr=c32t1115
                                                            server=server_2
 disk=d18 stor_dev=APM00042000818-001D addr=c0t1113
                                                            server=server 2
 disk=d18 stor_dev=APM00042000818-001D addr=c32t1113
                                                            server=server_2
 disk=d14 stor_dev=APM00042000818-0019 addr=c0t119
                                                            server=server_2
 disk=d14 stor dev=APM00042000818-0019 addr=c32t119
                                                            server=server 2
 disk=d11
           stor_dev=APM00042000818-0016 addr=c0t116
                                                            server=server_2
 disk=d11
            stor_dev=APM00042000818-0016 addr=c32t116
                                                            server=server_2
```

**EXAMPLE #1** and **EXAMPLE #3** provide a description of command outputs. The **clar\_r5\_performance** storage pool is available with the CLARiiON storage system.

# **EXAMPLE #12** To list all of the filesystems in the NAS database, type:

				<pre>\$ nas_fs</pre>	-list	
id	inuse	type	acl	volume	name	server
1	n	1	0	10	root_fs_1	
2	У	1	0	12	root_fs_2	1
3	У	1	0	14	root_fs_3	2

```
4
                 1
                     0
                            16
                                       root_fs_4
                                                              3
           У
5
                     0
                                                              4
                 1
                            18
                                       root_fs_5
           У
6
           n
                 1
                     0
                            20
                                       root_fs_6
7
           n
                 1
                     0
                            22
                                       root_fs_7
8
                 1
                    0
                            24
           n
                                       root_fs_8
9
                 1
                     0
                            26
                                       root fs 9
           n
10
                 1
                     0
                            28
                                       root_fs_10
           n
11
           n
                 1
                     0
                            30
                                       root_fs_11
12
                 1
                     0
                            32
                                       root fs 12
           n
13
                 1
                     0
                            34
                                       root_fs_13
           n
                 1
                     0
14
           n
                            36
                                       root_fs_14
15
                 1
                     0
           n
                            38
                                       root fs 15
16
                 1
                     0
                            40
                                       root_fs_common
                                                              2,4,3,1
           У
                 5
17
           n
                     0
                            73
                                       root_fs_ufslog
                 5
                     0
                            76
18
                                       root_fs_d3
19
                 5
                    0
                            77
                                       root_fs_d4
           n
20
                 5
                     0
                            78
                                       root_fs_d5
           n
                 5
21
                    0
                            79
                                       root_fs_d6
38
                 1
                     0
                            171
                                       ufs2
                                                              1
           У
40
           n
                 1
                     0
                            175
                                       ufs4
```

EXAMPLE #1 provides a description of command outputs.

## **EXAMPLE #13** To extend the filesystem, **ufs1**, with the volume, **emtv2b**, type:

\$ nas\_fs -xtend ufs1 emtv2b

```
= 38
id
name
          = ufs1
          = 0
acl
in_use
          = True
          = uxfs
type
          = off
worm
          = v171
volume
          = clar_r5_performance
member_of = root_avm_fs_group_3
rw servers= server 2
ro_servers=
rw_vdms
ro vdms
auto_ext = no, virtual_provision=no
deduplication = off
stor_devs = APM00042000818-001F, APM00042000818-001D, APM00042000818-0019, APM00042
000818-0016,APM00042000818-001C
disks
          = d20, d18, d14, d11, d17
 disk=d20
            stor_dev=APM00042000818-001F addr=c0t1115
                                                              server=server 2
 disk=d20
            stor_dev=APM00042000818-001F addr=c32t1115
                                                              server=server_2
 disk=d18
            stor_dev=APM00042000818-001D addr=c0t1113
                                                              server=server_2
 disk=d18
            stor_dev=APM00042000818-001D addr=c32t1113
                                                              server=server_2
 disk=d14
            stor_dev=APM00042000818-0019 addr=c0t119
                                                              server=server_2
            stor_dev=APM00042000818-0019 addr=c32t119
 disk=d14
                                                              server=server_2
            stor_dev=APM00042000818-0016 addr=c0t116
                                                              server=server_2
 disk=d11
 disk=d11
            stor_dev=APM00042000818-0016 addr=c32t116
                                                              server=server_2
```

```
disk=d17 stor_dev=APM00042000818-001C addr=c0t1112 server=server_2 disk=d17 stor_dev=APM00042000818-001C addr=c32t1112 server=server_2
```

**EXAMPLE** #1 provides a description of command outputs.

# **EXAMPLE #14** To extend the filesystem named **ufs2**, with the specified size of 100 GB, by using **clar\_r5\_performance** pool, type:

\$ nas\_fs -xtend ufs2 size=100G pool=clar\_r5\_performance

```
id
         = 38
         = ufs2
name
         = 0
acl
in_use
         = True
type
         = uxfs
         = off
WOrm
volume
         = v171
         = clar_r5_performance
member_of = root_avm_fs_group_3
rw_servers= server_2
ro_servers=
rw_vdms
ro vdms
auto_ext = no, virtual_provision=no
deduplication = off
stor_devs = APM00042000818-001F,APM00042000818-001D,APM00042000818-0019,APM00042
000818-0016,APM00042000818-001C
       = d20, d18, d14, d11, d17
disk=d20
            stor dev=APM00042000818-001F addr=c0t1115
                                                             server=server 2
disk=d20
           stor_dev=APM00042000818-001F addr=c32t1115
                                                             server=server_2
disk=d18
           stor_dev=APM00042000818-001D addr=c0t1113
                                                             server=server_2
disk=d18
           stor_dev=APM00042000818-001D addr=c32t1113
                                                             server=server_2
 disk=d14
           stor_dev=APM00042000818-0019 addr=c0t119
                                                             server=server_2
 disk=d14
            stor_dev=APM00042000818-0019 addr=c32t119
                                                             server=server_2
disk=d11
            stor_dev=APM00042000818-0016 addr=c0t116
                                                             server=server_2
 disk=d11
            stor_dev=APM00042000818-0016 addr=c32t116
                                                             server=server_2
 disk=d17
            stor_dev=APM00042000818-001C addr=c0t1112
                                                             server=server_2
 disk=d17
            stor_dev=APM00042000818-001C addr=c32t1112
                                                             server=server_2
```

**EXAMPLE #1** provides a description of command outputs. The **clar\_r5\_performance** storage pool is available with the CLARiiON storage system.

## **EXAMPLE #15** To set the access control level for the filesystem **ufs1**, type:

```
$ nas_fs -acl 1432 ufs1
```

acl = 1432, owner=nasadmin, ID=201
in\_use = True
type = uxfs
worm = off

= 38 = ufs1

id

name

```
volume = v171
pool
        = clar_r5_performance
member_of = root_avm_fs_group_3
rw_servers= server_2
ro_servers=
rw vdms
ro vdms
auto_ext = no, virtual_provision=no
deduplication = off
stor_devs = APM00042000818-001F, APM00042000818-001D, APM00042000818-0019, APM00042
000818-0016,APM00042000818-001C
disks
         = d20, d18, d14, d11, d17
disk=d20 stor_dev=APM00042000818-001F addr=c0t1115
                                                            server=server_2
disk=d20 stor_dev=APM00042000818-001F addr=c32t1115
                                                            server=server 2
disk=d18 stor_dev=APM00042000818-001D addr=c0t1113
                                                            server=server_2
disk=d18 stor_dev=APM00042000818-001D addr=c32t1113
                                                            server=server 2
disk=d14 stor_dev=APM00042000818-0019 addr=c0t119
                                                            server=server_2
disk=d14 stor dev=APM00042000818-0019 addr=c32t119
                                                            server=server 2
disk=d11 stor_dev=APM00042000818-0016 addr=c0t116
                                                            server=server_2
disk=d11 stor_dev=APM00042000818-0016 addr=c32t116
                                                            server=server_2
disk=d17 stor_dev=APM00042000818-001C addr=c0t1112
                                                           server=server_2
disk=d17 stor_dev=APM00042000818-001C addr=c32t1112
                                                            server=server_2
```

**Note:** The value 1432 specifies **nasadmin** as the owner and gives users with an access level of at least observer read access only, users with an access level of at least operator read/write access, and users with an access level of at least admin read/write/delete access.

EXAMPLE #1 provides a description of command outputs.

# **EXAMPLE #16** To convert a uxfs filesystem type named **ufs1** to a **rawfs**, type:

```
$ nas_fs -Type rawfs ufs1 -Force
```

```
name
         = ufs1
acl
        = 1432, owner=nasadmin, ID=201
in_use
        = True
         = rawfs
type
        = off
worm
volume = v171
        = clar_r5_performance
member_of = root_avm_fs_group_3
rw_servers= server_2
ro_servers=
rw_vdms =
ro vdms
auto_ext = no, virtual_provision=no
deduplication = off
stor_devs = APM00042000818-001F, APM00042000818-001D, APM00042000818-0019, APM00042
000818-0016,APM00042000818-001C
disks
         = d20, d18, d14, d11, d17
```

```
disk=d20
           stor_dev=APM00042000818-001F addr=c0t1115
                                                             server=server_2
disk=d20
           stor_dev=APM00042000818-001F addr=c32t1115
                                                             server=server_2
                                                             server=server_2
disk=d18
           stor_dev=APM00042000818-001D addr=c0t1113
disk=d18
           stor_dev=APM00042000818-001D addr=c32t1113
                                                             server=server_2
disk=d14
           stor_dev=APM00042000818-0019 addr=c0t119
                                                             server=server_2
disk=d14
           stor dev=APM00042000818-0019 addr=c32t119
                                                             server=server 2
disk=d11
           stor_dev=APM00042000818-0016 addr=c0t116
                                                             server=server 2
disk=d11
           stor_dev=APM00042000818-0016 addr=c32t116
                                                             server=server_2
disk=d17
           stor_dev=APM00042000818-001C addr=c0t1112
                                                             server=server 2
disk=d17
           stor_dev=APM00042000818-001C addr=c32t1112
                                                             server=server_2
```

EXAMPLE #1 provides a description of command outputs.

#### **EXAMPLE #17**

To start the conversion of the filesystem, **ufs2**, and to conform to the **MIXED** access policy mode, type:

\$ nas\_fs -translate ufs2 -access\_policy start -to MIXED
-from NT

```
= 38
id
          = ufs2
name
          = 1432, owner=nasadmin, ID=201
acl
in use
          = True
          = uxfs
type
worm
         = off
          = v171
volume
          = clar_r5_performance
member_of = root_avm_fs_group_3
rw servers= server 2
ro_servers=
rw_vdms
ro_vdms
auto_ext = no, virtual_provision=no
deduplication = off
stor_devs = APM00042000818-001F, APM00042000818-001D, APM00042000818-0019, APM00042
000818-0016,APM00042000818-001C
disks
          = d20, d18, d14, d11, d17
disk=d20
            stor_dev=APM00042000818-001F addr=c0t1115
                                                              server=server_2
 disk=d20
            stor_dev=APM00042000818-001F addr=c32t1115
                                                              server=server_2
disk=d18
            stor_dev=APM00042000818-001D addr=c0t1113
                                                              server=server_2
disk=d18
            stor_dev=APM00042000818-001D addr=c32t1113
                                                              server=server_2
 disk=d14
            stor_dev=APM00042000818-0019 addr=c0t119
                                                              server=server 2
 disk=d14
            stor_dev=APM00042000818-0019 addr=c32t119
                                                              server=server_2
disk=d11
            stor_dev=APM00042000818-0016 addr=c0t116
                                                              server=server_2
 disk=d11
            stor_dev=APM00042000818-0016 addr=c32t116
                                                              server=server_2
 disk=d17
            stor_dev=APM00042000818-001C addr=c0t1112
                                                              server=server_2
 disk=d17
            stor_dev=APM00042000818-001C addr=c32t1112
                                                              server=server 2
```

**EXAMPLE** #1 provides a description of command outputs. The **clar\_r5\_performance** storage pool is available with the CLARiiON storage system.

## **EXAMPLE #18** To display the status of access policy conversion for **ufs2**, type:

## **EXAMPLE #19** To create a nested mount filesystem, **nmfs1**, type:

```
$ nas_fs -name nmfs1 -type nmfs -create
id
        = nmfs1
name
acl
        = 0
in_use = False
      = nmfs
type
       = off
worm
volume = 0
loog
rw_servers=
ro_servers=
rw_vdms
ro_vdms
auto_ext = no,virtual_provision=no
deduplication = off
stor_devs =
disks
```

EXAMPLE #1 provides a description of command outputs.

## **EXAMPLE #20** To delete **ufs1**, type:

#### \$ nas fs -delete ufs1

```
id
        = 37
name
        = ufs1
acl
        = 0
in_use = False
        = uxfs
type
worm
        = off
volume = mtv1
rw_servers=
ro_servers=
rw_vdms =
ro_vdms
auto_ext = no,virtual_provision=no
deduplication = off
stor_devs = APM00042000818-0012, APM00042000818-0014
disks = d7, d9
```

EXAMPLE #1 provides a description of command outputs.

# **EXAMPLE #21**

To create a filesystem named **ufs3**, with a size of 20 GB, by using the **clar\_r5\_performance** pool, a maximum size of 100 GB and with auto-extend and Virtual Provisioning enabled, type:

\$ nas\_fs -name ufs3 -create size=20G
pool=clar\_r5\_performance -auto\_extend yes -max\_size 100G
-vp yes

```
id
          = 42
name
          = ufs3
acl
          = 0
        = False
in_use
type
        = uxfs
         = off
worm
volume
        = v180
        = clar_r5_performance
pool
member_of = root_avm_fs_group_3
rw_servers=
ro_servers=
rw_vdms
ro_vdms
auto_ext = hwm=90%, max_size=102400M, virtual_provision=yes
deduplication = off
stor_devs = APM00042000818-001F, APM00042000818-001D, APM00042000818-0019, APM00042
000818-0016
disks
          = d20, d18, d14, d11
```

EXAMPLE #1 provides a description of command outputs.

# **EXAMPLE #22** To disable the Virtual Provisioning on **ufs3**, type:

```
$ nas_fs -modify ufs3 -vp no
```

```
iд
          = 42
name
          = ufs3
          = 0
acl
         = False
in_use
type
         = uxfs
        = off
worm
         = v180
volume
        = clar_r5_performance
pool
member_of = root_avm_fs_group_3
rw_servers=
ro_servers=
rw_vdms
ro_vdms
auto_ext = hwm=90%, max_size=102400M, virtual_provision=no
deduplication = off
stor_devs = APM00042000818-001F, APM00042000818-001D, APM00042000818-0019, APM00042
000818-0016
disks
          = d20, d18, d14, d11
```

**EXAMPLE #1** provides a description of command outputs. The **clar\_r5\_performance** storage pool is available with the CLARiiON storage system.

# **EXAMPLE #23**

To query the current directory type and translation status for MPD, type:

\$ nas\_fs -info ufs1 -option mpd id = 35 = ufs1name acl = 0in use = True = uxfs type = off worm = mtv1volume pool rw\_servers= server\_3 ro\_servers= rw\_vdms ro\_vdms auto\_ext = no, virtual\_provision=no deduplication = off stor\_devs = disks = d59, d60, d61, d62stor\_dev=000190100559-0094 addr=c0t418-04-0 disk=d59 server=server\_3 disk=d59 stor dev=000190100559-0094 addr=c16t418-03-0 server=server 3 disk=d59 stor dev=000190100559-0094 addr=c32t418-08-0 server=server 3 disk=d59 stor\_dev=000190100559-0094 addr=c48t418-07-0 server=server\_3 disk=d60 stor dev=000190100559-0095 addr=c0t419-04-0 server=server 3 disk=d60 stor\_dev=000190100559-0095 addr=c16t419-03-0 server=server\_3 addr=c32t419-08-0 disk=d60 stor\_dev=000190100559-0095 server=server\_3 disk=d60 stor dev=000190100559-0095 addr=c48t419-07-0 server=server 3 disk=d61 stor\_dev=000190100559-0096 addr=c0t4l10-04-0 server=server\_3 disk=d61 stor\_dev=000190100559-0096 addr=c16t4l10-03-0 server=server 3 disk=d61 stor dev=000190100559-0096 addr=c32t4l10-08-0 server=server\_3 disk=d61 stor\_dev=000190100559-0096 addr=c48t4l10-07-0 server=server\_3 disk=d62 stor\_dev=000190100559-0097 addr=c0t4l11-04-0 server=server\_3 disk=d62 stor dev=000190100559-0097 addr=c16t4l11-03-0 server=server 3 disk=d62 stor\_dev=000190100559-0097 addr=c32t4l11-08-0 server=server\_3 disk=d62 stor\_dev=000190100559-0097 addr=c48t4l11-07-0 server=server\_3 Multi-Protocol Directory Information Default\_directory\_type = DIR3 Needs\_translation = False Translation\_state = Completed

= False

EXAMPLE #1 provides a description of filesystem command outputs.

Has\_translation\_error

# Where:

Value	Definition
Default_directory_type	The default directory type for the filesystem. Available types are: DIR3 and COMPAT.
Needs_translation	If true, then the filesystem may contain more than one directory type. If false, then all directories are of the filesystem default directory type.
Translation_state	The current state of the translation thread. Available states are: never, not requested, pending, queued, running, paused, completed, and failed.
Has_translation_error	Indicated if the most recent translation encountered any errors.

Default_directory_type	Needs_translation state	File system
DIR3	False	Is MPD. No action required.
DIR3	True	Requires translation or filesystem maintenance. Contact EMC Customer Service.
COMPAT	False	Is COMPAT and requires translation. Contact EMC Customer Service.
COMPAT	True	Requires translation. Contact EMC Customer Service.

The state where both Default\_directory\_type=DIR3 and Needs\_transalation=False assure that this filesystem's directories are all in MPD format, and there are no directories of the obsolete single-protocol format.

Any other combination of states, for example, Needs\_transalation=True, indicates that there could be non-MPD directories in the filesystem which may not be compatible with a future release.

# nas\_fsck

Manages fsck and aclchk utilities on specified filesystems.

# **SYNOPSIS**

```
nas_fsck
  -list
| -info {-all|<fs_name>|id=<fs_id>}
| -start {<fs_name>|id=<fs_id>} [-aclchkonly][-monitor][-mover <movername>]
        [-Force]
```

# **DESCRIPTION**

nas\_fsck uses the fsck and aclchk utilities to perform a check for consistency and errors on the specified filesystem. nas\_fsck also lists and displays the status of the fsck and aclchk utilities. Filesystems must be mounted read-write to use these utilities.

Depending on the size of the filesystem, the **FSCK** utility may use a significant portion of the system's memory and may affect overall system performance. Hence, it should not be run on a server under heavy load as it is possible that the server may run out of resources. In most cases, the user will be notified if sufficient memory is not available to run a filesystem check. In these cases, one of the following can be done:

- Start the filesystem during off-peak hours.
- Reboot the server and let nas fsck run on reboot.
- Run nas\_fsck on a different server if the filesystem is unmounted.

# **OPTIONS**

#### -list

Displays a list of all the filesystems undergoing **fsck** or **aclchk**.

```
-info {-all|<fs_name>|id=<fs_id>}
```

Queries the Data Mover and displays information about the status of the **fsck** or **aclchk** utilities for the specified filesystem.

```
-start {<fs_name>|id=<fs_id>}
```

Starts the **fsck** and the **aclchk** utilities on the specified filesystem.



# **CAUTION**

If filesystem check is started on a mounted filesystem, the filesystem will be unavailable for the duration of the check. NFS clients will display the message NFS server not responding and CIFS clients will lose connectivity with the server and will have to remap shares.

# [-aclchkonly]

Initiates the **aclchk** utility only, which checks and corrects any errors in the ACL database and removes duplicate ACL information stored on the specified filesystem. The **aclchkonly** option can only be used on a filesystem that is not exported. The default is for both **fsck** and **aclchk**.

**Note:** The NDMP backup process must be stopped on the Data Mover before using the **nas\_fsck -aclchkonly** command.

## [-monitor]

Displays the status of **fsck** and **aclchk** until the command completes.

**Note:** For a mounted filesystem, a <movername> is not required since the **fsck** and **aclchk** utilities are run on the Data Mover where the filesystem is mounted.

# [-Force]

Forces a **fsck** or **aclchk** to be run on a enabled filesystem.

**SEE ALSO** *Managing Celerra Volumes and File Systems Manually* and nas\_fs.

# **EXAMPLE #1**

To start filesystem check on **ufs1** and monitor the progress, type:

```
$ nas_fsck -start ufs1 -monitor
id
                           = 27
name
                           = ufs1
volume
                           = mtv1
fsck_server
                           = server 2
inode_check_percent
                           = 10..20..30..40..60..70..80..100
                         = 0..0..100
directory_check_percent
used_ACL_check_percent
                          = 100
free_ACL_check_status
                          = Done
cylinder_group_check_status = In Progress..Done
```

## Where:

Value	Definition	
id	Automatically assigned ID of a filesystem.	
name	Name assigned to the filesystem.	
volume	Volume on which the filesystem resided.	
fsck_server	Name of the Data Mover where the utility is being run.	
inode_check_percent	Percentage of inodes in the filesystem checked and fixed.	
directory_check_percent	Percentage of directories in the filesystem checked and fixed.	

used_ACL_check_percent	Percentage of used ACLs that have been checked and fixed.
free_ACL_check_status	Status of the ACL check.
cylinder_group_check_status	Status of the cylinder group check.

# **EXAMPLE #2** To start ACL check on **ufs1**, type:

```
$ nas_fsck -start ufs1 -aclchkonly
```

ACLCHK: in progress for file system ufs1

# **EXAMPLE #3** To start a filesystem check on **ufs2** using Data Mover **server\_5**, type:

```
$ nas_fsck -start ufs2 -mover server_5
= ufs2
```

 name
 = ufs2

 id
 = 23

 volume
 = v134

 fsck\_server
 = server\_5

 inode\_check\_percent
 = 40

 directory\_check\_percent
 = 0

# EXAMPLE #1 provides a description of command outputs.

# **EXAMPLE #4** To list all current filesystem checks, type:

# \$ nas fsck -list

id	type	state	volume	name	server
23	1	FSCK	134	ufs2	4
27	1	ACLCHK	144	ufs1	1

# Where:

Value	Definition
id	Automatically assigned ID of a filesystem.
type	Type of filesystem.
state	Utility being run.
volume	Volume on which the filesystem resided.
name	Name assigned to the filesystem.
server	Server on which <b>fsck</b> is being run.

# **EXAMPLE #5** To display information about filesystem check for **ufs2** that is currently running, type:

\$ nas\_fsck -info ufs2

```
directory_check_percent = 100
used_ACL_check_percent = 100
free_ACL_check_status = Done
cylinder_group_check_status = In Progress
```

EXAMPLE #1 provides a description of command outputs.

# **EXAMPLE #6** To display information about all filesystem checks that are currently running, type:

```
$ nas_fsck -info -all
                            = ufs2
name
                            = 23
id
volume
                            = v134
fsck_server
                            = server_5
inode_check_percent
                            = 30
directory_check_percent
                            = 0
used_ACL_check_percent
                            = 0
free_ACL_check_status
                            = Not Started
cylinder_group_check_status = Not Started
                            = ufs1
name
id
                            = 27
volume
                            = mtv1
fsck server
                            = server 2
inode_check_percent
                            = 100
directory_check_percent
used_ACL_check_percent
                            = 0
free_ACL_check_status
                            = Not Started
cylinder_group_check_status = Not Started
```

**EXAMPLE #1** provides a description of command outputs.

# nas halt

Performs a controlled halt of all Control Stations and Data Movers in the Celerra Network Server.

# **SYNOPSIS**

nas halt now

**DESCRIPTION** nas\_halt performs an orderly shutdown of the Control Stations and

Data Movers in the Celerra Network Server. nas\_halt must be

executed from the /nas/sbin directory.

OPTIONS now

Performs an immediate halt for the Celerra Network Server.

**SEE ALSO** *Celerra Network Server System Operations* and server\_cpu.

**EXAMPLE #1** To perform an immediate halt of the Celerra Network Server, type:

# /nas/sbin/nas\_halt\_now

usage: nas\_halt now

Perform a controlled halt of the Control Stations and Data Movers # /nas/sbin/nas\_halt now

You are about to HALT this Celerra including all of its Control Stations and Data Movers. DATA will be UNAVAILABLE when the system is halted. Note that this command does \*not\* halt the storage array.

```
ARE YOU SURE YOU WANT TO CONTINUE? [yes or no] : yes
Sending the halt signal to the Master Control Daemon...: Done
May 3 11:12:54 cs100 EMCServer: nas_mcd: Check and halt other CS...: Done
May 3 11:13:26 cs100 JSERVER: *** Java Server is exiting ***
May 3 11:13:31 cs100 ucd-snmp[11218]: Received TERM or STOP signal... shutting
May 3 11:13:31 cs100 snmpd: snmpd shutdown succeeded
May 3 11:13:32 cs100 setup_enclosure: Executing -dhcpd stop option
May 3 11:13:32 cs100 snmptrapd[11179]: Stopping snmptrapd
May 3 11:13:32 cs100 EV_AGENT[13721]: Signal TERM received
May 3 11:13:32 cs100 EV_AGENT[13721]: Agent is going down
May 3 11:13:40 cs100 DHCPDMON: Starting DHCPD on CS 0
May 3 11:13:41 cs100 setup_enclosure: Executing -dhcpd start option
May 3 11:13:41 cs100 dhcpd: Internet Software Consortium DHCP Server V3.0pl1
May 3 11:13:41 cs100 dhcpd: Copyright 1995-2001 Internet Software Consortium.
May 3 11:13:41 cs100 dhcpd: All rights reserved.
May 3 11:13:41 cs100 dhcpd: For info, please visit
```

http://www.isc.org/products/DHCP

```
May 3 11:13:41 cs100 dhcpd: Wrote 0 deleted host decls to leases file.
May 3 11:13:41 cs100 dhcpd: Wrote 0 new dynamic host decls to leases file.
May 3 11:13:41 cs100 dhcpd: Wrote 0 leases to leases file.
May 3 11:13:41 cs100 dhcpd: Listening on
LPF/eth2/00:00:f0:9d:04:13/128.221.253.0/24
May 3 11:13:41 cs100 dhcpd: Sending on
LPF/eth2/00:00:f0:9d:04:13/128.221.253.0/24
May 3 11:13:41 cs100 dhcpd: Listening on
LPF/eth0/00:00:f0:9d:01:e5/128.221.252.0/24
May 3 11:13:41 cs100 dhcpd: Sending on
LPF/eth0/00:00:f0:9d:01:e5/128.221.252.0/24
May 3 11:13:41 cs100 dhcpd: Sending on Socket/fallback/fallback-net
May 3 11:13:59 cs100 mcd_helper: : Failed to umount /nas (0)
May 3 11:13:59 cs100 EMCServer: nas_mcd: Failed to gracefully shutdown MCD and
halt servers. Forcing halt and reboot...
May 3 11:13:59 cs100 EMCServer: nas_mcd: Halting all servers...
May 3 11:15:00 cs100 get_datamover_status: Data Mover server_5: COMMAND doesnt
match.
```

# nas\_inventory

Provides detailed information about hardware components in the system.

# **SYNOPSIS**

```
nas_inventory
{
    -list [-location]
| {-info <location>|-all}
| -tree
}
```

# **DESCRIPTION**

**nas\_inventory** displays detailed information about the hardware components that are configured on a system.

# **OPTIONS**

## -list

Displays a list of all hardware components and their associated name, type, status, and system ID.

# [-location]

Displays the location string for each component in the output. The location string is a unique identifier for the component.

Specifies the location string with enclosed double quotes (" ") and displays a list of detailed information for the specific component for which the string is the unique ID.

```
-info <location_string>|-all
```

Displays a list of all the properties for a component, including the component name, type, status, variant, associated storage system, serial number, part number, and history.

The **-all** option lists detailed information for all components in the system.

#### -tree

Displays a hierarchical tree of components, including the status of each component.

# **EXAMPLE #1** To display a list of components on the system, type:

	<pre>\$ nas_invento</pre>	ry -list	
Component	Type	Status	System ID
Battery A	Battery	OK	CLARIION CX4-240FCNTR083000055
Celerra NS40G			
FCNTR083000055001A	Celerra	Warning	Celerra NS40GFCNTR083000055001A
CLARiiON CX4-240			
FCNTR083000055	CLARiiON	OK	CLARiiON CX4-240 FCNTR083000055
DME 0 Data Mover 2	Data Mover	OK	Celerra NS40G FCNTR083000055001A
DME 0 Data Mover 2			
Ethernet Module	Module	OK	Celerra NS40G FCNTR083000055001A
DME 0 Data Mover 2			
SFP BE0	SFP	OK	Celerra NS40G FCNTR083000055001A
DME 0 Data Mover 2			
SFP BE1	SFP	OK	Celerra NS40G FCNTR083000055001A
DME 0 Data Mover 2			
SFP FE0	SFP	OK	Celerra NS40G FCNTR083000055001A

# Where:

Value	Definition
Component	Description of the component.
Туре	The type of component. Possible types are: battery, blower, Celerra, CLARiiON, Control Station, Data Mover, and disk.
Status	The current status of the component. Status is component type specific. There are several possible status values, each of which is associated with a particular component type.
System ID	The identifier for the Celerra Network Server or the storage ID of the storage system containing the component.

# **EXAMPLE #2** To display a list of components and component locations, type:

# \$ nas\_inventory -list -location

Component Status System ID Type Location CLARiiON CX4-240 FCNTR083000055 Battery A Battery OK system:NS40G:FCNTR083000055001A|clariionSystem:CX4-240:FCNTR083000055|sps::A Warning Celerra NS40G Celerra NS40G FCNTR083000055001A Celerra system: NS40G: FCNTR083000055001A FCNTR083000055001A CLARIION CX4-240 FCNTR083000055 CLARIION OK CLARiiON CX4-240 FCNTR083000055 system:NS40G:FCNTR083000055001A|clariionSystem:CX4-240:FCNTR083000055 DME 0 Data Mover 2 Data Mover OK Celerra NS40G FCNTR083000055001A system: NS40G: FCNTR083000055001A enclosure: xpe:0 mover: NS40:2 DME 0 Data Mover 2 Ethernet Module Module OK Celerra NS40G FCNTR083000055001A system:NS40G:FCNTR083000055001A|enclosure:xpe:0|mover:NS40:2|module:ethernet: DME 0 Data Mover 2 SFP BE0 SFP OK Celerra NS40G FCNTR083000055001A system:NS40G:FCNTR083000055001A|enclosure:xpe:0|mover:NS40:2|sfp::BE0 DME 0 Data Mover 2 SFP BE1 SFP Celerra NS40G FCNTR083000055001A OK system:NS40G:FCNTR083000055001A|enclosure:xpe:0|mover:NS40:2|sfp::BE1 DME 0 Data Mover 2 SFP FE0 SFP OK Celerra NS40G FCNTR083000055001A system:NS40G:FCNTR083000055001A|enclosure:xpe:0|mover:NS40:2|sfp::FE0

# **EXAMPLE #3** To list information for a specific component, type:

\$ nas\_inventory -info "system:NS40G:FCNTR083000055001A|
clariionSystem:CX4-240:FCNTR083000055|iomodule::B0"

Location = system:NS40G:FCNTR083000055001A|clariionSystem:CX4-240:

FCNTR083000055|iomodule::B0
Component Name = IO Module B0
Type = IO Module

Status = OK

Variant = 4 PORT FIBRE IO MODULE

Storage System = CLARiiON CX4-240 FCNTR083000055

Serial Number = CF2YW082800426 Part Number = 103-054-100C

History

= EMC\_PART\_NUMBER:103-054-100C EMC\_ARTWORK\_REVISION:C01 EMC\_ASSEMBLY\_REVISION:C03

EMC\_SERIAL\_NUMBER: CF2YW082800426

VENDER\_PART\_NUMBER:N/A
VENDER\_ARTWORK\_NUMBER:N/A
VENDER\_ASSEMBLY\_NUMBER:N/A
VENDER\_SERIAL\_NUMBER:N/A

VENDOR\_NAME:N/A

LOCATION\_OF\_MANUFACTURE:N/A
YEAR\_OF\_MANUFACTURE:N/A
MONTH\_OF\_MANUFACTURE:N/A

DAY\_OF\_MONTH\_OF\_MANUFACTURE:N/A ASSEMBLY\_NAME:4 PORT FIBRE IO MODULE

Note: The location string must be enclosed in double quotes.

# Where:

Value	Definition
Location	The unique identifier of the component and where the component is located in the component hierarchy.
Component	The description of the component.
Type	The type of component. Possible types are: battery, blower, Celerra, CLARiiON, Control Station, Data Mover, and disk.
Status	The current condition of the component. Status is component type specific. There are several possible status values, each of which is associated with a particular component type.
Variant	The specific type of hardware.
Storage System	The model and serial number of the storage system.
Serial Number	The serial number of the hardware component.
Part Number	The part number of the hardware component.
History	If available, the history information of the component. Possible values are: part number, serial number, vendor, date of manufacture, and CPU information.

# **EXAMPLE #4** To display components in a tree structure, type:

# \$ nas\_inventory -tree

· · · · · · · · · · · · · · · · · · ·	-	
Component	Туре	Status
Celerra NS40G FCNTR083000055001A	Celerra	Warning
CLARIION CX4-240 FCNTR083000055	CLARiiON	OK
Battery A	Battery	OK
IO Module AO	IO Module	OK
IO Module A1	IO Module	OK
IO Module A2	IO Module	Empty
IO Module A3	IO Module	Empty
IO Module A4	IO Module	Empty
IO Module B0	IO Module	OK
IO Module B1	IO Module	OK
IO Module B2	IO Module	Empty
IO Module B3	IO Module	Empty
IO Module B4	IO Module	Empty
Power Supply A0	Power Supply	OK
Power Supply A1	Power Supply	OK
Power Supply B0	Power Supply	OK
Power Supply B1	Power Supply	OK

# nas\_license

Enables software packages.

# **SYNOPSIS**

# nas\_license -list | -create <package\_name>[|<key\_code>] | -delete <package\_name> | -init

# **DESCRIPTION**

**nas\_license** enables software packages that are available for use with your Celerra Network Server. The <key\_code> is supplied by EMC.

All entries are case-sensitive.

# **OPTIONS** No arguments

Displays a usage message containing all available and valid software packages that can be installed.

#### -list

Displays the site\_key as a string and any software packages for which a license has been installed. The site\_key is a permanent license and cannot be deleted. The failover license is legacy and should only be used only when instructed to do so by EMC personnel.

**Note:** Licenses installed on the Control Station are read by the system. The site\_key is a unique identifier which gets generated the first time nas\_license is run. The site\_key is also used to decode the key\_code supplied by EMC personnel for special packages.

```
-create <package_name>[=<key_code>]
Installs the license for the indicated <package_names>. Valid
<package_names> are:
```

```
site key
nfs
cifs
failover
snapsure
advancedmanager
replicator
iscsi
replicatorV2
filelevelretention
```

**Note:** These packages do not require key\_code as they can be enabled from the GUI. Special packages are supplied along with the required key\_code by the EMC Customer Service Representative. The package failover requires key\_code.

## -delete <package\_name>

Deletes the license for the specified <package\_name>.

#### -init

Initializes the database and re-creates the license file by using the site\_key that is already installed. The license file is located at /nas/site as nas\_license. It contains license keys in an encrypted format. The **-init** option should be run only if the license file containing all the license information has been lost and the following error message is received:

license table is not initialized

Once the license file has been re-created, the rest of the entries, if present, should be re-added by using the **-create** option.

# **EXAMPLE #1**

To install a license for the **iscsi** software package, type:

\$ nas\_license -create iscsi

done

## **EXAMPLE #2**

To display all software packages with currently installed licenses, type:

	<pre>\$ nas_license</pre>	-list
key	status	value
site_key	online	42 de 6f d1
advancedmanager	online	
nfs	online	
cifs	online	
iscsi	online	
snapsure	online	
replicator	online	
replicatorV2	online	
filelevelretention	online	

# **EXAMPLE #3**

To delete a license for specified software package, type:

\$ nas\_license -delete iscsi

done

# **EXAMPLE #4**

To initialize the database and re-create the license file, type:

\$ nas\_license -init

done

# nas\_logviewer

Displays the content of nas\_eventlog generated log files.

# **SYNOPSIS**

nas\_logviewer <file\_name>
 [-f][-v|-t]

# **DESCRIPTION**

nas\_logviewer displays the event log and other logs created by nas\_eventlog. The log files may be system generated, or created by the user. Information in the log file is read from oldest to newest.

# **OPTIONS**

# No arguments

Displays the contents of the specified log file.

-f

Monitors the growth of the log by entering into an endless loop, pausing and reading the log as it is being generated. To exit, press **Ctrl-C** together.

[-v|-t]

Displays the log files in verbose or terse format.

# SEE ALSO

Configuring Celerra Events and Notifications and server\_log.

## **EXAMPLE #1**

To view the contents of the sys log file, type:

# \$ nas\_logviewer /nas/log/sys\_log more

```
May 12 18:01:57 2007:CS_PLATFORM:NASDB:INFO:300:::::nasdb_backup: NAS_DB
checkpoint in progress
May 12 18:02:59 2007:CS_PLATFORM:NASDB:INFO:305:::::nasdb_backup: NAS_DB
Checkpoint done
May 12 18:03:00 2007:CS_PLATFORM:NASDB:ERROR:202:::::NAS database error
detectedMay 12 18:03:12 2007
:CS_PLATFORM:NASDB:INFO:306:::::nasdb_backup: NAS DB Backup done
May 12 19:01:52 2007:CS_PLATFORM:NASDB:INFO:300:::::nasdb_backup: NAS_DB
checkpoint in progress
May 12 19:02:50 2007:CS_PLATFORM:NASDB:INFO:305:::::nasdb_backup: NAS_DB
Checkpoint done
May 12 19:02:51 2007:CS_PLATFORM:NASDB:ERROR:202:::::NAS database error
detectedMay 12 19:03:02 2007
:CS_PLATFORM:NASDB:INFO:306:::::nasdb_backup: NAS DB Backup done
May 12 20:01:57 2007:CS_PLATFORM:NASDB:INFO:300:::::nasdb_backup: NAS_DB
checkpoint in progress
May 12 20:02:58 2007:CS_PLATFORM:NASDB:INFO:305:::::nasdb_backup: NAS_DB
Checkpoint done
May 12 20:02:59 2007:CS_PLATFORM:NASDB:ERROR:202:::::NAS database error
detectedMay 12 20:03:10 2007
```

```
:CS_PLATFORM:NASDB:INFO:306::::nasdb_backup: NAS DB Backup done
May 12 21:01:52 2007:CS_PLATFORM:NASDB:INFO:300::::nasdb_backup: NAS_DB
checkpoint in progress
May 12 21:02:51 2007:CS_PLATFORM:NASDB:INFO:305::::nasdb_backup: NAS_DB
Checkpoint done
May 12 21:02:52 2007:CS_PLATFORM:NASDB:ERROR:202::::NAS database error
detectedMay 12 21:03:03 2007
:CS_PLATFORM:NASDB:INFO:306:::::nasdb_backup: NAS_DB_Backup done
```

**Note**: This is a partial listing due to the length of the outputs.

# **EXAMPLE #2** To display the contents of the log files in terse format, type:

## \$ nas\_logviewer -t /nas/log/sys\_log

```
May 12 18:01:57 2007:96108871980:nasdb_backup: NAS_DB checkpoint in progress May 12 18:02:59 2007:96108871985:nasdb_backup: NAS_DB Checkpoint done May 12 18:03:00 2007:83223969994:NAS database error detected May 12 18:03:12 2007:96108871986:nasdb_backup: NAS_DB Backup done May 12 19:01:52 2007:96108871980:nasdb_backup: NAS_DB checkpoint in progress May 12 19:02:50 2007:96108871985:nasdb_backup: NAS_DB Checkpoint done May 12 19:02:51 2007:83223969994:NAS database error detected May 12 19:03:02 2007:96108871986:nasdb_backup: NAS_DB Backup done May 12 19:03:02 2007:96108871986:nasdb_backup: NAS_DB checkpoint in progress May 12 20:01:57 2007:96108871980:nasdb_backup: NAS_DB checkpoint in progress May 12 20:02:58 2007:96108871985:nasdb_backup: NAS_DB Checkpoint done May 12 20:03:10 2007:96108871986:nasdb_backup: NAS_DB Backup done May 12 20:03:10 2007:96108871986:nasdb_backup: NAS_DB checkpoint in progress May 12 21:01:52 2007:96108871980:nasdb_backup: NAS_DB checkpoint in progress May 12 21:02:51 2007:96108871980:nasdb_backup: NAS_DB checkpoint in progress May 12 21:02:51 2007:96108871980:nasdb_backup: NAS_DB checkpoint done
```

# **EXAMPLE #3** To display the contents of the log files in verbose format, type:

# \$ nas\_logviewer -v /nas/log/sys\_log|more

```
logged time = May 12 18:01:57 2007
creation time = May 12 18:01:57 2007
slot id =
id = 96108871980
severity = INFO
component = CS_PLATFORM
facility = NASDB
baseid = 300
type = EVENT
brief discription = nasdb_backup: NAS_DB checkpoint in progress
full discription = The Celerra configuration database is being checkpointed.
recommended action = No action required.
logged time = May 12 18:02:59 2007
creation time = May 12 18:02:59 2007
slot id =
id = 96108871985
severity = INFO
component = CS_PLATFORM
facility = NASDB
```

baseid = 305
type = EVENT
brief description = nasdb\_backup: NAS\_DB Checkpoint done
full description = The NAS DB backup has completed a checkpoint of the current
reparation for performing a backup of NAS system data.
recommended action = No action required.

# **EXAMPLE #4** To monitor the growth of the current log, type:

# \$ nas\_logviewer -f /nas/log/sys\_log more May 12 18:01:57 2007:CS\_PLATFORM:NASDB:INFO:300:::::nasdb\_backup: NAS\_DB checkpoint in progress May 12 18:02:59 2007:CS\_PLATFORM:NASDB:INFO:305:::::nasdb\_backup: NAS\_DB Checkpoint done May 12 18:03:00 2007:CS\_PLATFORM:NASDB:ERROR:202:::::NAS database error detectedMay 12 18:03:12 2007 :CS PLATFORM: NASDB: INFO: 306:::::nasdb backup: NAS DB Backup done May 12 19:01:52 2007:CS\_PLATFORM:NASDB:INFO:300:::::nasdb\_backup: NAS\_DB checkpoint in progress May 12 19:02:50 2007:CS\_PLATFORM:NASDB:INFO:305:::::nasdb\_backup: NAS\_DB Checkpoint done May 12 19:02:51 2007:CS\_PLATFORM:NASDB:ERROR:202:::::NAS database error detectedMay 12 19:03:02 2007 :CS\_PLATFORM:NASDB:INFO:306:::::nasdb\_backup: NAS\_DB\_Backup\_done May 12 20:01:57 2007:CS\_PLATFORM:NASDB:INFO:300:::::nasdb\_backup: NAS\_DB checkpoint in progress May 12 20:02:58 2007:CS\_PLATFORM:NASDB:INFO:305:::::nasdb\_backup: NAS\_DB Checkpoint done May 12 20:02:59 2007:CS\_PLATFORM:NASDB:ERROR:202:::::NAS database error detectedMay 12 20:03:10 2007 :CS\_PLATFORM:NASDB:INFO:306:::::nasdb\_backup: NAS\_DB\_Backup\_done May 12 21:01:52 2007:CS\_PLATFORM:NASDB:INFO:300:::::nasdb\_backup: NAS\_DB checkpoint in progress May 12 21:02:51 2007:CS\_PLATFORM:NASDB:INFO:305:::::nasdb\_backup: NAS\_DB Checkpoint done May 12 21:02:52 2007:CS\_PLATFORM:NASDB:ERROR:202:::::NAS database error detectedMay 12 21:03:03 2007 :CS\_PLATFORM:NASDB:INFO:306:::::nasdb\_backup: NAS DB Backup done

# nas\_message

Displays message description.

# **SYNOPSIS**

## nas message

-info <MessageId>

# **DESCRIPTION**

**nas\_message** provides detailed descriptions to a dedicated message. A brief description, full description, and recommended user action of the message are displayed.

# **OPTIONS**

-info <MessageId>

Displays detailed descriptions of the error message, including severity, component, facility, BaseID, and recommended user action. The message parameters are displayed in the form \${stateDesc,8,%s} and not as parameter values. The <MessageId> must be a positive integer.

## SEE ALSO

Celerra Network Server Error Messages Guide.

# **EXAMPLE #1**

To display detailed descriptions for error message 13421838337, type:

# \$ nas\_message -info 13421838337

MessageID = 13421838337

BaseID = 1
Severity = ERROR
Component = CS\_CORE
Facility = default
Type = STATUS

Brief Description = Operation not permitted\${arg0,8,%s}

Full\_Description = The operation is not permitted due to an ACL or ownership issue on the specified object.

Recommended\_Action = Check ownership or ACL of the object in question.If appropriate change the setting to resolve the conflict. Refer to the nas\_acl and chmod man page.

# nas\_mview

Performs MirrorView/Synchronous (MirrorView/S) operations on a Celerra Network Server attached to a CLARiiON CX-Series storage system.

## **SYNOPSIS**

#### nas mview

- -info
- -init <cel\_name>
- -activate
- -restore

## DESCRIPTION

nas\_mview retrieves MirrorView/S cabinet-level information, initializes the source and destination Celerra Network Servers for MirrorView/S, activates a failover to a destination Celerra, or restores the source site after a failover.

MirrorView/S is supported on a Celerra Network Server attached to a CLARiiON CX-Series storage array serving as the boot storage, not as the secondary storage. **nas\_mview** must be run from a Control Station in slot 0; it will report an error if run from a Control Station in slot 1.

nas\_mview must be issued as root from the /nas/sbin directory. For the -init and -info options, log in with your administrative username and use the su root command to log in as root. For the -activate and -restore options, you must log in to the destination Celerra Network Server using the remote administration account (for example, dradmin) and log in as root.

# **OPTIONS**

#### -info

Displays disaster recovery information such as the MirrorView/S device group eligible, displays the MirrorView/S Data Mover configuration for the current Celerra Network Server.

# -init <cel\_name>

Initializes the MirrorView/S relationship between the source and destination Celerra Network Servers based on if the configuration is active/passive (unidirectional) or active/active' (bidirectional).

**Note:** The apostrophe in active/active' indicates that both sites have source LUNs mirrored at the other site.

The passphrase-protected relationship between the source and destination Celerra Network Servers in the MirrorView/S configuration must be built prior to initialization using the **nas\_cel -create** command:

- On the destination Control Station in a MirrorView/S
   active/passive configuration, use the -init option to specify the
   name of the source Celerra Network Server.
- On the Control Station of each Celerra Network Server in a MirrorView/S active/active' configuration, use the -init option to specify the name of the remote Celerra Network Server. The active/active configuration is a bidirectional configuration in which a Celerra can serve both as source and destination for another Celerra.

### -activate

Executed from the destination Celerra Network Server using the remote administration account, initiates a failover from the source to the destination Celerra Network Server. The activation works as follows:

- If the source is available, the **-activate** option swaps the primary-secondary role for all mirrors in the MirrorView/S device group and makes the destination LUNs read/write. The standby Data Movers acquire the IP and MAC addresses, filesystems, and export tables of their source counterparts.
- If the original source site is unavailable, the destination LUNs are promoted to the primary role, making them visible to the destination Celerra. The original source LUNs cannot be converted to backup images; they stay visible to the source Celerra, and the original destination site is activated with new source (primary) LUNs only. If the source cannot be shut down in a disaster scenario, any writes occurring after the forced activation will be lost during a restore.

## -restore

Issued from the destination Celerra Network Server using the remote administration account, restores a source Celerra Network Server after a MirrorView/S failover, and fails back the device group to the source Celerra Network Server.

The restore process begins by checking the state of the device group. If the device group state is Local Only (where each mirror has only the source LUN), the device group will be fully synchronized and rebuilt before the failback can occur. If the device group condition is

fractured, an incremental synchronization is performed before the failback occurs. Source devices are then synchronized with the data on the original destination devices, I/O access is shut down, the original destination Data Movers are rebooted as remote standbys, and the mirrored devices are failed back. When the source side is restored, the source Data Movers and their services are restarted.

If the restore fails, the source Control Station is not reachable on the data network. To complete the restore, access the source, log in as **root**, and type **/nasmcd/sbin/nas\_mview -restore**.

# SEE ALSO

*Using MirrorView/Synchronous with Celerra for Disaster Recovery,* nas cel, and nas checkup.

# STORAGE SYSTEM OUTPUT

The number associated with the storage device reflects the attached storage system; for MirrorView/S, some CLARiiON storage systems display a prefix of APM before a set of integers, for example, APM00033900124-0019.

The CLARiiON storage system supports the following system-defined AVM storage pools for MirrorView/S only: cm\_r1, cm\_r5\_performance, cm\_r5\_economy, cmata\_archive, cmata\_r3, cm\_r6, and cmata\_r6.

# **EXAMPLE #1**

To initialize a destination Celerra in an active/passive configuration to communicate with source site source\_cs, from the destination Control Station, type:

# # /nas/sbin/nas\_mview -init source\_cs

Celerra with MirrorView/Synchronous Disaster Recovery

```
Initializing source_cs --> target_cs
```

Contacting source\_cs for remote storage info

Local storage system: APM00053001549 Remote storage system: APM00053001552

Enter the Global CLARiiON account information

Username: emc

Password: \*\*\* Retype your response to validate

Password: \*\*\*

Discovering storage on source\_cs (may take several minutes)
Setting security information for APM00053001549
Discovering storage APM00053001552 (may take several minutes)

Discovering storage (may take several minutes)

```
Contacting source_cs for remote storage info
Gathering server information...
Contacting source_cs for server capabilities...
Analyzing server information...
Source servers available to be configured for remote DR
______
    server_2:source_cs
1.
     server_3:source_cs [ local standby ]
2.
     Verify standby server configuration
v.
     Quit initialization process
q.
     Continue initialization
Select a source_cs server: 1
Destination servers available to act as remote standby
_____
      server_2:target_cs [ unconfigured standby ]
     server_3:target_cs [ unconfigured standby ]
     Back
Select a target_cs server: 1
Source servers available to be configured for remote DR
_____
1.
     server_2:source_cs [ remote standby is server_2:target_cs ]
     server_3:source_cs [ local standby ]
     Verify standby server configuration
v.
     Quit initialization process
q.
      Continue initialization
c.
Select a source_cs server: 2
Destination servers available to act as remote standby
_____
      server_2:target_cs [ is remote standby for server_2:source_cs ]
      server_3:target_cs [ unconfigured standby ]
b.
      Back
Select a target_cs server: 2
Source servers available to be configured for remote DR
_____
     server_2:source_cs [ remote standby is server_2:target_cs ]
2.
     server_3:source_cs [ remote standby is server_3:target_cs ]
     Verify standby server configuration
v.
     Quit initialization process
q.
     Continue initialization
Select a source_cs server: c
Standby configuration validated OK
Enter user information for managing remote site source_cs
Username: dradmin
Password: *****
                           Retype your response to validate
Password: ******
```

```
Active/Active configuration
Initializing (source_cs-->target_cs)

Do you wish to continue? [yes or no] yes

Updating MirrorView configuration cache
Setting up server_3 on source_cs
Setting up server_2 on source_cs
Creating user account dradmin
Setting acl for server_3 on target_cs
Setting acl for server_2 on target_cs
Updating the Celerra domain information
Creating device group mviewgroup on source_cs
done
```

**EXAMPLE #2** To get information about a source MirrorView configuration (for example, on new\_york configured as active/passive), type:

#### # /nas/sbin/nas\_mview -info

```
***** Device Group Configuration *****
                     = mviewgroup
name
description
                    = 50:6:1:60:B0:60:26:BC:0:0:0:0:0:0:0:0
                    = Consistent
state
role
                    = Primary
condition
                   = Active
recovery policy
                   = Automatic
number of mirrors
                    = 16
mode
                    = SYNC
owner
                    = 0
mirrored disks
root_disk,root_ldisk,d5,d8,d10,d11,d24,d25,d26,d27,d29,d30,d31,d32,d33,d39,
local clarid = APM00053001552
remote clarid
                    = APM00053001549
mirror direction
                   = local -> remote
**** Servers configured with RDFstandby *****
id
         = 1
name
        = server_2
acl
        = 1000, owner=nasadmin, ID=201
type
        = nas
slot
member_of =
standby = server_3, policy=auto
RDFstandby= slot=2
status :
  defined = enabled
  actual = online, active
```

```
id = 2
name = server_3
acl = 1000, owner=nasadmin, ID=201
type = standby
slot = 3
member_of =
standbyfor= server_2
RDFstandby= slot=3
status :
  defined = enabled
   actual = online, ready

***** Servers configured as standby *****
No servers configured as standby
```

# Where:

Value	Definition
Device group configuration:	
name	Name of the consistency (device) group.
description	Brief description of device group.
uid	UID assigned, based on the system.
state	State of the device group (for example, Consistent, Synchronized, Out-of-Sync, Synchronizing, Scrambled, Empty, Incomplete, or Local Only).
role	Whether the current system is the Primary (source) or Secondary (destination) for this group.
condition	Whether the group is functioning (Active), Inactive, Admin Fractured (suspended), Waiting on Sync, System Fractured (which indicates link down), or Unknown.
recovery policy	Type of recovery policy (Automatic is the default and recommended value for group during storage system configuration; if Manual is set, you must use -resume after a link down failure).
number of mirrors	Number of mirrors in group.
mode	MirrorView mode (always SYNC in this release).
owner	ACL ID assigned (0 indicates no control). nas_acl provides information.
mirrored disks	Comma-separated list of disks that are mirrored.
local clarid	APM number of local CLARiiON storage array.
remote clarid	APM number of remote CLARiiON storage array.
mirror direction	On primary system, local to remote (on primary system); on destination system, local from remote.

Value	Definition
Servers configured with RDFstandby/ Servers configured as standby:	
id	Server ID
name	Server name
acl	ACL value and owner
type	Server type (for example, nas or standby)
slot	Slot number for this Data Mover
member_of	If applicable, shows membership information.
standby	If this Data Mover is configured with local standbys, the server that is the local standby and any policy information.
RDFstandby	If this Data Mover is configured with a remote RDF standby, the slot number of the destination Data Mover that serves as the RDF standby.
standbyfor	If this Data Mover is also configured as a local standby, the server numbers for which it is a local standby.
status	Indicates whether the Data Mover is defined and online/ready.

**EXAMPLE #3** To activate a failover, log in to destination Control Station using **dradmin** account, **su** to **root**, and type:

## # /nas/sbin/nas\_mview -activate

```
Validating mirror group configuration ..... done
Is source site source_cs ready for complete shut down (power OFF)? [yes or no] yes
Contacting source site source_cs, please wait... done
Shutting down remote site source_cs ...... done
Sync with CLARiiON backend ..... done
STARTING an MV 'FAILOVER' operation.
Device group: mviewgroup ..... done
The MV 'FAILOVER' operation SUCCEEDED.
Failing over Devices ... done
Adding NBS access for server_2 ..... done
Adding NBS access for server_3 ..... done
Activating the target environment ... done
server_2 : going offline
rdf : going active
replace in progress ...done
failover activity complete
server_3 : going offline
rdf : going active
replace in progress ...done
failover activity complete
commit in progress (not interruptible)...done
```

Sync with CLARiiON backend ..... done

```
commit in progress (not interruptible)...done
commit in progress (not interruptible)...done
commit in progress (not interruptible)...done
```

done

# **EXAMPLE #4** To restore, log in to the destination Control Station using **dradmin** account, as **root** user, and type:

## # /nas/sbin/nas\_mview -restore

```
Sync with CLARiiON backend ..... done
Validating mirror group configuration ..... done
Contacting source site source_cs, please wait... done
Running restore requires shutting down source site source_cs.
Do you wish to continue? [yes or no] yes
Shutting down remote site source_cs ..... done
Is source site source_cs ready for storage restoration ? [yes or no] yes
Sync with CLARiiON backend ..... done
STARTING an MV 'RESUME' operation.
Device group: mviewgroup ..... done
The MV 'RESUME' operation SUCCEEDED.
Percent synchronized: 100
Updating device group ... done
Is source site ready for network restoration ? [yes or no] yes
Restoring servers ..... done
Waiting for servers to reboot ..... done
Removing NBS access for server_2 .. done
Removing NBS access for server_3 .. done
Waiting for device group ready to failback .... done
Sync with CLARiiON backend ..... done
STARTING an MV 'FAILBACK' operation.
Device group: mviewgroup ..... done
The MV 'FAILBACK' operation SUCCEEDED.
Restoring remote site source_cs ..... failed
Error 5008: -1:Cannot restore source_cs. Please run restore on site source_cs.
```

# Then on the Source Control Station, as the **root** user, type:

# # /nasmcd/sbin/nas\_mview -restore

Stopping NAS services. Please wait...

```
Powering on servers ( please wait ) ..... done
Sync with CLARiiON backend ..... done
STARTING an MV 'SUSPEND' operation.
Device group: mviewgroup ....... done
The MV 'SUSPEND' operation SUCCEEDED.
server_2 : going standby
rdf : going active
replace in progress ...done
failover activity complete
server_3 : going standby
rdf : going active
```

# The nas Commands

replace in progress ...done
failover activity complete
commit in progress (not interruptible)...done
commit in progress (not interruptible)...done
Sync with CLARiiON backend ..... done
STARTING an MV 'RESUME' operation.
Device group: mviewgroup ....... done
The MV 'RESUME' operation SUCCEEDED.
Restarting NAS services ..... done
commit in progress (not interruptible)...done
commit in progress (not interruptible)...done
done

# nas\_pool

Manages the user-defined and system-defined storage pools for the Celerra Network Server.

## **SYNOPSIS**

```
nas pool
  -list
  -info {<name>|id=<id>|-all}
  -size {<name>|id=<id>|-all} [-mover <mover>][-slice {y|n}]
    [-storage <system_name>]
  -create [-name <name>][-acl <acl>][-description <desc>]
    [-volumes <volume_name>[, <volume_name>, ...]]
    [-default_slice_flag {y | n}] [-is_greedy {y | n}]
 -create [-name <name>][-acl <acl>][-description <desc>]
    [-default_slice_flag {y|n}] [-is_greedy {y|n}]
    -size <integer>[M|G|T][-storage <system_name>]
    -template <system_pool_name> [-num_stripe_members <num>]
          [-stripe_size <num>]
  -modify {<name> | id=<id>} [-name <name>] [-acl <acl>]
    [-description <desc>][-default_slice_flag {y|n}]
    [-is\_dynamic \{y|n\}][-is\_greedy \{y|n\}]
  -delete {<name>|id=<id>} [-deep]
  -xtend {<name>|id=<id>} -volumes <volume_name>[,<volume_name>,...]
  -xtend {<name>|id=<id>} -size <integer> [M|G|T][-storage <system_name>]
  -shrink {<name> | id=<id>} -volumes <volume_name>[, <volume_name>, ...] [-deep]
```

# **DESCRIPTION**

**nas\_pool** creates, deletes, extends, shrinks, lists, displays, manages the access control level, and modifies a user-defined storage pool.

**nas\_pool** extends, shrinks, lists, displays, and modifies system-defined storage pools.

# **OPTIONS**

#### -list

Lists all storage pools on the Celerra Network Server.

```
-info {<name>|id=<id>|-all}
```

Displays detailed information for the specified storage pool, or all storage pools.

```
-size {<name>|id=<id>|-all}
```

Displays the size for the specified storage pool, or all storage pools.

```
[-mover <mover>]
```

Displays size information that is visible to the physical Data Mover or the virtual Data Mover (VDM).

# [-slice {y | n}]

If **y** is typed, displays size information when volumes in the storage pool are sliced. If **n** is typed, displays size information when volumes in the storage pool are not sliced. The **-slice** option defaults to the value of **default\_slice\_flag** for the storage pool.

```
[-storage <system_name>]
```

Displays size information for members that reside on a specified storage system.

#### -create

Creates a user-defined storage pool.

```
[-name <name>]
```

Assigns a name to the new storage pool. If no name is specified, assigns one by default.

```
[-acl <acl>]
```

Sets an access control level value that defines the owner of the storage pool, and the level of access allowed for users and groups defined in the access control level table. The nas\_acl command provides more information.

```
[-description <desc>]
```

Assigns a comment to the storage pool.

```
[-volumes <volume_name>[, <volume_name>, ...]
```

Designates the members to be added to the storage pool. The members can be any meta, slice, stripe, or disk volumes.

```
[-default_slice_flag {y | n}]
```

If set to y (default), then members of the storage pool might be sliced when space is allocated from the storage pool. If set to n, members of the storage pool will not be sliced when space is dispensed from the storage pool and the volumes specified cannot be built on a slice.

```
[-is\_greedy \{y | n\}]
```

If set to **n** (default), the system uses space from the user-defined storage pool's existing member volumes in the order that the volumes were added to the pool to create a new filesystem or extend an existing filesystem.

If set to y, the user-defined storage pool uses space from the least-used member volume to create a new filesystem. When there is more than one least-used member volume available, AVM selects the member volume that contains the most disk volumes. For example, if one member volume contains four disk volumes

and another member volume contains eight disk volumes, AVM selects the one with eight disk volumes. If there are two or more member volumes that have the same number of disk volumes, AVM selects the one with the lowest ID.

```
[-size <integer> {M|G|T}]
```

Creates a storage pool with the size specified. When specifying a size, type an integer between 1 and 1024, then specify T for terabytes, G for gigabytes (default), or M for megabytes.

```
[-storage <system_name>]
```

Specifies the storage system on which one or more volumes will be created, to be added to the storage pool.

```
[-template <system_pool_name>]
```

Specifies a system pool name, required when the **-size** option is specified. The user pool will be created using the profile attributes of the specified system pool template.

```
[-num_stripe_members <num>]
```

Specifies the number of stripe members for user pool creation by size. The **-num\_stripe\_members** option works only when both **-size** and **-template** options are specified. It overrides the number of stripe members attribute of the specified system pool template.

```
[-stripe_size <num>]
```

Specifies the stripe size for user pool creation by size. The **-stripe\_size** option works only when both **-size** and **-template** options are specified. It overrides the stripe size attribute of the specified system pool template.

```
-modify {<name> | id=<id>}
```

Modifies the attributes of the specified user-defined or system-defined storage pool.

Managing Celerra Volumes and File Systems with Automatic Volume Management lists the available system-defined storage pools.

```
[-name <name>]
```

Changes the name of the storage pool to the new name.

```
[-acl <acl>]
```

Sets an access control level value that defines the owner of the storage pool, and the level of access allowed for users and groups defined in the access control level table. The nas\_acl command provides more information.

```
[-description <desc>]
```

Changes the comment for the storage pool.

```
[-default_slice_flag {y | n}]
```

If set to y (default), then members of the storage pool might be sliced when space is dispensed from the storage pool. If set to n, members of the storage pool will not be sliced when space is dispensed from the storage pool and the volumes specified cannot be built on a slice.

```
[-is_dynamic {y|n}]
```

Allows a system-defined storage pool to automatically extend or shrink member volumes.

**Note:** The **-is\_dynamic** option is for system-defined storage pools only.

# [-is\_greedy {y|n}]

For system-defined storage pools, if set to y, then the storage pool attempts to create new member volumes before using space from existing member volumes. A system-defined storage pool that is not greedy (set to n), consumes all the space existing in the storage pool before trying to add additional member volumes. A y or n value must be specified when modifying a system-defined storage pool.

For user-defined storage pools, if set to  $\mathbf{n}$  (default), the system uses space from the user-defined storage pool's existing member volumes in the order that the volumes were added to the pool to create a new filesystemfilesystem.

For user-defined storage pools, if set to **y**, the system uses space from the least-used member volume in the user-defined storage pool to create a new filesystem. When there is more than one least-used member volume available, AVM selects the member volume that contains the most disk volumes. For example, if one member volume contains four disk volumes and another member volume contains eight disk volumes, AVM selects the one with eight disk volumes. If there are two or more member volumes that have the same number of disk volumes, AVM selects the one with the lowest ID.

For both system-defined and user-defined pools when extending a filesystem, the is\_greedy attribute is ignored unless there is not enough free space on the existing volumes that the filesystem is using to meet the requested extension size.

```
-delete {<name>|id=<id>}
```

Deletes a storage pool. Storage pools cannot be deleted if any members are in use. After deletion, the storage pool no longer exists on the system, however, members of the storage pool are not deleted.

## [-deep]

Deletes the storage pool and also recursively deletes each member of the storage pool. Each storage pool member is deleted unless it is in use or is a disk volume.

```
-xtend {<name>|id=<id>} -volumes <volume_name>
[, <volume_name>,...]
```

Adds one or more unused volumes to a storage pool. If the

**default\_slice\_value** is set to **n**, member volumes cannot contain slice volumes (for compatibility with TimeFinder/FS).

**Note:** Extending a storage pool by volume is for user-defined storage pools only.

-xtend  ${\text{-xiend}} {\text{-size}} {\text{-integer}} [M|G|T]$ 

Extends the specified storage pool with one or more volumes of the size equal to or greater than the size specified. When specifying the volume by size, type an integer between 1 and 1024, then specify T for terabytes, **G** for gigabytes (default), or **M** for megabytes.

```
[-storage <system_name>]
```

Specifies the storage system on which one or more volumes will be created, to be added to the storage pool.

**Note:** To successfully extend a system-defined storage pool by size, the **is\_dynamic** attribute must be set to **n**, and there must be enough available disk volumes to satisfy the request.

```
-shrink {<name>|id=<id>} -volumes <volume_name>
[,<volume_name>,...][-deep]
```

Shrinks the storage pool by the specified unused volumes. When the **-deep** option is used to shrink a user-defined storage pool, it removes the specified member volumes from the pool, and recursively deletes any unused volumes unless it is a disk volume. If the **-deep** option is not used to shrink a user-defined storage pool, the member volumes are left intact so that they can be reused. The **is\_dynamic** option must be set to **n** before shrinking system-defined storage pools.

**Note:** Shrinking of a system-defined storage pool by default deletes member volumes automatically. Specifying the **-deep** option on the system-defined storage pool shrink does not make any difference.

# **SEE ALSO**

Managing Celerra Volumes and File Systems with Automatic Volume Management, Managing Celerra Volumes and File Systems Manually, Controlling Access to Celerra System Objects, UsingTimeFinder/FS, NearCopy, and FarCopy with Celerra, fs\_timefinder, nas\_fs, nas\_volume, and nas\_slice.

# STORAGE SYSTEM OUTPUT

Celerra Network Servers with a CLARiiON storage system support the following traditional system-defined storage pools: clar\_r1, clar\_r5\_performance, clar\_r5\_economy, clar\_r6, clarata\_r3,

clarata\_r6, clarata\_r10, clarata\_archive, cm\_r1, cm\_r5\_performance, cm\_r5\_economy, cm\_r6, cmata\_r3, cmata\_archive, cmata\_r6, cmata\_r10, clarsas\_archive, clarsas\_r6, clarsas\_r10, clarefd\_r5, clarefd\_r10, cmsas\_archive, cmsas\_r6, cmsas\_r10, and cmefd\_r5.

Celerra Network Servers with a CLARiiON CX4 storage system also support virtually provisioned storage system Thin Pools. One CLARiiON storage system Thin Pool is mapped to one Celerra system-defined virtual AVM pool with a default naming convention of "<thin pool name>\_<backend id>".

Disk types when using CLARiiON are CLSTD, CLEFD, and CLATA, and for CLARiiON storage systems involving mirrored disks are: CMEFD, CMSTD, and CMATA.

Celerra Network Servers with a Symmetrix storage system support the following system-defined storage pools: symm\_std, symm\_std\_rdf\_src, symm\_ata\_rdf\_src, symm\_ata\_rdf\_tgt, symm\_std\_rdf\_tgt, and symm\_efd.

For user-defined storage pools, the difference in output is in the disk type. Disk types when using a Symmetrix are STD, R1STD, R2STD, BCV, R1BCV, R2BCV, ATA, R1ATA, R2ATA, BCVA, R1BCA, R2BCA, and EFD.

# **EXAMPLE #1**

id

To create a storage pool with the name, **marketing**, with a description, with the following disk members, **d12**, **d13**, and with the default slice flag set to **y**, type:

```
name
                   = marketing
description
                   = Storage Pool
                   = 0
acl
in_use
                   = False
clients
members
                   = d12.d13
default_slice_flag = True
is_user_defined = True
virtually_provisioned= True
disk_type
              = CLSTD
server_visibility = server_2, server_3, server_4, server_5
template_pool
              = N/A
num\_stripe\_members = N/A
stripe_size
                   = N/A
```

= 20

# Where:

Value	Definition
id	ID of the storage pool.
name	Name of the storage pool.
description	Comment assigned to the storage pool.
acl	Access control level value assigned to the storage pool.
in_use	Whether the storage pool is being used by a filesystem.
clients	Filesystems using the storage pool.
members	Volumes used by the storage pool.
default_slice_flag	Allows slices from the storage pool.
is_user_defined	User-defined as opposed to system-defined.
virtually_provisioned	Indicates whether Virtual Provisioning is enabled or disabled.
disk_type	Type of disk contingent on the storage system attached. CLSTD, CLATA, CMSTD, CLEFD, CMEFD, and CMATA are for CLARiiON, and STD, BCV, R1BCV, R2BCV, R1STD, R2STD, ATA, R1ATA, R2ATA, BCVA, R1BCA, R2BCA, and EFD are for Symmetrix.
server_visibility	Storage pool is visible to the physical Data Movers specified.
template_pool	System pool template used to create the user pool. Only applicable to user pools created by size or if the last member volume is a stripe or both.
num_stripe_members	Number of stripe members used to create the user pool. Applicable to system pools and user pools created by size or if the last member volume is a stripe or both.
stripe_size	Stripe size used to create the user pool. Applicable to system pools and user pools created by size or if the last member volume is a stripe or both.

# **EXAMPLE #2**

To change the description for the **marketing** storage pool to include a descriptive comment, type:

# \$ nas\_pool -modify marketing -description 'Marketing Storage Pool'

```
id
                    = 20
                   = marketing
name
description
                   = Marketing Storage Pool
                   = 0
acl
in_use
                   = False
clients
members
                   = d12, d13
default_slice_flag = True
is_user_defined = True
virtually_provisioned= True
disk_type
           = CLSTD
server_visibility = server_2, server_3, server_4, server_5
template_pool = N/A
num\_stripe\_members = N/A
stripe_size
                    = N/A
```

## **EXAMPLE #3**

To view the size information for the **engineer\_APM00084401666** pool, type:

\$ nas\_pool -size engineer\_APM00084401666 -mover server\_2
-slice y -storage APM00084401666

```
id = 40
```

name = engineer\_APM00084401666

Physical storage usage in Thin Pool engineer on APM00084401666:

used\_mb = 7168 avail\_mb = 1088578 total\_mb = 1095746

## Where:

Value	Definition
id	ID of the storage pool.
name	Name of the storage pool.
used_mb	Space in use by the storage pool specified.
avail_mb	Unused space still available in the storage pool.
total_mb	Total space in the storage pool (total of used and unused).
potential_mb	Available space that can be added to the storage pool.
Physical used_mb	Used physical size of a storage system Thin Pool in MB (some may be used by non-Celerra hosts).
Physical avail_mb	Available physical size of a storage system Thin Pool in MB.
Physical total_mb	Physical size of a storage system Thin Pool in MB (shared by Celerra and potential non-Celerra hosts).

**Note:** Each of the options used with the command **nas\_pool - size** is filters for the output of the command. For example, if you specify a Data Mover, the output will reflect only the space to which the specified Data Mover has visibility. Physical used\_mb, Physical avail\_mb, and Physical total\_mb are applicable for system-defined virtual AVM pools only.

## **EXAMPLE #4**

For Celerra Network Servers with a CLARiiON storage system, to change the **-is\_greedy** and **-is\_dynamic** options for the system defined, **clar\_r5\_performance** storage pool, type:

\$ nas\_pool -modify clar\_r5\_performance -is\_dynamic n
-is\_greedy y

id = 3

acl = 421in\_use = False

```
clients
                      = v120
members
default_slice_flag = True
is_user_defined = False
virtually_provisioned= False
                = CLSTD
disk_type
server_visibility = server_2, server_3, server_4, server_5
volume_profile = clar_r5_performance_vp
is_dynamic = False
is_greedy
                      = True
num_stripe_members = 4
                        = 32768
stripe_size
                          For Celerra Network Servers with a Symmetrix storage system, to
                          change the -is_greedy and -is_dynamic options for the
                          system-defined, symm_std storage pool, type:
                          $ nas_pool -modify symm_std -is_dynamic y -is_greedy y
id
                        = 1
name
                      = symm_std
description
                      = Symmetrix STD
```

 $disk\_type = STD$ 

server\_visibility = server\_2, server\_3, server\_4, server\_5

volume\_profile = symm\_std\_vp

#### Where:

Value	Definition
id	ID of the storage pool.
name	Name of the storage pool.
description	Comment assigned to the storage pool.
acl	Access control level value assigned to the storage pool.
in_use	Whether the storage pool is being used by a filesystem.
clients	Filesystems using the storage pool.
members	Disks used by the storage pool.
default_slice_flag	Allows slices from the storage pool.
is_user_defined	User-defined as opposed to system-defined.
virtually_provisioned	Indicates whether Virtual Provisioning is enabled or disabled.

disk_type	Contingent on the storage system attached.
server_visibility	Storage pool is visible to the physical Data Movers specified.
volume_profile	Volume profile used.
is_dynamic	Whether the system-defined storage pool can add or remove volumes.
is_greedy	Indicates whether the system-defined storage pool will use new member volumes as needed.
template_pool	System pool template used to create the user pool. Only applicable to user pools created by size or if the last member volume is a stripe or both.
num_stripe_members	Number of stripe members used to create the user pool. Applicable to system pools and user pools created by size or if the last member volume is a stripe or both.
stripe_size	Stripe size used to create the user pool. Applicable to system pools and user pools created by size or if the last member volume is a stripe or both.

To change the **-is\_greedy** option for the user-defined, **user\_pool** storage pool, type:

```
$ nas_pool -modify user_pool -is_greedy y
id
                    = 58
name
                    = user_pool
description
                    = 0
acl
in_use
                    = False
clients
members
                    = d21, d22, d23, d24
default_slice_flag = True
is_user_defined
                   = True
virtually_provisioned= False
disk_type
          = CLSTD
server_visibility = server_2
is_greedy
                    = True
template_pool
                   = N/A
num\_stripe\_members = N/A
stripe_size
                    = N/A
```

# **EXAMPLE #5** To add the volumes, **d7** and **d8**, to the **marketing** storage pool, type:

```
$ nas_pool -xtend marketing -volumes d7,d8
id
                     = 20
name
                     = marketing
description
                     = Marketing Storage Pool
                     = False
in_use
clients
members
                     = d12, d13, d7, d8
default_slice_flag
                     = True
is_user_defined
                    = True
virtually_provisioned= True
disk_type
                    = CLSTD
server_visibility = server_2, server_3, server_4, server_5
```

```
template_pool = N/A
num_stripe_members = N/A
stripe_size = N/A
```

### **EXAMPLE #6**

For Celerra Network Servers with a CLARiiON storage system, to extend the system-defined storage pool by a specified size with a specified storage storage system, type:

\$

-xtend clar\_r5\_performance -size 128M -storage APM00042000818

```
id = 3
```

 $egin{array}{lll} \mbox{name} & = \mbox{clar_r5\_performance} \ \mbox{description} & = \mbox{CLARiiON RAID5 4plus1} \end{array}$ 

acl = 1421, owner=nasadmin, ID=201

is\_user\_defined = False
virtually\_provisioned= False
disk\_type = CLSTD

server\_visibility = server\_2,server\_3,server\_4,server\_5
volume\_profile = clar\_r5\_performance\_vp

is\_dynamic = False is\_greedy = True

num\_stripe\_members = 4 stripe\_size = 32768

## **EXAMPLE #7**

For Celerra Network Servers with a CLARiiON storage system, to remove **d7** and **d8** from the marketing storage pool, type:

```
$ nas_pool -shrink marketing -volumes d7,d8
```

id = 20

name = marketing

description = Marketing Storage Pool

acl = 0 in\_use = False clients =

members = d12,d13
default\_slice\_flag = True
is\_user\_defined = True
virtually\_provisioned= True
disk\_type = CLSTD

server\_visibility = server\_2, server\_3, server\_4, server\_5

template\_pool = N/A
num\_stripe\_members = N/A
stripe\_size = N/A

# **EXAMPLE #8** To list the storage pools, type:

\$ nas\_pool -list

id	inuse	acl	name
1	n	421	symm_std
2	n	421	clar_r1
3	n	421	clar_r5_performance
4	n	421	clar_r5_economy
8	n	421	symm_std_rdf_src
10	n	421	clarata_archive
11	n	421	clarata_r3
20	n	0	marketing
40	У	0	engineer_APM0084401666

## Where:

Value	Definition
id	ID of the storage pool.
inuse	Whether the storage pool is being used by a filesystem.
acl	Access control level value assigned to the storage pool.
name	Name of the storage pool.

# **EXAMPLE #9** To display information about the user-defined storage pool called **marketing**, type:

```
$ nas_pool -info marketing
```

```
id = 20
```

name = marketing

description = Marketing Storage Pool

default\_slice\_flag = True
is\_user\_defined = True
virtually\_provisioned= True
disk\_type = CLSTD

server\_visibility = server\_2, server\_3, server\_4, server\_5

 $\begin{array}{lll} \texttt{template\_pool} & = & \texttt{N/A} \\ \texttt{num\_stripe\_members} & = & \texttt{N/A} \\ \texttt{stripe\_size} & = & \texttt{N/A} \\ \end{array}$ 

# **EXAMPLE #10** To display information about the system-defined clar\_r5\_performance storage pool, type:

## \$ nas\_pool -info clar\_r5\_performance

id = 3

acl = 1421, owner=nasadmin, ID=201

members = v120 default\_slice\_flag = True

```
is_user_defined
                  = False
virtually_provisioned= False
               = CLSTD
disk_type
server_visibility = server_2,server_3,server_4,server_5
volume_profile = clar_r5_performance_vp
is dynamic
                     = False
is_greedy
                     = True
num_stripe_members = 4
stripe_size
                     = 32768
```

# **EXAMPLE #11**

To display information about the system-defined engineer\_APM00084401666 virtual pool, type:

```
$ nas_pool -info engineer_APM00084401666
```

id = 40

name = engineer\_APM00084401666

description = Thin Pool engineer on APM00084401666

= 0acl in\_use = True

clients = DA\_BE\_VIRT\_FS, vp\_test, vp\_test1, vp\_test12, cvpfs1, cvpfs3

members = v363default\_slice\_flag = True is\_user\_defined = False virtually\_provisioned= True disk\_type = CLSTD

server\_visibility = server\_2,server\_3
volume\_profile = engineer\_APM00084401666\_vp

is\_dynamic = True = True is\_greedy  $num\_stripe\_members = N/A$ stripe\_size = N/A

## **EXAMPLE #12**

To delete the storage pool, **marketing**, and each of the storage pool member volumes recursively, type:

## \$ nas\_pool -delete marketing -deep

id = 20

name = marketing

description = Marketing Storage Pool

acl = 0in\_use = False

clients members

default\_slice\_flag = True
is\_user\_defined = True
virtually\_provisioned= True
template\_pool = N/A
num\_stripe\_members = N/A
stripe\_size = N/A

# nas\_quotas

Manages quotas for mounted filesystems.

## **SYNOPSIS**

```
nas_quotas
  -edit [-user|-group] {-mover <movername>|-fs <fs_name>} [-path
  <pathname>]} [[-proto <proto_id>]|[-block <hard_limit>[:<soft_limit>]]
  [-inode <hard_limit>[:<soft_limit>]]] <id> [<id>...]
-edit -config {-mover <movername>|-fs <fs_name>} [-path <pathname>]}
    [-option <options>]
-edit -tree -fs <fs_name>
    [[-proto <proto_id>] | [-block <hard_limit>[:<soft_limit>]]
    [-inode <hard_limit>[:<soft_limit>]]]
    [-comment <comment>] <id> [<id>...]
| -report [-user|-group] {-mover <movername>|-fs <fs_name>}
    -report -config {-mover <movername> | -fs <fs_name> } [-path <pathname>] }
 -report -tree -fs <fs_name> [<id> <id>...]
| {-on|-off|-clear} [-user|-group|-both]
    {-mover <movername>|-fs <fs_name>|[-path <pathname>] -all}
 -on -tree -fs <fs_name> -path <pathname> [-comment <comment>]
 -off -tree -fs <fs_name> -path <pathname>
 {-list|-clear} -tree -fs <fs_name>
```

### DESCRIPTION

nas\_quotas edits quotas for mounted filesystems, and displays a listing of quotas and disk usage at the filesystem level (by the user, group, or tree), or at the quota-tree level (by the user or group).

**nas\_quotas** turns quotas on and off, and clears quotas records for a filesystem, quota tree, or a Data Mover. When a Data Mover is specified, the action applies to all mounted filesystems on the Data Mover.



### CAUTION

Quotas should be turned on (enabled) before filesystems go into a production environment. Enabling (or disabling, or clearing) quotas in a production environment is time consuming and the process may disrupt filesystem operation. CIFS clients are disconnected during these events and NFS clients receive a message that the server is not responding. However, once enabled, quotas can be changed at any time without impact.

## **OPTIONS**

```
-edit [-user|-group] {-mover <movername>|-fs <fs_name>
[-path <pathname>]} [<id> [<id>...]
```

Sets the quota limits for users or groups on a specified Data Mover, mounted filesystem, or directory tree.

For a user, the ID can be a user ID or UID, however, if NIS or the local password file on the Data Mover is available, a username can also be used.

For a group, the ID can be a group ID or GID, however, if NIS or the local password file is available, a group name can also be used.

Upon execution, a **vi** session (unless the EDITOR environment variable specifies otherwise) is opened to edit the quota configuration file. Changes to the file are applied when the **vi** session is saved and exited.

```
[-proto <proto_id>] | [-block <hard_limit>
[:<soft_limit>]]
```

Applies the quota configuration defined for the prototype user for each specified ID, and sets a hard and soft limit for storage (block) usage in kilobytes.

```
[-inode <hard_limit>[:<soft_limit>]][<id>
[<id>...][-block <hard_limit>[:<soft_limit>]]
```

Edits the inode (file count) limits and the block (storage in KBs) limits directly into the quota configuration file without opening an editing session.

```
-edit -config {-mover <movername>|-fs <fs_name>} [-path
<pathname>] }
```

Edits the default quota configuration for all users/groups currently without quotas or subsequently added to the specified Data Mover or filesystem or quota tree. Also edits the grace periods for soft quotas, and the conditions upon which to generate a quotas-event message to the system log.

```
[-option <options>]
```

Specifies the following comma-separated options:

```
BGP=<integer>
```

Sets the block grace period in seconds.

```
IGP=<integer>
```

Sets the inode grace period in seconds.

```
DUBSL=<integer>
```

Sets the default user block soft limit in KB.

```
DUBHL=<integer>
```

Sets the default user block hard limit in KB.

DUISL = <integer>

Sets the default user inode soft limit.

**DUIHL**=<integer>

Sets the default user inode hard limit.

**DGBSL**=<integer>

Sets the default group block soft limit in KB.

DGBHL=<integer>

Sets the default group block hard limit in KB.

DGISL=<integer>

Sets the default group inode soft limit.

**DGIHL**=<integer>

Sets the default group inode hard limit.

HLE={True | False}

Specifies whether the hard limit is enforced.

ESFCS={True | False}

Specifies the event for check start has been sent.

ESFCE={True | False}

Specifies the event for check end has been sent.

ESFBSL={True | False}

Specifies that the event for block soft limits has been sent.

ESFBHL={True | False}

Specifies that the event for block hard limits has been sent.

```
-edit -tree -fs <fs_name> [[-proto <proto_id>] | [-block
<hard_limit> [:<soft_limit>]][-inode
<hard_limit>[:<soft_limit>]]][-comment <comment>] <id>
[<id>...]
```

Edits the quota limits for trees (inodes or blocks used by a tree directory) where the <id> is the tree ID. This option can only be applied on each filesystem basis. The **-list** option to display the tree IDs.

The **-proto** option applies the quota configuration of the prototype tree for each specified tree ID, or sets a hard and soft limit for blocks. The cproto\_id must be a tree ID.

The **-inode** and **-block** options edit the inode/block limits for the tree directly in the quota configuration file without opening an editing session.

The **-comment** option associates a comment with the quota tree. The comment is delimited by single quotes. Comment length is limited to 256 bytes (represented as 256 ASCII characters or a variable number of Unicode multibyte characters) and cannot include single quotes (' '), double quotes (" "), semicolons (;), NL (New Line), or FF (Form Feed).

```
-report [-user|-group] {-mover <movername>|-fs <fs_name>}
[-path <pathname>]} [<id> <id> ...]
```

Displays a summary of disk usage and quotas for the user or group, including the number of files and space in kilobytes for the specified <fs\_name>, or all filesystems mounted on the specified <movername>, or for the specified quota tree. The **-edit** option provides more information for the usage of UIDs and GIDs.

**Note:** The **nas\_quotas** can show report for maximum 1024 IDs at a time.

```
-report -config {-mover <movername>|-fs <fs_name>} [-path <pathname>]}
```

Displays quota configuration information as viewed from the specified Data Mover, filesystem, or quota-tree level, including:

- Active quota policy
- Quota status (user/group quotas enabled or disabled)
- Grace period
- Default limits currently set for users/groups
- Hard-quota enforcement option setting (deny disk space enabled or disabled)
- Quota conditions that trigger event-logging

```
-report -tree -fs <fs_name>[<id> <id>...]
Displays the quota limits for a specified quota tree in a filesystem.
The <id> is a tree ID.
```

**Note:** The <id> is either a user ID, a group ID, or a tree ID. If the quota type is not specified, the default is set to the '-user' ID.

```
{-on|-off|-clear} [-user|-group|-both] {-mover
<movername>|-fs <fs_name>|[-path <pathname>]|-all}
Turns quotas on, off, and clears quotas for the user, group, or both
```

(users and groups at once) on the <movername>, <fs\_name>, <pathname>, for all users, or groups on all filesystems on all Data Movers in the cabinet.

The **-clear** option permanently removes all quota records, deletes the quota configuration file, and turns quotas **off**.



## **CAUTION**

While quotas are being turned on, off, or cleared, other operations to a filesystem may be disrupted. CIFS clients are disconnected during this execution.

-on -tree -fs <fs\_name> -path <pathname>

Turns on (enables) tree quotas so that quota tracking and hard-limit enforcement (if enabled) can occur. When enabling tree quotas, the directory must not exist; it is created in this tree-quota-enabling process.

**Note:** The quota path length (which Celerra calculates as including the filesystem mountpoint) must be less than 1024 bytes. If Unicode is enabled on the selected Data Mover, **-path** accepts any characters defined by the Unicode 3.0 standard. Otherwise, it accepts only ASCII characters.

[-comment <comment>]

The **-comment** option associates a comment with the quota tree. The comment is delimited by single quotes. Comment length is limited to 256 bytes (represented as 256 ASCII characters or a variable number of Unicode multibyte characters) and cannot include single quotes (' '), double quotes (" "), semicolons (;), NL (New Line), or FF (Form Feed).

-off -tree -fs <fs\_name> -path <pathname>

Turns tree quotas off. When turning tree quotas off, the tree directory must be empty.

{-list | -clear} -tree -fs <fs\_name>

The **-list** option displays all active quota trees and their respective tree IDs used by **-edit** and **-report** with the specified filesystem.

Use the **-tree -clear** option to clear all the information from the database after you disable (turn off) quotas for all trees within a filesystem. Once cleared, the database information is not recoverable.



## **CAUTION**

The -clear option deletes the usage and the limit information for tree quotas. The limits cannot be recovered.

**SEE ALSO** 

Using Quotas on Celerra.

**EXAMPLE #1** 

To turn quotas on for users and groups of a filesystem, type:

\$ nas\_quotas -on -both -fs ufs1

done

EXAMPLE #2

To open a vi session to edit filesystem quotas on **ufs1** for the specified user, **1000**, type:

\$ nas\_quotas -edit -user -fs ufs1 1000

Userid : 1000

fs ufs1 blocks (soft = 2000, hard = 3000) inodes (soft = 0, hard = 0)

~

~

"/tmp/EdP.agGQuIz" 2L, 84C written done

**EXAMPLE #3** 

To change the block limit and inode limit for a filesystem without opening up a vi session, type:

\$ nas\_quotas -edit -user -fs ufs1 -block 7000:6000 -inode 700:600 2000

done

## **EXAMPLE #4**

To view a report of user quotas for **ufs1**, type:

## \$ nas\_quotas -report -user -fs ufs1

Report for user quotas on filesystem ufs1 mounted on /ufs1

User	   	Bytes Us	sed (1K)	)		Fil∈	es	
	Used	Soft	Hard	Timeleft	Used	Soft	Hard	Timeleft
#1000  #2000  #5000	1328    6992    141592	2000  6000  0	3000 7000 0	!!!	54  66  516	0 600 0	0 700 0	

done

# **EXAMPLE #5** To select user **300** as prototype user for **ufs1**, and assign other users the same limits, type:

\$ nas\_quotas -group -edit -fs ufs1 -proto 300 301 302 303

done

## **EXAMPLE #6** To display the group quotas information for **ufs1**, type:

## \$ nas\_quotas -report -group -fs ufs1

Report for group quotas on filesystem ufs1 mounted on /ufs1

Group	B		Files					
ļ	Used   Soft   Hard  Time:				Used	Soft	Hard	Timeleft
#1  #300  #301  #302  #303  #32772	296    6992    0    0    22296	0   6000   6000   6000   6000	7000 7000 7000 7000 7000		12 67 0 0 0 228	0  600  600  600  600	700 700 700 700 700	

done

## **EXAMPLE #7** To edit the default quota configuration for **server\_2**, type:

```
$ nas_quotas -edit -config -mover server_2
```

```
File System Quota Parameters:
fs "ufs1"
        Block Grace: (1.0 weeks)
        Inode Grace: (1.0 weeks)
        * Default Quota Limits:
           User: block (soft = 5000, hard = 8000) inodes (soft = 100, hard= 200)
           Group: block (soft = 6000, hard = 9000) inodes (soft = 200, hard= 400)
        Deny disk space to users exceeding quotas: (yes)
        * Generate Events when:
            Quota check starts:
                                          (no)
            Quota check ends:
                                          (no)
            soft quota crossed:
                                          (no)
            hard quota crossed:
                                          (no)
fs "ufs2"
        Block Grace: (1.0 weeks)
        Inode Grace: (1.0 weeks)
        * Default Quota Limits:
            User: block (soft = 0, hard = 0) inodes (soft = 0, hard= 0)
            Group: block (soft = 0, hard = 0) inodes (soft = 0, hard= 0)
        Deny disk space to users exceeding quotas: (yes)
        * Generate Events when:
            Quota check starts:
                                          (no)
            Quota check ends:
                                          (no)
            soft quota crossed:
                                          (no)
            hard quota crossed:
                                          (no)
```

```
~
~
"/tmp/EdP.ahCPdAB" 25L, 948C written
done
```

## **EXAMPLE #8** To open a vi session and edit the quotas for a filesystem, type:

```
$ nas_quotas -edit -config -fs ufs1
File System Quota Parameters:
fs "ufs1"
        Block Grace: (1.0 weeks)
        Inode Grace: (1.0 weeks)
        * Default Ouota Limits:
           User: block (soft = 5000, hard = 8000) inodes (soft = 100, hard= 200)
           Group: block (soft = 6000, hard = 9000) inodes (soft = 200, hard= 400)
        Deny disk space to users exceeding quotas: (yes)
        * Generate Events when:
            Quota check starts:
                                          (no)
            Ouota check ends:
                                         (no)
            soft quota crossed:
                                         (yes)
            hard quota crossed:
                                         (yes)
"/tmp/EdP.a4slhyg" 13L, 499C written
done
```

# **EXAMPLE #9** To view the quota configuration for the filesystem, **ufs1**, type:

## \$ nas\_quotas -report -config -fs ufs1

```
Quota parameters for filesystem ufs1 mounted on /ufs1:
 Ouota Policy: blocks
 User Quota: ON
 Group Quota: ON
 Block grace period: (1.0 weeks)
 Inode grace period: (1.0 weeks)
 Default USER quota limits:
     Block Soft: (
                        5000), Block Hard: (
                                                  8000)
     Inode Soft: (
                        100), Inode Hard: (
                                                  200)
 Default GROUP quota limits:
     Block Soft: (
                       6000), Block Hard: (
                                                  9000)
     Inode Soft: ( 200), Inode Hard: (
                                                   400)
 Deny Disk Space to users exceeding quotas:
 Log an event when ...
    Block hard limit reached/exceeded:
                                                 YES
    Block soft limit (warning level) crossed:
                                                 YES
    Quota check starts:
                                                  NO
    Quota Check ends:
done
```

**EXAMPLE #10** 

```
$ nas_quotas -on -tree -fs ufs1 -path /tree1 -comment
                        'Tree #1'
done
        EXAMPLE #11
                        To create a tree quota in a language that uses multibyte characters
                        (such as Japanese), type:
                        $ nas_quotas -on -tree -fs fs_22 -path / オリビアさん
done
        EXAMPLE #12
                        To list the tree quotas for ufs1, type:
                        $ nas_quotas -list -tree -fs ufs1
                 ______
 Quota trees for filesystem ufs1 mounted on /ufs1:
|TreeId| Quota tree path (Comment)
     1 | /tree1 (Tree #1)
     2 | /tree2 (Tree #2)
     3 | /<tree_path_in_local_language_text> (Tree #3)
done
        EXAMPLE #13
                        To edit or add a comment for a tree quota for ufs1, type:
                        $ nas quotas -edit -tree -fs ufs1 -comment 'Quota for
                        Tree1' 1
done
        EXAMPLE #14
                        To edit tree quotas for ufs1, type:
                        $ nas quotas -edit -tree -fs ufs1 1
treeid: 1
fs ufs1 blocks (soft = 6000, hard = 8000) inodes (soft = 200, hard = 300)
"/tmp/EdP.aiHKgh5" 2L, 85C written
done
        EXAMPLE #15
                        To edit tree quotas for ufs1 and change the block and inodes, type:
                        $ nas_quotas -edit -tree -fs ufs1 -block 8000:6000 -inode
                        900:800 1
done
        EXAMPLE #16
                        To edit tree quotas for ufs1 and apply the quota configuration of the
                        prototype tree, type:
                        $ nas_quotas -edit -tree -fs ufs1 -proto 1 2
done
```

To turn tree quotas on for **ufs1**, type:

## **EXAMPLE #17** To display any currently active trees on a filesystem, type:

## \$ nas\_quotas -report -tree -fs ufs1

Report for tree quotas on filesystem ufs1 mounted on /ufs1

Tree	+   E	Bytes Use	d (1K)	   		Fil∈	es	   
	Used	Soft	Hard	Timeleft	Used	Soft	Hard	Timeleft
#1  #2	384    7856	6000  6000	8000    8000	   7.0days	3   60	800   800	900 900	!

done

# **EXAMPLE #18** To turn tree quotas off, type:

\$ nas\_quotas -tree -off -fs ufs1 -path /tree1

done

**EXAMPLE #19** To turn quotas on for users and groups on tree quota, /tree3, of a filesystem, ufs1, type:

\$ nas\_quotas -on -both -fs ufs1 -path /tree3

done

50)

Userid: 1000

**EXAMPLE #20** To open a vi session to edit filesystem quotas on quota tree, /tree3, on ufs1 for the specified user, 1000, type:

fs ufs1 tree "/tree3" blocks (soft = 4000, hard = 6000) inodes (soft = 30, hard =

\$ nas\_quotas -edit -user -fs ufs1 -path /tree3 1000

~ ~ ~ ~ "/tmp/EdP.aMdtIQR" 2L, 100C written

# **EXAMPLE #21**

To change the block limit and inode limit on quota tree, /tree3, on ufs1 for the specified user, 1000, without opening up a vi session, type:

\$ nas\_quotas -edit -user -fs ufs1 -path /tree3 -block 6000:4000 -inode 300:200 1000

done

**EXAMPLE #22** To view a report of user quotas on tree quota, /tree3, for ufs1, type:

\$ nas\_quotas -report -user -fs ufs1 -path /tree3
Report for user quotas on quota tree /tree3 on filesystem ufs1 mounted on /ufs1

User	ļ	Bytes Used (1K)				Files			
	•			'	'	'		Timeleft	
#1000  #32768	2992   9824	4000	6000		34	200	300		
done		++							

**EXAMPLE #23** To open a vi session and edit the quota configuration for tree quota, /tree3, on a filesystem, ufs1, type:

```
$ nas_quotas -edit -config -fs ufs1 -path /tree3
Tree Quota Parameters:
fs "ufs1"
tree "/tree3"
        Block Grace: (1.0 weeks)
        Inode Grace: (1.0 weeks)
        * Default Quota Limits:
           User: block (soft = 8000, hard = 9000) inodes (soft = 200, hard= 300)
           Group: block (soft = 8000, hard = 9000) inodes (soft = 300, hard= 400)
        Deny disk space to users exceeding quotas: (yes)
        * Generate Events when:
            Quota check starts:
                                          (no)
            Quota check ends:
                                         (no)
            soft quota crossed:
                                         (yes)
            hard quota crossed:
                                         (yes)
"/tmp/EdP.aDTOKeU" 14L, 508C written
done
```

**EXAMPLE #24** To view the quota configuration for tree quota, /tree3, on filesystem, ufs1, type:

```
$ nas quotas -report -config -fs ufs1 -path /tree3
```

```
-----+
Quota parameters for tree quota /tree3 on filesystem ufs1 mounted on /ufs1:
Quota Policy: blocks
User Quota: ON
Group Quota: ON
Block grace period: (1.0 weeks)
Inode grace period: (1.0 weeks)
Default USER quota limits:
   Block Soft: (
                    8000), Block Hard: (
                                          9000)
   Inode Soft: (
                    200), Inode Hard: (
                                          300)
Default GROUP quota limits:
   Block Soft: ( 8000), Block Hard: (
                                         9000)
   Inode Soft: (
                    300), Inode Hard: (
                                         400)
```

Deny Disk Space to users exceeding quotas: YES
Log an event when ...

Block hard limit reached/exceeded: YES
Block soft limit (warning level) crossed: YES
Quota check starts: NO
Quota Check ends: NO

done

**EXAMPLE #25** To turn user quota and group quota off on tree quota, /tree3, type:

\$ nas\_quotas -off -both -fs ufs1 -path /tree3

done

**EXAMPLE #26** To turn group quotas off for **ufs1**, type:

\$ nas quotas -off -group -fs ufs1

done

**EXAMPLE #27** To clear all tree quotas for **ufs1**, type:

\$ nas quotas -clear -tree -fs ufs1

done

**EXAMPLE #28** To clear quotas for users and groups of a Data Mover, type:

\$ nas\_quotas -clear -both -mover server\_2

done

# nas\_rdf

Facilitates communication between two Celerra Network Servers. Its primary use is to manage Celerra systems and define the relationships needed for disaster recovery in a Symmetrix Remote Data Facility (SRDF) environment. It is also used to establish communication between Celerra systems and define the relationships needed for filesystem replication by using Celerra Replicator or TimeFinder/FS.

### **SYNPOSIS**

```
nas_rdf
   -init
| -activate [-reverse][-nocheck]
| -restore [-nocheck]
   -check {-all|<test>,...}
```

### DESCRIPTION

nas\_rdf establishes and manages relationships for Control Stations
and Data Movers that physically reside in separate Celerra cabinets.

For SRDF, nas\_rdf initializes the Celerra Network Servers, activates a failover to a destination Celerra, or restores a source Celerra. For Dynamic SRDF, nas\_rdf activates a failover and reverses the storage system from a destination volume (R2) to a source volume (R1). Configuration details depend on the type of SRDF: active/passive or active/active' SRDF/S for synchronous replication with disaster recovery, or active/passive SRDF/A for extended-distance, or asynchronous replication with a point-in-time replica.

**Note:** The apostrophe in active/active' indicates that both sites have a source volume mirrored at the other site.

SRDF is supported only on a Celerra Network Server attached to a Symmetrix system. Also, this command must be run from a primary Control Station in slot 0; it will report an error if run from a Control Station in slot 1.

**Note:** This command must be executed from the /nas/sbin directory, unless otherwise directed. Log in with your administrative username and password, and execute this functionality from **root**.

## OPTIONS -init

Initializes a source or destination (target) Celerra Network Server for SRDF/S or SRDF/A.

### -activate [-reverse]

Initiates an SRDF failover from the source Celerra Network Server to the destination. The **-activate** option is executed on the destination Celerra Network Server at the discretion of the user. The **-activate** option sets each SRDF-protected volume on the source Celerra Network Server as read-only, and each mirrored volume on the destination Celerra Network Server is set as read-write. The SRDF standby Data Movers acquire the IP and MAC addresses, filesystems, and export tables of their source counterparts. The **-reverse** option reverses SRDF direction by converting R2 volumes at destination site to R1 and synchronizing the destination and source sites. The **-reverse** option adds SYMCLI swap and establishes operations on the storage system after the normal activate operation is performed. When the **-activate** option is executed, an automatic, internal SRDF health check is performed before activating a failover. The **-nocheck** option allows you to skip this health check.

#### -restore

Restores a source Celerra Network Server after a failover. The **-restore** option is initially executed on the destination Celerra Network Server. The data on each destination volume is copied to the corresponding volume on the source Celerra Network Server. On the destination Celerra Network Server, services on each SRDF standby Data Mover are stopped. (NFS clients connected to these Data Movers see a "server unavailable" message; CIFS client connections time out.) Each volume on the source Celerra Network Server is set as read-write, and each mirrored volume on the destination Celerra Network Server is set as read-only.

Finally, nas\_rdf -restore can be remotely executed on the source Celerra Network Server to restore the original configuration. Each primary Data Mover reacquires its IP and MAC addresses, filesystems, and export tables. When the -restore option is executed, an automatic, internal SRDF health check is performed before restoring source and destination Celerras. The -nocheck option allows you to skip this health check.

## -check { -all | <test>, ...}

Runs SRDF health checks on the Celerra Network Server. The **-check** option can be executed either by using the **-all** option or by specifying one or more of the following individual checks: SRDF standby Data

Mover configuration check (r1\_dm\_config, r2\_dm\_config), SRDF session state check (r1\_session, r2\_session), Device group configuration check (r1\_dev\_group, r2\_dev\_group), Data Mover mirrored device accessibility check (r1\_dev\_avail, r2\_dev\_avail), Symmetrix device state check (dev\_not\_normal), and SRDF restored state check (restored). In these checks, r1 represents the source side and r2 represents the destination side.

When the **-all** option is used, all the checks are performed automatically. If the **-check** option detects invalid configurations or state issues, it prints relevant warning messages with recommended actions so that the issues can be resolved before running the activate or restore options. You can use the **-check** option to perform health checks at any time.

**Note:** To run the **-check** option, you must log in to the Celerra Network Server either as **nasadmin** and then switch (su) to root, or as **rdfadmin** and then switch (su) to root.

## SEE ALSO

Using SRDF/S with Celerra for Disaster Recovery, Using SRDF/A with Celerra, and nas cel.

## **EXAMPLE #1**

To start the initialization process on a source Celerra Network Server in an active/passive SRDF/S configuration, as **root** user, type:

## # /nas/sbin/nas\_rdf -init

Discover local storage devices ...

Discovering storage (may take several minutes)
done

## **EXAMPLE #2**

To initiate an SRDF failover from the source Celerra Network Server to the destination, as a **root** user, type:

### # /nas/sbin/nas\_rdf -activate

```
Is remote site CELERRA completely shut down (power OFF)?

Do you wish to continue? [yes or no]: yes

Successfully pinged (Remotely) Symmetrix ID: 000187430809

Successfully pinged (Remotely) Symmetrix ID: 000190100559

Successfully pinged (Remotely) Symmetrix ID: 000190100582

Write Disable devices on SA at source (R1).......Done.

Suspend RDF links......Done.

Read/Write Enable devices on RA at target (R2)......Done.

Waiting for nbs clients to die ... done

Waiting for nbs clients to start ... done

fsck 1.35 (28-Feb-2004)

/dev/ndj1: recovering journal
/dev/ndj1: clean, 13780/231360 files, 233674/461860 blocks

Waiting for nbs clients to die ... done
```

```
Waiting for nbs clients to start ... done
id
       type acl slot groupID state name
1
       1
            1000 2
                               0
                                   server_2
2
        4
            1000 3
                               0
                                   server_3
3
        1
            1000 4
                               0
                                   server 4
4
        4
            1000 5
                               0
                                   server 5
server_2 :
server_2 : going offline
rdf : going active
replace in progress ...done
failover activity complete
commit in progress (not interruptible)...done
done
server 3 :
server_3 : going offline
rdf : going active
replace in progress ...done
failover activity complete
commit in progress (not interruptible)...done
done
server_4 :
Error 4003: server_4 : standby is not configured
server_5 :
Error 4003: server_5 : standby is not configured
   Suspend RDF links......Done.
   Merge device track tables between source and target.....Started.
   Merge device track tables between source and target......Done.
   Resume RDF links......Started.
   EXAMPLE #3
                    To initiate an SRDF failover from the source Celerra Network Server
                    to the destination, without the SRDF health check, as a root user,
                     # /nas/sbin/nas_rdf -activate -nocheck
Skipping SRDF health check ....
Is remote site CELERRA completely shut down (power OFF)?
Do you wish to continue? [yes or no]: yes
Successfully pinged (Remotely) Symmetrix ID: 000187430809
Successfully pinged (Remotely) Symmetrix ID: 000190100559
Successfully pinged (Remotely) Symmetrix ID: 000190100582
Write Disable devices on SA at source (R1).....Done.
Suspend RDF links......Done.
Read/Write Enable devices on RA at target (R2)......Done.
Waiting for nbs clients to die ... done
Waiting for nbs clients to start ... done
fsck 1.35 (28-Feb-2004)
```

```
/dev/ndj1: recovering journal
/dev/ndj1: clean, 13780/231360 files, 233674/461860 blocks
Waiting for nbs clients to die ... done
Waiting for nbs clients to start ... done
id type acl slot groupID state name
1 1 1000 2 0 server 2
2 4 1000 3 0 server_3
3 1 1000 4 0 server_4
4 4 1000 5 0 server 5
server_2 :
server_2 : going offline
rdf : going active
replace in progress ...done
failover activity complete
commit in progress (not interruptible)...done
done
server_3 :
server_3 : going offline
rdf : going active
replace in progress ...done
failover activity complete
commit in progress (not interruptible)...done
done
server 4:
Error 4003: server_4 : standby is not configured
server_5 :
Error 4003: server_5 : standby is not configured
Suspend RDF links......Done.
Merge device track tables between source and target......Started.
Device: 045A in (0557,005)...... Merged.
Merge device track tables between source and target.....Done.
Resume RDF links......Started.
```

# **EXAMPLE #4** To initiate a Dynamic SRDF failover from the source Celerra Network Server to the destination, as a **root** user, type:

### # /nas/sbin/nas\_rdf -activate -reverse

```
Is remote site CELERRA completely shut down (power OFF)?
Do you wish to continue? [yes or no]: yes
Successfully pinged (Remotely) Symmetrix ID: 000280600118
   Write Disable devices on SA at source (R1).....Done.
   Read/Write Enable devices on RA at target (R2)......Done.
fsck 1.35 (28-Feb-2004)
/dev/sdj1: recovering journal
Clearing orphaned inode 37188 (uid=0, gid=0, mode=0100644, size=0)
/dev/sdj1: clean, 12860/219968 files, 194793/439797 blocks
       type acl slot groupID state name
id
1
       1
            1000 2
                              0
                                  server_2
       4
            1000 3
                              0
                                   server 3
```

```
4
          2000 4
                         0
                             server_4
          2000 5
                             server_5
server_2 :
server_2 : going offline
rdf : going active
replace in progress ...done
failover activity complete
commit in progress (not interruptible)...done
done
server_3 :
server 3 : going offline
rdf : going active
replace in progress ...done
failover activity complete
commit in progress (not interruptible)...done
An RDF 'Swap Personality' operation execution is
in progress for device group '1R2_500_1'. Please wait...
   Swap RDF Personality......Started.
   Swap RDF Personality......Done.
The RDF 'Swap Personality' operation successfully executed for
device group '1R2_500_1'.
An RDF 'Incremental Establish' operation execution is
in progress for device group '1R2_500_1'. Please wait...
   Suspend RDF links.....Done.
   Resume RDF links......Started.
   Merge device track tables between source and target......Started.
   Devices: 0009-000B ..... Merged.
   Devices: 0032-0034 ..... Merged.
   Devices: 0035-0037 ..... Merged.
   Devices: 0038-003A ..... Merged.
   Devices: 003B-003D ..... Merged.
   Devices: 003E-0040 ..... Merged.
   Devices: 0041-0043 ..... Merged.
   Devices: 0044-0046 ..... Merged.
  Devices: 0047-0049 ..... Merged.
  Merge device track tables between source and target......Done.
   Resume RDF links......Done.
The RDF 'Incremental Establish' operation successfully initiated for
device group '1R2_500_1'.
```

# **EXAMPLE #5** To restore a source Celerra Network Server after failover, as a **root** user, type:

## # /nas/sbin/nas\_rdf -restore

Is remote site CELERRA ready for Storage restoration? Do you wish to continue? [yes or no]: yes Contact Joker\_R1\_CSO ... is alive

Restore will now reboot the source site control station. Do you wish to continue? [yes or no]: yes

Device Group (DG) Name : 1R2\_500\_5

DG's Type : RDF2

DG's Symmetrix ID : 000190100557

Target (R2) View							Sour	ce (R1)	View	MODES	
		ST			LI		ST				
Standar	đ	A			N		A				
Logical		Т	R1 Inv	R2 Inv	K		Т	R1 Inv	R2 Inv		RDF Pair
Device		E	Tracks	Tracks	S	Dev	E	Tracks	Tracks		STATE
DEV001	045A		10	0		045A	WD	0		S	R1 Updated
DEV002	045B		2054	0		045B	WD	0	0	S	Failed Over
DEV003	045C		0	0		045C	WD	0	0	~	Failed Over
DEV004	045D	RW	0	0		045D	WD	0	0	S	Failed Over
DEV005	045E	RW	1284	0	NR	045E	WD	0	0	S	Failed Over
DEV006	045F	RW	0	0	NR	045F	WD	0	0	S	Failed Over
DEV007	0467	RW	0	0	NR	0467	WD	0	0	S	Failed Over
DEV008	0468	RW	2	0	NR	0468	WD	0	0	S	Failed Over
DEV009	0469	RW	0	0	NR	0469	WD	0	0	S	Failed Over
DEV010	046A	RW	0	0	NR	046A	WD	0	0	S	Failed Over
DEV011	046B	RW	2	0		046B	WD	0	0	~	Failed Over
DEV012	046C	RW	0	0	NR	046C	WD	0	0	S	Failed Over
DEV013	046D	RW	0	0	NR	046D	WD	0	0	S	Failed Over
DEV014	046E	RW	0	0	NR	046E	WD	0	0	S	Failed Over
DEV015	046F	RW	2	0	NR	046F	WD	0	0	S	Failed Over
DEV016	0470	RW	0	0	NR	0470	WD	0	0	S	Failed Over
DEV017	0471	RW	2	0	NR	0471	WD	0	0	S	Failed Over
DEV018	0472	RW	0	0	NR	0472	WD	0	0	S	Failed Over
DEV019	0473	RW	0	0	NR	0473	WD	0	0	S	Failed Over
DEV020	0474	RW	0	0	NR	0474	WD	0	0	S	Failed Over
DEV021	0475	RW	0	0	NR	0475	WD	0	0	S	Failed Over
DEV022	0476	RW	0	0	NR	0476	WD	0	0	S	Failed Over
DEV023	0477	RW	2	0	NR	0477	WD	0	0	S	Failed Over
DEV024	0478	RW	2	0	NR	0478	WD	0	0	S	Failed Over
DEV025	0479	RW	0	0	NR	0479	WD	0	0	S	Failed Over
DEV026	047A	RW	0	0	NR	047A	WD	0	0	S	Failed Over
DEV027	047B	RW	0	0	NR	047B	WD	0	0	S	Failed Over
DEV028	047C	RW	0	0	NR	047C	WD	0	0	S	Failed Over
DEV029	047D	${\sf RW}$	0	0	NR	047D	WD	0	0	S	Failed Over

```
0
DEV030 047E RW
                           0 NR 047E WD
                                            0
                                                   0 S..
                                                          Failed Over
DEV031
      047F RW
                  Ο
                           0 NR 047F WD
                                            0
                                                   0 S..
                                                          Failed Over
      0480 RW
                  0
                           0 NR 0480 WD
                                            0
                                                   0 S.,
                                                          Failed Over
DEV032
DEV033
      0481 RW
                  0
                          0 NR 0481 WD
                                            0
                                                   0 S..
                                                          Failed Over
DEV034
      0482 RW
                  0
                          0 NR 0482 WD
                                            Ω
                                                   0 S..
                                                          Failed Over
                  0
DEV035
      0483 RW
                          0 NR 0483 WD
                                            0
                                                   0 S..
                                                          Failed Over
DEV036
      0484 RW
                  0
                          0 NR 0484 WD
                                            0
                                                  0 S..
                                                          Failed Over
                  0
                          0 NR 0485 WD
                                                   0 S..
DEV037
      0485 RW
                                            0
                                                           Failed Over
DEV038
      0486 RW
                  0
                          0 NR 0486 WD
                                            0
                                                   0 S..
                                                          Failed Over
DEV039
      0487 RW
                  0
                          0 NR 0487 WD
                                            0
                                                   0 S..
                                                          Failed Over
DEV040
      0488 RW
                  0
                          0 NR 0488 WD
                                            0
                                                   0 S..
                                                          Failed Over
                  0
                                            0
      0489 RW
                          0 NR 0489 WD
                                                   0 S..
                                                          Failed Over
DEV041
DEV042
      048A RW
                  0
                          0 NR 048A WD
                                            0
                                                  0 S..
                                                          Failed Over
                                                  0 S..
DEV043
      048B RW
                  0
                          0 NR 048B WD
                                            0
                                                          Failed Over
DEV044
      048C RW
                  0
                          0 NR 048C WD
                                            0
                                                  0 S..
                                                          Failed Over
DEV045
      048D RW
                  Ω
                          0 NR 048D WD
                                            0
                                                   0 S..
                                                          Failed Over
                          0 NR 048E WD
                  0
                                            0
                                                   0 S..
                                                          Failed Over
DEV046
      048E RW
      048F RW
                  2
                          0 NR 048F WD
                                           0
                                                  0 S..
                                                          Failed Over
DEV047
      0490 RW
DEV048
                  0
                          0 NR 0490 WD
                                           0
                                                  0 S..
                                                          Failed Over
                  0
                                                  0 S..
DEV049
      0491 RW
                          0 NR 0491 WD
                                            0
                                                          Failed Over
                                                 0 S..
0 S..
DEV050
     0492 RW
                  0
                          0 NR 0492 WD
                                            0
                                                          Failed Over
DEV051
      0493 RW
                  0
                          0 NR 0493 WD
                                            0
                                                  0 S..
                                                          Failed Over
                  0
                                                  0 S..
      0494 RW
                          0 NR 0494 WD
                                            0
                                                          Failed Over
DEV052
                  0
                         0 NR 0495 WD
                                            0
                                                          Failed Over
DEV053
      0495 RW
                                                  0 S..
DEV054
      0496 RW
                  0
                          0 NR 0496 WD
                                           0
                                                  0 S.. Failed Over
                         0 NR 0497 WD
      0497 RW
                  2
                                                  0 S..
                                                          Failed Over
DEV055
                                           0
                                                0 S..
0 S..
0 S..
                  2
                          0 NR 0498 WD
DEV056
      0498 RW
                                            0
                                                          Failed Over
DEV057
      0499 RW
                  0
                          0 NR 0499 WD
                                            0
                                                          Failed Over
      049A RW
                  0
                          0 NR 049A WD
                                            0
                                                  0 S..
                                                          Failed Over
DEV058
DEV059
      049B RW
                  0
                         0 NR 049B WD
                                            0
                                                  0 S..
                                                          Failed Over
                  0
                                           0
DEV060
      049C RW
                         0 NR 049C WD
                                                  0 S..
                                                         Failed Over
DEV061
      049D RW
                  0
                          0 NR 049D WD
                                           0
                                                  0 S..
                                                          Failed Over
                                                0 S..
0 S..
                  0
                          0 NR 049E WD
DEV062
      049E RW
                                            0
                                                          Failed Over
                  0
DEV063
     049F RW
                         0 NR 049F WD
                                            0
                                                          Failed Over
                  0
                          0 NR 04A0 WD
                                            0
                                                  0 S..
DEV064
      04A0 RW
                                                          Failed Over
                  0
                          0 NR 04A1 WD
                                            0
DEV065
      04A1 RW
                                                   0 S..
                                                          Failed Over
                  0
                         0 NR 04A2 WD
DEV066
      04A2 RW
                                           0
                                                  0 S.. Failed Over
      04A3 RW
                  0
                         0 NR 04A3 WD
                                           0
                                                  0 S.. Failed Over
DEV067
                  0
                         0 NR 04A4 WD
                                                  0 S..
                                                          Failed Over
                                           0
DEV068
      04A4 RW
      04A5 RW
DEV069
                  Ο
                          0 NR 04A5 WD
                                            0
                                                   0 S.. Failed Over
DEV070 04A6 RW
                   0
                       0 NR 04A6 WD
                                            0
                                                  0 S.. Failed Over
Total
 Tracks
                 3366
                          0
                                            Ω
                                                    0
 MBs
                105.2
                         0.0
                                           0.0
                                                   0.0
Legend for MODES:
M(ode of Operation): A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
D(omino) : X = Enabled, . = Disabled
A(daptive Copy)
                : D = Disk Mode, W = WP Mode, . = ACp off
   Suspend RDF link(s)......Done.
   Merge device track tables between source and target......Started.
   Devices: 045A-045F, 0467-0477 in (0557,005)..... Merged.
   Devices: 0478-0489 in (0557,005)...... Merged.
```

```
Devices: 048A-049B in (0557,005)..... Merged.
   Devices: 049C-04A6 in (0557,005)..... Merged.
   Merge device track tables between source and target......Done.
   Resume RDF link(s)......Started.
   Is remote site CELERRA ready for Network restoration?
Do you wish to continue? [yes or no]: yes
server_2 : done
server 3 : done
server_4 :
Error 4003: server_4 : standby is not configured
Error 4003: server_5 : standby is not configured
fsck 1.35 (28-Feb-2004)
/dev/ndj1: clean, 13836/231360 files, 233729/461860 blocks
Waiting for nbs clients to die ... done
Waiting for nbs clients to start ... done
Waiting for nbs clients to die ... done
Waiting for nbs clients to start ... done
Waiting for 1R2_500_5 access ...done
   Write Disable device(s) on RA at target (R2)......Done.
   Merge device track tables between source and target......Started.
   Devices: 045A-045F, 0467-0477 in (0557,005)..... Merged.
   Devices: 048A-049B in (0557,005)...... Merged.
   Devices: 049C-04A6 in (0557,005)..... Merged.
   Merge device track tables between source and target......Done.
   Resume RDF link(s)......Started.
   Resume RDF link(s)......Done.
   Read/Write Enable device(s) on SA at source (R1)......Done.
Waiting for 1R2_500_5 sync ...done
Starting restore on remote site CELERRA ...
Waiting for nbs clients to start ... done
Waiting for nbs clients to start ... done
   Suspend RDF link(s)......Done.
server_2 :
server_2 : going standby
rdf : going active
replace in progress ...done
failover activity complete
commit in progress (not interruptible)...done
done
server_3 :
server_3 : going standby
rdf : going active
replace in progress ...done
failover activity complete
commit in progress (not interruptible)...done
done
server 4:
Error 4003: server_4 : standby is not configured
```

# **EXAMPLE #6** To restore a source Celerra Network Server after failover, without the SRDF health check, as a **root** user, type:

### # /nas/sbin/nas\_rdf -restore -nocheck

```
Skipping SRDF health check ....
Is remote site CELERRA ready for Storage restoration?
Do you wish to continue? [yes or no]: yes
Contact Joker_R1_CS0 ... is alive
Restore will now reboot the source site control station.
Do you wish to continue? [yes or no]: yes
Device Group (DG) Name: 1R2_500_5
DG's Type : RDF2
DG's Symmetrix ID: 000190100557
Target (R2) View Source (R1) View MODES
ST LI ST
Standard A N A
Logical T R1 Inv R2 Inv K T R1 Inv R2 Inv RDF Pair
Device Dev E Tracks Tracks S Dev E Tracks Tracks MDA STATE
DEV001 045A RW 10 0 RW 045A WD 0 0 S.. R1 Updated
DEV002 045B RW 2054 0 NR 045B WD 0 0 S.. Failed Over
DEV003 045C RW 0 0 NR 045C WD 0 0 S.. Failed Over
DEV004 045D RW 0 0 NR 045D WD 0 0 S.. Failed Over
DEV005 045E RW 1284 0 NR 045E WD 0 0 S.. Failed Over
DEV006 045F RW 0 0 NR 045F WD 0 0 S.. Failed Over
DEV007 0467 RW 0 0 NR 0467 WD 0 0 S.. Failed Over
DEV008 0468 RW 2 0 NR 0468 WD 0 0 S.. Failed Over
DEV009 0469 RW 0 0 NR 0469 WD 0 0 S.. Failed Over
DEV010 046A RW 0 0 NR 046A WD 0 0 S.. Failed Over
DEV011 046B RW 2 0 NR 046B WD 0 0 S.. Failed Over
DEV012 046C RW 0 0 NR 046C WD 0 0 S.. Failed Over
DEV013 046D RW 0 0 NR 046D WD 0 0 S.. Failed Over
DEV014 046E RW 0 0 NR 046E WD 0 0 S.. Failed Over
DEV015 046F RW 2 0 NR 046F WD 0 0 S.. Failed Over
DEV016 0470 RW 0 0 NR 0470 WD 0 0 S.. Failed Over
DEV017 0471 RW 2 0 NR 0471 WD 0 0 S.. Failed Over
DEV018 0472 RW 0 0 NR 0472 WD 0 0 S.. Failed Over
DEV019 0473 RW 0 0 NR 0473 WD 0 0 S.. Failed Over
DEV020 0474 RW 0 0 NR 0474 WD 0 0 S.. Failed Over
DEV021 0475 RW 0 0 NR 0475 WD 0 0 S.. Failed Over
```

DEV022 0476 RW 0 0 NR 0476 WD 0 0 S.. Failed Over

```
DEV023 0477 RW 2 0 NR 0477 WD 0 0 S.. Failed Over
DEV024 0478 RW 2 0 NR 0478 WD 0 0 S.. Failed Over
DEV025 0479 RW 0 0 NR 0479 WD 0 0 S.. Failed Over
DEV026 047A RW 0 0 NR 047A WD 0 0 S.. Failed Over
DEV027 047B RW 0 0 NR 047B WD 0 0 S.. Failed Over
DEV028 047C RW 0 0 NR 047C WD 0 0 S.. Failed Over
DEV029 047D RW 0 0 NR 047D WD 0 0 S.. Failed Over
DEV030 047E RW 0 0 NR 047E WD 0 0 S.. Failed Over
DEV031 047F RW 0 0 NR 047F WD 0 0 S.. Failed Over
DEV032 0480 RW 0 0 NR 0480 WD 0 0 S.. Failed Over
DEV033 0481 RW 0 0 NR 0481 WD 0 0 S.. Failed Over
DEV034 0482 RW 0 0 NR 0482 WD 0 0 S.. Failed Over
DEV035 0483 RW 0 0 NR 0483 WD 0 0 S.. Failed Over
DEV036 0484 RW 0 0 NR 0484 WD 0 0 S.. Failed Over
DEV037 0485 RW 0 0 NR 0485 WD 0 0 S.. Failed Over
DEV038 0486 RW 0 0 NR 0486 WD 0 0 S.. Failed Over
DEV039 0487 RW 0 0 NR 0487 WD 0 0 S.. Failed Over
DEV040 0488 RW 0 0 NR 0488 WD 0 0 S.. Failed Over
DEV041 0489 RW 0 0 NR 0489 WD 0 0 S.. Failed Over
DEV042 048A RW 0 0 NR 048A WD 0 0 S.. Failed Over
DEV043 048B RW 0 0 NR 048B WD 0 0 S.. Failed Over
DEV044 048C RW 0 0 NR 048C WD 0 0 S.. Failed Over
DEV045 048D RW 0 0 NR 048D WD 0 0 S.. Failed Over
DEV046 048E RW 0 0 NR 048E WD 0 0 S.. Failed Over
DEV047 048F RW 2 0 NR 048F WD 0 0 S.. Failed Over
DEV048 0490 RW 0 0 NR 0490 WD 0 0 S.. Failed Over
DEV049 0491 RW 0 0 NR 0491 WD 0 0 S.. Failed Over
DEV050 0492 RW 0 0 NR 0492 WD 0 0 S.. Failed Over
DEV051 0493 RW 0 0 NR 0493 WD 0 0 S.. Failed Over
DEV052 0494 RW 0 0 NR 0494 WD 0 0 S.. Failed Over
DEV053 0495 RW 0 0 NR 0495 WD 0 0 S.. Failed Over
DEV054 0496 RW 0 0 NR 0496 WD 0 0 S.. Failed Over
DEV055 0497 RW 2 0 NR 0497 WD 0 0 S.. Failed Over
DEV056 0498 RW 2 0 NR 0498 WD 0 0 S.. Failed Over
DEV057 0499 RW 0 0 NR 0499 WD 0 0 S.. Failed Over
DEV058 049A RW 0 0 NR 049A WD 0 0 S.. Failed Over
DEV059 049B RW 0 0 NR 049B WD 0 0 S.. Failed Over
DEV060 049C RW 0 0 NR 049C WD 0 0 S.. Failed Over
DEV061 049D RW 0 0 NR 049D WD 0 0 S.. Failed Over
DEV062 049E RW 0 0 NR 049E WD 0 0 S.. Failed Over
DEV063 049F RW 0 0 NR 049F WD 0 0 S.. Failed Over
DEV064 04A0 RW 0 0 NR 04A0 WD 0 0 S.. Failed Over
DEV065 04A1 RW 0 0 NR 04A1 WD 0 0 S.. Failed Over
DEV066 04A2 RW 0 0 NR 04A2 WD 0 0 S.. Failed Over
DEV067 04A3 RW 0 0 NR 04A3 WD 0 0 S.. Failed Over
DEV068 04A4 RW 0 0 NR 04A4 WD 0 0 S.. Failed Over
DEV069 04A5 RW 0 0 NR 04A5 WD 0 0 S.. Failed Over
DEV070 04A6 RW 0 0 NR 04A6 WD 0 0 S.. Failed Over
Total ----- -----
Track(s) 3366 0 0 0he nas Commands
MB(s) 105.2 0.0 0.0 0.0
Legend for MODES:
```

```
M(ode of Operation): A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
D(omino) : X = Enabled, . = Disabled
A(daptive Copy) : D = Disk Mode, W = WP Mode, . = ACp off
Suspend RDF link(s)......Done.
Merge device track tables between source and target......Started.
Devices: 045A-045F, 0467-0477 in (0557,005)..... Merged.
Devices: 0478-0489 in (0557,005)..... Merged.
Devices: 049C-04A6 in (0557,005)..... Merged.
Merge device track tables between source and target......Done.
Resume RDF link(s)......Started.
Resume RDF link(s)......Done.
Is remote site CELERRA ready for Network restoration?
Do you wish to continue? [yes or no]: yes
server_2 : done
server_3 : done
server_4 :
Error 4003: server_4 : standby is not configured
server 5 :
Error 4003: server_5 : standby is not configured
fsck 1.35 (28-Feb-2004)
/dev/ndj1: clean, 13836/231360 files, 233729/461860 blocks
Waiting for nbs clients to die ... done
Waiting for nbs clients to start ... done
Waiting for nbs clients to die ... done
Waiting for nbs clients to start ... done
Waiting for 1R2_500_5 access ...done
Write Disable device(s) on RA at target (R2)......Done.
Merge device track tables between source and target......Started.
Devices: 045A-045F, 0467-0477 in (0557,005)..... Merged.
Merge device track tables between source and target......Done.
Resume RDF link(s)......Started.
Resume RDF link(s)......Done.
Read/Write Enable device(s) on SA at source (R1).....Done.
Waiting for 1R2_500_5 sync ...done
Starting restore on remote site CELERRA ...
Waiting for nbs clients to start ... done
Waiting for nbs clients to start ... done
Suspend RDF link(s)......Done.
server_2 :
server_2 : going standby
rdf : going active
replace in progress ...done
failover activity complete
commit in progress (not interruptible)...done
doneThe nas Commands
server 3 :
server_3 : going standby
```

# **EXAMPLE #7** To restore a source Celerra Network Server after failover, when using Dynamic SRDF, as a **root** user, type:

### # /nas/sbin/nas\_rdf -restore

Is remote site CELERRA ready for Storage restoration? Do you wish to continue? [yes or no]: yes Contact eng17335 ... is alive

Restore will now reboot the source site control station. Do you wish to continue? [yes or no]: yes

Device Group (DG) Name : 1R2\_500\_1 DG's Type : RDF1

DG's Symmetrix ID : 000280600187 (Microcode Version: 5568) Remote Symmetrix ID : 000280600118 (Microcode Version: 5568)

RDF (RA) Group Number : 1 (00)

Source (	(R1) View		Ta	rget (R2)	View MC	DDES
	ST	L	I S	ST		
Standard	A	I	N	A		
Logical	T R1 Inv	R2 Inv K		T R1 Inv	R2 Inv	RDF Pair
Device Dev	E Tracks	Tracks S	Dev	E Tracks	Tracks N	MDA STATE
		0			0	
DEV001 0056			0030 1			S Synchronized
DEV002 0057	RW 0	0 RW	0031 1	WD 0	0 5	S Synchronized
DEV003 0032	RW 0	0 RW	000C 1	WD 0	0 9	S Synchronized
BCV008 0069	RW 0	0 RW	005F T	WD 0	0 5	S Synchronized
BCV009 006A	RW 0	0 RW	0060 7	WD 0	0 9	S Synchronized
BCV010 006B	RW 0	0 RW	0061 7	WD 0	0 5	S Synchronized
Total						
Track(s)	0	0		0	0	

```
MB(s)
               0.0
                     0.0
                                   0.0
                                         0.0
Legend for MODES: M(ode of Operation): A = Async, S = Sync, E = Semi-sync, C =
Adaptive Copy
D(omino)
                : X = Enabled, . = Disabled
A(daptive Copy)
               : D = Disk Mode, W = WP Mode, . = ACp off
Is remote site CELERRA ready for Network restoration?
Do you wish to continue? [yes or no]: yes
server 2 : done
server_3 : done
server_4 :
Error 4003: server 4: standby is not configured
server_5 :
Error 4003: server_5 : standby is not configured
fsck 1.35 (28-Feb-2004)
/dev/sdj1: clean, 12956/219968 files, 188765/439797 blocks
An RDF 'Failover' operation execution is
in progress for device group '1R2_500_1'. Please wait...
   Write Disable device(s) on SA at source (R1)......Done.
   Swap RDF Personality......Started.
   Suspend RDF link(s)......Done.
   Read/Write Enable device(s) on SA at source (R1).....Done.
   Resume RDF link(s)......Done.
   Read/Write Enable device(s) on SA at target (R2).....Done.
The RDF 'Failover' operation successfully executed for
device group '1R2_500_1'.
Waiting for 1R2_500_1 sync ...done
Starting restore on remote site CELERRA ...
Suspend RDF link(s)......Done.
server 2 :
server_2 : going standby
rdf : going active
replace in progress ...done
failover activity complete
commit in progress (not interruptible)...done
done
server_3 :
server_3 : going standby
rdf : going active
replace in progress ...done
failover activity complete
commit in progress (not interruptible)...done
done
```

```
server_4 :
Error 4003: server_4: standby is not configured
server_5 :
Error 4003: server_5: standby is not configured
done
```

# **EXAMPLE #8** To run all available checks on a source Celerra Network Server, as a **root** user, type:

## # /nas/sbin/nas\_rdf -check -all

			SRDF Health Checks	
SRDF:	Checking	dev	vice is normal	Pass
SRDF:	Checking	sys	stem is restored	Pass
SRDF:	Checking	R1	SRDF session is Synch or Consistent	Pass
SRDF:	Checking	R1	Data Mover configuration is valid	Pass
SRDF:	Checking	R1	devices are available	Pass
SRDF:	Checking	R1	device group has all devices	Pass
SRDF:	Checking	R2	SRDF session is Synch or Consistent	Pass
SRDF:	Checking	R2	Data Mover configuration is valid	Pass
SRDF:	Checking	R2	devices are available	Pass
SRDF:	Checking	R2	device group has all devices	Pass

# **EXAMPLE #9** To run one or more specific available checks on a source Celerra Network Server, as a **root** user, type:

## # /nas/sbin/nas\_rdf -check r1\_dev\_group,r2\_dev\_group

----- SRDF Health Checks ------- SRDF: Checking R1 device group has all devices...... Pass SRDF: Checking R2 device group has all devices...... Pass

# nas\_replicate

Manages loopback, local, and remote Celerra Replicator V2 sessions.

## **SYNOPSIS**

```
nas_replicate
  -list [-id]
-info {-all|id=<sessionId>|<name>}
  -create <name>
   -source -fs {<fsName>|id=<fsId>}
    [-sav {<srcSavVolStoragePool>|id=<srcSavVolStoragePoolId>}]
   -destination {-fs {id=<dstFsId>|<existing_dstFsName>}
      -pool {id=<dstStoragePoolId> | <dstStoragePool>} [-vdm <dstVdmName>]}
    [-sav {id=<dstSavVolStoragePoolId> | <dstSavVolStoragePool> }]
   -interconnect {<name> | id=<interConnectId>}
    [-source_interface {ip=<ipAddr> | <nameServiceInterfaceName>}]
    [-destination interface {ip=<ipAddr> | <nameServiceInterfaceName>}]
    [{-max_time_out_of_sync <maxTimeOutOfSync>|-manual_refresh}]
    [-overwrite_destination] [-tape_copy] [-background]
 -create <name>
   -source -vdm <vdmName>
   -destination {-vdm <existing_dstVdmName>|-pool
    {id=<dstStoragePoolId>|<dstStoragePool>}}
   -interconnect {<name> | id=<interConnectId>}
    [-source_interface {ip=<ipAddr> | <nameServiceInterfaceName>}]
    [-destination_interface {ip=<ipAddr>|<nameServiceInterfaceName>}]
    [{-max_time_out_of_sync <maxTimeOutOfSync>|-manual_refresh}]
    [-overwrite_destination][-background]
-create <name>
   -source -lun <lunNumber> -target <targetIqn>
   -destination -lun <lunNumber> -target <targetIgn>
   -interconnect {<name>|id=<interConnectId>}
    [-source_interface {ip=<ipAddr> | <nameServiceInterfaceName>}]
    [-destination_interface {ip=<ipAddr>|<nameServiceInterfaceName>}]
    [{-max time out of sync <maxTimeOutOfSync>|-manual_refresh}]
    [-overwrite_destination][-background]
-start {<name>|id=<sessionId>}
    [-interconnect {<name>|id=<interConnectId>}]
    [-source_interface {ip=<ipAddr> | <nameServiceInterfaceName>}]
    [-destination interface {ip=<ipAddr> | <nameServiceInterfaceName>}]
    [{-max_time_out_of_sync <maxTimeOutOfSync>|-manual_refresh}]
    [-overwrite_destination][-reverse][-full_copy][-background]
 -modify {<name> | id=<sessionId>} [-name <new name>]
    [-source_interface {ip=<ipAddr> | <nameServiceInterfaceName>}]
    [-destination interface {ip=<ipAddr> | <nameServiceInterfaceName>}]
    [{-max time out of sync <maxTimeOutOfSync> | -manual refresh}]
```

**Note**: This command manages replication sessions using Celerra Replicator V2. For a one-time filesystem copy using Celerra Replicator V2, use the nas\_copy command. For ongoing filesystem, use the fs\_ckpt command.

## **DESCRIPTION**

nas\_replicate creates, manages, or displays session information for ongoing Celerra Replicator V2 replication of a filesystem, Virtual Data Mover (VDM), or iSCSI LUN at a destination using an existing Data Mover interconnect. Each session handles a single source object and destination, and is assigned a globally unique ID, fixed for the life of the session.

In response to a potential disaster scenario, use **nas\_replicate** to perform a failover of a specified replication session with possible data loss. The **-switchover** option switches over a replication relationship and performs synchronization of the source and destination without data loss. Use **nas\_replicate** to also reverse the direction of a replication session or refresh the destination side with updates to the source based on a time-out of synchronization value or on demand.

## **OPTIONS**

#### -list [-id]

Displays all configured (or stopped) replication sessions (V2 filesystem, VDM, iSCSI LUN, or copy sessions) on each Data Mover in the Celerra cabinet. Each session is represented by either a name or a session ID that is generated automatically whenever a session is configured and is globally unique.

Use this option to obtain the session ID needed for another command. Since session IDs are lengthy, the session ID obtained from this command can be copied and pasted into the command.

```
-info {-all|id=<sessionId>|<name>}
```

Displays the status of a specific configured (or stopped) replication session (V2 filesystem, VDM, iSCSI LUN, or copy session, or the status of all replication sessions).

# CREATING FILE SYSTEM REPLICATION

#### -create <name>

Assigns a name to the filesystem replication session. The name must be unique for each Data Mover pair, which is defined by the interconnect.

```
-source -fs {<fsName>|id=<fsId>} [-sav
<srcSavVolStoragePoolId>|id=<srcSavVolStoragePoolId>}
]
```

Specifies the name or ID of the existing source filesystem to replicate. The source filesystem must be mounted as read-only or read/write.

**Note:** If the source filesystem is mounted to a VDM and the goal is to replicate a CIFS environment for disaster recovery (that is, replicate a VDM and the filesystems mounted to the VDM), create a session to replicate the VDM first, before replicating a filesystem mounted to the VDM.

The **-sav** option allocates a storage pool for all subsequent checkpoints for the filesystem. By default, if checkpoint storage (the checkpoint SavVol) needs to be allocated for checkpoints of the filesystem, the command uses the same storage pool used to create the source filesystem.

```
-destination {-fs {<existing_dstFsName>|
id=<dstFsId>| -pool <dstStoragePool>|
id=<dstStoragePoolId>[-vdm <dstVdmName>]}[-sav
{<dstSavVolStoragePool>|id=<dstSavVolStoragePoolId>}]
Specifies an existing destination filesystem or the storage needed to create the destination filesystem. An existing destination filesystem must be mounted as read-only and the same size as the source. Specifying a storage pool or ID creates the read-only, destination filesystem automatically, using the same name and size as the source filesystem.
```

Specifying a pool with the **-vdm** option mounts the destination filesystem to an existing VDM as part of replication in a CIFS environment.

The **-sav** option allocates a storage pool for all subsequent checkpoints of the destination filesystem. By default, if destination checkpoint storage needs to be allocated for checkpoints, the command uses the same storage pool used to create the destination filesystem.

By default, the destination filesystem name will be the same as the source filesystem name. If a filesystem with the same name as the source filesystem already exists on the destination, the naming convention <source\_fs\_name>\_replica<#> will be used. A number 1–4 is assigned according to how many replicas of that filesystem already exist.

-interconnect {<name>|id=<interConnectId>}
Specifies the local (source) side of an established Data Mover
interconnect to use for this replication session.

Use the **nas\_cel -interconnect -list** command on the source Celerra to list the interconnects available to the replication sessions.

#### [-source interface

{<nameServiceInterfaceName>|ip=<ipAddr>}]

Instructs the replication session to use a specific local interface defined for the interconnect on the source Celerra instead of selecting the local interface supporting the lowest number of sessions (the default). If this local interface was defined for the interconnect using a name service interface name, specify the name service interface name; if it was defined using an IP address, specify the IP address. If you define an interface using an IP address, make sure that the destination interface uses the same IPv4/IPv6 protocol. An IPv4 interface cannot connect to an IPv6 interface and vice versa. Both sides of the connection must use the same protocol.

The **source\_interfaces** field of the output from the **nas\_cel -interconnect -info** command shows how the source interface was defined. This option does not apply to a loopback interconnect, which always uses 127.0.0.1.

If no source interface is specified, the system will select an interface. This ensures that the interface selected can communicate with the destination interface.

[-destination\_interface{<nameServiceInterfaceName>|ip
=<ipaddr>}]

Instructs the replication session to use a specific peer interface

defined for the interconnect on the destination Celerra instead of selecting the peer interface supporting the lowest number of sessions (the default). If this peer interface was defined for the interconnect using a name service interface name, specify the name service interface name; if it was defined using an IP address, specify the IP address. If you define an interface using an IP address, make sure that the source interface uses the same IPv4/IPv6 protocol. An IPv4 interface cannot connect to an IPv6 interface and vice versa. Both sides of the connection must use the same protocol.

The **destination\_interfaces** field of the output from the **nas\_cel -interconnect -info** command shows how the peer interface was defined. This option does not apply to a loopback interconnect, which always uses 127.0.0.1.

If no destination interface is specified, the system will select an interface. This ensures that the interface selected can communicate with the source interface.

```
[{-max_time_out_of_sync <maxTimeOutOfSync>|
-manual_refresh}]
```

Specifies the time, in 1—1440 minutes (up to 24 hours), that the source and destination can be out of synchronization before an update occurs. If you do not specify a <code>max\_time\_out\_of\_sync</code> value, use the <code>-manual\_refresh</code> option to indicate that the destination will be updated on demand using the <code>nas\_replicate-refresh</code> command. If no option is selected, the refresh default time for a filesystem replication is 10 minutes.

# [-overwrite\_destination]

For an existing destination object, discards any changes made to the destination object and restores it from the established common base, thereby starting the replication session from a differential copy. If this option is not specified and the destination object contains different content than the established common base, an error is returned.

### [-tape\_copy]

For filesystem replication only, creates and stops the replication session to enable an initial copy using the physical tape backup and process instead of an initial copy over the network. *Using Celerra Replicator* (*V*2) describes the procedures for performing a tape copy, which involves a manually issued

backup to tape from the source filesystem, a restore from tape to the destination filesystem, and a start of the replication session.

### [-background]

Executes the command in asynchronous mode. Use the **nas\_task** command to check the status of the command.

# CREATING VDM REPLICATION

#### -create <name>

Assigns a name to the VDM replication session. The name must be unique for each Data Mover pair, which is defined by the interconnect.

```
-source -vdm <vdmName> | [id=<VdmId>]
```

Specifies the name or ID of an existing VDM to replicate. This replicates the CIFS working environment information contained in the root filesystem of the VDM. The source VDM must be in a loaded read/write or mounted read-only state. The source VDM can be the source or destination VDM of another replication session.

**Note:** Any filesystem mounted to a VDM must be replicated using filesystem replication. VDM replication affects the VDM only.

```
-destination -vdm {<existing_dstVdmName>|
id=<dstVdmId>|-pool {id=<dstStoragePoolId>|
<dstStoragePool>}
```

Specifies either an existing destination VDM or the storage needed to create the destination VDM. An existing destination VDM must be mounted as read-only, the same size as the source, and not loaded. The destination VDM can be the source of another replication but cannot be the destination of another replication. Specifying a storage pool creates the destination VDM automatically, as read-only, using the same name and size as the source VDM.

-interconnect {<name>|id=<interConnectId>}
Specifies the local (source) side of an established Data Mover
interconnect to use for this replication session.

Use the **nas\_cel -interconnect -list** command on the source Celerra to list the interconnects available to replication sessions. The **nas\_cel -interconnect -create** command is executed twice, one from each side, to create an interconnect between a pair of Data Movers (two local Data Movers for local replication, or one

local and one remote, for remote replication). Loopback interconnects are created for each Data Mover and are named automatically.

```
[-source_interface {<nameServiceInterfaceName>
|ip=<ipAddr>}]
```

Instructs the replication session to use a specific local interface defined for the interconnect on the source Celerra instead of selecting the local interface supporting the lowest number of sessions (the default). If this local interface was defined for the interconnect using a name service interface name, specify the name service interface name; if it was defined using an IP address, specify the IP address. If you define an interface using an IP address, make sure that the destination interface uses the same IPv4/IPv6 protocol. An IPv4 interface cannot connect to an IPv6 interface and vice versa. Both sides of the connection must use the same protocol.

The **source\_interfaces** field of the output from the **nas\_cel -interconnect -info** command shows how the source interface was defined. This option does not apply to a loopback interconnect, which always uses 127.0.0.1.

If no source interface is specified, the system will select an interface. This ensures that the interface selected can communicate with the destination interface.

```
[-destination_interface{<nameServiceInterfaceName>|ip
=<ipaddr>}]
```

Instructs the replication session to use a specific peer interface defined for the interconnect on the destination Celerra instead of selecting the peer interface supporting the lowest number of sessions (the default). If this peer interface was defined for the interconnect using a name service interface name, specify the name service interface name; if it was defined using an IP address, specify the IP address. If you define an interface using an IP address, make sure that the source interface uses the same IPv4/IPv6 protocol. An IPv4 interface cannot connect to an IPv6 interface and vice versa. Both sides of the connection must use the same protocol.

The destination\_interfaces field of the output from the nas\_cel -interconnect -info command shows how the peer interface was defined. This option does not apply to a loopback interconnect, which always uses 127.0.0.1.

If no destination interface is specified, the system will select an interface. This ensures that the interface selected can communicate with the source interface.

# [{-max\_time\_out\_of\_sync <maxTimeOutOfSync> |-manual\_refresh}]

Specifies the time, in 1–1440 minutes (up to 24 hours), that the source and destination can be out of synchronization before an update occurs. If you do not specify a <code>max\_time\_out\_of\_sync</code> value, use the <code>-manual\_refresh</code> option to indicate that the destination will be updated on demand using the <code>nas\_replicate-refresh</code> command. If no option is selected, the refresh default time for a VDM replication is 5 minutes.

### [-overwrite destination]

For an existing destination object, discards any changes made to the destination object and restores it from the established common base, thereby starting the replication session from a differential copy. If this option is not specified, and the destination object contains different content than the established common base, an error is returned.

# [-background]

Executes the command in asynchronous mode. Use the **nas\_task** command to check the status of the command.

# CREATING ISCSI LUN REPLICATION

#### -create <name>

Assigns a name to the V2 iSCSI LUN replication session. The name must be unique for each Data Mover pair, which is defined by the interconnect.

-destination -lun {<lunNumber> -target <targetIqn>} Specifies an available iSCSI LUN and iSCSI target at the destination. The destination iSCSI LUN cannot be created automatically, unlike a destination filesystem or VDM. The destination iSCSI LUN must be the same size as the source, and it must be read-only and marked protected. For -lun, specify a LUN number up to 256. For -target, specify a fully qualified name (IQN) of the iSCSI target with the destination LUN.

-interconnect {<name>|id=<interConnectId>}
Specifies the local (source) side of an established Data Mover
interconnect to use for this replication session.

Use the nas\_cel -interconnect -list command on the source Celerra to list the interconnects available to replication sessions. The nas\_cel -interconnect -create command is executed twice, one from each side, to create an interconnect between a pair of Data Movers (two local Data Movers for local replication, or one local and one remote for remote replication). Loopback interconnects are created for each Data Mover and are named automatically.

```
[-source_interface {<nameServiceInterfaceName>
|ip=<ipAddr>}]
```

Instructs the replication session to use a specific local interface defined for the interconnect on the source Celerra instead of selecting the local interface supporting the lowest number of sessions (the default). If this local interface was defined for the interconnect using a name service interface name, specify the name service interface name; if it was defined using an IP address, specify the IP address. If you define an interface using an IP address, make sure that the destination interface uses the same IPv4/IPv6 protocol. An IPv4 interface cannot connect to an IPv6 interface and vice versa. Both sides of the connection must use the same protocol.

The **source\_interfaces** field of the output from the **nas\_cel -interconnect -info** command shows how the source interface
was defined. This option does not apply to a loopback
interconnect, which always uses 127.0.0.1.

If no source interface is specified, the system will select an interface. This ensures that the interface selected can communicate with the destination interface.

```
[-destination_interface{<nameServiceInterfaceName>|ip
=<ipaddr>}]
```

Instructs the replication session to use a specific peer interface defined for the interconnect on the destination Celerra instead of selecting the peer interface supporting the lowest number of sessions (the default). If this peer interface was defined for the interconnect using a name service interface name, specify the name service interface name; if it was defined using an IP address, specify the IP address. If you define an interface using an IP address, make sure that the source interface uses the same

IPv4/IPv6 protocol. An IPv4 interface cannot connect to an IPv6 interface and vice versa. Both sides of the connection must use the same protocol.

The **destination\_interfaces** field of the output from the **nas\_cel -interconnect -info** command shows how the peer interface was defined. This option does not apply to a loopback interconnect, which always uses 127.0.0.1.

If no destination interface is specified, the system will select an interface. This ensures that the interface selected can communicate with the source interface.

```
[{-max_time_out_of_sync <maxTimeOutOfSync>|
-manual_refresh}]
```

Specifies the time, in 1–1440 minutes (up to 24 hours), that the source and destination can be out of synchronization before an update occurs. If you do not specify a <code>max\_time\_out\_of\_sync</code> value, use the <code>-manual\_refresh</code> option to indicate that the destination will be updated on demand using the <code>nas\_replicate-refresh</code> command. If no option is selected, the refresh default time for an iSCSI LUN replication is 10 minutes.

#### [-overwrite destination]

For an existing destination object, discards any changes made to the destination object and restores it from the established common base, thereby starting the replication session from a differential copy. If this option is not specified, and the destination object contains different content than the established common base, an error is returned.

### [-background]

Executes the command in asynchronous mode. Use the **nas\_task** command to check the status of the command.

# **START OPTIONS**

```
-start {<name>|id=<sessionId>}
```

From the source side only, specifies the name or session ID needed to start the replication session. A replication name is unique for each Data Mover pair; if a duplicate name is detected on the system, the session ID is required. To get the session ID, use nas\_replicate -list.

```
[-interconnect {<name>|id=<interConnectId>}]
Specifies an established source-side (local) Data Mover
interconnect to use for the replication session. Use the nas_cel
-interconnect -list command to list the interconnects available to
replication sessions. The nas_cel -interconnect -create command
```

creates an interconnect between a pair of Data Movers (two local Data Movers for local replication, or one local and one remote, for remote replication). Loopback interconnects are created and named automatically, and always use IP address 127.0.0.1.

```
[-source_interface {<nameServiceInterfaceName>
|ip=<ipaddr>}]
```

As the source interface for the replication session, uses a specific local interface defined for the interconnect instead of any local interface defined for the interconnect (the default, which enables the software to select the interface supporting the lowest number of sessions). If this interface was defined for the interconnect using a name service interface name, specify the name service interface name; if it was defined using an IP address, specify the IP address (IPv4 or IPv6). If you define an interface using an IP address, make sure that the destination interface uses the same IPv4/IPv6 protocol. An IPv4 interface cannot connect to an IPv6 interface and vice versa. Both sides of the connection must use the same protocol.

# [-destination\_interface{<nameServiceInterfaceName>|ip =<ipaddr>}]

As the destination interface for the replication session, uses a specific peer interface defined for the interconnect instead of any peer interface defined for the interconnect (the default, which enables the software to select the interface supporting the lowest number of sessions). If this interface was defined for the interconnect using a name service interface name, specify the name service interface name; if it was defined using an IP address, specify the IP address (IPv4 or IPv6). If you define an interface using an IP address, make sure that the source interface uses the same IPv4/IPv6 protocol. An IPv4 interface cannot connect to an IPv6 interface and vice versa. Both sides of the connection must use the same protocol.

```
[{-max_time_out_of_sync <maxtimeOutOfSync>
|-manual_refresh}]
```

Specifies the time, in 1—1440 minutes (up to 24 hours), that the source and destination can be out of synchronization before an update occurs. If you do not specify a max\_time\_out\_of\_sync value, use the -manual\_refresh option to indicate that the destination will be updated on demand using the nas\_replicate -refresh command. If no option is selected, the refresh default time for both filesystem replication and iSCSI LUN is 10 minutes, and 5 minutes for VDM replication sessions.

# [-overwrite\_destination]

For an existing destination object, discards any changes made to the destination object and restores the destination object from the established, internal common base checkpoint, thereby starting the replication session from a differential copy. If this option is not specified and the destination object has different content than the established common base, an error is returned.

#### [-reverse]

Reverses the direction of the replication session when invoked from the new source side (the original destination). A reverse operation continues to use the established replication name or replication session ID. Use this option to restart replication after a failover or switchover.

### [-full\_copy]

For an existing destination object that contains content changes, performs a full copy of the source object to the destination object. If replication cannot be started from a differential copy using the **-overwrite\_destination** option, omitting this option causes the command to return an error.

# [-background]

Executes the command in asynchronous mode. Use the **nas\_task** command to check the status of the command.

# **MODIFY OPTIONS**

```
-modify {<name> | id=<sessionId>}
```

From the source side only, specifies the name or session ID of the replication session to be modified. If a duplicate name is detected on the system, the session ID (fixed for the life of the session) is required. To get the session ID, use nas\_replicate -list.

**Note:** A session cannot be modified if a **-stop**, **-delete**, **-reverse**, **-failover**, **-switchover**, **-create**, or **-start** operation is running. However, once a **-modify** operation is running, any other operation is permitted.

```
[-name <newName>]
```

Renames the replication session to the new name specified. When renaming a session, note that the name must be unique for each Data Mover pair.

```
[-source_interface {<nameServiceInterfaceName>
|ip=<ipaddr>}]
```

Changes the source interface used for the session to another local interface from the list defined for the interconnect. If this interface

was defined for the interconnect using a name service interface name, specify the name service interface name; if it was defined using an IP address, specify the IP address (IPv4 or IPv6). If you change an IP address, make sure that the destination interface uses the same IPv4/IPv6 protocol. An IPv4 interface cannot connect to an IPv6 interface and vice versa. Both sides of the connection must use the same protocol.

[-destination\_interface{<nameServiceInterfaceName>|ip
=<ipaddr>}]

Changes the destination interface used for the session to another peer interface from the list defined for the interconnect. If this interface was defined for the interconnect using a name service interface name, specify the name service interface name; if it was defined using an IP address, specify the IP addres (IPv4 or IPv6). If you change an IP address, make sure that the source interface uses the same IPv4/IPv6 protocol. An IPv4 interface cannot connect to an IPv6 interface and vice versa. Both sides of the connection must use the same protocol.

```
[-max_time_out_of_sync <maxTimeOutOfSync>
|-manual_refresh]
```

Specifies the time, from 1—1440 minutes (up to 24 hours), that the source and destination can be out of synchronization before an update occurs. If you do not specify a <code>max\_time\_out\_of\_sync</code> value, use the <code>-manual\_refresh</code> option to indicate that the destination will be updated on demand using the <code>nas\_replicate-refresh</code> command. If no option is selected, the refresh default time for both filesystem replication and iSCSI LUN is 10 minutes, and 5 minutes for VDM replication sessions.

# **STOP OPTIONS**

```
-stop {<name>|id=<session_id>}
```

Executed from the Control Station on the source Celerra, stops the specified replication session but retains the session's configuration information. Any data transfer in progress is terminated immediately and the destination object is restored to a consistent state.

**Note:** A session cannot be stopped if the **-delete option** is already running for the session. Once a stop operation is in progress, only the options **-list**, **-info**, and the **nas\_task** command are permitted.

```
[-mode {source|destination|both}]
```

When stopping a session handling a local or remote replication from the source side, the **-mode both** option immediately stops

both sides of the replication session. The **-mode source** option stops only the replication session on the source and ignores the other side of the replication relationship. If the destination side is not operational, the **-mode source** option is required to stop the session. From the destination side, only the **-mode destination** option can be issued. When stopping a session handling a loopback replication, you can specify any **-mode** option to stop the session.

### [-background]

Executes the command in asynchronous mode. Use the **nas\_task** command to check progress.

### **DELETE OPTIONS**

# -delete {<name>|id=<session\_id>}

Executed from the Control Station on the source Celerra, cancels replication data transfer if it is in progress, and deletes the replication session specified by the **-mode** options.

# [-mode {source|destination|both}]

When deleting a local or remote replication session from the source side, the **-mode both** option deletes both sides of the replication session. The **-mode source** option immediately aborts only the replication session on the source and ignores the other side of the replication relationship. If the destination side is not operational, the **-mode source** option is required to delete the session. From the destination side, only the **-mode destination** option can be issued. When deleting a loopback replication, you can specify any **-mode** option to stop the session.

#### [-background]

Executes the command in asynchronous mode. Use the **nas\_task** command to check progress.

The execution of the **-delete** option is asynchronous and can be delayed if there is a network problem. During the delete process, other operations on the replication session are not allowed.

# **FAILOVER OPTIONS**

### -failover {<name> | id=<session\_id>}

In response to a potential disaster scenario, performs a failover of the specified replication session with possible data loss. Execute this command from the Control Station on the destination Celerra only. This command cancels any data transfer that is in process and marks the destination object as read-write so that it can serve as the new source object. When the original source Data Mover becomes reachable, the source object is changed to read-only (and for an iSCSI LUN, protected mode).



# **CAUTION**

The execution of the failover operation is asynchronous and results in data loss if all the data was not transferred to the destination site prior to issuing the failover.

For an iSCSI LUN, the administrator needs to make sure the LUN masks are set correctly on the destination target, so that after failover, hosts can connect to the destination LUN.

If there are multiple sessions using the same source object, only one replication session can be failed over. After the selected session is failed over, the other sessions become inactive until the session is restarted or failed back.

# [-background]

Executes the command in asynchronous mode. Use the **nas\_task** command to check progress.

# **SWITCHOVER OPTIONS**

```
-switchover {<name>|id=<session_id>}
```

For test or migration purposes, switches over the specified replication relationship and performs synchronization of the source and destination without data loss. Execute this command from the Control Station on the source Celerra only. This command stops replication, mounts the source object as read-only, and marks the destination object as read-write so that it can act as the new source object.

Unlike a reverse operation, a switchover operation does not restart replication.

# [-background]

Executes the command in asynchronous mode. Use the **nas\_task** command to check progress.

# **REVERSE OPTIONS**

```
-reverse {<name>|id=<session_id>}
```

If executed from the source side of a replication session, reverses the direction of the specified replication session without data loss. A reverse synchronizes the destination with the source, mounts the source object as read-only, stops replication, marks the destination object as read-write so that it can act as the new source object, then restarts replication in the reverse direction from a differential copy (using the same configuration parameters established originally for the session).

# [-background]

Executes the command in asynchronous mode. Use the **nas\_task** command to check progress.

#### **REFRESH OPTIONS**

### -refresh{<name>|id=<session\_id>}

Updates the destination side of the specified replication session based on changes to the source side. Execute this command from the Control Station on the source side only. A refresh operation handles updates on demand; as an alternative, the <code>-max\_time\_out\_of\_sync</code> option performs an update automatically after a specified number of minutes.

If the data changes on the source are large, this command can take a long time to complete. Consider running this command in background mode.

```
[-source{<ckptName>|id=<ckptId>}
-destination{<ckptName>|id=<ckptId>}]
```

Instructs the replication -refresh option to use a specific checkpoint on the source side and a specific checkpoint on the destination side.

Specifying source and destination checkpoints for the **-refresh** option is optional. However, if you specify a source checkpoint, you must also specify a destination checkpoint. Replication transfers the contents of the user-specified source checkpoint to the destination filesystem. This transfer can be either a full copy or a differential copy depending on the existing replication semantics. After the transfer, the replication internally refreshes the user specified destination checkpoint and marks the two checkpoints as common bases.

After the replication refresh operation completes successfully, both the source and destination checkpoints have the same view of their filesystems. The replication continues to use these checkpoints as common bases until the next transfer is completed. After a user checkpoint is marked with a common base property, the property is retained until the checkpoint is refreshed or deleted. A checkpoint that is already paired as a common base with another checkpoint propagates its common base property when it is specified as the source in a replication refresh operation. This propagation makes it possible for filesystems without a direct replication relationship to have common base checkpoints.

#### [-background]

Executes the command in asynchronous mode. Use the **nas\_task** command to check progress.

# STORAGE SYSTEM OUTPUT

The number associated with the storage device is dependent on the attached storage system. CLARiiON storage systems display a prefix of APM before a set of integers, for example, APM00033900124-0019. For example, Symmetrix storage systems appear as 002804000190-003C. The outputs displayed in the examples use a CLARiiON storage system.

# **EXAMPLE #1** To list all the Celerra Replicator V2 sessions, type:

# \$ nas\_replicate -list

Name	Type	Local Mover Interconnect	Celerra	Status
ufs1_rep1	filesystem	server_3>NYs3_LAs2	cs110	OK
vdm1_rep1	vdm	server_3>NYs3_LAs2	cs110	OK
iscsi_rep1	iscsiLun	server_3>NYs3_LAs2	cs110	OK

# Where:

Value	Definition
Name	Either then name of the session or the globally unique session ID for the session, if there are duplicate names on the system.
Туре	The type of V2 replication session (ongoing filesystem (fs), copy, LUN, or VDM).
Source Mover	The source Data Mover for the session.
Interconnect	The name of the source-side interconnect used for the session.
Celerra	The name of the Celerra Network Server.
Status	The status of the session (OK, Active, Idle, Stopped, Error, Waiting) Info, Critical.

# **EXAMPLE #2**

To create a filesystem replication session **ufs1\_rep1** on the source filesystem ufs1 and destination pool clar\_r5\_performance on the interconnect **NYs3\_LAs2** using the specified source and destination IP addresses to be updated automatically every **5** minutes, type:

```
$ nas_replicate -create ufs1_rep1 -source -fs ufs1
-destination -pool clar_r5_performance -interconnect
NYs3_LAs2 -source_interface ip=10.6.3.190
-destination_interface ip=10.6.3.173
-max_time_out_of_sync 5
```

OK

# **EXAMPLE #3** To display information for a replication session **ufs1\_rep1**, type:

```
$ nas_replicate -info ufs1_rep1
```

```
TD
                               = 184_APM00064600086_0000_173_APM00072901601_0000
                               = ufs1_rep1
Name
                              = OK
Source Status
Network Status
                              = OK
Destination Status
                              = OK
Last Sync Time
                              = Thu Dec 13 14:47:16 EST 2007
Type
                              = filesystem
Celerra Network Server
                             = cs110
```

```
Dart Interconnect
Peer Dart Interconnect
                                        = NYs3_LAs2
                                        = 20004
Replication Role
                                        = source
Source Filesystem
                                        = ufs1
Source Data Mover
Source Interface
                                        = server_3
= 10.6.3.190
Source Interface = 10.6.3.190
Source Control Port = 0
Source Current Data Port = 0
Destination Filesystem = ufs1_replica3
Destination Data Mover = server_2
Destination Interface = 10.6.3.173
Destination Control Port = 5081
Destination Data Port = 8888
Max Out of Sync Time (minutes) = 5
Next Transfer Size (Kb) = 0
Latest Snap on Source
Latest Snap on Destination = Current Transfer Size (KB) = 0
Current Transfer Remain (KB) = 0
Estimated Completion Time
Current Transfer is Full Copy = No
Current Transfer Rate (KB/s) = 76
Current Read Rate (KB/s) = 115
Current Write Rate (KB/s) = 580
                                        = 11538
Previous Transfer Rate (KB/s) = 0
Previous Read Rate (KB/s) = 0
Previous Write Rate (KB/s)
Average Read Rate (KB/s)
Average Transfer Rate (KB/s) = 6277
                                        = 0
                                         = 0
```

# **EXAMPLE #4**

To create a VDM replication session **vdm\_rep1** on source VDM **vdm1** and destination pool **clar\_r5\_performance** on the interconnect **NYs3\_LAs2** with the given source and destination IP addresses to be updated automatically every 5 minutes, type:

```
$ nas_replicate -create vdm1_rep1 -source -vdm vdm1
-destination -pool clar_r5_performance -interconnect
NYs3_LAs2 -source_interface ip=10.6.3.190
-destination_interface ip=10.6.3.173
-max time out of sync 5
```

OK

# **EXAMPLE #5**

To create an iSCSI replication session **iscsi\_rep1** with the source LUN number 0, and using specific interfaces to use for the interconnect, type:

```
$ nas_replicate -create iscsi_rep1 -source -lun 0 -target
iqn.1992-05.com.emc:apm000646000860000-4 -destination
-lun 0 -target iqn.1992-05.com.emc:apm000729016010000-2
-interconnect NYs3_LAs2 -source_interface ip=10.6.3.190
-destination_interface ip=10.6.3.173
-max_time_out_of_sync 60 -overwrite_destination
```

OK

# **EXAMPLE #6** To list existing replication sessions, type:

# \$ nas replicate -list

Name	Type	Local Mover	Interconnect	Celerra	Status
ufs1_rep1	filesystem	server_3	>NYs3_LAs2	cs110	OK
vdm1_rep1	vdm	server_3	>NYs3_LAs2	cs110	OK
iscsi_rep1	iscsiLun	server_3	>NYs3_LAs2	cs110	OK

# **EXAMPLE #7** To manually synchronize source and destination for the replication session **iscsi\_rep1**, type:

\$ nas\_replicate -refresh iscsi\_rep1

OK

# **EXAMPLE #8** To manually synchronize source and destination for the replication session **iscsi\_rep1** by using user checkpoints on the source and the destination, type:

\$ nas\_replicate -refresh iscsi\_rep1 -source id=101
-destination id=102

OK

# **EXAMPLE #9** To stop replication on both source and destination for the replication session **icsci\_rep1**, type:

\$ nas\_replicate -stop iscsi\_rep1 -mode both

OK

# **EXAMPLE #10** To start stopped replication session **iscsi\_rep1** on interconnect **NYs3\_LAs2**, specify manual refresh and to overwrite the destination LUN performing a full copy, type:

\$ nas\_replicate -start iscsi\_rep1 -interconnect NYs3\_LAs2
-manual\_refresh -overwrite\_destination -full\_copy

# **EXAMPLE #11** To display information for the iSCSI replication session **iscsi\_rep1**, type:

\$ nas\_replicate -info iscsi\_rep1

```
Source Status
                                = OK
                              = OK
Network Status
                              = OK
Destination Status
Last Sync Time
                              = iscsiLun
Type
Celerra Network Server
                              = cs110
Dart Interconnect
                              = NYs3 LAs2
Peer Dart Interconnect = 20004
Replication Role = source
Source Target = iqn.1992-05.com.emc:apm000646000860000-4
Source Target
                               = 0
Source LUN
                              = server_3
= 10.6.3.190
Source Data Mover
Source Interface
Source Control Port
                              = 0
                             = 0
= iqn.1992-05.com.emc:apm000729016010000-2
Source Current Data Port
Destination Target
Destination LUN
                               = 0
                          = server_2
Destination Data Mover
Destination Interface
                              = 10.6.3.173
Destination Control Port = 5081
Destination Data Port = 8888
                              = 5081
Max Out of Sync Time (minutes) = Manual Refresh
Application Data
Next Transfer Size (Kb)
                            = 0
Latest Snap on Source
Latest Snap on Destination
Current Transfer Size (KB)
Current Transfer Remain (KB) = 0
Estimated Completion Time
Current Transfer is Full Copy = No
Current Transfer Rate (KB/s) = 78
Current Read Rate (KB/s) = 12578
Current Write Rate (KB/s) = 590
Previous Transfer Rate (KB/s) = 0
                           = 0
= 0
Previous Read Rate (KB/s)
Previous Write Rate (KB/s)
Average Transfer Rate (KB/s) = 6497
Average Read Rate (KB/s)
                           = 0
                               = 0
Average Write Rate (KB/s)
```

# **EXAMPLE #12** To display information for the VDM replication session **vdm\_rep1**, type:

```
$ nas_replicate -info vdm1_rep1
```

```
Source Data Mover
                              = server_3
                              = 10.6.3.190
Source Interface
                              = 0
Source Control Port
Source Current Data Port
Destination VDM
                              = vdm1
Destination Data Mover
                              = server_2
Destination Interface
                              = 10.6.3.173
Destination Control Port
                             = 5081
Destination Data Port
                              = 8888
Max Out of Sync Time (minutes) = 5
Next Transfer Size (Kb) = 0
Latest Snap on Source
Latest Snap on Destination
Current Transfer Size (KB)
                              = 0
Current Transfer Remain (KB) = 0
Estimated Completion Time
Current Transfer is Full Copy = No
Current Transfer Rate (KB/s) = 313
Current Read Rate (KB/s)
                              = 19297
Current Write Rate (KB/s)
                              = 469
Previous Transfer Rate (KB/s) = 0
Previous Read Rate (KB/s)
Previous Write Rate (KB/s)
Average Transfer Rate (KB/s) = 155
Average Read Rate (KB/s)
                              = 0
Average Write Rate (KB/s)
        EXAMPLE #13
                       To change the session name iscsi rep1 to iscsi rep2, and to change
                       max time out of sync value to 90, type:
                        $ nas_replicate -modify iscsi_rep1 -name iscsi_rep2
                        -max time out of sync 90
OK
        EXAMPLE #14
                       To failover the replication session ufs1_rep1, type on destination:
                        $ nas replicate -failover ufs1 rep1
OK
        EXAMPLE #15
                       To start failed over replication in the reverse direction, type:
                        $ nas_replicate -start ufs1_rep1 -interconnect LAs2_NYs3
                        -reverse -overwrite_destination
OK
        EXAMPLE #16
                       To reverse direction of the replication session ufs1_rep1, type:
                        $ nas_replicate -reverse ufs1_rep1
```

= NYs3 LAs2

= 20004

= vdm1

= source

Dart Interconnect

Source VDM

Peer Dart Interconnect

Replication Role

OK

**EXAMPLE #17** To switch over the replication session **ufs1\_rep1** using the background option, type:

\$ nas\_replicate -switchover ufs1\_rep1 -background

Info 26843676673: In Progress: Operation is still running. Check task id 4058 on the Task Status screen for results.

\*\*\* Comment: Use nas\_task -info command to find out the status of background task.

**EXAMPLE #18** To delete the replication session **fs1\_rep1** on both source and destination, type:

\$ nas\_replicate -delete fs1\_rep1 -mode both

OK

# nas\_server

Manages the Data Mover (server) table.

# **SYNOPSIS**

# **DESCRIPTION**

nas\_server manages the server tables for both physical and virtual Data Movers (VDMs), creates a VDM, sets an access control value for a physical Data Mover or VDM, renames a Data Mover and displays attributes for a specified Data Mover or all Data Movers, deletes a physical Data Mover entry from the server table, and deletes the VDM configuration for a Data Mover.

# **OPTIONS**

#### -list

Lists the Data Mover server table. The server table displays the ID, type, access control level value, slot number, group ID, state, and name of a Data Mover. Virtual Data Movers (VDMs) have their own server table and do not have a numeric reference in the general server table.

**Note:** The ID of the object is an integer and is assigned automatically. The name of the Data Mover might be truncated if it is too long for the display. To display the full name, use the **-info** option with the Data Mover ID.

Valid Data Mover types are:

1=nas
2=not used
3=not used
4=standby
5=not used

# 6=rdf

**Note:** The **nas** type is set automatically, **vdm** is set using **nas\_server**, **rdf** and **standby** are set up using **server\_standby**.

# [-all|-vdm]

The **-all** option displays the physical Data Mover and VDMs. The **-vdm** option lists the VDMs only.

#### -delete <movername>

Deletes the specified physical Data Mover entry from the server table or deletes the VDM configuration. A Data Mover that is being deleted cannot contain mounted filesystems.

Deleting a physical Data Mover requires the **root** command. Use **/nas/sbin/rootnas\_server** to execute a delete.

-info {-all|<movername>|<slot\_number>|id =<mover\_id>}
Displays attributes for all physical Data Movers, or a Data Mover
specified by its <movername>, <slot\_number>, or <mover\_id>.

-info -vdm {-all|<vdm\_name>|id=<vdm\_id>}
Displays attributes for all VDMs, or a specified VDM, including the network interfaces that are being used by the CIFS servers.

-rename <old movername> <new movername>

Changes the name of the physical Data Mover or the VDM to the specified name. The **-create** option provides more information for rules applicable to naming a Data Mover.

-acl <acl value> <movername>

Sets an access control level value that defines the owner of the physical Data Mover or the Virtual Data Mover, and the level of access allowed for users and groups defined in the access control level table. The <a href="mailto:nas\_acl">nas\_acl</a> command provides more information.

[-name <name>] [-type vdm] -create <movername> Creates a VDM with an optional name for the specified physical or VDM. The movername is case-sensitive and supports the following characters: a through z, A through Z, 0 through 9, \_(underscore), -(hyphen) though names may not start with a hyphen. The default type is nas.

[-setstate <state>]

Sets the Data Mover to **loaded** or **mounted**.

The **loaded** option installs the image of the VDM onto the physical Data Mover, but does not mount the non-root filesystems. The **mounted** option mounts the **root\_fs** as read-only, but the VDM image is not installed. The **-setstate** option is for use with replication.

**Note:** Before a VDM image is loaded, the image must be unloaded from the previous Data Mover, and the CIFS server must be joined using server\_cifs. The network interfaces used by the CIFS servers on the VDM must be available on the destination Data Mover.

```
[-fs <fs_name> | pool=<pool>]
```

Specifies a filesystem or assigns a rule set known as a storage **pool** for the VDM root filesystem.

For the **-fs** option, the filesystem must be unmounted, clean (nas\_fsck provides more information), and be either of type **uxfs** or **rawfs**. For a loaded state VDM, only an **uxfs** filesystem type can be used, but for mounted state VDM, both **uxfs** and **rawfs** can be used. The filesystem to be provided as the VDM root filesystem is renamed to **root\_fs\_vdm\_**<vdm\_name>. This is deleted when the VDM is deleted.

The storage **pool** option assigns a rule set for the root filesystem of the VDM that contains automatically created volumes and defines the type of disk volumes used and how they are aggregated. Storage pools are system defined (storage pool description provides more information) or user defined. nas\_pool provides a description of pool types.

```
[-option <options>]
```

Specifies the following comma-separated list of options:

# fstype=rawfs

Specifies the filesystem type as **rawfs** when creating a mounted VDM.

```
-vdm <vdm_name> -setstate <state>
```

Sets the state of the VDM to **loaded**, **mounted**, **tempunloaded**, or **permunloaded**.

The **loaded** option installs the image of the VDM onto the physical Data Mover, but does not mount the non-root filesystems. The **mounted** option mounts the **root\_fs** read-only, but the VDM image is not installed.

The **tempunloaded** option, temporarily unloads the VDM image, while the **permunloaded** option permanently unloads the image.

[<movername>]

Specifies a physical Data Mover for the VDM.

#### [-ConvertI18N]

When loading the VDM image, forces the conversion of the I18N mode of the VDM's root filesystem from ASCII to UNICODE. The I18N mode of the Data Mover can be either ASCII or UNICODE. The mode of the VDM must be the same as the physical Data Mover, for example, when performing the **-move** option, or when replicating.

This mode is used when the mode of the VDM root filesystem is different from that of the physical Data Mover.

-vdm <vdm\_name> -move <movername>

Moves the image of the VDM onto the physical Data Mover, and mounts the non-root filesystems.

**Note:** Before a VDM image is loaded, the image must be unloaded from the previous Data Mover, and the CIFS server must be joined using server\_checkup. The network interfaces used by the CIFS servers on the VDM must be available on the destination Data Mover.

#### [-ConvertI18N]

When loading the VDM image, forces the conversion of the I18N mode of the VDM's root filesystem from ASCII to UNICODE. The I18N mode of the Data Mover can be either ASCII or UNICODE. The mode of the VDM must be the same as the physical Data Mover, for example, when performing the **-move** option, or when replicating.

This mode is used when the mode of the VDM root filesystem is different from that of the physical Data Mover.

SEE ALSO Configuring Virtual Data Movers for Celerra, Using International Character Sets with Celerra, nas\_fs, nas\_volume, and server\_checkup.

# STORAGE SYSTEM OUTPUT

Celerra Network Servers with a CLARiiON storage system support the following system-defined storage pools: clar\_r1, clar\_r5\_performance, clar\_r5\_economy, clar\_r6, clarata\_r3, clarata\_r6, clarata\_r10, clarata\_archive, cm\_r1, cm\_r5\_performance, cm\_r5\_economy, cm\_r6, cmata\_r3, cmata\_archive, cmata\_r6, cmata\_r10, clarsas\_archive, clarsas\_r6, clarsas\_r10, clarefd\_r5, clarefd\_r10, cmsas\_archive, cmsas\_r6, cmsas\_r10, and cmefd\_r5.

Disk types when using CLARiiON are **CLSTD**, **CLEFD**, and **CLATA**, and for CLARiiON storage systems involving mirrored disks are: **CMEFD**, **CMSTD**, and **CMATA**.

Celerra Network Servers with a Symmetrix storage system supports the following system-defined storage pools: symm\_std, symm\_std\_rdf\_src, symm\_ata\_rdf\_src, symm\_ata\_rdf\_tgt, symm\_ata\_rdf\_tgt, symm\_ata\_rdf\_tgt, symm\_std\_rdf\_tgt, and symm\_efd.

For user-defined storage pools, the difference in output is in the disk type. Disk types when using a Symmetrix are: STD, R1STD, R2STD, BCV, R1BCV, R2BCV, ATA, R1ATA, R2ATA, BCVA, R1BCA, R2BCA, and EFD.

# **EXAMPLE #1** To list the physical Data Mover table, type:

# \$ nas\_server -list

id	type	acl	slot	groupID	state	name
1	1	1000	2		0	server_2
2	1	1000	3		0	server_3
3	1	1000	4		0	server_4
4	4	1000	5		0	server 5

### Where:

Value	Definition
id	ID of the Data Mover.
type	Type assigned to Data Mover.
acl	Access control level value assigned to the Data Mover or VDM.
slot	Physical slot in the cabinet where the Data Mover resides.
groupID	ID of the Data Mover group.
state	Whether the Data Mover is enabled=0, disabled=1, failed over=2.
name	Name given to the Data Mover.

# **EXAMPLE #2** To list the physical Data Mover and VDM table, type:

			\$	nas_serv	er -list	-all
id	type	acl	slot	groupID	state	name
1	1	1000	2		0	server_2
2	1	1000	3		0	server_3
3	1	1000	4		0	server_4
4	4	1000	5		0	server_5
id	acl	server	mount	edfs	rootfs	name
3	0	1			31	vdm_1

**EXAMPLE #1** provides a description of outputs for the physical Data Movers. The following table provides a description of the command output for the VDM table.

### Where:

Value	Definition
id	ID of the Data Mover.
acl	Access control level value assigned to the Data Mover or VDM.
server	Server the VDM is loaded on.
mountedfs	Filesystems that are mounted on this VDM.
rootfs	ID number of the root filesystem.
name	Name given to the Data Mover or VDM.

# **EXAMPLE #3** To list the VDM server table, type:

# **EXAMPLE #4** To list information for a Data Mover, type:

```
$ nas_server -info server_2
```

```
id = 1
name = server_2
acl = 1000, owner=nasadmin, ID=201
type = nas
slot = 2
member_of =
standby = server_5, policy=auto
status :
  defined = enabled
  actual = online, ready
```

### Where:

Value	Definition
id	ID of the Data Mover.
name	Name given to the Data Mover.

acl	Access control level value assigned to the Data Mover or VDM.	
type	Type assigned to Data Mover.	
slot	Physical slot in the cabinet where the Data Mover resides.	
member_of	Group to which the Data Mover is a member.	
standby	If the Data Mover has a local standby associated with it.	
status	Whether the Data Mover is enabled or disabled, and whether it is active.	

# **EXAMPLE #5** To display detailed information for all servers, type:

```
$ nas_server -info -all
```

```
id
          = 1
         = server_2
name
        = 1000, owner=nasadmin, ID=201
acl
type
         = nas
         = 2
slot
member_of =
standby = server_5, policy=auto
status
  defined = enabled
   actual = online, active
id
          = 2
         = server_3
name
acl
         = 1000, owner=nasadmin, ID=201
type
        = nas
         = 3
slot
member_of =
standby = server_5, policy=auto
status
  defined = enabled
  actual = online, ready
id
         = 3
         = server_4
name
         = 1000, owner=nasadmin, ID=201
acl
type
         = nas
slot
         = 4
member_of =
standby = server_5, policy=auto
status
  defined = enabled
   actual = online, ready
id
         = 4
        = server_5
         = 1000, owner=nasadmin, ID=201
acl
        = standby
type
          = 5
slot
member_of =
standbyfor= server_4,server_2,server_3
```

```
status :
  defined = enabled
  actual = online, ready
```

EXAMPLE #4 provides a description of command outputs.

# **EXAMPLE #6** To display information for all VDMs, type:

```
$ nas_server -info -vdm -all
```

```
id = 3
name = vdm_1
acl = 0
type = vdm
server = server_2
rootfs = root_fs_vdm_1
I18N mode = UNICODE
mountedfs =
member_of =
status :
  defined = enabled
   actual = mounted
Interfaces to services mapping:
```

# Where:

Value	Definition
id	ID of the Data Mover.
name	Name of the Data Mover.
acl	Access control level value assigned to the VDM.
type	For VDM server, the type is always VDM.
server	Server on which the VDM loaded.
rootfs	Root filesystem of the VDM.
I18N mode	L18N mode of the VDM. I18N mode is either ASCII or UNICODE.
mountedfs	Filesystems that are mounted on this VDM.
member_of	If it is a member of a cluster, then this field will show the cluster name.
status	Whether the VDM is enabled or disabled, and whether it can be loaded ready, loaded active, mounted, temporarily unloaded and permanently unloaded.
Interfaces to services mapping	List of interfaces that are used for the services configured on this VDM. Currently, only CIFS service is provided, so this field lists all the interfaces used in the CIFS servers configured on this VDM.

# **EXAMPLE #7**

To create a mounted VDM named **vdm\_1** on **server\_2** using the storage pool, **clar\_r5\_performance** with a **rawfs**, type:

\$ nas\_server -name vdm\_1 -type vdm -create server\_2
-setstate mounted pool=clar\_r5\_performance -option
fstype=uxfs

```
id
         = 3
         = vdm_1
name
acl
         = 0
type
        = vdm
        = server_2
server
rootfs = root fs vdm 1
I18N mode = UNICODE
mountedfs =
member_of =
status
  defined = enabled
  actual = mounted
Interfaces to services mapping:
```

**EXAMPLE** #6 provides a description of command outputs.

# **EXAMPLE #8** To set the state of a **vdm\_1** to mounted, type:

\$ nas\_server -vdm vdm\_1 -setstate mounted

```
= 3
id
          = vdm 1
name
acl
         = 0
        = vdm
type
server
        = server_2
rootfs
       = root_fs_vdm_1
I18N mode = UNICODE
mountedfs =
member_of =
status
  defined = enabled
   actual = mounted
Interfaces to services mapping:
```

EXAMPLE #6 provides a description of command outputs.

# **EXAMPLE #9** To move the image of **vdm\_1** onto **server\_4**, type:

\$ nas\_server -vdm vdm\_1 -move server\_4

```
member_of =
status :
  defined = enabled
  actual = loaded, ready
Interfaces to services mapping:
```

EXAMPLE #6 provides a description of command outputs.

# **EXAMPLE #10** To rename a Data Mover entry from **server\_2** to **dm2**, type:

```
$ nas_server -rename server_2 dm2
```

```
id = 1
name = dm2
acl = 1000, owner=nasadmin, ID=201
type = nas
slot = 2
member_of =
standby = server_5, policy=auto
status :
  defined = enabled
  actual = online, active
```

**EXAMPLE** #4 provides a description of command outputs.

# **EXAMPLE #11** To set the access control level for **server\_2**, type:

```
$ nas_server -acl 1432 server_2
```

```
id = 1
name = server_2
acl = 1432, owner=nasadmin, ID=201
type = nas
slot = 2
member_of =
standby = server_5, policy=auto
status :
  defined = enabled
  actual = online, ready
```

**Note:** The value 1432 specifies **nasadmin** as the owner, gives users with an access level of at least observer read-only access, users with an access level of at least operator read/write access, and users with an access level of at least admin read/write/delete access.

EXAMPLE #4 provides a description of command outputs.

# **EXAMPLE #12** To delete **vdm\_1**, type:

```
$ nas_server -delete vdm_1
id
         = 3
name
         = vdm_1
acl
         = 0
        = vdm
type
server
       = root_fs_vdm_1
rootfs
I18N mode = UNICODE
mountedfs =
member_of =
status
  defined = enabled
  actual = permanently unloaded
Interfaces to services mapping:
```

EXAMPLE #6 provides a description of command outputs.

# **EXAMPLE #13** To delete a physical Data Mover using **root** command, type:

```
$ /nas/sbin/rootnas_server -delete server_3
```

```
id = 2
name = server_3
acl = 0
type = nas
slot = 3
member_of =
standby = server_5, policy=auto
status :
  defined = disabled
  actual = boot_level=0
```

EXAMPLE #6 provides a description of command outputs.

# nas slice

Manages the slice table.

# **SYNOPSIS**

```
nas_slice
   -list
| -delete <slice_name>
| -info {<slice_name>|id=<slice_id>}
| -rename <old_name> <new_name>
| [-name <name>] -create <volume_name> <size>[<offset>]
```

### **DESCRIPTION**

nas\_slice creates, deletes, and renames a slice. nas\_slice also displays
the slice table and displays the attributes of a specified slice.

# **OPTIONS**

-list

Lists the slice table.

**Note:** The ID of the slice is an integer and is assigned automatically. The name of the slice may be truncated if it is too long for the display. To display the full name, use the **-info** option with the slice ID.

```
-delete <slice_name>
```

Deletes an unused slice. Slices cannot be used by a storage pool.

```
-info {<slice_name>|id=<slice_id>}
```

Displays attributes for the slice as specified by the <slice\_name> or <slice\_id>.

```
-rename <old_name> <new_name>
```

Changes the current name of a slice to the new name.

```
[-name <name>] -create <volume_name> <size>
```

Creates a slice with the specified size in megabytes. Also assigns an optional name for the slice. Names are case-sensitive.

```
[<offset>]
```

Specifies the offset which is the distance in megabytes from the beginning of the volume to the start of a slice. If an offset is not specified, the system defaults to the next best-fit size.

#### SEE ALSO

Managing Celerra Volumes and File Systems with Automatic Volume Management, Managing Celerra Volumes and File Systems Manually, nas\_volume to manage metavolumes, stripe volumes, and disk volumes and nas\_cel.

# **EXAMPLE #1** To display the slice table, type:

		-			
SI	กลร	S 1 '	100	-list	Ε.

			<pre>\$ nas_sli</pre>	ice -list	
id	inuse	slice_of	offsetMB	sizeMB	name
1	n	1	0	134	root_dos
2	n	1	134	1	root_layout
3	У	1	135	16	root_slice_1
4	У	1	151	16	root_slice_2
5	У	1	167	16	root_slice_3
6	У	1	183	16	root_slice_4
7	У	1	199	16	root_slice_5
8	У	1	215	16	root_slice_6
9	У	1	231	16	root_slice_7
10	У	1	247	16	root_slice_8
11	У	1	263	16	root_slice_9
12	У	1	279	16	root_slice_10
13	У	1	295	16	root_slice_11
14	У	1	311	16	root_slice_12
15	У	1	327	16	root_slice_13
16	У	1	343	16	root_slice_14
17	У	1	359	16	root_slice_15
18	У	1	375	16	root_slice_16
19	n	1	391	1	root_log_1
20	n	1	392	1	root_log_2
21	n	1	393	1	root_log_3
22	n	1	394	1	root_log_4
23	n	1	395	1	root_log_5
24	n	1	396	1	root_log_6
25	n	1	397	1	root_log_7
26	n	1	398	1	root_log_8
27	n	1	399	1	root_log_9
28	n	1	400	1	root_log_10
29	n	1	401	1	root_log_11
30	n	1	402	1	root_log_12
31	n	1	403	1	root_log_13
32	n	1	404	1	root_log_14

Note: This is a partial listing of the slice table.

# Where:

	WHOIC.
Value	Definition
id	ID of the slice.
inuse	Whether the slice is used.
slice_of	Volume the slice is sliced from.
offsetMB	Distance in megabytes from the beginning of the disk to the start of the slice.
sizeMB	Slice size in megabytes.
name	Name of the slice.

# EXAMPLE #2

To create a slice named slv1 on volume d8 of 1024 MB in size, type:

```
$ nas_slice -name slv1 -create d8 1024
```

```
id = 76
name = slv1
acl = 0
in_use = False
slice_of = d8
offset(MB) = 0
size (MB) = 1024
volume_name = slv1
```

# Where:

Value	Definition	
id	ID of the slice.	
name	Name of the slice.	
acl	Access control level value assigned to the slice.	
inuse	Whether the slice is used.	
slice_of	Volume the slice is sliced from.	
offset(MB)	Distance in Megabytes from the beginning of the disk to the start of the slice.	
size (MB)	Slice size in Megabytes.	
volume_name	Volume used to create the slice.	

# **EXAMPLE #3**

To create a slice volume of 1 GB (1024 MB) with an offset of 1 GB called **slv2** from volume **d8**, type:

```
$ nas_slice -name slv2 -create d8 1024 1024
```

```
id = 77
name = slv2
acl = 0
in_use = False
slice_of = d8
offset(MB) = 1024
size (MB) = 1024
volume name = slv2
```

EXAMPLE #2 provides a description of command outputs.

# **EXAMPLE #4**

To display attributes of **slv1**, type:

\$ nas\_slice -info slv1

```
id = 76
name = slv1
acl = 0
in_use = False
slice_of = d8
offset(MB) = 0
size (MB) = 1024
volume_name = slv1
```

EXAMPLE #2 provides a description of command outputs.

# **EXAMPLE #5**

To rename a slice from **slv** to **slv1**, type:

\$ nas\_slice -rename slv slv1

```
id = 76
name = slv1
ac1 = 0
in_use = False
slice_of = d8
offset(MB) = 0
size (MB) = 1024
volume name = slv1
```

EXAMPLE #2 provides a description of command outputs.

# **EXAMPLE #6**

To delete **slv1**, type:

\$ nas\_slice -delete slv1

id = 76
name = slv1
acl = 0
in\_use = False
slice\_of = d8
offset(MB) = 0
size (MB) = 1024

EXAMPLE #2 provides a description of command outputs.

# nas stats

Manages Statistics Groups.

# **SYNOPSIS**

```
nas_stats
 -groups
   { -list
     -info [-all|<statgroup_name>[,...]]
     -create <statgroup_name>
          [-description "<description_line>"]
            {<statpath_name>|<statgroup_name>}[,...]
     -modify <statgroup_name>
       { [-rename < new_statgroup_name>]
        [-description "<description_line>"]
        [{<statpath_name> | <statgroup_name>}[,...]]}
    -add <statgroup_name>
       {<statpath_name> | <statgroup_name>}[,...]
     -remove <statgroup_name>
      {<statpath_name>|<statgroup_name>}[,...]
     -delete <statgroup_name> [-Force]
     -database
      { -recover [-Force]
       -verify }
```

### DESCRIPTION

**nas\_stats** allows the user to manage Statistics Groups. There are two types of Statistics Groups: System-defined and User-defined groups.

System-defined statistics groups are created as part of the installation (or upgrade) process and include the following statistics, which correspond to the summary and table collections used by server\_stats:

System-defined statistics group name	server_stats collection name
basic-std	-summary basic
caches-std	-summary caches
cifs-std	-summary cifs
nfs-std	-summary nfs
cifsOps-std	-table cifs
diskVolumes-std	-table dvol
metaVolumes-std	-table fsvol
netDevices-std	-table net
nfsOps-std	-table nfs

**Note:** server\_stats collection names are deprecated and will not be supported in future releases.

Statistics Groups can be created to include any combination of statpath names, displayed through the server\_stats command, as well as other existing statgroup names.

Any Statistics Group name can be used with server\_stats -monitor to collect statistics as defined in its member\_stats list.

#### **OPTIONS**

#### -list

Lists system and user-defined Statistics Groups.

#### -info

Provides detailed information on all (or specified) Statistics Groups.

#### -create

Creates a statistics group and specifies what statpath names it includes. It also allows the nesting of statgroups by adding existing statgroups to new statgroups.

Statgroup names can be used with the -info request. A statgroup name is limited to 255 characters. Space, slash, back slash, quote, double quote, and comma are the illegal characters in it.

#### [-description]

The **-description** option is optional and defaults to the statgroup name. If the **-description** option is used, its argument must be enclosed in quotation marks.

#### -modify

Allows you to modify a statgroup's member\_stats list by specifying the new member statistics of the group, overriding the previous contents.

#### -add

Allows you to add statpath and existing statgroup names to a statgroup by specifying additional items to be appended to the statgroup's list member\_stats list.

#### -remove

Allows you to remove member statpath and statgroup names from a statgroup by specifying the items to remove from the statgroup's member\_stats list.

#### -delete

Allows you to delete a statgroup. However, this option does not delete any statsgroups that are members of the statgroup.

#### -recover

Attempts to recover the latest uncorrupted copy of the Statistics Groups database from the NAS database backups. nas\_stats searches through the available backups and restores the latest copy. In this event, NAS database backups do not contain a healthy version of the Statistics Groups; a new Statistics Groups database is installed. In the case of a new Statistics Groups database, all user-defined information is lost. NAS database backups run hourly and Celerra maintains the last 12 backups.

#### [-Force]

Use the **-Force** option with the -recover option to skip the warning prompt.

#### -verify

Checks the health status of the Statistics Groups database.

## **SEE ALSO** *server\_stats*

# **EXAMPLE #1** To list

To list the system-defined and user-defined Statistics Groups, type:

\$ nas\_stats -groups -list

```
Type Name

System basic-std
System basicCifs-std
... ...
User basic
User nfsNet
... ...
```

# **EXAMPLE #2** To provide detailed information on all (or specified) Statistics Groups, type:

```
name
                = basic3
description
                = CPU and Memory
                = User-defined
type
member_stats
                = kernel.cpu.utilization.cpuUtil,kernel.memory.freeBytes
member_elements =
member of
                = caches-std
name
description
                = The caches system-defined group.
                = System-defined
type
member_stats
fs.dnlc.hitRatio,fs.ofCache.hitRatio,kernel.memory.bufferCache.hitRatio
member_elements =
member_of
                = cifs-std
name
                = The cifs system-defined group.
description
                = System-defined
type
member_stats
cifs.global.basic.totalCalls,cifs.global.basic.reads,cifs.global.basic.readBytes
,cifs.global.basic.readAvgSize,cifs.global.basic.writes,cifs.global.basic.writeB
ytes, cifs.qlobal.basic.writeAvqSize, cifs.qlobal.usaqe.currentConnections, cifs.ql
obal.usage.currentOpenFiles
member_elements =
member_of
                = newSG
                = cifsOps-std
name
description
                = The cifs table system-defined group.
                = System-defined
type
                = cifs.smb1.op,cifs.smb2.op
member_stats
member_elements =
member_of
                = diskVolumes-std
name
                = The disk volume table system-defined group.
description
                = System-defined
type
member_stats
                = store.diskVolume
member_elements =
name
                = metaVolumes-std
description
                = The meta volume table system-defined group.
                = System-defined
type
member_stats
                = store.logicalVolume.metaVolume
member_elements =
member_of
name
                = netDevices-std
                = The net table system-defined group.
description
                = System-defined
type
               = net.device
member_stats
member_elements =
member_of
```

```
name
               = newSG
description
              = newSG
               = User-defined
type
member_stats = cifs-std,nfs.v3.op,nfs.v4.op
member_elements =
member of
               = nfs-std
name
Description
             = The nfs system-defined group.
                = System-defined
type
member_stats
nfs.totalCalls,nfs.basic.reads,nfs.basic.readBytes,nfs.basic.readAvqSize,nfs.bas
ic.writes, nfs.basic.writeBytes, nfs.basic.writeAvgSize, nfs.currentThreads
member_elements =
member_of
         = nfsOps-stu
on = The nfs table system-defined group.
name
description
type
              = System-defined
member_stats = nfs.v2.op,nfs.v3.op,nfs.v4.op
member_elements =
member_of
            = statgroup1
name
description
              = My first group
               = User-defined
type
member_stats
net.basic.inBytes,net.basic.outBytes,store.readBytes,store.writeBytes
member_elements =
member_of = statgroup2
               = statgroup2
name
description
               = My first group
                = User-defined
member_stats
net.basic.inBytes,net.basic.outBytes,store.readBytes,store.writeBytes,kernel.cpu
.utilization.cpuUtil,statgroup1
member_elements =
member of
         EXAMPLE #3
                       To provide detailed information on all (or specified) Statistics
                       Groups, type:
                       $ nas_stats -groups -info statsA
name
                = statsA
description
              = My group # 2
               = user-defined
type
member_stats = statpath1, statpath2, statpath3, statsC
member_elements =
member_of = statsB
```

## **EXAMPLE #4** To create a statistics group called **basic3**, type:

\$ nas\_stats -groups -create basic3 -description "CPU and Memory" kernel.cpu.utilization.cpuUtil,kernel.memory. freeBytes

'basic3' created successfully.

## **EXAMPLE #5** To create a statistics group called **statgroup2**, type:

\$ nas\_stats -groups -create statgroup2 statgroup1,nfs,net
'statgroup2' created successfully.

## **EXAMPLE #6** To use an existing statgroup, type:

\$ nas\_stats -groups -create statgroup1 -description "My first group" kernel.cpu.utilization.cpuUtil, net.basic.inBytes,net.basic.outBytes,store.readBytes, store.writeBytes

ERROR (13421969439): 'statgroup1' already exists.

# **EXAMPLE #7** To modify a statgroup by specifying the new contents of the group, overriding the previous contents, type:

\$ nas\_stats -groups -modify statgroup2 cifs,nfs-std 'statgroup2' modified successfully.

## **EXAMPLE #8** To modify the description of a statgroup, type:

\$ nas\_stats -groups -modify basic1 -description "My basic
group"

'basic1' modified successfully.

## **EXAMPLE #9** To rename a user-defined statgroup, type:

\$ nas\_stats -groups -modify statgroup2 -rename basic2 'statgroup2' modified successfully.

# **EXAMPLE #10** To add to the member\_stats list of a statgroup, type:

\$ nas\_stats -groups -add statgroup2
kernel.cpu.utilization.cpuUtil,statgroup1

Adding the following statistics:
... kernel.cpu.utilization.cpuUtil
... statgroup1
Statistics added to 'statgroup2' successfully.

# **EXAMPLE #11** To remove from the member\_stats list of a statgroup, type:

\$ nas\_stats -groups -remove statgroup1
kernel.cpu.utilization.cpuUtil

Removing the following statistics:
... kernel.cpu.utilization.cpuUtil
Statistics removed from 'statgroup1' successfully.

## **EXAMPLE #12** To delete a statgroup, type:

\$ nas\_stats -groups -delete statgroup1

'statgroup1' deleted successfully.

**EXAMPLE #13** To delete reference from other groups using **statgroupA**, type:

\$ nas\_stats -groups -delete statgroupA

'statgroupA' is used in group (s): mystats1, mystats2. Clear 'statgroupA' from other groups? [Y/N] Y 'statgroupA' deleted successfully.

**EXAMPLE #14** To delete reference from other groups using **statgroupA** and the **-Force** option to skip the warning prompt, type:

\$ nas\_stats -groups -delete statgroupA -F

'statgroupA' is used in group (s): mystats1, mystats2. 'statgroupA' deleted successfully.

**EXAMPLE #15** To recover the latest healthy (uncorrupted) copy of a statgroup database from the NAS database backups, type:

\$ nas\_stats -groups -database -recover

Latest healthy database modified last on Tue Apr 7 17:29:06 EDT 2009. Any updates performed after the latest backup will be lost. Continue? [Y/N] Y The nas\_stats command recover operation is completed successfully.

**EXAMPLE #16** To recover the latest healthy (uncorrupted) copy of the statgroup database from the NAS database backups using the -**Force** option to skip the warning prompt, type:

\$ nas\_stats -groups -database -recover -Force

Latest healthy database modified last on Tue Apr 7 17:29:06 EDT 2009. The nas\_stats command recover operation is completed successfully.

**EXAMPLE #17** To check the health status of the Statistics Groups database, type:

\$ nas\_stats -groups -database -verify

Database is healthy.

# nas\_storage

Controls storage system access and performs some management tasks.

#### **SYNOPSIS**

**Note:** Output from this command is determined by the type of storage system attached to the Celerra Network Server.

#### **DESCRIPTION**

**nas\_storage** sets the name for a storage system, assigns an access control value, displays attributes, performs a health check, synchronizes the storage system with the Control Station, and performs a failback for CLARiiON systems.

#### **OPTIONS**

#### -list

Displays a list of all attached storage systems available for the Celerra Network Server.

**Note:** The ID of the object is an integer and is assigned automatically. The name of the storage system may be truncated if it is too long for the display. To display the full name, use the **-info** option with the storage system ID.

```
-info {-all|<name>|id=<storage_id>}
```

Displays the configuration of the attached storage system.

```
[-option <options>]
```

Specifies a comma-separated list of options.

```
sync={yes | no}
```

Synchronizes the Control Station's view with that of the storage system before displaying configuration information. Default=yes.

```
-rename <old_name> <new_name>
```

Renames the current storage system name to a new name. By default, the storage system name is its serial number.

```
-acl <acl_value> <name>
```

Sets an access control level value that defines the owner of the storage system, and the level of access allowed for users and groups defined in the access control level table (nas\_acl provides information).

```
-delete {<name>|id=<storage_id> [-spare
<spindle-id>|-group <diskgroup-id>]
```

Deletes an entry from the storage system table. The storage system can only be deleted after all disks on the storage system have been deleted using nas\_disk. The storage system and disks can be rediscovered using the server\_devconfig command. The -spare option deletes the hot spare disk from the hot spare pool on the CLARiiON storage used by NAS. The -group option deletes the disk group specified. This deletes and unbinds the LUNs in the RAID groups used by Celerra. If there are other LUNs in the RAID group not allocated to the Celerra, the RAID group is not unbound. After removing the Celerra LUNs, the RAID group is empty and it will be destroyed.

```
-sync {-all|<name>|id=<storage_id>}
```

Synchronizes the Control Station's view with that of the storage system.

```
-check {-all|<name>|id=<storage_id>}
```

Performs a health check on the storage system to verify if it is configured for, and in a state to provide the level of high availability that is required.

Use this option after making any management changes to your storage system (for example, changes to CLARiiON array properties, such as enabling/disabling statistics polling).

# For CLARiiON storage systems only

```
-failback {<name> | id=<storage_id>}
```

Returns the storage system's normal operating state by returning ownership of all disk volumes to their default storage processor.

To verify that the storage system failed over, type the **-info** option. If the value appears as failed\_over=True, then the system has failed over.

```
-modify {<name>|id=<storage_id>} -network {-spa|-spb}
<IP>
```

Modifies the IP address of the CLARiiON storage system in the Celerra Network Server database.

```
-modify {<name>|id=<storage_id>} -security [-username
<username>][-password <password>]
```

Updates the login information the Celerra uses to authenticate with the CLARiiON and changes the CLARiiON username, or password if the CLARiiON account is changed or the following error is reported:

```
Error 5010: APM00055105668: Storage API code=4651:
SYMAPI_C_CLAR_NOT_PRIVILEGED
```

Operation denied by Clariion array - you are not privileged to perform the requested operation

[-newpassword <new\_password]</pre>

Assigns a new password to the username on the CLARiiON.

**Note:** This operation is not supported for Symmetrix storage systems.

### **SEE ALSO**

*Celerra Network Server System Operations*, nas\_rdf, nas\_disk, and server\_devconfig.

# STORAGE SYSTEM OUTPUT

The number associated with the storage device is dependent on the attached storage system. CLARiiON storage systems display a prefix of APM before a set of integers, for example, APM00033900124-0019. For example, Symmetrix storage systems appear as 002804000190-003C.

## **EXAMPLE #1**

For the Celerra Network Server with a CLARiiON storage system, to list all attached storage systems, type:

### \$ nas\_storage -list

id	acl	name	serial_number
1	0	APM00042000818	APM00042000818

For the Celerra Network Server with a Symmetrix storage system, to list all attached storage systems, type:

#### \$ nas\_storage -list

id	acl	name	serial_number
1	0	000187940260	000187940260

#### Where:

Value	Definition
id	ID number of the attached storage system.
acl	Access control level value assigned to the attached storage system.
name	Name assigned to the attached storage system.
serial_number	Serial number of the attached storage system.

# **EXAMPLE #2** For the Celerra Network Server with a CLARiiON storage system, to display information for the attached storage system, type:

```
$ nas_storage -info APM00042000818
id
                           = APM00042000818
arrayname
name
                           = APM00042000818
                         = Clariion
type
                        = RACKMOUNT
model_type
model_num
                          = 700
db_sync_time = 1131986667 == Mon Nov 14 11:44:27 EST 2005
API_version = V6.0-629
API_version
                         = 60
num_disks
                        = 34
num_devs
num_pdevs = 8
num_storage_grps = 1
num_raid_grps = 16
                         = 16
cache_paye____
wr_cache_mirror
cache_page_size
                         = 8
                        = True
= 60
high_watermark = 80
unassigned_cache = 0
is_local
failed_over
                          = True
                         = False
captive_storage
                         = False
Active Software
-AccessLogix
FLARE-Operating-Environment= 02.16.700.5.004
-NavisphereManager
Storage Processors
SP Identifier = A

signature = 1057303

microcode_version = 2.16.700.5.004

serial_num = LKE00040201171

prom_rev = 3.30.00

agent_rev = 6.16.0 (4.80)
                         = 3967
phys_memory
                         = 773
sys_buffer
```

```
read_cache
                      = 122
write_cache
                      = 3072
free_memory
                      = 0
                      = 0
raid3_mem_size
failed_over
                      = False
hidden
                      = False
network_name
                      = spa
                      = 172.24.102.5
ip_address
subnet mask
                      = 255.255.255.0
gateway_address
                     = 172.24.102.254
num_disk_volumes
                     = 20 - root_disk root_ldisk d3 d4 d5 d6 d7 d8 d9 d10 d11 d12
d13 d14 d15 d16 d17 d18 d19 d20
  Port Information
    Port 1
                      = 50:6:1:60:B0:60:1:CC:50:6:1:61:30:60:1:CC
    uid
    link_status
                      = UP
    port_status
                      = ONLINE
    switch_present
                      = True
    switch_uid
                      = 10:0:8:0:88:A0:36:F3:20:42:8:0:88:A0:36:F3
                      = 6373907
    sp_source_id
<...removed...>
    Port 2
                      = 50:6:1:60:B0:60:1:CC:50:6:1:62:30:60:1:CC
    uid
    link status
                      = UP
    port_status
                      = ONLINE
    switch_present
                      = True
    switch uid
                      = 10:0:8:0:88:A0:36:F3:20:41:8:0:88:A0:36:F3
    sp_source_id
                      = 6373651
SP Identifier
                      = B
                      = 1118484
signature
microcode_version
                    = 2.16.700.5.004
                      = LKE00041700812
serial num
prom_rev
                      = 3.30.00
agent_rev
                      = 6.16.0 (4.80)
phys_memory
                      = 3967
sys_buffer
                      = 773
                      = 122
read_cache
write_cache
                      = 3072
free_memory
                      = 0
raid3_mem_size
                      = 0
failed_over
                      = False
hidden
                      = False
network_name
                     = spb
ip_address
                     = 172.24.102.6
```

= 255.255.255.0

subnet\_mask

```
gateway_address
                      = 172.24.102.254
num_disk_volumes
 Port Information
   Port 1
   uid
                      = 50:6:1:60:B0:60:1:CC:50:6:1:69:30:60:1:CC
   link status
                      = UP
   port_status
                      = ONLINE
   switch_present
                     = True
                      = 10:0:8:0:88:A0:36:F3:20:3E:8:0:88:A0:36:F3
   switch uid
    sp_source_id
                      = 6372883
<...removed...>
   Port 2
   uid
                      = 50:6:1:60:B0:60:1:CC:50:6:1:6A:30:60:1:CC
   link_status
                      = UP
                     = ONLINE
   port_status
   switch_present
                     = True
                      = 10:0:8:0:88:A0:36:F3:20:3D:8:0:88:A0:36:F3
    switch_uid
   sp_source_id
                      = 6372627
Storage Groups
   id
                          = A4:74:8D:50:6E:A1:D9:11:96:E1:8:0:1B:43:5E:4F
   name
                          = ns704g-cs100
   num hbas
                          = 18
   num_devices
                         = 24
   shareable
                          = True
   hidden
                          = False
        Hosts
        uid
                          = 50:6:1:60:90:60:3:49:50:6:1:60:10:60:3:49
        storage_processor = B
        port
        server
                          = server_4
        uid
                          = 50:6:1:60:90:60:3:49:50:6:1:60:10:60:3:49
        storage_processor = A
        port
                          = 0
        server
                          = server_4
                          = 50:6:1:60:80:60:4:F0:50:6:1:61:0:60:4:F0
        uid
        storage_processor = B
        port
                          = server_2
        server
 <...removed...>
```

```
= 50:6:1:60:80:60:4:F0:50:6:1:68:0:60:4:F0
        uid
        storage_processor = B
        port
                          = 1
                          = server_3
        server
                          = 20:0:0:0:0:C9:2B:98:77:10:0:0:0:C9:2B:98:77
        uid
        storage_processor = B
        port
                          = 0
        uid
                          = 20:0:0:0:0:C9:2B:98:77:10:0:0:0:C9:2B:98:77
        storage_processor = A
        port
                          = 0
        ALU
              HLU
        0000 -> 0000
        0001 -> 0001
        0002 -> 0002
        0003 -> 0003
        0004 -> 0004
        0005 -> 0005
        0018 -> 0018
        0019 -> 0019
        0020 -> 0020
        0021 -> 0021
        0022 -> 0022
        0023 -> 0023
        0024 -> 0024
        0025 -> 0025
        0026 -> 0026
        0027 -> 0027
        0028 -> 0028
        0029 -> 0029
        0030 -> 0030
        0031 -> 0031
        0032 -> 0032
        0033 -> 0033
        0034 -> 0034
        0035 -> 0035
Disk Groups
                      = 0000
storage profiles
                      = 2 - clar_r5_performance,cm_r5_performance
                      = RAID5
raid_type
                      = 1068997528
logical_capacity
num_spindles
                      = 5 - 0_0_0 0_0_1 0_0_2 0_0_3 0_0_4
                     = 6 - 0000 0001 0002 0003 0004 0005
num_luns
num_disk_volumes
                     = 6 - root_disk root_ldisk d3 d4 d5 d6
spindle_type
                      = FC
```

id

bus

```
raw_capacity
                   = 1336246910
used_capacity
                  = 62914560
free_capacity
                  = 1006082968
hidden
                   = False
<...removed...>
                   = 2 0 14
product
                   = ST314670 CLAR146
revision
                   = 6A06
serial
                  = 3KS02RHM
capacity
                  = 280346624
used_capacity
                   = 224222822
                  = 0014
disk_group
hidden
                   = False
                    = FC
type
bus
                    = 2
enclosure
                   = 0
                   = 14
slot
vendor
                   = SEAGATE
remapped_blocks
                  = -1
state
                   = ENABLED
```

= 0

For the Celerra Network Server with a Symmetrix storage system, to display information for the attached storage system, type:

```
$ nas_storage -info 000187940260
id
                         = 1
serial_number
                         = 000187940260
name
                        = 000187940260
                       = Symmetrix
type
ident
                        = Symm6
model
                        = 800-M2
microcode_version = 5670
microcode_version_num = 16260000
microcode_date = 03012004
microcode_patch_level = 69
microcode_patch_date = 03012004
symmetrix_pwron_time = 1130260200 == Tue Oct 25 13:10:00 EDT 2005
db_sync_time = 1133215405 == Mon Nov 28 17:03:25 EST 2005
db_sync_bcv_time = 1133215405 == Mon Nov 28 17:03:25 EST 2005
db_sync_rdf_time = 1133215405 == Mon Nov 28 17:03:25 EST 2005
last_ipl_time = 1128707062 == Fri Oct 7 13:44:22 EDT 2005
last_fast_ipl_time = 1130260200 == Tue Oct 25 13:10:00 EDT 2005
API_version = V6.0-629
cache_size
                       = 32768
cache_slot_count
                       = 860268
max_wr_pend_slots = 180000
max_da_wr_pend_slots = 90000
max_dev_wr_pend_slots = 6513
permacache_slot_count = 0
```

```
num_disks
                      = 60
num_symdevs
                      = 378
num_pdevs
                      = 10
sddf_configuration = ENABLED config_checksum = 0x01ca544
num_powerpath_devs
                     = 0
config_crc
                      = 0x07e0ba1e6
is_local
                       = True
Physical Devices
/nas/dev/c0t0l15s2
/nas/dev/c0t0115s3
/nas/dev/c0t0l15s4
/nas/dev/c0t0l15s6
/nas/dev/c0t0l15s7
/nas/dev/c0t0115s8
```

/nas/dev/c16t0115s2 /nas/dev/c16t0115s3 /nas/dev/c16t0115s4 /nas/dev/c16t0115s8

#### Director Table

type	num	slot	ident	stat	scsi	vols	ports	p0_stat	p1_stat	p2_stat	p3_stat
DA	1	1	DF-1A	On	NA	21	2	On	On	NA	NA
DA	2	2	DF-2A	On	NA	8	2	On	On	NA	NA
DA	15	15	DF-15A	On	NA	21	2	On	On	NA	NA
DA	16	16	DF-16A	On	NA	8	2	On	On	NA	NA
DA	17	1	DF-1B	On	NA	8	2	On	On	NA	NA
DA	18	2	DF-2B	On	NA	21	2	On	On	NA	NA
DA	31	15	DF-15B	On	NA	152	2	On	On	NA	NA
DA	32	16	DF-16B	On	NA	165	2	On	On	NA	NA
FA	33	1	FA-1C	On	NA	0	2	On	On	NA	NA
FA	34	2	FA-2C	On	NA	0	2	On	On	NA	NA
FA	47	15	FA-15C	On	NA	0	2	On	On	NA	NA
FA	48	16	FA-16C	On	NA	0	2	On	On	NA	NA
FA	49	1	FA-1D	On	NA	0	2	On	On	NA	NA

**Note:** This is a partial listing due to the length of the outputs.

# **EXAMPLE #3** To rename a storage system, type:

```
$ nas_storage -rename APM00042000818 cx700_1
```

```
id = 1
serial_number = APM00042000818
name = cx700_1
acl = 0
```

## **EXAMPLE #4** To set the access control level for the storage system **cx700\_1**, type:

\$ nas\_storage -acl 1000 cx700\_1

```
id
        = 1
serial_number = APM00042000818
name = cx700 1
acl
         = 1000, owner=nasadmin, ID=201
                        Note: The value 1000 specifies nasadmin as the owner and gives read, write,
                        and delete access only to nasadmin.
         EXAMPLE #5
                        To change the existing password on the CLARiiON system, type:
                        $ nas_storage -modify APM00070204288 -security -username
                        nasadmin -password nasadmin -newpassword abc
Changing password on APM00070204288
         EXAMPLE #6
                        To avoid specifying passwords in clear text on the command line,
                        type:
                        $ nas_storage -modify APM00070204288 -security
                        -newpassword
Enter the Global CLARiiON account information
Username: nasadmin
Password: *** Retype your response to validate
Password: ***
New Password
Password: ****** Retype your response to validate
Password: ******
Changing password on APM00070204288
Done
         EXAMPLE #7
                        To failback a CLARiiON system, type:
                        $ nas_storage -failback cx700_1
         = 1
serial number = APM00042000818
name = cx700_1
acl
         = 1000, owner=nasadmin, ID=201
         EXAMPLE #8
                        To display information for a CLARiiON system and turn
                        synchronization off, type:
                        $ nas_storage -info cx700_1 -option sync=no
id
                      = APM00042000818
arrayname
                      = cx700_1
name
type
                      = Clariion
model_type
                     = RACKMOUNT
model_num
                     = 700
db_sync_time
                     = 1131986667 == Mon Nov 14 11:44:27 EST 2005
API_version
                     = V6.0-629
num_disks
                      = 60
```

= 34

num\_devs

```
num_pdevs
                     = 8
                     = 1
num_storage_grps
num_raid_grps
                     = 16
                     = 8
cache_page_size
wr_cache_mirror
                    = True
low watermark
                    = 60
high_watermark
                    = 80
                     = 0
unassigned_cache
is local
                     = True
failed_over
                     = False
captive_storage
                     = False
Active Software
-AccessLogix
                     = -
FLARE-Operating-Environment= 02.16.700.5.004
-NavisphereManager
Storage Processors
SP Identifier
                    = A
                     = 1057303
signature
microcode_version
                   = 2.16.700.5.004
serial_num
                    = LKE00040201171
prom_rev
                    = 3.30.00
                     = 6.16.0 (4.80)
agent_rev
phys_memory
                    = 3967
sys_buffer
                    = 773
read_cache
                     = 122
                    = 3072
write_cache
free_memory
                    = 0
raid3 mem size
                     = 0
failed_over
                    = False
hidden
                    = False
network_name
                     = spa
ip_address
                     = 172.24.102.5
subnet_mask
                     = 255.255.255.0
gateway address
                     = 172.24.102.254
num_disk_volumes
                    = 20 - root_disk root_ldisk d3 d4 d5 d6 d7 d8 d9 d10
d11 d12 d13 d14 d15 d16 d17 d18 d19 d20
  Port Information
    Port 1
                     = 50:6:1:60:B0:60:1:CC:50:6:1:61:30:60:1:CC
   uid
    link_status
                     = UP
   port_status
                     = ONLINE
    switch_present
                    = True
    switch_uid
                    = 10:0:8:0:88:A0:36:F3:20:42:8:0:88:A0:36:F3
    sp_source_id
                     = 6373907
```

```
<...removed...>
   Port 2
   uid
                     = 50:6:1:60:B0:60:1:CC:50:6:1:62:30:60:1:CC
   link_status
                     = UP
                     = ONLINE
   port_status
   switch_present
                    = True
   switch_uid
                     = 10:0:8:0:88:A0:36:F3:20:41:8:0:88:A0:36:F3
    sp_source_id
                     = 6373651
SP Identifier
                     = B
signature
                     = 1118484
microcode_version = 2.16.700.5.004
serial_num
                   = LKE00041700812
prom_rev
                    = 3.30.00
                    = 6.16.0 (4.80)
agent_rev
phys_memory
                    = 3967
                     = 773
sys_buffer
read_cache
                     = 122
write_cache
                    = 3072
                     = 0
free_memory
                     = 0
raid3_mem_size
failed_over
                     = False
hidden
                     = False
network_name
                     = spb
ip_address
                     = 172.24.102.6
                     = 255.255.255.0
subnet_mask
gateway_address
                     = 172.24.102.254
num_disk_volumes
                    = 0
  Port Information
    Port 1
   uid
                     = 50:6:1:60:B0:60:1:CC:50:6:1:69:30:60:1:CC
   link_status
                     = UP
                     = ONLINE
   port status
   switch_present = True
                    = 10:0:8:0:88:A0:36:F3:20:3E:8:0:88:A0:36:F3
   switch_uid
    sp_source_id
                     = 6372883
 <...removed...>
   Port 2
   uid
                     = 50:6:1:60:B0:60:1:CC:50:6:1:6A:30:60:1:CC
    link_status
                     = UP
   port_status
                     = ONLINE
   switch_present
                     = True
   switch_uid
                     = 10:0:8:0:88:A0:36:F3:20:3D:8:0:88:A0:36:F3
```

```
sp\_source\_id = 6372627
Storage Groups
   id
                         = A4:74:8D:50:6E:A1:D9:11:96:E1:8:0:1B:43:5E:4F
   name
                        = ns704q-cs100
                        = 18
   num_hbas
   num_devices
                         = 24
   shareable
                         = True
   hidden
                         = False
       Hosts
                         = 50:6:1:60:90:60:3:49:50:6:1:60:10:60:3:49
       uid
       storage_processor = B
       port
                         = 1
       server
                         = server_4
       uid
                         = 50:6:1:60:90:60:3:49:50:6:1:60:10:60:3:49
       storage_processor = A
                         = 0
       port
       server
                         = server_4
                         = 50:6:1:60:80:60:4:F0:50:6:1:61:0:60:4:F0
       uid
       storage_processor = B
                         = 0
       port
       server
                         = server_2
 <...removed...>
       uid
                         = 50:6:1:60:80:60:4:F0:50:6:1:68:0:60:4:F0
       storage_processor = B
                         = 1
       port
       server
                         = server_3
       uid
                         = 20:0:0:0:0:C9:2B:98:77:10:0:0:0:C9:2B:98:77
       storage\_processor = B
       port
                         = 20:0:0:0:C9:2B:98:77:10:0:0:0:C9:2B:98:77
       uid
       storage_processor = A
                         = 0
       port
       ALU HLU
        _____
       0000 -> 0000
       0001 -> 0001
       0002 -> 0002
       0003 -> 0003
       0004 -> 0004
       0005 -> 0005
       0018 -> 0018
```

id

bus

id

bus

```
0019 -> 0019
       0020 -> 0020
       0021 \rightarrow 0021
       0022 -> 0022
       0023 -> 0023
       0024 -> 0024
       0025 -> 0025
       0026 -> 0026
       0027 -> 0027
       0028 -> 0028
       0029 -> 0029
       0030 -> 0030
       0031 -> 0031
       0032 -> 0032
       0033 -> 0033
       0034 -> 0034
       0035 -> 0035
Disk Groups
                     = 0000
storage profiles
                   = 2 - clar_r5_performance,cm_r5_performance
raid_type
                    = RAID5
logical_capacity
                   = 1068997528
num_spindles
                   = 5 - 0_0_0 0_0_1 0_0_2 0_0_3 0_0_4
                   = 6 - 0000 0001 0002 0003 0004 0005
num_luns
num_disk_volumes = 6 - root_disk root_ldisk d3 d4 d5 d6
spindle_type
                   = FC
                    = 0
                   = 1336246910
raw_capacity
used_capacity
                   = 62914560
free_capacity
                   = 1006082968
hidden
                    = False
<...removed...>
                    = 0205
storage profiles
                   = 0
                    = SPARE
raid type
logical_capacity = 622868992
num_spindles
                   = 1 - 0 1 0
                    = 1 - 0205
num_luns
num_disk_volumes
                   = 0
spindle_type
                   = ATA
                    = 0
raw_capacity
                   = 622868992
used_capacity
                   = 622868992
free_capacity
                   = 0
hidden
                     = False
```

Spindles

```
id
                  = 0_0_0
product
                  = ST314670 CLAR146
revision
                  = 6A06
serial
                  = 3KS088SQ
capacity
                  = 280346624
used_capacity
                  = 12582912
disk_group
                   = 0000
hidden
                  = False
type
                  = FC
                   = 0
bus
                  = 0
enclosure
slot
                  = 0
vendor
                   = SEAGATE
remapped_blocks
                  = -1
state
                  = ENABLED
<...removed...>
                   = 2_0_14
id
product
                  = ST314670 CLAR146
revision
                  = 6A06
serial
                  = 3KS02RHM
capacity
                  = 280346624
used_capacity
                  = 224222822
disk_group
                  = 0014
hidden
                   = False
type
                  = FC
                   = 2
bus
enclosure
                   = 0
slot
                  = 14
vendor
                  = SEAGATE
remapped_blocks
                  = -1
state
                   = ENABLED
```

**Note:** This is a partial display due to the length of the outputs.

# **EXAMPLE #9** To delete a storage system with no attached disks, type:

\$ nas\_storage -delete APM00035101740

```
id = 0
serial_number = APM00035101740
name = APM00035101740
acl = 0
```

# **EXAMPLE #10** To turn synchronization on for all systems, type:

\$ nas\_storage -sync -all

done

## **EXAMPLE #11** To perform a health check on the storage system, type:

\$ nas\_storage -check -all

Discovering storage (may take several minutes)

done

## EXAMPLE #12

To set the access control level for the storage system APM00042000818, type:

\$ nas\_storage -acl 1432 APM00042000818

id = 1

serial\_number = APM00042000818

name = APM00042000818

acl = 1432, owner=nasadmin, ID=201

**Note:** The value **1432** specifies **nasadmin** as the owner and gives users with an access level of at least **observer** read access only, users with an access level of at least **operator** read/write access, and users with an access level of at least **admin** read/write/delete access.

#### **EXAMPLE #13**

To modify the IP address of the CLARiiON storage system on the Celerra Network Server, type:

\$ nas\_storage -modify APM00072303347 -network -spa 10.6.4.225

Changing IP address for APM00072303347 Discovering storage (may take several minutes) done

# nas\_task

Manages in-progress or completed tasks.

#### **SYNOPSIS**

```
nas_task
  -list [-remote_system {<remoteSystemName>|id=<id>}]
| -info {-all|<taskId>}
     [-remote_system {<remoteSystemName>|id=<id>}]
| -abort <taskId>
     [-mover <moverName>][-remote_system {<remoteSystemName>|id=<id>}]
| -delete <taskId>
        -remote_system {<remoteSystemName>|id=<id>}]
```

#### **DESCRIPTION**

nas\_task lists the tasks associated with commands currently in progress or completed, reports information about a particular task, aborts a task, or deletes a task. Each task can be uniquely identified by its task ID and the remote Celerra system name or ID.

Use the **nas\_task** command to monitor, abort, and delete long running tasks and tasks started in asynchronous mode.

#### OPTIONS -

-list

Lists all local tasks that are in progress, or completed tasks that have not been deleted. For each task, lists the task ID, remote system name, a description of the task, and the task state (running, recovering, succeeded, or failed).

-remote\_system {<remoteSystemName>|id=<id>}
Lists local tasks initiated by the specified remote Celerra system.
Specify the remote system name or ID.

```
-info {-all|<taskId>}
```

Provides more detailed status information for all tasks or for a particular task. Displays the run time status, estimated completion time, and percent complete for running tasks. Displays the completion status and actual end time for completed tasks.

The taskID is the ID returned from a command run in the background mode or from the **nas\_task** -list command.

**Note:** The ID of the task is an integer and is assigned automatically. The task ID is unique to the Celerra server.

```
[-remote_system {<remoteSystemName>|id=<id>}]
```

Provides more detailed status information of local tasks initiated by the specified remote Celerra system. Specify the remote system name or remote system ID. The remote system name is returned from the **nas\_task-list** command.

#### -abort <taskId>

Aborts the specified task leaving the system in a consistent state. For example, it aborts a one-time copy in progress. This might take a long time to complete because a remote system may be unavailable or the network may be down. You should check the status of the task to verify that the task was aborted. This command can be executed from the source only.



#### **CAUTION**

This option might leave the system in an inconsistent state. Use caution when using this option.

```
[-mover <moverName>]
```

Aborts a task running locally on the specified Data Mover. This command can be executed from the source or destination side. Use this command when the source and destination Celerra systems cannot communicate. You should run this command on both sides.

```
[-remote_system {<remoteSystemName>|id=<id>]
Aborts a task that was initiated on a remote Celerra leaving the
```

source side intact. Specify the Data Mover to abort a task from the destination side. Specify the Data Mover and remote system name or remote system id along with the task id.

```
-delete id <taskId>
```

Based on the task ID, deletes a completed task from the database on the Control Station.

```
[-remote_system {<remoteSystemName>|id=<id>]
Deletes a task that was initiated on a remote Celerra. Specify the remote system name or remote system id along with the task id.
```

**SEE ALSO** *Using Celerra Replicator (V2)*, nas\_copy, nas\_replicate, and nas\_cel.

## **EXAMPLE #1** To display detailed information about the task with taskID **4241**, type:

```
$ nas_task -info 4241
Task Id = 4241
Celerra Network Server = cs100
```

Task State = Running

Percent Complete = 95

Description = Create Replication ufs1\_replica1.

Originator = nasadmin@cli.localhost

Start Time = Mon Dec 17 14:21:35 EST 2007 Estimated End Time = Mon Dec 17 19:24:21 EST 2007

Schedule = n/a

#### Where:

Value	Definition
Task Id	Globally unique character string used as the identifier of the task.
Celerra Network Server	When set, local.
Remote Task Id	When set, identifies a remote task.
State	Running, Recovering, Completed, or Failed. Running could be a combination of completed and failed.
Current Activity	Displays state property when available.
Percent Completed	Appears only when set and not complete.
Description	Appears if details are set.
Originator	User or host that initiated the task.
Start Time/End Time	The starting time and ending time (or status) for the task.
Estimated End Time	Appears instead of previous line when available and task is incomplete.
Schedule	The schedule in effect, or n/a for a task that is not a scheduled checkpoint refresh.
Response Statuses	Displayed list of messages, if any. A completed task should always have one.

## **EXAMPLE #2** To display the list of all tasks, type:

#### \$ nas\_task -list

Task State Originator Start Time escription Schedule Remote System 4241 Running nasadmin@cli+ Mon Dec 17 14:21:35 EST 2007 Create Replication ufs1\_r+ cs100 4228 Succeeded nasadmin@cli+ Mon Dec 17 14:04:02 EST 2007 Delete task NONE: 4214. cs100 Failed nasadmin@cli+ Mon Dec 17 13:59:26 EST 2007 Create Replication 4177 ufs1\_r+ cs100 4150 Succeeded nasadmin@cli+ Mon Dec 17 13:55:39 EST 2007 Delete task NONE: 4136. cs100 4127 Succeeded nasadmin@cli+ Mon Dec 17 11:38:32 EST 2007 Delete task NONE: 4113. cs100 4103 Succeeded nasadmin@cli+ Mon Dec 17 11:21:00 EST 2007 Delete task NONE: 4098. 4058 Succeeded nasadmin@cli+ Fri Dec 14 16:43:23 EST 2007 Switchover Replication NONE. cs100 2277 Succeeded nasadmin@cli+ Fri Dec 14 16:42:08 EST 2007 Reverse Replication cs110 NONE. 2270 Succeeded nasadmin@cli+ Fri Dec 14 16:40:29 EST 2007 Start Replication NONE. cs110

#### The nas Commands

2265 Failed nasadmin@cli+ Fri Dec 14 16:40:11 EST 2007 Start Replication NONE. cs110

EXAMPLE #1 provides a description of the outputs.

**EXAMPLE #3** To abort task **4267** running locally on server\_3, type:

\$ nas\_task -abort 4267 -mover server\_3

OK

**EXAMPLE #4** To delete the existing task **4267**, type:

\$ nas\_task -delete 4267

OK

# nas\_version

Displays the software version running on the Control Station.

#### **SYNOPSIS**

nas\_version [-h|-1]

#### **DESCRIPTION**

**nas\_version** displays the Control Station version in long form or short form. When used during a software upgrade, informs the user about the upgrade in progress.

#### OPTIONS N

**No arguments**Displays the software version running on the Control Station.

-h

Displays command usage.

-1

Displays detailed software version information for the Control Station.

#### **EXAMPLE #1**

To display the software version running on the Control Station during a software upgrade, type:

\$ nas\_version

5.6.25-0

#### **EXAMPLE #2**

To display the system output during a software upgrade, type:

\$ nas\_version

5.6.19-0

Warning!!Upgrade is in progress from 5.6.19-0 to 5.6.20-0 Warning!!Please log off IMMEDIATELY if you are not upgrading the Celerra

**EXAMPLE #3** To display the usage for nas\_version, type:

#### \$ nas\_version -h

usage: /nas/bin/nas\_version [-h|-l]

-h help

-l long\_format

#### **EXAMPLE #4**

To display detailed software version information for the Control Station, type:

#### \$ nas\_version -1

Name : emcnas Relocations: /nas

Version : 5.6.19 Vendor: EMC

Release : 0 Build Date: Tue 19 Dec 2006 08:53:31 PM EST

Size : 454239545 License: EMC Copyright

#### **The nas Commands**

Signature : (none)

Packager : EMC Corporation
URL : http://www.emc.com
Summary : EMC nfs base install
Description : EMC nfs base install

# **EXAMPLE #5** To display detailed software version information for the Control Station during a software upgrade, type:

#### \$ nas\_version -1

Name : emcnas Relocations: /nas

Version : 5.6.19 Vendor: EMC

Release : 0 Build Date: Wed 14 Mar 2007 12:36:55 PM EDT

Size : 500815102 License: EMC Copyright

Signature : (none)

Packager : EMC Corporation
URL : http://www.emc.com
Summary : EMC nfs base install
Description : EMC nfs base install

Warning!!Upgrade is in progress from 5.6.19-0 to 5.6.20-0

Warning!!Please log off IMMEDIATELY if you are not upgrading the Celerra

# nas\_volume

Manages the volume table.

#### **SYNOPSIS**

#### DESCRIPTION

nas\_volume creates metavolumes and stripe volumes and lists, renames, extends, clones, and deletes metavolumes, stripe, and slice volumes. nas\_volume sets an access control value for a volume, and displays detailed volume attributes, including the total size of the volume configuration.

#### **OPTIONS**

#### -list

Displays the volume table.

**Note:** The ID of the object is an integer and is assigned automatically. The name of the volume may be truncated if it is more than 17 characters. To display the full name, use the **-info** option with the volume ID.

```
-delete <volume_name>
```

Deletes the specified volume.

```
-info [-size] {-all|<volume_name>} [-tree]
```

Displays attributes and the size for all volumes, or the specified <volume\_name>. The **-tree** option recursively displays the volume set, that is, the list of component volumes for the specified volume or all volumes.

```
-rename <old_name> <new_name>
```

Changes the current name of a volume to a new name.

```
-size <volume_name>
```

Displays the total size in MB of the <volume\_name>, including used and available space.

```
-acl <acl value> <volume name>
```

Sets an access control level value that defines the owner of the volume, and the level of access allowed for users and groups defined in the access control level table. The <a href="mailto:nas\_acl">nas\_acl</a> command provides information.

```
-xtend <volume_name> {<volume_name>,...}
```

Extends the specified metavolume by adding volumes to the configuration. The total size of the metavolume increases by the sum of all the volumes added.

**Note:** Only metavolumes can be extended. The volume that was added remains in use until the original metavolume is deleted. Volumes containing mounted filesystems cannot be extended using this option. The nas\_fs command provides information to extend a volume that is hosting a mounted filesystem.

```
-create {<volume_name>,...}
```

Creates a volume configuration from the specified volumes. Unless otherwise specified, volumes are automatically created as metavolumes.

```
[-name <name>]
```

Assigns a <name> to volume. If a name is not specified, one is assigned automatically. The name of a volume is case-sensitive.

```
[-Stripe <stripe_size>|-Meta]
```

Sets the type for the volume to be either a stripe volume or metavolume (default). If **-Stripe** is specified, a stripe size must be typed in multiples of 8192 bytes with a recommended size of 32,768 bytes (default) for NFS and CIFS environments. A 256 KB stripe size is recommended for EMC HighRoad<sup>®</sup> environments; while RAID 3 and 64 KB stripe size are recommended for ATA-based filesystems.

nas\_slice provides information to create a slice volume.

```
[-Force] {<volume_name>,...}
```

Forces the creation of a volume on a mixed storage system.

```
-Clone <volume_name>
```

Creates an exact clone of the specified <volume\_name>. Volumes can be cloned from slice, stripe, or metavolumes. The name automatically assigned to the clone is derived from the ID of the volume.

```
[{<svol>:<dvol>,...}]
```

Sets a specific disk volume set for the source volume and the destination volume. The size of the destination volume must be the same as the source volume.

## -option disktype=<type>

Specifies the type of disk to be created.

Disk types when using CLARiiON are CLSTD, CLEFD, and CLATA, and for CLARiiON storage systems involving mirrored disks are: CMEFD, CMSTD, and CMATA.

Disk types when using a Symmetrix are STD, R1STD, R2STD, BCV, R1BCV, R2BCV, ATA, R1ATA, R2ATA, BCVA, R1BCA, R2BCA, and EFD.

#### SEE ALSO

Managing Celerra Volumes and File Systems with Automatic Volume Management, Managing Celerra Volumes and File Systems Manually, Using Time Finder/FS, Near Copy, and Far Copy with Celerra, Controlling Access to Celerra System Objects, nas\_slice, nas\_disk, nas\_acl, and nas\_fs.

# **EXAMPLE #1** To list all volumes, type:

				<pre>\$ nas_volume</pre>	-list	
id	inuse	type	acl	name	cltype	clid
1	У	4	0	root_disk	0	1,2,3,4,5,6,7,8,9,10,11,
						12,13,14,15,16,17,18,19,20,
						21,22,23,24,25,26,27,28,29,
						30,31,32,33,34,51
2	У	4	0	root_ldisk	0	35,36,37,38,39,40,41,42,
	_			_		43,44,45,46,47,48,49,50,52
3	У	4	0	d3	1	76
4	У	4	0	d4	1	77
5	У	4	0	d5	1	78
6	У	4	0	d6	1	79
7	n	1	0	root_dos	0	
8	n	1	0	root_layout	0	
9	У	1	0	root_slice_1	1	10
10	У	3	0	root_volume_1	2	1
11	У	1	0	root slice 2	1	12
12	У	3	0	root_volume_2	2	2
13	_	1	0	root slice 3	1	14
10	У		U	1000_51106_3	Т.	14

**Note:** This is a partial listing due to the length of the outputs.

#### Where:

Value	Definition
id	ID of the volume.
inuse	Whether the volume is used.
type	Type assigned to the volume. Available types are: 1=slice, 2=stripe, 3=meta, 4=disk, and 100=pool.
acl	Access control level assigned the volume.
name	Name assigned to the volume.
cltype	The client type of the volume. Available values are:
	0 — If the clid field is not empty then the client is a slice.
	1 — The client is another volume (meta, stripe, volume_pool).
	• 2 — The client is a filesystem.
clid	ID of the client.

# **EXAMPLE #2** To create a metavolume named, **mtv1**, on disk volume, **d7**, type:

\$ nas\_volume -name mtv1 -create d7

#### Where:

Value	Definition	
id	ID of the volume.	
name	Name assigned to the volume.	
acl	Access control level value assigned to the volume.	
in_use	Whether the volume is used.	
type	Type assigned to the volume. Types are meta, stripe, slice, disk, and pool.	
volume_set	Name assigned to the volume.	
disks	Disks used to build a filesystem.	

# **EXAMPLE #3** To display configuration information for **mtv1**, type:

\$ nas\_volume -info mtv1

### **EXAMPLE #4** To ren

To rename a **mtv1** to **mtv2**, type:

\$ nas\_volume -rename mtv1 mtv2

#### **EXAMPLE #5**

To create a stripe volume named, **stv1**, with a size of **32768** bytes on disk volumes **d10**, **d12**, **d13**, and **d15**, type:

\$ nas\_volume -name stv1 -create -Stripe 32768
d10,d12,d13,d15

volume\_set = d10,d12,d13,d15 disks = d10,d12,d13,d15

#### Where:

Value	Definition
stripe_size	Specified size of the stripe volume.

# **EXAMPLE #6** To clone **mtv1**, type:

\$ nas\_volume -Clone mtv1

```
id
            = 146
name
            = mtv1
acl
            = False
in_use
type
            = meta
volume set = d7
disks
            = d7
id
            = 148
            = v148
name
acl
            = False
in_use
            = meta
type
volume_set = d8
disks
            = d8
```

# **EXAMPLE #7**

To clone the volume **mtv1** and set the disk type to **BCV**, type:

\$ /nas/sbin/rootnas\_volume -Clone mtv1 -option
disktype=BCV

id = 322name = mtv1acl = 0in\_use = False type = meta  $volume\_set = d87$ disks = d87= 323 id = v323name acl = 0 in\_use = False = meta type volume\_set = rootd99 disks = rootd99

# **EXAMPLE #8** To extend **mtv1** with **mtv2**, type:

\$ nas\_volume -xtend mtv1 mtv2

# **EXAMPLE #9** To display the size of **mtv1**, type:

\$ nas\_volume -size mtv1

total = 547418 avail = 547418 used = 0 (0%) (sizes in MB)

### Where:

Value	Definition
total	Total size of the volume.
avail	Amount of unused space on the volume.
used	Amount of space used on the volume.

### **EXAMPLE #10**

To set the access control level for the metavolume **mtv1**, type:

\$ nas\_volume -acl 1432 mtv1

**Note:** The value **1432** specifies **nasadmin** as the owner and gives users with an access level of at least observer read access only, users with an access level of at least operator read/write access, and users with an access level of at least admin read/write/delete access.

#### EXAMPLE #11

To delete mtv2, type:

\$ nas\_volume -delete mtv1

id = 146 name = mtv1

acl = 1432, owner=nasadmin, ID=201

 $\begin{array}{lll} \text{in\_use} & = & \text{False} \\ \text{type} & = & \text{meta} \\ \text{volume\_set} & = & \text{d7,mtv2} \\ \text{disks} & = & \text{d7,d8} \\ \end{array}$ 

The nas Commands	

# The server Commands

This chapter lists the Celerra Network Server Command Set provided for managing, configuring, and monitoring Data Movers. The commands are prefixed with server and appear alphabetically. The command line syntax (Synopsis), a description of the options, and an example of usage are provided for each command. Topics included are:

- server archive
- server\_arp
- ◆ server\_certificate
- server cdms
- server\_cepp
- server\_checkup
- server\_cifs
- server\_cifssupport
- server\_cpu
- server date
- server\_dbms
- server\_devconfig
- server df
- server\_dns
- server\_export
- server\_extlog
- server\_file
- server\_ftp
- server\_http
- server\_ifconfig
- server\_ip
- server\_iscsi

- server\_kerberos
- server\_ldap
- server\_log
- server\_mount
- server\_mountpoint
- server\_mpfs
- server\_mt
- server\_name
- server\_netstat
- server\_nfs
- server\_nis
- ◆ server\_param
- server\_pax
- server\_ping
- server\_ping6
- server\_rip
- server\_route
- server\_security
- server\_setup
- server\_snmpd
- server\_ssh
- server\_standby
- server\_stats
- server\_sysconfig
- server\_sysstat
- server\_tftp
- server\_umount
- server\_uptime
- server\_user
- server\_usermapper
- server\_version
- server\_viruschk
- server\_vtlu

# server\_archive

Reads and writes file archives, and copies directory hierarchies.

# **SYNOPSIS**

```
server\_archive < movername > [-cdnvN] - f < archive_file > [-J [p] [w | d | u]]
  [-I <client_dialect>]
            [-e <archive_name>][-s <replstr>] ...
            [-T [<from_date>][,<to_date>]][<pattern>] ...
  server_archive <movername> -r [-cdiknuvDNYZ][-E <limit>]
            [-J [w|d|u]][-C d|i|m][-I < client_dialect>]
            [-f <file_name>][-e <archive_name>]
            [-p <string>] ... [-s <replstr>] ...
            [-T [<from_date>][,<to_date>]] ... [<pattern> ...]
  server_archive <movername> -w [-dituvLNPX]
            [-J [w|d|u]][-I <client_dialect>]
            [-b <block_size>][-f <file_name>][-e <archive_name>]
            [-x <format>][-B bytes][-s <replstr>] ...
            [-T [<from_date>][,<to_date>][/[c][m]]] ...
            [[-0]|[-1]][<file> ...]
  server_archive -r -w [-diklntuvDLPXYZ]
            [-J [w|d|u]][-C d|i|m]
            [-p <string>] ... [-s <replstr>] ...
            [-T [<from_date>][,<to_date>][/[c][m]]] ...
            [<file> ...] <directory>
```

# **DESCRIPTION**

**server\_archive** reads, writes, and lists the members of an archive file, and copies directory hierarchies. The **server\_archive** operation is independent of the specific archive format, and supports a variety of different archive formats.

**Note:** A list of supported archive formats can be found under the description of the **-x** option.

The presence of the **-r** and the **-w** options specifies the following functional modes: list, read, write, and copy.

```
-List (no arguments)
```

**server\_archive** writes a table of contents of the members of the archive file read from archive to the standard output, whose pathnames match the specified patterns.

**Note:** If no options are specified, **server\_archive** lists the contents of the archive.

### Read (-r)

**server\_archive** extracts the members of the archive file, read from the archive, with pathnames matching the specified patterns. The archive format and blocking are automatically determined on input. When an extracted file is a directory, the entire file hierarchy rooted at that directory is extracted.

**Note:** Ownership, access, modification times, and file mode of the extracted files are discussed in more detail under the **-p** option.

### Write (-w)

**server\_archive** writes an archive containing the file operands to archive using the specified archive format. When a file operand is also a directory, the entire file hierarchy rooted at that directory is included.

## Copy (-r -w)

**server\_archive** copies the file operands to the destination directory. When a file operand is also a directory, the entire file hierarchy rooted at that directory is included. The effect of the copy is as if the copied files were written to an archive file and then subsequently extracted, except that there may be hard links between the original and the copied files. The **-1** option provides more information.



# **CAUTION**

The destination directory must exist and must not be one of the file operands or a member of a file hierarchy rooted at one of the file operands. The result of a copy under these conditions is unpredictable.

While processing a damaged archive during a read or list operation, **server\_archive** attempts to recover from media defects and searches through the archive to locate and process the largest number of archive members possible (the **-E** option provides more details on error handling).

# **OPERANDS**

The directory operand specifies a destination directory pathname. If the directory operand does not exist, or if it is not writable by the user, or if it is not a directory name, **server\_archive** exits with a non-zero exit status.

The pattern operand is used to select one or more pathnames of archive members. Archive members are selected using the pattern matching notation described by **fnmatch 3**. When the pattern

operand is not supplied, all members of the archive are selected. When a pattern matches a directory, the entire file hierarchy rooted at that directory is selected. When a pattern operand does not select at least one archive member, **server\_archive** writes these pattern operands in a diagnostic message to standard error and then exits with a non-zero exit status.

The file operand specifies the pathname of a file to be copied or archived. When a file operand does not select at least one archive member, **server\_archive** writes these file operand pathnames in a diagnostic message to standard error and then exits with a non-zero exit status.

The archive\_file operand is the name of a file where the data is stored (write) or read (read/list). The archive\_name is the name of the streamer on which the data will be stored (write) or read (read/list).

Note: To obtain the device name, you can use server\_devconfig -scsi.

# **OPTIONS**

The following options are supported:

-r

Reads an archive file from an archive and extracts the specified files. If any intermediate directories are needed to extract an archive member, these directories will be created as if **mkdir 2** was called with the bit-wise inclusive OR of S\_IRWXU, S\_IRWXG, and S\_IRWXO, as the mode argument. When the selected archive format supports the specification of linked files and these files cannot be linked while the archive is being extracted, **server\_archive** writes a diagnostic message to standard error and exits with a non-zero exit status at the completion of operation.

-w

Writes files to an archive in the specified archive format.

-0 (zero)

With this option, a full referenced backup is performed with the time and date of launching is set in a reference file. This reference file is an ASCII file and is located in /.etc/BackupDates. The backup is referenced by the pathname of the files to back up and the time and date when the backup was created. This file is updated only if the backup is successful.

Backup files can be copied using the server\_file command.

#### -<x>

Level x (x=1-9) indicates a backup of all files in a filesystem that have been modified since the last backup of a level smaller than the previous backup.

For example, a backup is performed for:

Monday: level 0 = full backup Tuesday: level 3 = files modified since Monday Friday: level 5 = files modified since Tuesday Saturday: level 4 = files modified since Tuesday Sunday: level 4 = files modified since Tuesday

**Note:** If the backup type is not indicated, a full backup is performed automatically.

-b <block\_size>

When writing an archive, blocks the output at a positive decimal integer number of bytes per write to the archive file. The <br/>
<b

Note: To remain POSIX-compatible, do not exceed 32256 Bytes.

A <block\_size> can end with k or b to specify multiplication by 1024 (1K) or 512, respectively. A pair of <block\_size> can be separated by x to indicate a product. A specific archive device may impose additional restrictions on the size of blocking it will support. When blocking is not specified, the default for <block\_size> is dependent on the specific archive format being used. The -x option provides more information.

#### -c

Matches all file or archive members except those specified by the pattern and file operands.

### -đ

Causes files of type directory being copied or archived, or archive members of type directory being extracted, to match only the directory file or archive member, and not the file hierarchy rooted at the directory.

-e <archive\_name>

Specifies the archive name when it is streamed.

**Note:** To prevent the tape from rewinding at the end of command execution, use the **-N** option with the **-e** <archive\_name> option.

# -f <archive\_file>

Specifies the archive name when it is a file.

**Note:** A single archive may span multiple files and different archive devices. When required, **server\_archive** prompts for the pathname of the file or device of the next volume in the archive.

### -i

Interactively renames files or archive members. For each archive member matching a pattern operand, or each file matching a file operand, server\_archive prompts to /dev/tty giving the name of the file, its file mode, and its modification time. Then server\_archive reads a line from /dev/tty. If this line is blank, the file or archive member is skipped. If this line consists of a single period, the file or archive member is processed with no modification to its name. Otherwise, its name is replaced with the contents of the line. The server\_archive immediately exits with a non-zero exit status if <EOF> is encountered when reading a response or if /dev/tty cannot be opened for reading and writing.

### -k

Does not allow overwriting to existing files.

### -1

Links files. In the copy mode (-r, -w), hard links are made between the source and destination file hierarchies whenever possible.

# -I <client\_dialect>

Allows filename information recovered from an archive to be translated into UTF-8.

### -n

Selects the first archive member that matches each pattern operand. No more than one archive member is matched for each pattern. When members of type directory are matched, the file hierarchy rooted at that directory is also matched (unless **-d** is also specified).

### -p <string>

Specifies one or more file characteristic options (privileges). The <string> option-argument is a string specifying file characteristics to be retained or discarded on extraction. The string consists of the specification characters **a**, **e**, **m**, **o**, and **p**. Multiple characteristics can

be concatenated within the same string and multiple **-p** options can be specified. The meaning of the specification characters is as follows:

#### а

Do not preserve file access times. By default, file access times are preserved whenever possible.

#### е

Preserve everything (default mode), the user ID, group ID, file mode bits, file access time, and file modification time.

**Note:** The **e** flag is the sum of the **o** and **p** flags.

#### m

Do not preserve file modification times. By default, file modification times are preserved whenever possible.

## 0

Preserve the user ID and group ID.

### р

Preserve the file mode bits. This specification character is intended for a user with regular privileges who wants to preserve all aspects of the file other than the ownership. The file times are preserved by default, but two other flags are offered to disable this and use the time of extraction instead.

In the preceding list, preserve indicates that an attribute stored in the archive is given to the extracted file, subject to the permissions of the invoking process. Otherwise, the attribute of the extracted file is determined as part of the normal file creation action. If neither the **e** nor the **o** specification character is specified, or the user ID and group ID are not preserved for any reason, **server\_archive** will not set the S\_ISUID (setuid) and S\_ISGID (setgid) bits of the file mode. If the preservation of any of these items fails for any reason, **server\_archive** writes a diagnostic message to standard error.

**Note:** Failure to preserve these items will affect the final exit status, but will not cause the extracted file to be deleted.

If the file characteristic letters in any of the string option-arguments are duplicated, or in conflict with one another, the ones given last will take precedence. For example, if you specify **-p eme**, file modification times are still preserved.

### -s <replstr>

Modifies the file or archive member names specified by the pattern or <file> operand according to the substitution expression <replstr> using the syntax of the **ed** utility regular expressions.

**Note:** The **ed 1** manual page provides information.

Multiple -s expressions can be specified. The expressions are applied in the order they are specified on the command line, terminating with the first successful substitution. The optional trailing **g** continues to apply the substitution expression to the pathname substring, which starts with the first character following the end of the last successful substitution.

The optional trailing **p** causes the final result of a successful substitution to be written to standard error in the following format:

```
<original pathname> >> <new pathname>
```

File or archive member names that substitute the empty string are not selected and are skipped.

#### -t

Resets the access times of any file or directory read or accessed by **server\_archive** to be the same as they were *before* being read or accessed by **server\_archive**.

#### -u

Ignores files that are older (having a less recent file modification time) than a pre-existing file, or archive member with the same name. During read, an archive member with the same name as a file in a filesystem is extracted if the archive member is newer than the file. During copy, the file in the destination hierarchy is replaced by the file in the source hierarchy, or by a link to the file in the source hierarchy if the file in the source hierarchy is newer.

#### -v

During a list operation, produces a verbose table of contents using the format of the **ls 1** utility with the **-l** option. For pathnames representing a hard link to a previous member of the archive, the output has the format:

```
<ls -1 listing> == <link name>
```

For pathnames representing a symbolic link, the output has the format:

```
<ls -1 listing> => <link name>
```

where <ls -l listing> is the output format specified by the ls 1 utility when used with the -l option. Otherwise, for all the other operational modes (read, write, and copy), pathnames are written and flushed a standard error without a trailing <newline> as soon as processing begins on that file or archive member. The trailing <newline> is not buffered, and is written only after the file has been read or written.

### -x format

Specifies the output archive format, with the default format being **ustar**. The **server\_archive** command currently supports the following formats:

# cpio

The extended cpio interchange format specified in the **-p1003.2** standard. The default blocksize for this format is 5120 bytes. Inode and device information about a file (used for detecting file hard links by this format) which may be truncated by this format is detected by **server\_archive** and is repaired.

**Note:** To be readable by **server\_archive**, the archive must be built on another machine with the option **-c** (write header information in ASCII).

### bcpio

The old binary cpio format. The default blocksize for this format is 5120 bytes.

**Note:** This format is not very portable and should not be used when other formats are available.

Inode and device information about a file (used for detecting file hard links by this format) which may be truncated by this format is detected by **server\_archive** and is repaired.

### sv4cpio

The System V release 4 cpio. The default blocksize for this format is 5120 bytes. Inode and device information about a file (used for detecting file hard links by this format) which may be truncated by this format is detected by **server\_archive** and is repaired.

### sv4crc

The System V release 4 cpio with file crc checksums. The default blocksize for this format is 5120 bytes. Inode and device information about a file (used for detecting file hard links by this format) which may be truncated by this format is detected by **server\_archive** and is repaired.

#### tar

The old BSD tar format as found in BSD4.3. The default blocksize for this format is 10240 bytes. Pathnames stored by this format must be 100 characters or less in length. Only regular files, hard links, soft links, and directories will be archived (other filesystem types are not supported).

#### ustar

The extended tar interchange format specified in the **-p1003.2** standard. The default blocksize for this format is 10240 bytes.

**Note:** Pathnames stored by this format must be 250 characters or less in length (150 for basename and 100 for <file\_name>).

### emctar

This format is not compatible with **-p1003.2** standard. It allows archiving to a file greater than 8 GB. Pathnames stored by this format are limited to 3070 characters. The other features of this format are the same as **ustar**.

**server\_archive** detects and reports any file that it is unable to store or extract as the result of any specific archive format restrictions. The individual archive formats may impose additional restrictions on use.

**Note:** Typical archive format restrictions include (but are not limited to) file pathname length, file size, link pathname length, and the type of the file.

### -B bytes

Limits the number of bytes written to a single archive volume to bytes. The bytes limit can end with m, k, or b to specify multiplication by 1048576 (1M), 1024 (1K) or 512, respectively. A pair of bytes limits can be separated by x to indicate a product.

Note: The limit size will be rounded up to the nearest block size.

# -C [d|i|m]

When performing a restore, this allows you to choose PAX behaviors on CIFS collision names.

d: deletei: ignorem: mangle

### -D

Ignores files that have a less recent file inode change time than a pre-existing file, or archive member with the same name. The **-u** option provides information.

**Note:** This option is the same as the **-u** option, except that the file inode change time is checked instead of the file modification time. The file inode change time can be used to select files whose inode information (such as uid, gid, and so on) is newer than a copy of the file in the destination directory.

### -E limit

Has the following two goals:

- ◆ In case of medium error, to limit the number of consecutive read faults while trying to read a flawed archive to limit. With a positive limit, server\_archive attempts to recover from an archive read error and will continue processing starting with the next file stored in the archive. A limit of 0 (zero) will cause server\_archive to stop operation after the first read error is detected on an archive volume. A limit of "NONE" will cause server\_archive to attempt to recover from read errors forever.
- ◆ In case of no medium error, to limit the number of consecutive valid header searches when an invalid format detection occurs. With a positive value, server\_archive will attempt to recover from an invalid format detection and will continue processing starting with the next file stored in the archive. A limit of 0 (zero) will cause server\_archive to stop operation after the first invalid header is detected on an archive volume. A limit of "NONE" will cause server\_archive to attempt to recover from invalid format errors forever. The default limit is 10 retries.



### **CAUTION**

Using this option with NONE requires extreme caution as server\_archive may get stuck in an infinite loop on a badly flawed archive.

### -J

Backs up, restores, or displays CIFS extended attributes.

**p:** displays the full pathname for alternate names (for listing and archive only)

**u**: specifies UNIX name for pattern search

w: specifies M256 name for pattern search

**d:** specifies M83 name for pattern search

### -L

Follows all symbolic links to perform a logical filesystem traversal.

### -N

Used with the **-e archive\_name** option, prevents the tape from rewinding at the end of command execution.

# -P

Does *not* follow symbolic links.

**Note:** Performs a physical filesystem traversal. This is the default mode.

### -T [from\_date][,to\_date][/[c][m]]

Allows files to be selected based on a file modification or inode change time falling within a specified time range of from\_date to to\_date (the dates are inclusive). If only a from\_date is supplied, all files with a modification or inode change time equal to or less than are selected. If only a to\_date is supplied, all files with a modification or inode change time equal to or greater than will be selected. When the from\_date is equal to the to\_date, only files with a modification or inode change time of exactly that time will be selected.

When **server\_archive** is in the write or copy mode, the optional trailing field [c][m] can be used to determine which file time (inode change, file modification or both) is used in the comparison. If neither is specified, the default is to use file modification time only. The m specifies the comparison of file modification time (the time when the file was last written). The c specifies the comparison of inode change time; the time when the file inode was last changed. For example, a change of owner, group, mode, and so on. When c and m are both

specified, then the modification and inode change times are both compared. The inode change time comparison is useful in selecting files whose attributes were recently changed, or selecting files which were recently created and had their modification time reset to an older time as what happens when a file is extracted from an archive and the modification time is preserved. Time comparisons using *both* file times are useful when **server\_archive** is used to create a time-based incremental archive. Only files that were changed during a specified time range are archived.

A time range is made up of six different fields and each field must contain two digits. The format is:

# [yy[mm[dd[hh]]]]mm[ss]

Where yy is the last two digits of the year, the first mm is the month (from 01 to 12), dd is the day of the month (from 01 to 31), hh is the hour of the day (from 00 to 23), the second mm is the minute (from 00 to 59), and ss is seconds (from 00 to 59). The minute field mm is required, while the other fields are optional, and must be added in the following order: hh, dd, mm, yy. The ss field may be added independently of the other fields. Time ranges are relative to the current time, so -T 1234/cm selects all files with a modification or inode change time of 12:34 P.M. today or later. Multiple -T time range can be supplied, and checking stops with the first match.

#### -x

When traversing the file hierarchy specified by a pathname, does not allow descending into directories that have a different device ID. The st\_dev field as described in **stat 2** for more information about device IDs.

#### -Y

Ignores files that have a less recent file inode change time than a pre-existing file, or archive member with the same name.

**Note:** This option is the same as the **-D** option, except that the inode change time is checked using the pathname created after all the filename modifications have completed.

### $-\mathbf{Z}$

Ignores files that are older (having a less recent file modification time) than a pre-existing file, or archive member with the same name.

**Note:** This option is the same as the **-u** option, except that the modification time is checked using the pathname created after all the filename modifications have completed.

The options that operate on the names of files or archive members (-c, -i, -n, -s, -u, -v, -D, -T, -Y, and -Z) interact as follows.

When extracting files during a read operation, archive members are selected, based only on the user-specified pattern operands as modified by the -c, -n, -u, -D, and -T options. Then any -s and -i options will modify, in that order, the names of those selected files. Then the -Y and -Z options will be applied based on the final pathname. Finally, the -v option will write the names resulting from these modifications.

When archiving files during a write operation, or copying files during a copy operation, archive members are selected, based only on the user specified pathnames as modified by the -n, -u, -D, and -T options (the -D option applies only during a copy operation). Then any -s and -i options will modify, in that order, the names of these selected files. Then during a copy operation, the -Y and the -Z options will be applied based on the final pathname. Finally, the -v option will write the names resulting from these modifications.

When one or both of the **-u** or **-D** options are specified along with the **-n** option, a file is *not* considered selected unless it is newer than the file to which it is compared.

**SEE ALSO** *Using the Celerra server\_archive Utility.* 

**EXAMPLE #1** To archive the contents of the root directory to the device **rst0**, type:

\$ server\_archive <movername> -w -e rst0

**EXAMPLE #2** To display the verbose table of contents for an archive stored in <file\_name>, type:

\$ server\_archive <movername> -v -f <file\_name>

**EXAMPLE #3** To copy the entire olddir directory hierarchy to newdir, type:

\$ server\_archive <movername> -rw <olddir newdir>

**EXAMPLE #4** To interactively select the files to copy from the current directory to dest\_dir, type:

\$ server\_archive <movername> -rw -i <olddir dest\_dir>

# **EXAMPLE #5**

To extract *all* files from the archive stored in <file\_name>, type:

```
$ server_archive <movername> -r -f <file_name>
```

# **EXAMPLE #6**

To update (and list) only those files in the destination directory **/backup** that are older (less recent inode change or file modification times) than files with the same name found in the source file tree **home**, type:

\$ server\_archive <movername> -r -w -v -Y -Z home /backup

# **STANDARDS**

The **server\_archive** utility is a superset of the **-p1003.2** standard.

**Note:** The archive formats bcpio, sv4cpio, sv4crc, and tar, and the flawed archive handling during list and read operations are extensions to the **POSIX** standard.

# **ERRORS**

The **server\_archive** command exits with one of the following system messages:

```
All files were processed successfully.

or

An error occurred.
```

Whenever **server\_archive** cannot create a file or a link when reading an archive, or cannot find a file when writing an archive, or cannot preserve the user ID, group ID, or file mode when the **-p** option is specified, a diagnostic message is written to standard error, and a non-zero exit status is returned. However, processing continues.

In the case where **server\_archive** cannot create a link to a file, this command will *not* create a second copy of the file.

If the extraction of a file from an archive is prematurely terminated by a signal or error, **server\_archive** may have only partially extracted a file the user wanted. Additionally, the file modes of extracted files and directories may have incorrect file bits, and the modification and access times may be wrong.

If the creation of an archive is prematurely terminated by a signal or error, **server\_archive** may have only partially created the archive, which may violate the specific archive format specification.

If while doing a copy, **server\_archive** detects a file is about to overwrite itself, the file is *not* copied, a diagnostic message is written to standard error and when **server\_archive** completes, it exits with a non-zero exit status.

# server\_arp

Manages the Address Resolution Protocol (ARP) table for Data Movers.

# **SYNOPSIS**

# **DESCRIPTION**

**server\_arp** displays and modifies the IP-to-MAC address translation tables used by the ARP for the specified Data Mover.

The **ALL** option executes the command for all Data Movers.

# **OPTIONS**

<ip\_addr>

Displays the ARP entry for the specified IP address.

### -a11

Displays the first 64 of the current ARP entries.

-delete <ip\_addr>
Deletes an ARP entry.

-set <ip\_addr> <physaddr>

Creates an ARP entry with an IP address and physical address.

# **EXAMPLE #1** To

To create an ARP entry, type:

```
$ server_arp server_2 -set 172.24.102.20
00:D0:B7:82:98:E0
```

server\_2 : added: 172.24.102.20 at 0:d0:b7:82:98:e0

# **EXAMPLE #2**

To display all ARP entries for a specified Data Mover, type:

```
$ server_arp server_2 -all
```

```
server_2 :
172.24.102.254 at 0:d0:3:f9:37:fc
172.24.102.20 at 0:d0:b7:82:98:e0
172.24.102.24 at 0:50:56:8e:1d:5
128.221.253.100 at 0:4:23:a7:b1:35
```

# **EXAMPLE #3** To display an ARP entry specified by IP address, type:

```
$ server_arp server_2 172.24.102.20
```

server\_2 : 172.24.102.20 at 0:d0:b7:82:98:e0

# **EXAMPLE #4** To delete an ARP entry, type:

\$ server\_arp server\_2 -delete 172.24.102.24 server\_2 : deleted: 172.24.102.24 at 0:50:56:8e:1d:5

# server certificate

Manages Celerra's Public Key Infrastructure (PKI) for the specified Data Movers.

### **SYNOPSIS**

```
server_certificate {<movername> | ALL}
  -ca_certificate
     [-list]
      -info {-all|<certificate_id>}
      -import [-filename <path>]
      -delete {-all|<certificate_id>}}
  -persona
     [-list]
      -info {-all|<persona_name>|id=<persona_id>}
      -generate {<persona_name>|id=<persona_id>} -key_size {2048|4096}
       [-cs_sign_duration <# of months>]
       {-cn | -common_name} <common_name>[;<common_name>]
       [-ou <org_unit>[;<org_unit>]]
       [-organization <organization>]
       [-location <location>]
       [-state <state>]
       [-country <country>]
       [-filename <output_path>]
      -clear {<persona_name>|id=<persona_id>) {-next|-current|-both}
      -import {<persona_name>|id=<persona_id>} [-filename <path>]
```

### DESCRIPTION

server\_certificate manages the use of public key certificates between Data Movers acting as either clients or servers. server\_certificate -ca\_certificate manages the Certificate Authority (CA) certificates Celerra uses to confirm a server's identity when the Data Mover is acting as a client. server\_certificate -persona manages the certificates presented by the Data Mover to a client application when the Data Mover is acting as a server as well as the certificates presented by the Data Mover to a server configured to require client authentication.

# **OPTIONS**

### -ca\_certificate

Lists the CA certificates currently available on the Celerra Network Server. The ouput from this command is identical to the output from the **-list** option.

### -ca certificate -list

Lists the CA certificates currently available on the Celerra Network Server.

-ca\_certificate -info {-all|<certificate\_id>}
Displays the properties of a specified CA certificate or all CA
certificates.

## -ca\_certificate -import [-filename <path>]

Imports a CA certificate. You can only paste text in PEM format at the command prompt. Specify **-filename** and provide a path to import a CA certificate in either DER or PEM format.

-ca\_certificate -delete {-all|<certificate\_id>}
Deletes a specified CA certificate or all CA certificates.

### -persona

Lists the key sets and associated certificates currently available on the Celerra Network Server. The ouput from this command is identical to the output from the **-list** option.

### -persona -list

Lists the key sets and associated certificates currently available on the Celerra Network Server.

**-persona -info** {-all|<persona\_name>|id= <persona\_id>} Displays the properties of the key sets and associated certificates, including the text of a pending certificate request, of a specified persona or all personas.

```
-persona -generate {<persona_name>|id=<persona_id>}
-key_size <bits> {-cn|-common_name} <common_name>
[;<common_name>]
```

Generates a public/private key set along with a request to sign the certificate. Specify either the persona name or ID. The ID is automatically generated when the persona is created. You can determine the ID using the **-list** or **-info** options. The key size can be either 2048 or 4096 bits. Use either **-cn** or **-common\_name** to specify the commonly used name. The common name is typically a hostname that describes the Data Mover with which the persona is associated. Multiple common names are allowed but must be separated by semicolons.

```
[-cs_sign_duration <# of months>]
```

Specifies the number of months the certificate is valid. A month is defined as 30 days. This option is valid only if the certificate is signed by the Control Station. If this option is specified, you cannot save the request to a file using the **-filename** option.

```
[-ou <org_unit>[;<org_unit>]]
```

Identifies the organizational unit. Multiple organizational units are allowed but must be separated by semicolons.

```
[-organization <organization>]
```

Identifies the organization.

```
[-location <location>]
```

Identifies the physical location of the organizational unit.

```
[-state <state>]
```

Identifies the state where the organizational unit is located.

```
[-country <country>]
```

Identifies the country where the organization unit is located. This value is limited to two characters.

```
[-filename <output_path>]
```

Provides a path to where the request should be saved to a file. This option is valid only if the certificate is signed by an external CA. If this option is specified, you cannot specify the number of months the certificate is valid using the **-cs\_sign\_duration** option.

```
-persona -clear {<persona_name>|id=<persona_id>}
{-next|-current|-both}
```

Deletes a key set and the associated certificate. You can delete the current key set and certificate, the next key set and certificate, or both.

```
-persona -import {<persona_name>|id=<persona_id>}
[-filename <path>]
```

Imports a CA-signed certificate. You can only paste text in PEM format at the command prompt. Specify **-filename** and provide a path to import a CA-signed certificate in either DER or PEM format.

# **SEE ALSO** nas\_ca\_certificate.

# **EXAMPLE #1** To import a CA certificate, specifying a filename and path, type:

```
$ server_certificate server_2 -ca_certificate -import
-filename "/tmp/ca_cert.pem"
```

done

# **EXAMPLE #2** To list all the CA certificates currently available on the Celerra, type:

```
$ server_certificate ALL -ca_certificate -list
server_2 :
id=1
subject=0=Celerra Certificate Authority; CN=sorento
issuer=O=Celerra Certificate Authority;CN=sorento
expire=20120318032639Z
id=2
subject=C=US;O=VeriSign, Inc.;OU=Class 3 Public Primary Certification Author
issuer=C=US;O=VeriSign, Inc.;OU=Class 3 Public Primary Certification Author
expire=20280801235959Z
server_3 :
subject=0=Celerra Certificate Authority; CN=zeus-cs
issuer=O=Celerra Certificate Authority;CN=zeus-cs
expire=20120606181215Z
         EXAMPLE #3
                        To list the properties of the CA certificate identified by certificate ID
                        2, type:
                        $ server_certificate server_2 -ca_certificate -info 2
server 2 :
id=2
subject = C=US; O=VeriSign, Inc.; OU=Class 3 Public Primary Certification Authority
issuer = C=US; O=VeriSign, Inc.; OU=Class 3 Public Primary Certification Authority
start = 199601290000002
expire = 20280801235959Z
signature alg. = md2WithRSAEncryption
public key alg. = rsaEncryption
public key size = 1024 bits
serial number = 70ba e41d 10d9 2934 b638 ca7b 03cc babf
version
         EXAMPLE #4
                        To generate a key set and certificate request to be sent to an external
                        CA for the persona identified by the persona name default, type:
                        $ server_certificate server_2 -persona -generate default
                        -key_size 2048 -common_name division.xyz.com
server_2 :
Starting key generation. This could take a long time ...
done
         EXAMPLE #5
                        To list all the key sets and associated certificates currently available
                        on the Celerra, type:
                        S server certificate ALL -persona -list
server_2 :
id=1
name=default
next state=Request Pending
request subject=CN=name; CN=1.2.3.4
```

```
server_3 :
id=1
name=default
next state=Not Available
CURRENT CERTIFICATE:
id=1
subject=CN=test;CN=1.2.3.4
expire=20070706183824Z
issuer=O=Celerra Certificate Authority; CN=eng173100
```

### **EXAMPLE #6** To list the properties of the key set and certificate identified by persona ID 1, type:

\$ server\_certificate server\_2 -persona -info id=1

```
server_2 :
id=1
name=default
next state=Request Pending
request subject=CN=name; CN=1.2.3.4
Request:
----BEGIN CERTIFICATE REQUEST----
MIIEZjCCAk4CAQAwITENMAsGA1UEAxMEbmFtZTEQMA4GA1UEAxMHMS4yLjMuNDCC
```

AiIwDQYJKoZIhvcNAQEBBQADggIPADCCAgoCggIBANKW3Q/F6eMqIxrCO5IeXLET bWkm5RzrbI51HxLNuhobR5S9G2o+k47X0QZFkGzg/2F7kR06vVIH7CPH9X2qGAzV 7GmZaF00wPcktPJYzjQE8guNhcL1qZpPl4IZrbnSGEAWcAAE0nvNwLp9aN0WSC+N TDJZY4A9yTURiUc+Bs8plhQh16wLLL0zjUKIvKjAqiTE0F3RApVJEE/9y6N+Idsb Vwf/rvzP6/z0wZW5H184HKXInJaHTBDK59G+e/Y2JqvUY1UNBZ5SODunOakHabex k6COFYjDu7Vd+yHpvcyTalHJ2RcIavpQuM02o+VVpxqUyX7M1+VXJXTJm0yb4j4q tZITOSVZ2FqEpOkoIpzqoAL7A9B69WpFbbpIX8danhReafDh4oj4yWocvSwMKYv1 33nLak3+wpMQNrwJ2L9FIHP2fXClnvThBqupm7uqqHP3TfNBbBPTYY3qkNPZ78wx /njUrZKbfWd81Cc+ngUi33hbMuBR3FFsQNASYZUzg15+JexALH5jhBahd2aRXBag itOLhvxYK0dEgIEwDfdDedx7i+yro2gbNxhLLdtkuBtKrmOnuT5g2WWXNKzNa/H7 KWv8JSwCv1mW1N/w7V9aEbDizBBfer+ZdMPkGLbyb/EVXZnHABeWH3iKC6/ecnRd 4Kn7KO9F9qXVH1zzTeYVAgMBAAGgADANBgkqhkiG9w0BAQUFAAOCAgEAzSS4ffYf 2WN0vmZ0LgsSBcVHPVEVg+rP/aU9iNM9KDJ4P4OK41UDU8tOGy09Kc8EvklBUm59 fyjt2T/3RqSqvvkCEHJsVW3ZMnSsyjDo6Ruc0HmuY4q+tuLl+di1SQnZGUxt8asw dhEpdEzXA6o9cfmVZMSt5QicfAmmBNr4BaO96+VAlq59fu/chU1pvKWWMGXz4I2s 7z+UdMBYO4pEfyG1i34Oof/z4K0SVNICn3CEkW5TIsSt8qA/E2JXX1LhbMYWKYuY 9ur/qspHuWzkIXZFx4SmTK9/RsE1Vy7fBztIoN8myFN0nma84D9pyqls/yhvXZ/D iDF6Tgk4RbNzuanRBSYiJFu4Tip/nJlK8uv3ZyFJ+3DK0c8oz1BLuQdadxHcJglt m/T4FsHa3JS+D8CdA3uDPfIvvVNcwP+4RBK+Dk6EyQe8uKrVL7ShbacQCUXn0AAd Ol+DQYFQ7Mczcm84L98srhov3JnIEKcjaPseB7S9KtHvHvvs4q11Q5U2RjQppykZ qpSFnCbYDGjOcqOrsqNehV9F4h9fTszEdUY1UuLqvtRj+FTT2Ik7nMK641wfVtSO LCial6kuYsZg16SFxncnH5gKHtQMWxd9nv+UyJ5VwX3aN12N0ZQbaIDcQp75Em2E aKjd28cZ6FEavimn69sz0B8PHQV+6dPwywM=

----END CERTIFICATE REQUEST----

# **EXAMPLE #7**

To generate a key set and certificate request that is automatically received by the Control Station for the persona identified by the persona name default, type:

\$ server\_certificate server\_2 -persona -generate default
-key\_size 2048 -cs\_sign\_duration 12 -common\_name
division.xyz.com

### server\_2 :

Starting key generation. This could take a long time ...

# **EXAMPLE #8**

To generate a key set and certificate request to be sent to an external CA specifying subject information, type:

\$ server\_certificate server\_2 -persona -generate default
-key\_size 2048 -common\_name division.xyz.com -ou QA
-organization XYZ -location Bethesda -state Maryland
-country US -filename /tmp/server\_2.1.request.pem

### server\_2 :

Starting key generation. This could take a long time  $\dots$  done

# **EXAMPLE #9** To import a signed certificate and paste the certificate text, type:

\$ server\_certificate server\_2 -persona -import default server\_2 : Please paste certificate data. Enter a carriage return and on the new line type 'end of file' or 'eof' followed by another carriage return.

# server\_cdms

Provides Celerra Data Migration Service (CDMS) functionality for the specified Data Movers.

### **SYNOPSIS**

```
server_cdms {<movername> | ALL}
    -connect <mgfs> -type {nfsv2 | nfsv3} -path <localpath>
    -source <srcName>:/<srcPath>[-option <options>]
| -connect <mgfs> -type cifs -path <localpath> -netbios <netbios> -source
    \\<srcServer>[.<domain>]\<srcShare>[\<srcPath>] -admin
    [<domain>\]<admin_name> [-wins <wins>]
| -disconnect <mgfs> {-path <localpath> | -path <cid> | -all}
| -verify <mgfs> [-path {<localpath> | <cid> }]
| -Convert <mgfs>
| -start <mgfs> -path <localpath> [-Force] -log <logpath>
    [-include <include_path>][-exclude <exclude_path>]
| -halt <mgfs> -path <localpath>
| -info [<mgfs>][-state {START | STOP | ON GOING | ERROR | SUCCEED | FAIL}]
```

### **DESCRIPTION**

**server\_cdms** establishes and removes connections to remote systems, and allows users to start on-access migration.

**server\_cdms** creates an auto-migration process on the Data Mover to ensure that all data has been migrated from the remote system. **server\_cdms** also checks the state of the migrated filesystem (MGFS), all auto-migration processes, and the connection, and reports if all data has been migrated successfully.

CDMS supports NFSv2 and NFSv3 only.

The ALL option executes the command for all Data Movers.

# **OPTIONS**

```
-connect <mgfs> -type {nfsv2 | nfsv3} -path <localpath>
-source <srcName>:/<srcPath>
```

Provides a connection for the Celerra Network Server with the remote NFS server. The **-type** option specifies the protocol type to be used for communication with the remote NFS server. The directory <localpath> in the filesystem must be unique for that filesystem.

The **-source** option specifies the source file server name or IP address of the remote server as the <srcName> and the export path for migration. For example, nfs\_server:/export/path

**Note:** After the **-connect** command completes, the filesystem must be exported.

```
[-option <options>]
```

Specifies the following comma-separated options:

```
[useRootCred={true | false}]
```

When the filesystem is mounted, **true** ensures that the MGFS reads from the source file server using root access UID=0, GID=0. This assumes that the source file server path is exported to allow root access from the specified Data Mover. When **false** (default), the MGFS uses the owner's UID and GID to access data.

```
[proto={TCP | UDP}]
```

Sets the connection protocol type. The default is TCP.

```
[nfsPort=<port>]
```

Sets a remote NFS port number in case the Portmapper or RPC bind is not running, and the port is not the default of 2049.

```
[mntPort=<port>]
```

Sets a remote mount port number in case Portmapper or RPC bind is not running.

```
[mntVer={1 | 2 | 3}]
```

Sets the version used for mount protocol. By default, NFSv2 uses mount version 2, unless user specified version 1; NFSv3 uses mount version 3.

```
[localPort=<port>]
```

Sets the port number used for NFS services, if it needs to be different from the default. The default port number is always greater than 1024.

```
-connect <mgfs> -type cifs -path <localpath> -netbios
<netbios> -source \\<srcServer>[.<domain>]
\<srcShare>[\<srcPath>] -admin [<domain>\]
<admin_name>[-wins <wins>]
```

Provides a connection for the Celerra Network Server with the remote CIFS server as specified by its NetBIOS name. The directory <localpath> in the filesystem must be unique for that filesystem. The **-source** option specifies the source file server name of the remote server as the <srcName> and the share path for migration that is not at the root of the share. For example, \\share\\dir1...

The **-source** and **-admin** option strings must be enclosed by quotes when issued in a Linux shell.

The **-admin** option specifies an administrator for the filesystem. A password is asked interactively when the command is issued. The **-wins** option specifies an IP address for the WINS server.

**Note:** This is required only for Windows NT 4.0.

-disconnect <mgfs> {-path <localpath> | -path <cid> | -all} Removes a connection without migrating the data. The <localpath> is not removed nor is any partially migrated data.

The administrator should manually remove this data before attempting a **-verify** or **-Convert** command. It may require the administrator to handle a partial migration of old data as well as potentially new data created by users.

It is recommended not to use the **-disconnect** option if the administrator has exported this directory for user access.

```
-verify <mgfs>
```

Checks that all data has completed the migration for the <mgfs>.

```
[-path {<localpath>|<cid>}]
```

If the **-path** option is provided, it can check on a communication basis. If no path is provided, the system defaults to checking all connections on the filesystem.

## -Convert <mgfs>

Performs a verify check on the entire filesystem, then changes the filesystem type from MGFS to UxFS. After the **-Convert** option succeeds, no data migration can be done on that filesystem.

**-start** <mgfs> **-path** <localpath> [**-Force**] **-log** <logpath> Directs the Data Mover to migrate all files from the source file server to the Celerra Network Server. The **-log** option provides detailed information on the state of the migration, and any failures that might occur. The **-log** option is the path where the migration thread is started. The **-Force** option is used if you need to start a migration thread a second time on the same **-localpath>** where a previous migration thread had already finished. For example, **-Force** would be needed to start a thread which had no include file (that is, to migrate all remaining files) on **-localpath>** where a thread with an include file had already been run.

```
[-include <include_path>]
```

Starts the thread in the <include\_path> which is the path of the file containing the specified directories.

```
[-exclude <exclude_path>]
```

Excludes files or directories from migration. The <include\_path> is the path of the file containing the specified directories.

```
-halt <mgfs> -path <localpath>
```

Stops a running thread, and halts its execution on the Data Mover. The <mgfs> is the name of the migration filesystem and the <localpath> is the full path where the migration thread was started.

The **-start** option resumes thread execution.

#### -info

Displays a status on the migration filesystem and the threads.

```
[<mqfs>]
```

Specifies the migration filesystem.

```
[-state {START | STOP | ON_GOING | ERROR | SUCCEED | FAIL}]
```

Displays only the threads that are in the state that is specified.

# **SEE ALSO** *Celerra CDMS Version 2.0 for NFS and CIFS*, server\_export, server mount, and server setup.

# **EXAMPLE #1** To provide a connection for the migration filesystem to communicate with the remote NFS server, type:

```
$ server_cdms server_2 -connect ufs1 -type nfsv3 -path
/nfsdir -source 172.24.102.144:/srcdir -option proto=TCP
```

server\_2 : done

done

# **EXAMPLE #2** To provide a connection for the migration filesystem to communicate with the remote CIFS server, type:

**EXAMPLE #3** To display a status on the migration filesystem, type:

```
$ server_cdms server_2
```

```
server_2 :
CDMS enabled with 32 threads.

ufs1:
path = /nfsdir
  cid = 0
  type = NFSV3
  source = 172.24.102.144:/srcdir
  options= proto=TCP
path = /dstdir
  cid = 1
```

```
type = CIFS
 source = \\winserver1.nasdocs.emc.com\srcdir\
 netbios = DM112-CGE0.NASDOCS.EMC.COM
 admin = nasdocs.emc.com\administrator
                       When migration is started:
                       $ server_cdms server_2
server_2 :
CDMS enabled with 32 threads.
ufs1:
path = /nfsdir
 cid
      = 0
 type = NFSV3
 source = 172.24.102.144:/srcdir
 options= proto=TCP
path
      = /dstdir
 cid
 type = CIFS
 source = \\winserver1.nasdocs.emc.com\srcdir\
 netbios = DM112-CGE0.NASDOCS.EMC.COM
 admin = nasdocs.emc.com\administrator
 threads:
             = /dstdir
    path
     state = ON_GOING
      log
           = /
      cid
             = NONE
```

### Where:

Value	Definition
ufs1	Migration filesystem.
path	Directory in the local filesystem.
cid	Connection ID (0 through 1023).
type	Protocol type to be used to communicate with the remote server.
source	Source file server name or IP address of the remote server and the export path for migration.
options	Connection protocol type.
netbios	NetBIOS name of the remote CIFS server.
admin	Administrator for the filesystem.
threads	Currently existing migration threads.
state	Current status of migration threads.
log	Location of the log file that provides detailed information.

# **EXAMPLE #4** To direct **server\_2** to migrate all files from the source file server to the Celerra Network Server, type:

```
$ server_cdms server_2 -start ufs1 -path /dstdir -log /
```

server\_2 : done

**EXAMPLE #5** 

```
type:
                        $ server_cdms server_2 -info ufs1 -state ON_GOING
server_2 :
ufs1:
      = /nfsdir
path
 cid
       = 0
 type = NFSV3
 source = 172.24.102.144:/srcdir
 options= proto=TCP
path
       = /dstdir
 cid
       = 1
 type = CIFS
 source = \\winserver1.nasdocs.emc.com\srcdir\
 netbios = DM112-CGE0.NASDOCS.EMC.COM
 admin = nasdocs.emc.com\administrator
 threads:
             = /dstdir
    path
      state = ON_GOING
      loa
            = /
              = NONE
      cid
         EXAMPLE #6
                        To stop data migration on server_2 for ufs1, type:
                        $ server_cdms server_2 -halt ufs1 -path /dstdir
server_2 : done
         EXAMPLE #7
                        To check that all data has completed the migration, type:
                        $ server_cdms server_2 -verify ufs1 -path /dstdir
server_2 : done
         EXAMPLE #8
                        To disconnect the path on server_2 for data migration, type:
                        $ server_cdms server_2 -disconnect ufs1 -path /nfsdir
server_2 : done
         EXAMPLE #9
                        To disconnect all paths for data migration, type:
                        $ server_cdms server_2 -disconnect ufs1 -all
server_2 : done
        EXAMPLE #10
                        To perform a verify check on ufs1, and then convert it to a uxfs, type:
                        $ server_cdms server_2 -Convert ufs1
server_2 : done
```

To display information about migration with the specified status,

# server\_cepp

Manages the Celerra Event Publishing Agent (CEPA) service on the specified Data Mover.

# **SYNOPSIS**

```
server_cepp {<movername>|ALL}
   -service {-start|-stop|-status|-info}
| -pool {-info|-stats}
```

# **DESCRIPTION**

**server\_cepp** starts or stops the CEPA service on the specified Data Mover or all Data Movers and displays information on the status, configuration, and statistics for the service and the pool. The CEPA service is set up in the cepp.conf configuration file. The CEPA configuration is displayed using **-service -status**, but changes can only be made by directly editing the file with a text editor.

**ALL** executes the command for all Data Movers.

# **OPTIONS**

```
-service {-start|-stop|-status|-info}
```

The **-start** option starts the CEPA service on the specified Data Mover. **-stop** stops the CEPA service, **-status** returns a message indicating whether the CEPA service has started or been stopped, and **-info** displays information about the CEPA service including key properties of the configured pool.

```
-pool {-info|-stats}
```

Displays properties or statistics for the CEPA pool on the specified Data Mover.

SEE ALSO

Using Celerra Event Enabler and Using Celerra Event Publishing Agent

# **EXAMPLE #1**

To start the CEPA service on a Data Mover, type:

```
$ server_cepp server_2 -service -start
```

server\_2 : done

# **EXAMPLE #2**

To display the status of the CEPA service, type:

```
$ server_cepp server_2 -service -status
```

server\_2 : CEPP Stopped

# **EXAMPLE #3** To display the configuration of the CEPA service, type:

# Where:

Value	Definition
CIFS share name	The name of the shared directory and CIFS server used to access files in the Data Movers.
cifs_server	CIFS server to access files.
heartbeat_interval	The time taken to scan each CEPA server.
ft level	Fault tolerance level assigned. This option is required. 0 (continue and tolerate lost events; default setting), 1 (continue and use a persistence file as a circular event buffer for lost events), 2 (continue and use a persistence file as a circular event buffer for lost events until the buffer is filled and then stop CIFS), or 3 (upon heartbeat loss of connectivity, stop CIFS).
ft location	Directory where the persistence buffer file resides relative to the root of a filesystem. If a location is not specified, the default location is the root of the filesystem.
ft size	Maximum size in MB of the persistence buffer file. The default is 1 MB and the range is 1 MB to 100 MB.
msrpc user	Name assigned to the user account that the CEPA service is running under on the CEE machine. For example, ceeuser.
msrpc client name	Domain name assigned if the msrpc user is a member of a domain. For example, domain.ceeuser.
pool_name	Name assigned to the pool that will use the specified CEPA options.
server_required	Displays availability of the CEPA server. If a CEPA server is not available and this option is yes, an error is returned to the requestor that access is denied. If a CEPA server is not available and this option is no, an error is not returned to the requestor and access is allowed.
access_checks_ignored	The number of CIFS requests processed when a CEPA server is not available and the server_required option is set to "no." This option is reset when the CEPA server becomes available.
req_timeout	Time out in ms to send a request that allows access to the CEPA server.
retry_timeout	Time out in ms to retry the access request sent to the CEPA server.

# **EXAMPLE #4** To display information about the CEPA pool, type:

```
$ server_cepp server_2 -pool -info
server_2 :
pool_name = pool1
server_required = yes
access_checks_ignored = 0
req_timeout = 5000 ms
retry_timeout = 25000 ms
pre_events = OpenFileNoAccess, OpenFileRead
post_events = CreateFile, DeleteFile
post_err_events = CreateFile, DeleteFile
CEPP Servers:
IP = 10.171.10.115, state = ONLINE, vendor = Unknown
```

### Where:

Value	Definition
pre_events	Sends notification before selected event occurs. An empty list indicates that no pre-event messages are generated.
post_events	Sends notification after selected event occurs. An empty list indicates that no post-event messages are generated.
post_err_events	Sends notification if selected event generates an error. An empty list indicates that no post-error-event messages are generated.
CEPP Servers	IP addresses of the CEPA servers; state of the CEPA servers; vendor software installed on CEPA servers.

# **EXAMPLE #5** To display statistics for the CEPA pool, type:

Requests

2

2

# \$ server\_cepp server\_2 -pool -stats

Max(us)

758

635

Min(us)

659

604

pool\_name = pool1
Event Name
OpenFileWrite
CloseModified
Total Requests = 4
Min(us) = 604
Max(us) = 758
Average(us) = 664

server\_2 :

. . .

Average(us)

709

620

# server\_checkup

Checks the configuration parameters, state of a Data Mover and its dependencies.

# **SYNOPSIS**

```
server_checkup {<movername> | ALL}
    {[-test <component> [-subtest <dependency>]][-quiet][-full]}
| -list
| -info {<component> | all}
```

# **DESCRIPTION**

**server\_checkup** performs a sanity check of a specific Data Mover component and its dependencies by checking configuration parameters, and the current state of the component and dependencies.

A component is any basic feature that is available on the Data Mover, for example, CIFS. A dependency is a configuration component of a Data Mover that the proper operation of a Data Mover functionality (like CIFS) is depending upon. This configuration component can be owned by multiple Data Mover components. For example, proper operation of a CIFS service depends on correctly specified DNS, WINS, Antivirus, and so on.

**server\_checkup** displays a report of errors and warnings detected in the specified Data Mover component and its dependencies.

## **OPTIONS**

# No arguments

Performs a sanity check of all the components and all their dependencies on the specified Data Mover or all Data Movers.

```
-test <component>
```

Performs a sanity check of a specific component and all of its dependencies.

```
[-subtest <dependency>]
```

Performs a sanity check of a specific component and its specified dependency only. If the dependency is not defined, executes the command for all the dependencies of the component.

### -quiet

Displays only the number of errors and warnings for the sanity check.

#### -full

Provides a full sanity check of the specified Data Movers.

### -list

Lists all available components that can be checked on a Data Mover.

# -info <component>

Lists all dependencies of the specified component, with details of checks that can be performed on each dependency.

### **EXAMPLE #1**

To list the available component in the Data Mover, type:

\$ server\_checkup server\_2 -list

server 2 : done

REPV2 HTTPS CIFS FTPDS

# **EXAMPLE #2** To execute the check of the CIFS component, type:

\$ server\_checkup server\_2 -test CIFS

server\_2 :

-----Checks------

### Component CIFS :

```
: Checking the number of ACL per file system.....*Pass
Connection: Checking the load of TCP connections of CIFS..... Pass
Credential: Checking the validity of credentials...... Pass
        : Checking the connectivity and configuration of the DCs.....*Fail
DFS
        : Checking the DFS configuration files and DFS registry..... Pass
        : Checking the DNS configuration and connectivity to DNS servers. Pass
DNS
        : Checking the configuration of Windows Event Logs..... Pass
        : Checking if all file systems are all DIR3 type..... Pass
FS Type
        : Checking the GPO configuration..... Pass
GPO
HomeDir
        : Checking the configuration of home directory share..... Pass
I18N
        : Checking the I18N mode and the Unicode/UTF8 translation tables. Pass
Kerberos : Checking machine password update for Kerberos..... Fail
LocalGrp : Checking the local groups database configuration..... Fail
        : Checking the connectivity to the NIS servers, if defined..... Pass
NIS
        : Checking the connectivity to theNTP servers, if defined...... Pass
NTP
        : Checking the ntxmap configuration file...... Pass
Ntxmap
Security : Checking the CIFS security settings..... Pass
        : Checking the CIFS files servers configuration..... Pass
Server
        : Checking the network shares database..... Pass
Share
        : Checking the range availability of SMB ID.....*Pass
SmbList.
        : Checking for CIFS blocked threads..... Pass
Threads
UM_Client : Checking for the connectivity to usermapper servers, if any.... Pass
UM_Server : Checking the consistency of usermapper database, if primary....*Pass
        : Checking for unsupported client network OS...... Pass
UnsupProto: Checking for unsupported client network protocols...... Pass
VC.
        : Checking the configuration to Virus Checker servers..... Pass
```

WINS : Checking for the connectivity to WINS servers, if defined Pass			
NB: a result with a '*' means that some tests were not executed. use -full to run them $ \\$			
CIFS : Kerberos Warnings			
Warning 17451974742: server_2 : No update of the machine password of server 'DM102-CGE1'. hold.			
> Check the log events to find out the reason of this issue.			
Warning 17451974742: server_2 : No update of the machine password of server 'DM102-CGE0'. hold.			
> Check the log events to find out the reason of this issue.			
CIFS : LocalGrp Warnings			
Warning 17451974726: server_2: The local group 'Guests' of server 'DM102-CGE1' contains an unmapped member: S-1-5-15-60415a8a-335a7a0d-6b635f23-202.The access to some network resources may be refused> According the configured resolver of your system (NIS, etc config files, usermapper, LDAP), add the missing members.			
CIFS : DC Errors			
Error 13160939577: server_2 : pingdc failed due to NT error ACCESS_DENIED at step SAMR lookups			
> check server configuration and/or DC policies according to reported error.			
Error 13160939577: $server_2$ : pingdc failed due to NT error ACCESS_DENIED at step SAMR lookups			
> check server configuration and/or DC policies according to reported error.			
<b>EXAMPLE #3</b> To execute only the check of the DNS dependency of the CIFS			
component, type:			
<pre>\$ server_checkup server_2 -test CIFS -subtest DNS server_2 :</pre>			
Checks			
Component CIFS:			
DNS : Checking the DNS configuration and connectivity to DNS servers. Pas			

# **EXAMPLE #4** To list the available dependencies of the CIFS component, type:

#### \$ server\_checkup server\_2 -info CIFS

server\_2 : done

COMPONENT : CIFS DEPENDENCY : ACL

DESCRIPTION : Number of ACL per file system.

TESTS

In full mode, check if the number of ACL per file system doesn't exceed 90% of the maximum limit.

COMPONENT : CIFS

DEPENDENCY : Connection

DESCRIPTION : TCP connection number

TESTS

Check if the number of CIFS TCP connections doesn't exceed 80% of the maximum

number.

COMPONENT : CIFS

DEPENDENCY : Credential

DESCRIPTION : Users and groups not mapped

TESTS

Check if all credentials in memory are mapped to a valid SID.

COMPONENT : CIFS DEPENDENCY : DC

DESCRIPTION : Connectivity to the domain controllers

TESTS :

Check the connectivity to the favorite DC (DCPing),

In full mode, check the connectivity to all DC of the domain,

Check if DNS site information are defined for each computer name,

Check if the site of each computer name has an available DC, Check if trusted domain of each computer name can be reached,

Check the ds.useDCLdapPing parameter is enabled,

Check the ds.useADSite parameter is enabled.

COMPONENT : CIFS DEPENDENCY : DFS

DESCRIPTION : DFS service configuration on computer names

TESTS

Check the DFS service is enabled in registry if DFS metadata exists,  $\ensuremath{\mathsf{C}}$ 

Check the DFS metadata of each share with DFS flag are correct,

Check if share names in DFS metadata are valid and have the DFS flag,

Check if each DFS link is valid and loaded,

Check in the registry if the WideLink key is enabled and corresponds to a valid

share name.

COMPONENT : CIFS DEPENDENCY : DNS

DESCRIPTION : DNS domain configuration

```
TESTS
Check if each DNS domain has at least 2 defined servers,
Check the connectivity to each DNS server of each DNS domain,
Check if each DNS server of each DNS domain supports really the DNS service,
Check the ds.useDSFile parameter (automatic discovery of DC),
Check the ds.useDSFile parameter is enabled if the directoryservice file exists.
COMPONENT
            : CIFS
DEPENDENCY : EventLog
DESCRIPTION : Event Logs parameters on servers
TESTS
Check if the pathnames of each event logs files are valid (application, system and
security),
Check if the maximum file size of each event logs file doesn't exceed 1GB,
Check if the retention time of each event logs file doesn't exceed 1 month.
COMPONENT
             : CIFS
DEPENDENCY
           : FS_Type
DESCRIPTION : DIR3 mode of filesystems
TESTS
Check if each file system is configured in the DIR3 mode.
             : CIFS
COMPONENT
DEPENDENCY
            : GPO
DESCRIPTION : GPO configuration on Win2K servers
Check if the size of the GPO cache file doesn't exceed 10% of the total size of
the root file system,
Check the last modification date of the GPO cache file is up-to-date,
Check the cifs.gpo and cifs.gpoCache parameters have not been changed,
COMPONENT
            : CIFS
DEPENDENCY
            : HomeDir
DESCRIPTION : Home directory shares configuration
TESTS
Check if the home directory shares configuration file exists, the feature is
enabled,
Check if the home directory shares configuration file is optimized (40 lines
Check the syntax of the home directory shares configuration file.
COMPONENT
             : CIFS
DEPENDENCY
            : I18N
DESCRIPTION : Internationalization and translation tables
TESTS
Check if computer name exists, the I18N mode is enabled,
Check the .etc_common file system is correctly mounted,
Check the syntax of the definition file of the Unicode characters,
Check the uppercase/lowercase conversion table of Unicode character is valid.
COMPONENT
             : CIFS
```

: Kerberos

DEPENDENCY

DESCRIPTION : Kerberos configuration

TESTS

Check the machine password update is enabled and up-to-date.

COMPONENT : CIFS
DEPENDENCY : LocalGrp

DESCRIPTION : Local groups and local users

TESTS :

Check the local group database doesn't contain more than 80% of the maximum number of servers,

Check if the servers in the local group database are all valid servers, Check the state of the local group database (initialized and writable),

Check if the members of built-in local groups are all resolved in the domain,

Check the number of built-in local groups and built-in local users,

Check if the number of defined local users doesn't exceed 90% of the maximum number.

COMPONENT : CIFS DEPENDENCY : NIS

DESCRIPTION: Network Information System (NIS) configuration

TESTS :

If NIS is configured, check at least 2 NIS servers are defined (redundancy check), Check if each NIS server can be contacted on the network,

Check if each NIS server really supports the NIS service.

COMPONENT : CIFS DEPENDENCY : NTP

DESCRIPTION : Network Time Protocol (NTP) configuration

TESTS :

If NTP is configured, check at least 2 NTP servers are defined (redundancy check), Check if each NIS server can be contacted on the network,

If computer names exist, check if NTP is configured and is running.

COMPONENT : CIFS
DEPENDENCY : Ntxmap

DESCRIPTION : Checking the ntxmap.conf file.

TESTS :

Check the data consistency of the ntxmap configuration file.

COMPONENT : CIFS
DEPENDENCY : Security

DESCRIPTION : Security settings

TESTS

If the I18N mode is enabled, check the share/unix security setting is not in use, Discourage to use the share/unix security setting,

Check the cifs.checkAcl parameter is enabled if the security setting is set to NT.

COMPONENT : CIFS
DEPENDENCY : Server

DESCRIPTION : Files servers

TESTS

Check if each CIFS server is configured with a valid IP interface,

Check if each computer name has joined its domain, Check if each computer name is correctly registered in their DNS servers, Check if the DNS servers have the valid IP addresses of each computer name, Check if a DNS domain exists if at least one computer name exists, : CIFS COMPONENT DEPENDENCY : Share DESCRIPTION : Network shares Check the available size and i-nodes on the root file system are at least 10% of the total size, Check the size of the share database doesn't exceed 30% of the total size of the root file system, Check if the pathname of each share is valid and is available, Check if each server in the share database really exists, Check if the I18N mode is enabled, all the share names are UTF-8 compatible, Check the list of ACL of each share contains some ACE, Check the length of each share name doesn't exceed 80 Unicode characters. COMPONENT : CIFS DEPENDENCY : SmbList DESCRIPTION : 64k UID, TID and FID limits TESTS In full mode, check the 3 SMB ID lists (UID, FID and TID) don't exceed 90% of the maximum ID number. COMPONENT : CIFS DEPENDENCY : Threads DESCRIPTION : Blocked threads and overload TESTS Check CIFS threads blocked more than 5 and 30 seconds, Check the maximum number of CIFS threads in use in the later 5 minutes doesn't exceed 90% of the total number, Check the number of threads reserved for Virus Checker doesn't exceed 20% of the total number of CIFS threads. COMPONENT : CIFS DEPENDENCY : UM\_Client DESCRIPTION : Connectivity to the usermapper server TESTS If usermapper servers are defined, check each server can be contacted, Check if usermapper servers are defined, NIS is not simultaneously activated. COMPONENT : CIFS : UM\_Server DEPENDENCY DESCRIPTION : Primary usermapper server TESTS If a primary usermapper is defined locally, check its database size doesn't exceed 30% of the total size, Check if configuration file is in use, the filling rate of the ranges doesn't exceed 90%,

Check if configuration file is in use, 2 ranges do not overlap,

Check if secmap is enabled, In full mode, check the SID/UID and SID/GID mappings and reverses are correct and coherent. COMPONENT : CIFS DEPENDENCY : UnsupOS DESCRIPTION : Client OS not supported TESTS Check for unsupported client network OS. : CIFS COMPONENT DEPENDENCY : UnsupProto DESCRIPTION : Unsupported protocol commands detected TESTS Check for unsupported client network protocol commands. COMPONENT : CIFS DEPENDENCY : VC DESCRIPTION : Virus checker configuration TESTS If VC is enabled, check the syntax of the VC configuration file, Check if the VC 'enable' file and the VC configuration are compatible, Check the number of VC servers. Make sure at least 2 servers are defined, for redundancy, Check if there are offline VC servers, Check if the VC high watermark has not been reached, Check the connection of VC servers to the Data Mover. COMPONENT : CIFS DEPENDENCY : WINS DESCRIPTION : WINS servers. TESTS If NetBIOS names are defined, check if at least one WINS server is defined, Check the number of WINS servers. check if two servers are defined for redundancy, Check if each WINS server can be contacted on the network, Check these servers are really WINS servers, Check if the NetBIOS are correctly registered on the servers. EXAMPLE #5 To execute additional tests, type: \$ server\_checkup server\_2 -full server\_2 : -----Checks------Component REPV2: F\_RDE\_CHEC: Checking the F-RDE compatibilty of Repv2 sessions..... Fail Component HTTPS: HTTP : Checking the configuration of HTTP applications..... Pass

SSL	:	Checking	the configuration of SSL applications Fai
Component	C:	IFS :	
ACL	:	Checking	the number of ACLs per file system Pas
Connection	1:	Checking	the load of CIFS TCP connections Pas
Credential	L:	Checking	the validity of credentials Fai
DC			the connectivity and configuration of Domain Controlle Fai
DFS	:	Checking	the DFS configuration files and DFS registry Fai
DNS			the DNS configuration and connectivity to DNS servers. Fai
EventLog			the configuration of Windows Event Logs Fai
FS_Type			if all file systems are in the DIR3 format Pas
GPO			the GPO configuration Pas
HomeDir	:	Checking	the configuration of home directory shares Fai
I18N			the I18N mode and the Unicode/UTF8 translation tables. Fai
Kerberos			password updates for Kerberos Fai
LDAP			the LDAP configuration Pas
LocalGrp			the database configuration of local groups Fai
NIS			the connectivity to the NIS servers Pas
NS		_	the naming services configuration Fai
NTP		_	the connectivity to the NTP servers Fai
Ntxmap		_	the ntxmap configuration file Pas
Secmap			the SECMAP database Fai
Security	:	Checking	the CIFS security settings Pas
Server		_	the CIFS file servers configuration Fai
Share			the network shares database Fai
SmbList			the range availability of SMB IDs Fai
Threads			for CIFS blocked threads Pas
			the connectivity to usermapper servers Pas
UM_Server		_	the usermapper server database Pas
Unsup0S			for unsupported client network operating systems Pas
			for unsupported client network protocols Pas
VC			the configuration of Virus Checker servers Pas
WINS	:	Checking	the connectivity to WINS servers Fai
Component	F.	TPDS :	
FS_Type	:	Checking	if all file systems are in the DIR3 format Pas
FTPD			the configuration of FTPD Fai
NIS			the connectivity to the NIS servers Pas
NS		_	the naming services configuration Fai
NTP	:	Checking	the connectivity to the NTP servers Fai
SSL	:	Checking	the configuration of SSL applications Fai
			HTTPS : SSL Warnings

Warning 17456169084: server $_2$ : The SSL feature 'DHSM' can not get certificate from the persona default. Because this feature needs a certificate and a private key, it can not start,

--> Run the server\_certificate command to generate a new key set and certificate for this persona. Or run the appropriate command (like server\_http for instance) to set a correct persona for this SSL feature.

Warning 17456169084: server\_2: The SSL feature 'DIC' can not get certificate from the persona default. Because this feature needs a certificate and a private key, it can not start,

--> Run the server\_certificate command to generate a new key set and certificate for this persona. Or run the appropriate command (like server\_http for instance) to set a correct persona for this SSL feature.

Warning 17456169084: server\_2: The SSL feature 'DIC\_S' can not get certificate from the persona default. Because this feature needs a certificate and a private key, it can not start,

--> Run the server\_certificate command to generate a new key set and certificate for this persona. Or run the appropriate command (like server\_http for instance) to set a correct persona for this SSL feature.

Warning 17456169084: server\_2: The SSL feature 'DIC\_L' can not get certificate from the persona default. Because this feature needs a certificate and a private key, it can not start,

--> Run the server\_certificate command to generate a new key set and certificate for this persona. Or run the appropriate command (like server\_http for instance) to set a correct persona for this SSL feature.

Warning 17456169084: server\_2: The SSL feature 'DBMS\_FILE\_TRANSFER' can not get certificate from the persona default. Because this feature needs a certificate and a private key, it can not start,

--> Run the server\_certificate command to generate a new key set and certificate for this persona. Or run the appropriate command (like server\_http for instance) to set a correct persona for this SSL feature.

-----CIFS : Credential Warnings-----

Warning 17456168968: server\_2: The CIFS service is currently stopped. Many CIFS sanity check tests cannot be done as all CIFS servers are currently disabled on this Data Mover.

--> Start the CIFS server by executing the 'server\_setup' command, and try again.

-----CIFS : DC Warnings-----

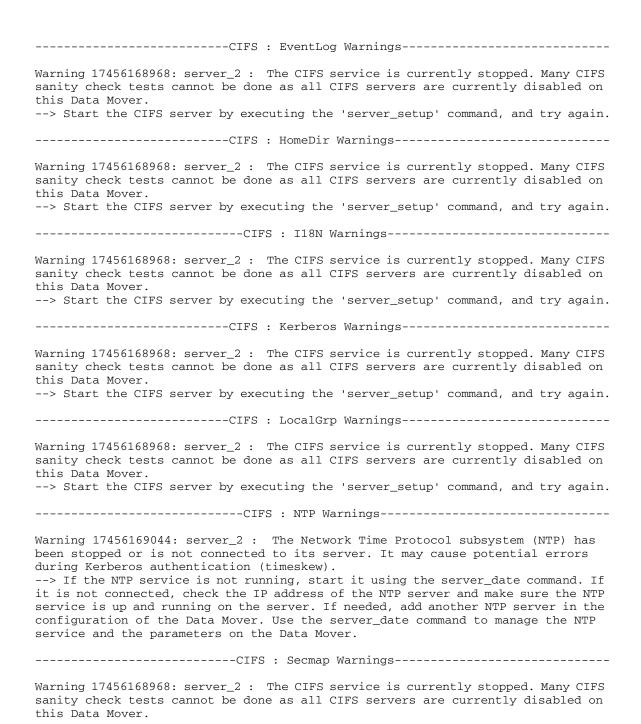
Warning 17456168968: server\_2: The CIFS service is currently stopped. Many CIFS sanity check tests cannot be done as all CIFS servers are currently disabled on this Data Mover.

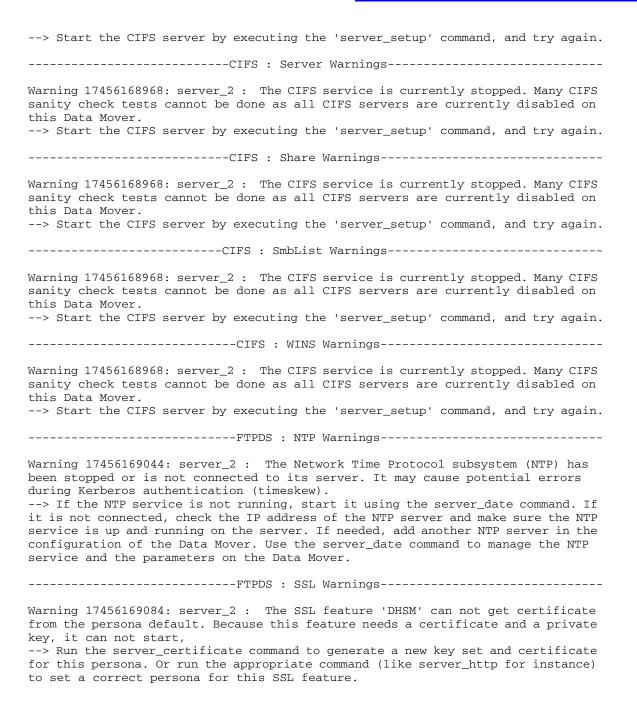
--> Start the CIFS server by executing the 'server\_setup' command, and try again.

-----CIFS : DFS Warnings-----

Warning 17456168968: server\_2: The CIFS service is currently stopped. Many CIFS sanity check tests cannot be done as all CIFS servers are currently disabled on this Data Mover.

--> Start the CIFS server by executing the 'server\_setup' command, and try again.





Warning 17456169084: server\_2: The SSL feature 'DIC' can not get certificate from the persona default. Because this feature needs a certificate and a private key, it can not start,

--> Run the server\_certificate command to generate a new key set and certificate for this persona. Or run the appropriate command (like server\_http for instance) to set a correct persona for this SSL feature.

Warning 17456169084: server\_2: The SSL feature 'DIC\_S' can not get certificate from the persona default. Because this feature needs a certificate and a private key, it can not start,

--> Run the server\_certificate command to generate a new key set and certificate for this persona. Or run the appropriate command (like server\_http for instance) to set a correct persona for this SSL feature.

Warning 17456169084: server\_2: The SSL feature 'DIC\_L' can not get certificate from the persona default. Because this feature needs a certificate and a private key, it can not start,

--> Run the server\_certificate command to generate a new key set and certificate for this persona. Or run the appropriate command (like server\_http for instance) to set a correct persona for this SSL feature.

Warning 17456169084: server\_2: The SSL feature 'DBMS\_FILE\_TRANSFER' can not get certificate from the persona default. Because this feature needs a certificate and a private key, it can not start,

--> Run the server\_certificate command to generate a new key set and certificate for this persona. Or run the appropriate command (like server\_http for instance) to set a correct persona for this SSL feature.

-----

-----REPV2 : F\_RDE\_CHECK Errors-----

Error 13160415855: server\_2: For the Replication session: rep1, Data Mover version on the source fs: 5.6.47 Data Mover version on the destination fs: 5.5.5 Minimum required Data Mover version on the destination fs: 5.6.46

The Data Mover version on the destination file system is incompatible with the Data Mover version on the source file system. After data transfer, the data in the destination file system may appear to be corrupt, even though the data is in fact intact.

Upgrade the Data Mover where the destination file system resides to at least 5.6.46.

Error 13160415855: server\_2: For the Replication session:rsd1, F-RDE version on the source fs: 5.6.46 F-RDE version on the destination fs: 5.5.5 Minimum required F-RDE version on the destination fs: 5.6.46 The F-RDE versions are incompatible. After data transfer, the data in the dst FS may appear to be corrupt.

--> Upgrade the DataMover where the dst fs resides to atleast the version on the source.

Error 13160415855: server\_2 : For the Replication session:rsd2,

F-RDE version on the source fs: 5.6.46

F-RDE version on the destination fs: 5.5.5

Minimum required F-RDE version on the destination fs: 5.6.46

The F-RDE versions are incompatible.

After data transfer, the data in the dst FS may appear to be corrupt.

 ${ ext{--->}}$  Upgrade the DataMover where the dst fs resides to atleast the version on the source.

Error 13160415855: server\_2 : For the Replication session:rsd3, F-RDE version on the source fs: 5.6.46

F-RDE version on the destination fs: 5.5.5 Minimum required

F-RDE version on the destination fs: 5.6.46

The F-RDE versions are incompatible.

After data transfer, the data in the dst FS may appear to be corrupt.

--> Upgrade the DataMover where the dst fs resides to atleast the version on the source.

-----HTTPS : SSL Errors-----

Error 13156876314: server\_2: The persona 'default' contains nor certificate neither private keys sets. So, this persona can not be used by a SSL feature on the Data Mover.

 $\operatorname{\mathsf{---}}$  Run the server\_certificate command to generate a new key set and certificate for this persona.

-----CIFS : DNS Errors-----

Error 13161070637: server\_2 : The DNS service is currently stopped and does not contact any DNS server. The CIFS clients may not be able to access the Data Mover on the network.

--> Start the DNS service on the Data Mover, using the 'server\_dns' command.

-----CIFS : NS Errors-----

Error 13156352011: server\_2: None of the naming services defined for the entity 'host' in nsswitch.conf is configured.

--> Make sure each entity (e.g. host, passwd..) in the nsswitch.conf file contains naming services, (e.g. local files, NIS or usermapper), and make sure these services are configured. Use the corresponding commands like server\_nis, server\_dns or server\_ldap to make sure they are configured.

Error 13156352011: server\_2: None of the naming services defined for the entity 'group' in nsswitch.conf is configured.

--> Make sure each entity (e.g. host, passwd..) in the nsswitch.conf file contains naming services, (e.g. local files, NIS or usermapper), and make sure these services are configured. Use the corresponding commands like server\_nis, server\_dns or server\_ldap to make sure they are configured.

Error 13156352011: server\_2: None of the naming services defined for the entity 'netgroup' in nsswitch.conf is configured.

--> Make sure each entity (e.g. host, passwd..) in the nsswitch.conf file contains naming services, (e.g. local files, NIS or usermapper), and make sure these services are configured. Use the corresponding commands like server\_nis, server\_dns or server\_ldap to make sure they are configured.

-----FTPDS : FTPD Errors-----

Error 13156876314: server\_2: The persona 'default' contains nor certificate neither private keys sets. So, this persona can not be used by a SSL feature on the Data Mover.

--> Run the server\_certificate command to generate a new key set and certificate for this persona.

-----FTPDS : NS Errors------

Error 13156352011: server\_2: None of the naming services defined for the entity 'host' in nsswitch.conf is configured.

--> Make sure each entity (e.g. host, passwd..) in the nsswitch.conf file contains naming services, (e.g. local files, NIS or usermapper), and make sure these services are configured. Use the corresponding commands like server\_nis, server\_dns or server\_ldap to make sure they are configured.

Error 13156352011: server\_2: None of the naming services defined for the entity 'group' in nsswitch.conf is configured.

--> Make sure each entity (e.g. host, passwd..) in the nsswitch.conf file contains naming services, (e.g. local files, NIS or usermapper), and make sure these services are configured. Use the corresponding commands like server\_nis, server\_dns or server\_ldap to make sure they are configured.

Error 13156352011: server\_2 : None of the naming services defined for the entity 'netgroup' in nsswitch.conf is configured.

--> Make sure each entity (e.g. host, passwd..) in the nsswitch.conf file contains naming services, (e.g. local files, NIS or usermapper), and make sure these services are configured. Use the corresponding commands like server\_nis, server\_dns or server\_ldap to make sure they are configured.

-----FTPDS : SSL Errors-------

Error 13156876314: server\_2: The persona 'default' contains nor certificate neither private keys sets. So, this persona can not be used by a SSL feature on the Data Mover.

--> Run the server\_certificate command to generate a new key set and certificate for this persona.

\_\_\_\_\_\_

Total: 14 errors, 25 warnings

# **EXAMPLE #6** To display only the number of errors and warnings for a Data Mover and dependency, type:

\$ server\_checkup server\_2 -quiet

```
server_2 :
------Checks------
Component REPV2:
F_RDE_CHEC: Checking the F-RDE compatibilty of Repv2 sessions..... Fail
Component HTTPS :
        : Checking the configuration of HTTP applications..... Pass
HTTP
SSL
        : Checking the configuration of SSL applications..... Pass
Component CIFS :
ACL
        : Checking the number of ACLs per file system.....*Pass
Connection: Checking the load of CIFS TCP connections..... Pass
Credential: Checking the validity of credentials..... Fail
        : Checking the connectivity and configuration of Domain Controlle Fail
DC
        : Checking the DFS configuration files and DFS registry..... Fail
DFS
DNS
        : Checking the DNS configuration and connectivity to DNS servers. Fail
       : Checking the configuration of Windows Event Logs..... Fail
EventLog
FS_Type
        : Checking if all file systems are in the DIR3 format...... Pass
GPO
        : Checking the GPO configuration...... Pass
        : Checking the configuration of home directory shares..... Fail
HomeDir
I18N
        : Checking the I18N mode and the Unicode/UTF8 translation tables. Fail
Kerberos
        : Checking password updates for Kerberos..... Fail
        : Checking the LDAP configuration..... Pass
LDAP
        : Checking the database configuration of local groups..... Fail
LocalGrp
        : Checking the connectivity to the NIS servers..... Pass
NIS
NS
        : Checking the naming services configuration...... Fail
NTP
        : Checking the connectivity to the NTP servers..... Fail
        : Checking the ntxmap configuration file..... Pass
Ntxmap
        : Checking the SECMAP database..... Fail
Secmap
       : Checking the CIFS security settings..... Pass
Security
Server
        : Checking the CIFS file servers configuration..... Fail
Share
        : Checking the network shares database..... Fail
SmbList
        : Checking the range availability of SMB IDs.....*Pass
Threads
        : Checking for CIFS blocked threads..... Pass
UM_Client : Checking the connectivity to usermapper servers..... Pass
UM_Server : Checking the usermapper server database.....*Pass
        : Checking for unsupported client network operating systems..... Pass
Unsup0S
UnsupProto: Checking for unsupported client network protocols..... Pass
VC
        : Checking the configuration of Virus Checker servers...... Pass
WINS
        : Checking the connectivity to WINS servers..... Fail
Component FTPDS:
```

# **The server Commands**

FS_Type		Checking if all file systems are in the DIR3 format P	
FTPD	:	Checking the configuration of FTPD F	ʻail
NIS	:	Checking the connectivity to the NIS servers P	ass
NS	:	Checking the naming services configuration F	'ail
NTP	:	Checking the connectivity to the NTP servers F	'ail
SSL	:	Checking the configuration of SSL applications P	ass
NB: a resu	ılt	with a '*' means that some tests were not executed. use -full to r	
Total :	12	errors, 14 warnings	
		Charks	

# server\_cifs

Manages the CIFS configuration for the specified Data Movers or Virtual Data Movers (VDMs).

#### **SYNOPSIS**

```
server_cifs {<movername>|ALL} [<options>]
  'options' can be one of the following:
 -option {{audit [,user=<user_name>][,client=<client_name>][,full]}
      homedir[=NO]}
 -add netbios=<netbios_name>, domain=<domain_name>[,alias=<alias_name>...]
    [,hidden={y|n} [[,interface=<if_name>[,wins=<ip>[:<ip>]]]...]
    [,local_users][-comment <comment>]
  -add
    compname = < comp_name > , domain = < full_domain_name > [ , alias = < alias_name > . . . ]
    [, hidden={y|n}][, authentication={kerberos|all}]
    [,netbios=<netbios_name>][[,interface=<if_name>[,wins=<ip>[:<ip>]]
    [,dns=<if_suffix>]]...][,local_users][-comment <comment>]
-add standalone=<netbios_name>, workgroup=<workgroup_name>
    [,alias=<alias_name>...][,hidden={y|n}]
    [[,interface=<if_name>[,wins=<ip>[:<ip>]]...][,local_users]
    [-comment <comment>]
  -rename -netbios <old_name> <new_name>
  -Join compname = <comp_name > , domain = <full_domain_name > , admin = <admin_name >
    [, ou=<organizational_unit>]
    [-option {reuse|resetserverpasswd|addservice=nfs}]
  -Unjoin compname = < comp_name > , domain = < full_domain_name > , admin = < admin_name >
  -add security={NT | UNIX | SHARE} [,dialect=<dialect_name>]
  -add wins=<ip_addr>[,wins=<ip_addr>...]
  -add usrmapper=<ip_addr>[,usrmapper=<ip_addr>...]
  -Disable <interface>[,<interface>...]
  -Enable <interface>[,<interface>...]
  -delete netbios=<netbios_name> [-remove_localgroup]
    [,alias=<alias_name>...][,interface=<if_name>]
  -delete compname=<comp_name> [-remove_localgroup]
    [,alias=<alias_name>...][,interface=<if_name>]
  -delete wins=<ip_addr>[,wins=<ip_addr>...]
  -delete usrmapper=<ip_addr>[,usrmapper=<ip_addr>...]
  -delete standalone=<netbios_name> [-remove_localgroup]
    [,alias=<alias_name>...][,interface=<if_name>]
  -update {<share_name> | <path>} [mindirsize=<size>] [force]
  -Migrate {<fs_name> -acl <netbios_servername> -localgroup}
    <src_domain>{:nb=<netbios>|:if=<interface>}
    <dst_domain>{:nb=<netbios>|:if=<interface>}
  -Replace {<fs_name> -acl | <netbios_servername> -localgroup}
    {:nb=<netbios>|:if=<interface>}
  -stats [-full [-zero]] | [-summary]
```

# **DESCRIPTION**

**server\_cifs** manages the CIFS configuration for the specified <movername> which can be the physical Data Mover or VDMs.

Most command options are used with both VDMs and physical Data Movers, whereas others are only used with physical Data Movers. Options available for physical Data Movers only are:

- -add security/dialect...
  -add/delete usrmapper
  -enable/disable interface
- The **ALL** option executes the command for all Data Movers.

# OPTIONS

# No arguments

Displays the CIFS protocol configuration. Certain inputs are not casesensitive; however, variables may be automatically converted to uppercase.

<options>

CIFS options include:

#### -option audit

Audits the CIFS configuration by testing for live connections to a Data Mover.

[,user=<user\_name>][,client=<client\_name>][,full] Audits the live connections created when the session is initiated by the specified <client\_name> or audits the live connections for those owned by the specified <user\_name>. The full option can be used to identify open files. The <client\_name> can be a string or an IPV4 address and the <user\_name> can be a string of maximum 20 characters.

## -option homedir[=NO]

Enables and disables (default) the home directory feature. The Data Mover reads information from the homedir map file.

-add netbios=<netbios\_name>, domain=<domain\_name>
Configures a Windows NT 4.0-like CIFS server on a Data Mover,
assigning the specified <netbios\_name> and <domain\_name> to the
server. The domain name is limited to 15 bytes.



#### **CAUTION**

Each NetBIOS name must be unique to the domain and the Data Mover.

```
[,alias=<alias_name>...]
```

Assigns a NetBIOS alias to the <netbios\_name> associated with the NetBIOS name. The <alias\_name> must:

- Be unique on a Data Mover
- Be limited to 15 bytes
- Not begin with an @ (at sign) or (dash) character
- Not include spaces, tab characters, or the following symbols: /
   ; , = \* + | [] ? < > "

```
[, hidden = {y | n}]
```

By default, the <netbios\_name> is displayed in the Network Neighborhood. If **hidden**=**y** is specified, the <netbios\_name> does not appear.

```
[[, interface=<if_name>[, wins=<ip>[:<ip>]]]...]
Specifies a logical IP interface for the CIFS server in the Windows
NT 4.0 domain and associates up to two WINS IP addresses with
each interface. The interface name is case-sensitive.
```

**Note:** When configuring a CIFS server without any interfaces for a Data Mover, it becomes the default CIFS server and is available on all interfaces not used by other CIFS servers. The default CIFS server can be deleted at any time. It is recommended that IP interfaces should always be specified. VDMs do not have default CIFS servers.

```
[,local_users]
```

Enables local user support that allows the creation of a limited number of local user accounts on the CIFS server. When this command executes, type and confirm a password that is assigned to the local Administrator account on the CIFS server. In addition to the Administrator account, a Guest account is also created. The Guest account is disabled by default. The Administrator account password must be changed before the Administrator can log in to the CIFS server.

After initial creation of the stand-alone server, the **local\_users** option resets the local Administrator account password. The password can only be reset if it has not been changed through a Windows client. If the password has already been changed through Windows, the reset will be refused.

```
[-comment <comment>]
```

Assigns a comment to the configuration. The comment is delimited by quotes. Comment length is limited to 48 bytes

(represented as 48 ASCII characters or a variable number of Unicode multibyte characters) and cannot include colons since they are recognized as delimiters.

-add compname=<comp\_name>, domain= <full\_domain\_name>
Configures a CIFS server as the <comp\_name> in the specified
Windows Active Directory workgroup. A default NetBIOS name is
automatically assigned to the <comp\_name>. Since the default for
<netbios\_name> is derived from the <comp\_name>, the
<comp\_name> must not contain any characters that are invalid for a
<netbios\_name>.

In the case of disjoint namespaces, you must use the fully qualified domain name for the <comp\_name>. For example, dm112-cge0.emc.com, not just dm112-cge0.

The <comp\_name> is limited to 63 bytes. The fully qualified domain name is limited to 155 bytes. The <full\_domain\_name> must contain a dot (.). There cannot be a @ (at sign) or - (dash) character. The name also cannot include spaces, tab characters, or the symbols: / : ; , = \* + | []? <> "



## **CAUTION**

Each computer name must be unique to the domain and the Data Mover.

**Note:** *Using International Character Sets with Celerra* provides details. Only Windows NT security mode can be configured when UTF-8 is enabled.

[,alias=<alias\_name>...]

Assigns an alias to the NetBIOS name. The <alias\_name> must:

- Be unique on a Data Mover
- Be limited to 15 bytes
- Not begin with an @ (at sign) or (dash) character
- Not include spaces, tab characters, or the following symbols: /
   :;, = \* + | [] ? < > "

 $[, hidden = \{y | n\}]$ 

By default, the computer name appears in the Network Neighborhood. If **hidden=y** is specified, then the computer name does not appear.

## [,authentication={kerberos|all}]

Specifies the type of user authentication. The **kerberos** option limits the server usage to Kerberos authentication; the **all** option (default) allows both Kerberos and NTLM authentication.

#### [, netbios=<netbios\_name>]

Specifies a <netbios\_name> for the <comp\_name> in place of the default. The default for <netbios\_name> is assigned automatically and is derived from the first 15 bytes of the <comp\_name>. The <netbios\_name> cannot begin with an @ (at sign) or - (dash) character. The name also cannot include spaces, tab characters, or the symbols: / \ :; , = \* + | [] ? <> "

[[, interface=<if\_name>[, wins=<ip>[:<ip>]]]...]
Specifies a logical IP interface for the CIFS server in the Active
Directory domain and associates up to two WINS IP addresses
with each interface. The interface name is case-sensitive.

**Note:** When configuring a CIFS server without any interfaces for a Data Mover, it becomes the default CIFS server and is available on all interfaces not used by other CIFS servers. The default CIFS server can be deleted at any time. It is recommended that IP interfaces should always be specified. VDMs do not have default CIFS servers.

#### [,dns=<if\_suffix>]

Specifies a different DNS suffix for the interface for DNS updates. By default, the DNS suffix is derived from the domain. This DNS option does not have any impact on the DNS settings of the Data Mover.

#### [,local\_users]

Enables local user support that allows the creation of a limited number of local user accounts on the CIFS server. When this command executes, type and confirm a password that is assigned to the local Administrator account on the CIFS server. In addition to the Administrator account, a Guest account is also created. The Guest account is disabled by default. The Administrator account password must be changed before the Administrator account can log in to the CIFS server.

After initial creation of the stand-alone server, the **local\_users** option resets the local Administrator account password. The password can only be reset if it has not been changed through a Windows client. If the password has already been changed through Windows, the reset will be refused.

```
[-comment <comment>]
```

Assigns a comment to the configuration. The comment is delimited by quotes. Comment length is limited to 48 bytes (represented as 48 ASCII characters or a variable number of Unicode multibyte characters) and cannot include colons since they are recognized as delimiters.

```
-add standalone=<netbios_name>, workgroup=
<workgroup name>
```

Creates or modifies a stand-alone CIFS server on a Data Mover, assigning the specified <netbios\_name> and <workgroup\_name> to the server. The NetBIOS and workgroup names are limited to 15 bytes. When creating a stand-alone CIFS server for the first time, the <code>,local\_users</code> option must be typed, or the command will fail. It is not required when modifying the CIFS server. A stand-alone CIFS server does not require any Windows domain infrastructure. A stand-alone server has local user accounts on the Data Mover and NTLM is used to authenticate users against the local accounts database.



#### **CAUTION**

Each NetBIOS name must be unique to the workgroup and the Data Mover.

```
[,alias=<alias_name>...]
```

Assigns an alias to the NetBIOS name. The <alias\_name> must:

- Be unique on a Data Mover
- Be limited to 15 bytes
- Not begin with an @ (at sign) or (dash) character
- Not include spaces, tab characters, or the following symbols: /
   \:;, = \* + | [] ? < > "

```
[, hidden = {y | n}]
```

By default, the <netbios\_name> is displayed in the Network Neighborhood. If **hidden**=**y** is specified, the <netbios\_name> does not appear.

```
[[, interface<if_name>[, wins<iip>[:<ip>]]]...] Specifies a logical IP interface for the CIFS server and associates up to two WINS IP addresses with each interface. The interface name is case-sensitive.
```

**Note:** When configuring a CIFS server without any interfaces for a Data Mover, it becomes the default CIFS server and is available on all interfaces not used by other CIFS servers. The default CIFS server can be deleted at any time. It is recommended that IP interfaces should always be specified. VDMs do not have default CIFS servers.

#### [,local\_users]

Enables local user support that allows the creation of a limited number of local user accounts on the CIFS server. When this command executes, type and confirm a password that is assigned to the local Administrator account on the CIFS server. In addition to the Administrator account, a Guest account is also created. The Guest account is disabled by default. The Administrator account password must be changed before the Administrator can log in to the CIFS server.

After initial creation of the stand-alone server, the <code>local\_users</code> option resets the local Administrator account password. The password can only be reset if it has not been changed through a Windows client. If the password has already been changed through Windows, the reset will be refused.

#### [-comment <comment>]

Assigns a comment to the configuration. The comment is delimited by quotes. Comment length is limited to 48 bytes (represented as 48 ASCII characters or a variable number of Unicode multibyte characters) and cannot include colons since they are recognized as delimiters.

#### -rename -netbios <old\_name> <new\_name>

Renames a NetBIOS name. For Microsoft Windows Server, renames a Compname after the CIFS server is unjoined from the domain. *Managing Celerra for the Windows Environment* provides the procedure to rename a Compname.

**Note:** Before performing a rename, the new NetBIOS name must be added to the domain using the Microsoft Windows Server Users and Computers MMC snap-in.

#### -Join compname=<comp\_name>, domain=

<full\_domain\_name>, admin=<admin\_name>

Creates an account for the CIFS server in the Active Directory. By default, the account is created under the domain **root** as **ou=Computers,ou=EMC Celerra**.



#### **CAUTION**

Before performing a -Join, CIFS service must be started using server\_setup.

The <comp\_name> is limited to 63 bytes and represents the name of the server to be registered in DNS. The <full\_domain\_name> is the full domain name to which the server belongs. This means the name must contain at least one period (.). The <admin\_name> is the logon name of the user with the right to create and manage computer accounts in the Organizational Unit that the CIFS server is being joined to. If a domain is given as part of the admin username it should be of the form: admin@FQDN. If no domain is given the admin user account is assumed to be part of the domain the CIFS Server is being joined to. The user is prompted to type a password for the admin account.

An Active Directory and a DNS can have the same domain name, or a different domain name (disjoint namespace). For each type of Active Directory and DNS domain relationship, specific Celerra parameters and command values must be used. For example, for a disjoint namespace, you must always specify the fully qualified domain name (FQDN) with the computer name when joining a CIFS server to a domain.



#### **CAUTION**

Time services must be synchronized using server\_date.

[, ou=<organizational\_unit>]

Specifies the organizational unit or container where computer accounts are created in the Active Directory. By default, computer accounts are created in an organizational unit called Computers. The name must be in a valid distinguished name format, for example, ou="cn=My\_mover". The name may contain multiple nested elements, such as ou="cn=comp:ou=mach". The colon (:) must be used as a separator for multiple elements. By default, ou=Computers,ou=EMC Celerra is used. The organizational unit name is limited to 256 bytes.

[-option {reuse|resetserverpasswd|addservice= nfs}] The reuse option reuses the existing computer account with the original principal or joins a CIFS server to the domain where the computer account has been created manually.

The **resetserverpasswd** option resets the CIFS server password and encryption keys on a domain controller. This option could be used for security reasons, such as changing the server password in the Kerberos Domain Controller.

The addservice option adds the NFS service to the CIFS server, making it possible for NFS users to access the Windows Kerberos Domain Controller. Before adding NFS service, the <comp\_name> must already be joined to the domain, otherwise the command will fail.

# -Unjoin compname = < comp\_name > , domain =

<full\_domain\_name>,admin=<admin\_name>

Deletes the account for the CIFS server as specified by its <comp\_name> from the Active Directory database. The user is prompted to type a password for the admin account.

## -add security={NT|UNIX|SHARE}

Defines the user authentication mechanism used by the Data Mover for CIFS services. NT (default) security mode uses standard Windows domain based user authentication. The local password and group files, NIS, EMC Active Directory UNIX users and groups extension, or UserMapper are required to translate Windows user and group names into UNIX UIDs and GIDs. NT security mode is required for the Data Mover to run Windows 2000 or later native environments. Unicode should be enabled for NT security mode.



# **CAUTION**

EMC does not recommend the use of UNIX or SHARE security modes.

For UNIX security mode, the client supplies a username and a plain-text password to the server. The server uses the local (password or group) file or NIS to authenticate the user. To use **UNIX** security mode, CIFS client machines must be configured to send user passwords to the Data Mover unencrypted in plain text. This requires a registry or security policy change on every CIFS client machine.

For VDM, UNIX and SHARE security modes are global to the Data Mover and cannot be set for each VDM. Unicode must not be enabled.

For **SHARE** security mode, clients supply a read-only or read-write password for the share. No user authentication is performed using **SHARE** security. Since this password is sent through the network in

clear text, you must modify the Client Registry to allow for clear text passwords.



# **CAUTION**

Before adding or changing a security mode, CIFS service must be stopped using server\_setup, then restarted once options have been set.

```
[,dialect=<dialect_name>]
```

Specifies a dialect. Optimum dialects are assigned by default. Options include **CORE**, **COREPLUS**, **LANMAN1** (default for UNIX and SHARE security modes), **LANMAN2**, and **NT1** (which represents SMB1 and is the default for NT security mode), and **SMB2**.

```
-add wins=<ip_addr>[,wins=<ip_addr>...]
```

Adds the WINS servers to the CIFS configuration. The list of WINS servers is processed in the order in which they are added. The first one is the preferred WINS server. If after 1500 milliseconds, the first WINS server times out, the next WINS server on the list is used.

-add usrmapper=<ip\_addr>[,usrmapper=<ip\_addr>...] Adds the IP address(es) of a secondary Usermapper hosts to the CIFS configuration. A single IP address can point to a primary or secondary Usermapper host. If you are using distributed Usermappers, up to eight subsequent IP addresses can point to secondary Usermapper hosts.

```
-Disable <interface> [<interface>,...]
```

Disables the specified IP interfaces for CIFS service. Interface names are case-sensitive. All unused interfaces should be disabled.

```
-Enable <interface> [<interface>,...]
```

Enables the specified IP interfaces for CIFS service. Interface names are case-sensitive.

```
-delete standalone=<netbios_name>
[-remove_localgroup][,alias=<alias_name>...][,interface=
<if_name>]
```

Deletes the stand-alone CIFS server as identified by its NetBIOS name from the CIFS configuration of the Data Mover.

```
-delete netbios=<netbios_name>
[-remove_localgroup][,alias=<alias_name>...][,interface=
<if_name>]
```

Deletes the CIFS server as identified by its NetBIOS name from the CIFS configuration of the Data Mover.

```
-delete compname=<comp_name> [-remove_localgroup]
[,alias=<alias_name>...][,interface=<if_name>]
```

Deletes the CIFS server as identified by its compname from the CIFS configuration of the Data Mover. This does not remove the account from the Active Directory. It is recommended that an **-Unjoin** be executed prior to deleting the computer name.



# **CAUTION**

The -remove\_localgroup option permanently deletes the local group information of the CIFS server from the permanent storage of the Data Mover. The alias and interface options delete the alias and the interface only, however, the CIFS server exists. The alias and interface options can be combined in the same delete command.

-delete wins=<ip\_addr>[,wins=<ip\_addr>...]
Deletes the WINS servers from the CIFS configuration.

-delete usrmapper=<ip\_addr>[,usrmapper= <ip\_addr>...] Deletes the IP addresses of a secondary Usermapper hosts from the CIFS configuration.

-update {<share\_name>|<path>}

Updates the attributes and their CIFS names for **COMPAT** filesystems. For every filesystem, CIFS maintains certain attributes for which there are no NFS equivalents. Updating CIFS attributes updates file attributes and CIFS names by searching the subdirectories of the defined share or path, generating a listing of Microsoft clients filenames (M8.3 and M256), and converting them to a format that CIFS supports. It is not necessary to use this command for DIR3 filesystems. Options include:

[mindirsize=<size>]

Updates the directories with the minimum size specified. Size must be typed in multiples of 512 bytes. A value of 0 ensures that all directories are rebuilt.

[force]

Forces a previous update to be overwritten.



#### **CAUTION**

The initial conversion of a directory can take considerable time when the directory contains a large number of files. Although the process is designed to take place in the background, an update should be run only during periods of light system usage.

```
-Migrate {<fs_name> -acl|<netbios_servername> -localgroup}<src_domain>{:nb=<netbios>|:if=<interface>}<dst_domain>{:nb=<netbios>|:if=<interface>}
Updates all security IDs (SIDs) from a <src_domain> to the SIDs of a <dst_domain> by matching the user and group account names in the source domain to the user and group account names in the destination domain. The interface that is specified in this option queries the local server, then its corresponding source and target Domain Controllers to look up each object's SID.
```

If **-acl** is specified, all secure IDs in the ACL database are migrated for the specified filesystem.

The **-localgroup option** must be used to migrate the SID members of local group defined for the specified NetBIOS name.

On the source domain, an interface specified to issue a lookup of the SID is defined by either the NetBIOS name or the interface name. On the destination domain, an interface specified to issue a lookup of the SID is defined by either the NetBIOS name or the interface name.

```
-Replace {<fs_name> -acl|<netbios_servername>
-localgroup}{::nb=<netbios>|:if=<interface>}
Replaces the history SIDs from the old domain with the new SIDS in the new domain. An interface that can be specified to issue a lookup of the SIDs is defined by the interface name or the NetBIOS name.
```

The **-localgroup** option must be used to migrate the SID members of the local group defined for the specified NetBIOS name. When the **-Replace** option is used, the user or group migrated in the new domain keeps their old SID in addition to the new SID created in the new domain.

The **-localgroup** option does the same kind of migration for a specified NetBIOS name in the local groups (instead of the ACL in a filesystem for the history argument).

#### -stats

Displays all non-zero CIFS statistics, including SMB statistics.

[-full [-zero]]

Lists all SMB statistics or zeroes out the values for SMB statistics.

[-summary]

Displays a summary of SMB statistics.

SEE ALSO

Configuring CIFS on Celerra, Managing Celerra for the Windows Environment, Using EMC Utilities for the CIFS Environment, Configuring CIFS on Celerra, Managing Celerra for a Multiprotocol Environment, Replicating Celerra CIFS Environments (V1), Using International Character Sets with Celerra, server\_date, server\_export, server\_mount, and server\_setup.

**OUTPUT NOTE** 

The network interface that appears in the output is dependent on the type of network interface cards that are installed. Dates appearing in the output are in UTC format.

**EXAMPLE #1** To display the number and names of open files on server\_2, type:

```
$ server_cifs server_2 -o audit,full
```

AUDIT Ctx=0xdffcc404, ref=2, Client(fm-main07B60004) Port=36654/139 NS40\_1[BRCSLAB] on if=cge0\_new CurrentDC 0xceeab604=W2K3PHYAD Proto=NT1, Arch=UNKNOWN, RemBufsz=0xfefb, LocBufsz=0xffff, popupMsg=1 0 FNN in FNNlist NbUsr=1 NbCnx=0 Uid=0x3f NTcred(0xcf156a04 RC=1 NTLM Capa=0x401) 'BRCSLAB\qustavo' CHECKER AUDIT Ctx=0xde05cc04, ref=2, XP Client(BRCSBARREGL1C) Port=1329/445 NS40\_1[BRCSLAB] on if=cge0\_new CurrentDC 0xceeab604=W2K3PHYAD Proto=NT1, Arch=Win2K, RemBufsz=0xffff, LocBufsz=0xffff, popupMsg=1 0 FNN in FNNlist NbUsr=1 NbCnx=2 Uid=0x3f NTcred(0xceeabc04 RC=3 NTLMSSP Capa=0x11001) 'BRCSLAB\qustavo' CHECKER Cnxp(0xceeaae04), Name=IPC\$, cUid=0x3f Tid=0x3f, Ref=1, Aborted=0 readOnly=0, umask=22, opened files/dirs=0 Cnxp(0xde4e3204), Name=gustavo, cUid=0x3f Tid=0x41, Ref=1, Aborted=0 readOnly=0, umask=22, opened files/dirs=2 Fid=64, FNN=0x1b0648f0(FREE,0x0,0), FOF=0x0 DIR=\ Notify commands received: Event=0x17, wt=0, curSize=0x0, maxSize=0x20, buffer=0x0 Tid=0x41, Pid=0xb84, Mid=0xec0, Uid=0x3f, size=0x20 Fid=73, FNN=0x1b019ed0(FREE,0x0,0), FOF=0xdf2ae504 (CHECK) FILE=\New Wordpad Document.doc

**EXAMPLE #2** 

To configure CIFS service on **server\_2** with a NetBIOS name of **dm110-cge0**, in the NT4 domain **NASDOCS**, with a NetBIOS alias of **dm110-cge0a1**, hiding the NetBIOS name in the Network Neighborhood, with the interface for CIFS service as **cge0**, the WINS

```
server as 172.24.102.25, and with the comment string EMC Celerra,
                        type:
                        $ server_cifs server_2 -add
                        netbios=dm110-cge0,domain=NASDOCS,alias=dm110-cge0a1,hid
                        den=y,interface=cge0,wins=172.24.102.25 -comment "EMC
                        Celerra"
server_2 : done
         EXAMPLE #3
                        To enable the home directory on server_2, type:
                        $ server_cifs server_2 -option homedir
server_2 : done
         EXAMPLE #4
                        To add the WINS servers, 172.24.103.25 and 172.24.102.25, type:
                        $ server_cifs server_2 -add
                        wins=172.24.103.25, wins=172.24.102.25
server_2 : done
         EXAMPLE #5
                        To rename the NetBIOS name from dm110-cge0 to dm112-cge0, type:
                        $ server_cifs server_2 -rename -netbios dm110-cge0
                        dm112-cge0
server_2 : done
         EXAMPLE #6
                        To display the CIFS configuration for NT4 with Internal Usermapper,
                        type:
                        $ server_cifs server_2
server_2 :
256 Cifs threads started
Security mode = NT
Max protocol = NT1
I18N mode = UNICODE
Home Directory Shares ENABLED, map=/.etc/homedir
Usermapper auto broadcast enabled
Usermapper[0] = [127.0.0.1] state:active port:14640 (auto discovered)
Default WINS servers = 172.24.103.25:172.24.102.25
Enabled interfaces: (All interfaces are enabled)
Disabled interfaces: (No interface disabled)
DOMAIN NASDOCS RC=3
 SID=S-1-5-15-99589f8d-9aa3a5f-338728a8-ffffffff
>DC=WINSERVER1(172.24.102.66) ref=2 time=0 ms
CIFS Server DM112-CGE0[NASDOCS] RC=2 (Hidden)
 Alias(es): DM110-CGE0A1
 Comment='EMC Celerra'
 if=cge0 l=172.24.102.242 b=172.24.102.255 mac=0:60:16:4:35:4f
```

wins=172.24.102.25
Password change interval: 0 minutes

# Where:

To a contract of the contract	
Value	Definition
Cifs threads started	Number of CIFS threads used when the CIFS service was started.
Security mode	User authorization mechanism used by the Data Mover.
Max protocol	Maximum dialect supported by the security mode.
I18N mode	L18N mode (unicode or ASCII).
Home Directory Shares	Whether Home Directory shares are enabled.
map	Home directory used by the Data Mover.
Usermapper auto broadcast enabled	Usermapper is using its broadcast mechanism to discover its servers. This only displays when the mechanism is active. It is disabled when you manually set the Usermapper server addresses.
Usermapper	IP address of the servers running the Usermapper service.
state	Current state of Usermapper.
Default WINS servers	Addresses of the default WINS servers.
Enabled interfaces	Data Mover's enabled interfaces.
Disabled interfaces	Data Mover's disabled interfaces.
Unused Interface(s)	Interfaces not currently used by the Data Mover.
RC	Reference count indicating the number of internal objects (such as client contexts) using the CIFS server.
SID	Security ID of the domain.
DC	Domain controllers used by the Data Mover. Depending on the number of DCs in the domain, this list may be large.
ref	Number of internal objects using the Domain Controller.
time	Domain Controller response time.
Alias(es)	Alternate NetBIOS names assigned to the CIFS server configuration.
if	Interfaces used by the CIFS server.
Password change interval:	The amount of time between password changes.

# **EXAMPLE #7** To display the CIFS configuration for NT4, type:

# \$ server\_cifs server\_2

```
server_2 :
256 Cifs threads started
Security mode = NT
Max protocol = NT1
I18N mode = UNICODE
Home Directory Shares ENABLED, map=/.etc/homedir
Usermapper auto broadcast suspended
Usermapper[0] = [172.24.102.20] state:available
```

```
Default WINS servers = 172.24.103.25:172.24.102.25
Enabled interfaces: (All interfaces are enabled)

Disabled interfaces: (No interface disabled)

DOMAIN NASDOCS RC=3
   SID=S-1-5-15-99589f8d-9aa3a5f-338728a8-fffffffff
>DC=WINSERVER1(172.24.102.66) ref=2 time=0 ms

CIFS Server DM112-CGE0[NASDOCS] RC=2 (Hidden)
   Alias(es): DM110-CGE0A1
   Comment='EMC Celerra'
   if=cge0 l=172.24.102.242 b=172.24.102.255 mac=0:60:16:4:35:4f
        wins=172.24.102.25
   Password change interval: 0 minutes
```

#### **EXAMPLE #8**

To add Microsoft Windows server using the compname dm112-cge0, in the Active Directory domain nasdocs.emc.com, with a NetBIOS alias of dm112-cge0a1, hiding the NetBIOS name in the Network Neighborhood, with the interface for CIFS service as cge0, the WINS servers as 172.24.102.25 and 172.24.103.25, in the DNS domain nasdocs.emc.com, and with the comment string EMC Celerra, type:

\$ server\_cifs server\_2 -add
compname=dm112-cge0,domain=nasdocs.emc.com,alias=dm112-c
ge0a1,hidden=y,interface=cge0,wins=172.24.102.25:172.24.
103.25,dns=nasdocs.emc.com -comment "EMC Celerra"

server 2 : done

#### **EXAMPLE #9**

To join dm112-cge0 into the Active Directory domain nasdocs.emc.com, using the Administrator account, and to add this server to Engineering \Computers organizational unit, type:

\$ server\_cifs server\_2 -Join
compname=dm112-cge0,domain=nasdocs.emc.com,admin=adminis
trator,ou="ou=Computers:ou=Engineering"

server\_2 : Enter Password:\*\*\*\*\*\*
done

#### EXAMPLE #10

To add the NFS service to the CIFS server to make it possible for NFS users to access the Windows KDC, type:

\$ server\_cifs server\_2 -Join
compname=dm112-cge0,domain=nasdocs.emc.com,admin=adminis
trator -option addservice=nfs

server\_2 : Enter Password:\*\*\*\*\*\*

done

# **EXAMPLE #11** To enable the cge1 interface, type:

```
$ server_cifs server_2 -Enable cge1
server_2 : done
        EXAMPLE #12
                        To display CIFS information for a Data Mover in a Windows domain
                        with internal usermapper, type:
                        $ server_cifs server_2
server_2 :
256 Cifs threads started
Security mode = NT
Max protocol = NT1
I18N mode = UNICODE
Home Directory Shares ENABLED, map=/.etc/homedir
Usermapper auto broadcast enabled
Usermapper[0] = [127.0.0.1] state:active (auto discovered)
Default WINS servers = 172.24.103.25:172.24.102.25
Enabled interfaces: (All interfaces are enabled)
Disabled interfaces: (No interface disabled)
Unused Interface(s):
 if=cge1 1=172.24.102.243 b=172.24.102.255 mac=0:60:16:4:35:4e
DOMAIN NASDOCS FODN=nasdocs.emc.com SITE=Default-First-Site-Name RC=3
 SID=S-1-5-15-99589f8d-9aa3a5f-338728a8-ffffffff
>DC=WINSERVER1(172.24.102.66) ref=3 time=1 ms (Closest Site)
CIFS Server DM112-CGE0[NASDOCS] RC=2 (Hidden)
 Alias(es): DM112-CGEA1
 Full computer name=dm112-cge0.nasdocs.emc.com realm=NASDOCS.EMC.COM
 Comment='EMC Celerra'
 if=cge0 l=172.24.102.242 b=172.24.102.255 mac=0:60:16:4:35:4f
         wins=172.24.102.25:172.24.103.25
  FQDN=dm112-cge0.nasdocs.emc.com (Updated to DNS)
 Password change interval: 30 minutes
 Last password change: Thu Oct 27 15:59:17 2005
 Password versions: 2
        EXAMPLE #13
                        To display CIFS information for a Data Mover in a Window domain,
                        type:
                        $ server_cifs server_2
server 2 :
256 Cifs threads started
Security mode = NT
Max protocol = NT1
I18N mode = UNICODE
Home Directory Shares ENABLED, map=/.etc/homedir
Usermapper auto broadcast suspended
```

```
Usermapper[0] = [172.24.102.20] state:available
Default WINS servers = 172.24.103.25:172.24.102.25
Enabled interfaces: (All interfaces are enabled)
Disabled interfaces: (No interface disabled)
Unused Interface(s):
if=cge1 1=172.24.102.243 b=172.24.102.255 mac=0:60:16:4:35:4e
DOMAIN NASDOCS FODN=nasdocs.emc.com SITE=Default-First-Site-Name RC=3
SID=S-1-5-15-99589f8d-9aa3a5f-338728a8-ffffffff
>DC=WINSERVER1(172.24.102.66) ref=3 time=1 ms (Closest Site)
CIFS Server DM112-CGE0[NASDOCS] RC=2 (Hidden)
Alias(es): DM112-CGEA1
Full computer name=dm112-cge0.nasdocs.emc.com realm=NASDOCS.EMC.COM
Comment='EMC Celerra'
if=cge0 1=172.24.102.242 b=172.24.102.255 mac=0:60:16:4:35:4f
         wins=172.24.102.25:172.24.103.25
 FQDN=dm112-cge0.nasdocs.emc.com (Updated to DNS)
Password change interval: 30 minutes
Last password change: Thu Oct 27 16:29:21 2005
Password versions: 3, 2
        EXAMPLE #14
                       To display CIFS information for a Data Mover when CIFS service is
                       not started, type:
                       $ server_cifs server_2
server_2 :
Cifs NOT started
Security mode = NT
Max protocol = NT1
I18N mode = UNICODE
Home Directory Shares ENABLED, map=/.etc/homedir
Usermapper auto broadcast suspended
Usermapper[0] = [172.24.102.20] state:available
Default WINS servers = 172.24.103.25:172.24.102.25
Enabled interfaces: (All interfaces are enabled)
Disabled interfaces: (No interface disabled)
Unused Interface(s):
if=cge1 1=172.24.102.243 b=172.24.102.255 mac=0:60:16:4:35:4e
CIFS Server DM112-CGE0[NASDOCS] RC=2 (Hidden)
Alias(es): DM112-CGEA1
Full computer name=dm112-cge0.nasdocs.emc.com realm=NASDOCS.EMC.COM
Comment='EMC Celerra'
if=cge0 1=172.24.102.242 b=172.24.102.255 mac=0:60:16:4:35:4f
```

```
wins=172.24.102.25:172.24.103.25
 FQDN=dm112-cge0.nasdocs.emc.com (Updated to DNS)
 Password change interval: 30 minutes
Last password change: Thu Oct 27 16:29:21 2005
 Password versions: 3, 2
        EXAMPLE #15
                       To add Microsoft Windows server named, dm112-cge0, in the Active
                       Directory domain nasdocs.emc.com, with the interface for CIFS
                       service as cge0, and enable local users support, type:
                       $ server_cifs server_2 -add
                       compname=dm112-cge0, domain=nasdocs.emc.com, interface=cge
                       0,local users
server_2 : Enter Password: ******
Enter Password Again: ******
done
        EXAMPLE #16
                       To set a security mode to NT for a Data Mover, type:
                       $ server_cifs server_2 -add security=NT
server_2 : done
        EXAMPLE #17
                       To disable a CIFS interface, type:
                       $ server_cifs server_2 -Disable cge1
server_2 : done
        EXAMPLE #18
                       To display CIFS audit information for a Data Mover, type:
                       $ server_cifs server_2 -option audit
server 2 :
|||| AUDIT Ctx=0xad3d4820, ref=1, W2K3 Client(WINSERVER1) Port=1638/139
| | DM112-CGE0[NASDOCS] on if=cge0
| | CurrentDC 0xad407620=WINSERVER1
 Proto=NT1, Arch=Win2K, RemBufsz=0xffff, LocBufsz=0xffff
| | 0 FNN in FNNlist NbUsr=1 NbCnx=1
| | Uid=0x3f NTcred(0xad406a20 RC=2 KERBEROS Capa=0x2) 'NASDOCS\administrator'
 Cnxp(0xad3d5420), Name=IPC$, cUid=0x3f Tid=0x3f, Ref=1, Aborted=0
readOnly=0, umask=22, opened files/dirs=1
| | DM112-CGE0[NASDOCS] on if=cge0
| | CurrentDC 0xad407620=WINSERVER1
Proto=NT1, Arch=Win2K, RemBufsz=0xffff, LocBufsz=0xffff
 | 0 FNN in FNNlist NbUsr=1 NbCnx=1
| | Uid=0x3f NTcred(0xad362c20 RC=2 KERBEROS Capa=0x2) 'NASDOCS\user1'
| Cnxp(0xaec21020), Name=IPC$, cUid=0x3f Tid=0x3f, Ref=1, Aborted=0
readOnly=0, umask=22, opened files/dirs=2
```

# Where:

Value	Definition
Ctx	Address in memory of the Stream Context.
ref	Reference counter of components using this context at this time.
Port	The client port and the Data Mover port used in the current TCP connection.
CurrentDC	Specify the address of the Domain Controller that is currently used.
Proto	Dialect level that is currently used.
Arch	Type of the client OS.
RemBufsz	Max buffer size negotiated by the client.
LocBufsz	Max buffer size we have negotiated.
FNN/FNNlist	Number of blocked files that has not yet been checked by Virus Checker.
NbUsr	Number of sessions connected to the stream context (TCP connection).
NbCnx	Number of connections to shares for this TCP connection.
Uid/NTcred	User Id (this number is not related to the UNIX UID used to create a file), the credential address, and the type of authentication.
Cnxp/Name	Share connection address' and the name of the share the user is connecting to.
cUid	User Id who has opened the connection first.
Tid	Tree Id (number which represents the share connection in any protocol request).
Aborted	Status of the connection.
readOnly	If the share connection is read-only.
umask	A user file-creation mask.
opened files/dirs	Number of files or directories opened on this share connection.

# **EXAMPLE #19**

To unjoin the computer **dm112-cge0** from the **nasdocs.emc.com** domain, type:

\$ server\_cifs server\_2 -Unjoin
compname=dm112-cge0,domain=nasdocs.emc.com,admin=adminis
trator

server\_2 : Enter Password:\*\*\*\*\*\*
done

## EXAMPLE #20

To delete WINS servers, 172.24.102.25, and 172.24.103.25, type:

\$ server\_cifs server\_2 -delete
wins=172.24.102.25,wins=172.24.103.25

server\_2 : done

## **EXAMPLE #21**

To delete a NetBIOS name, **dm112-cge0**, type:

\$ server\_cifs server\_2 -delete netbios=dm112-cge0

server\_2 : done

#### **EXAMPLE #22**

To delete the compname, **dm112-cge0**, type:

\$ server\_cifs server\_2 -delete compname=dm112-cge0

server\_2 : done

**EXAMPLE #23** To delete the usermapper, **172.24.102.20**, type:

\$ server\_cifs server\_2 -delete usrmapper=172.24.102.20

server\_2 : done

**EXAMPLE #24** To add Microsoft Windows server in disjoint DNS and Windows domains, type:

\$ server\_cifs server\_2 -add
compname=dm112-cge0,domain=nasdocs.emc.com,netbios=dm112
-cge0,interface=cge0,dns=eng.emc.com -comment "EMC
Celerra"

server\_2 : done

**EXAMPLE #25** To add Microsoft Windows server using a delegated account from a trusted domain, type:

\$ server\_cifs server\_2 -Join
compname=dm112-cge0,domain=nasdocs.emc.com,admin=delegat
eduser@it.emc.com

server\_2 : Enter Password: \*\*\*\*\*\*

done

**EXAMPLE #26** To add Microsoft Windows server in the Active Directory domain using a pre-created computer account, type:

\$ server\_cifs server\_2 -Join
compname=dm112-cge0,domain=nasdocs.emc.com,admin=adminis
trator -option reuse

server\_2 : Enter Password: \*\*\*\*\*\*

done

**EXAMPLE #27** To update the directory **/ufs1/users** with a new minimum directory size of **8192**, type:

\$ server\_cifs server\_2 -update /ufs1/users
mindirsize=8192

server\_2 : done

**EXAMPLE #28** To migrate all SIDs in the ACL database for filesystem, **ufs1**, from the <src\_domain>, **eng.emc.com:nb=dm112-cge1:if=cge1** to the <dst\_domain>, **nasdocs.emc.com:nb=dm112-cge0:if=cge0**, type:

\$ server\_cifs server\_2 -Migrate ufs1 -ac1
eng.emc.com:nb=dm112-cge1:if=cge1
nasdocs.emc.com:nb=dm112-cge0:if=cge0

server\_2 : done

# **EXAMPLE #29** To migrate SIDs of members of the local group defined for the specified NetBIOS name, from the <src\_domain>, eng.emc.com:nb=dm112-cge1:if=cge1 to the <dst\_domain>, nasdocs.emc.com:nb=dm112-cge0:if=cge0, type: \$ server\_cifs server\_2 -Migrate dm112-cge1 -localgroup eng.emc.com:nb=dm112-cge1:if=cge1 nasdocs.emc.com:nb=dm112-cge0:if=cge0 server\_2 : done EXAMPLE #30 To replace the SIDs for **ufs1**, type: \$ server\_cifs server\_2 -Replace ufs1 -acl :nb=dm112-cge0:if=cge0 server\_2 : done **EXAMPLE #31** To configure a stand-alone CIFS server on **server\_2** with a NetBIOS name of **dm112-cge0**, in the workgroup **NASDOCS**, with a NetBIOS alias of dm112-cge0a1, hiding the NetBIOS name in the Network Neighborhood, with the interface for CIFS service as **cge0**, the WINS servers as 172.24.102.25 and 172.24.103.25, and with enabled local users support, type: \$ server\_cifs server\_2 -add standalone=dm112-cge0, workgroup=NASDOCS, alias=dm112-cge0 a1, hidden=y, interface=cge0, wins=172.24.102.25:172.24.103 .25,local\_users server\_2 : Enter Password: \*\*\*\*\*\* Enter Password Again: \*\*\*\*\*\* EXAMPLE #32 To delete the stand-alone CIFS server, **dm112-cge0**, type: \$ server\_cifs server\_2 -delete standalone=dm112-cge0

server\_2 : done

done

**EXAMPLE #33** To display a summary of SMB statistics, type:

\$ server cifs server 2 -stats -summary

server\_2 : State info: Open connection Open files

2

SMB total requests:

totalAllSmb totalSmb totalTrans2Smb totalTransNTSmb 6593 10038 3437

> **EXAMPLE #34** To display all non-zero CIFS statistics, type:

> > \$ server\_cifs server\_2 -stats

server\_2 :

SMB statistics:						
proc	ncalls	%totcalls	maxT	ime	ms/call	
Close	1305	7.96	46.2	1	2.16	
Rename	2	0.01	0.81		0.50	
Trans	314	1.91	0.77		0.08	
Echo	21	0.13	0.01		0.00	
ReadX	231	1.41	0.03		0.00	
WriteX	3697	22.54	39.9	6	0.98	
Trans2Prim	9375	57.16	34.2	7	0.46	
TreeDisco	10	0.06	0.06		0.00	
NegProt	29	0.18	0.42		0.24	
SessSetupX	47	0.29	60.5	5	5.81	
UserLogoffX	9	0.05	0.01		0.00	
TreeConnectX	13	0.08	0.39		0.23	
TransNT	8	0.05	0.01		0.00	
CreateNTX	1338	8.16	47.1	1	0.81	
CancelNT	1	0.01	0.03		0.00	
Trans2 SMBs:						
proc	ncalls	%totcalls	maxT	limo	ms/call	
FindFirst	22	0.23	0.22		0.09	
OFsInfo	3154	33.65	0.22		0.05	
QPathInfo	1113	11.87	6.73		0.03	
OFileInfo	2077	22.16	0.04		0.13	
SetFileInfo	3007	32.08	34.2		1.28	
Secrifeino	3007	32.00	34.2	0	1.20	
NT SMBs:						
proc	ncalls	%totcalls	maxT		ms/call	
NotifyChange	8	100.00	0.01		0.00	
Performance inf	o:					
	Re/s	Write	Wr/s	All		Ops/sec
	231000.00	3697	1021.27	25783		1575.40
State info:						
Open connection	Open files					
2	2					
Q11						
Shadow info:	T-7	0-1:5-				_
Reads	Writes	Splits		nsert	Truncates	Ď.
0	0	0	0		0	
SMB total requests:						
totalAllSmb	totalSmb	totalTrans2		lTransNTSmb		rted)
25783	16400	9375	8		2	
N. 17						

# Where:

Value	Definition
proc	Name of CIFS requests received.
ncalls	Number of requests received.
%totcalls	Percentage of this type of request compared to all requests.

### **The server Commands**

maxTime	Maximum amount of time used.
ms/call	Average time in milliseconds taken to service calls.
failures	Number of times the call has failed.
Read	Total number of read operations.
Re/s	Number of read operations per second.
Write	Total number of write operations.
Wr/s	Number of write operations per second.

**EXAMPLE #35** To reset to zero the values for all SMB statistics, type:

\$ server\_cifs server\_2 -stats -zero

server\_2 : done

**EXAMPLE #36** 

To configure CIFS service in a language that uses multibyte characters (such as Japanese), type:

\$ server\_cifs server\_2 -add compname= かれんの共有名,domain=nasdocs.emc.com, -comment まだかいな / オリビアさん

server\_2 : done

# server\_cifssupport

Provides support services for CIFS users.

## **SYNOPSIS**

```
server_cifssupport {<movername> | ALL}
  -accessright
     {-name <name> [-domain <domain_name>]
        | -sid <SID>|-uname <unix_name>|-uid <user_id>}
        {-path <pathname>|-share <sharename>}
        [-policy {mixed|native|secure|nt|unix}]
        [-build [-admin <admin_name>]]
        [-netbios <netbios_servername> | -compname <comp_name>
         -standalone <netbios_ name>]
   -acl {-path <pathname> | -share <sharename>} [-verbose]
   -cred
     {-name <name> [-domain <domain_name>]
         -sid <SID> | -uname <unix_name> | -uid <user_id> }
        [-build [-ldap][-admin <admin_name>]]
        [-netbios <netbios_servername> | -compname <comp_name>
          -standalone <netbios_name>]
  -pingdc
     {-netbios <netbios_servername>|-compname <comp_name>}
        [-dc <netbios_Dcname>]
        [-verbose]}
  -secmap
     -list
        [-name <name> [-domain<domain_name>
          -domain <domain_name>
          -sid <SID>
          -uid <user id>
          -gid <qroup_id>]
     -create
        {-name <name>-domain<domain_name>]}
         -sid <SID>}
     -verify
        {-name <name>[-domain<domain name>]}
        -sid <SID>}
     -update
        {-name <name>[-domain<domain_name>]}
         -sid <SID>}
      -delete
        {-name <name> [-domain<domain_name>]}
        -sid <SID>}
      -export [-file <filename>]
      -import -file <filename>
      -report
      -migration
```

## **DESCRIPTION**

**server\_cifssuport** checks network connectivity between a CIFS server and domain controller, manages access rights, generates credentials, and manages secure mapping cache.

# The **-accessright** option:

- Displays user access rights to a file, directory, or share in a Windows permission mask.
- Rebuilds and displays a credential for users of a file, directory, or share, who do not have a session opened in one of the CIFS servers.
- Without impact on the actual access-checking policy of a filesystem object, shows how user permissions are reset if you were to change the policy on that object.

The **-acl** option displays the access control list (ACL) of files, directories, or shares in plain text form.

The **-cred** option generates a credential containing all groups to which a user belongs, including local groups, without the user being connected to a CIFS server. This allows you to verify if user's SIDs are being correctly mapped to UNIX UIDs and GIDs and to troubleshoot any user access control issues.

The **-pingdc** option checks the network connectivity between a CIFS server and a domain controller then verifies that a CIFS server can access and use the following domain controller services:

- ♦ IPC\$ share logon
- Secure Channel when verifying domain users during NT LAN Manager (NTLM) authentication
- Local Security Authority (LSA) pipe information when mapping Windows SIDs to UNIX UIDs and GIDs
- SAMR (Remote Security Account Manager) pipe when merging a user's UNIX and Windows groups together to create a credential
- Trusted domain information
- Privilege names for internationalization: pingdc

The **-secmap** option manages the secure mapping (secmap) cache. Secmap contains all mapping between SIDs and UID/GIDs used by a Data Mover or Virtual Data Mover (VDM). The Data Mover permanently caches all mappings it receives from any mapping mechanism (local files, NIS, iPlanet, Active Directory, and Usermapper) in the secmap database, making the response to

subsequent mapping requests faster and less susceptible to network problems. Reverse mapping provides better quota support.

# ACCESS RIGHT OPTIONS

```
-accessright {-name <name> [-domain <domain_name>] |
-sid <SID>|-uname <unix_name>|-uid <user_id>}
{-path <pathname>|-share <sharename>}
```

Displays user access rights to a file, directory, or share in a Windows permission mask for the specified:

 Windows username and the optional domain to which the user belongs

or

◆ <SID> which is the the user's Windows security identifier

or

<unix\_name>

or

<user\_id> which is the user's UNIX identifier

The **-path** option specifies the path of the file or directory to check for user permissions, or the absolute path of the share to check for user permissions.

```
[-policy {mixed|native|secure|nt|unix}]
```

Specifies an access-checking policy for the specified file, directory, or share. This does not change the current access-checking policy, instead it helps you anticipate any access problems before actually resetting the policy on a filesystem object. server\_cifs provides more information.

```
[-build [-admin <admin_name>]]
```

Rebuilds a credential for a user of a file, directory, or share, who does not have a session opened in one of the CIFS servers. If **-build** is not specified, the system searches the known user credentials in cache. If none are found, an error message is generated. The **-admin** option specifies the name of an administrative user to use for creating the access right list. The password of the admin\_name user is prompted when executing the command.

```
[-netbios <netbios_servername>|-compname <comp_
name>|-standalone <netbios_ name>]}
```

Indicates the CIFS server, as specified by its NetBIOS name or computer name to use when rebuilding the user credential.

The **-standalone** option specifies the stand-alone CIFS server, as specified by its name, to use when rebuilding a user credential.

**Note:** If no CIFS server is specified, the system uses the default CIFS server, which uses all interfaces not assigned to other CIFS servers on the Data Mover.

### **ACL OPTIONS**

-acl {-path <pathname>|-share <sharename>} [-verbose] Displays the ACL of a file, directory, or a share in plain text form. Windows or UNIX access control data are both displayed in their native forms. The -verbose option displays the ACE access rights mask in plain text form in addition to their native forms.

#### **CREDENTIAL OPTIONS**

```
-cred {-name <name> [-domain <domain_name>] |-sid
<SID>|-uname <unix_name>|-uid <user_id>}
Generates a credential containing all of the groups to which a user
```

Generates a credential containing all of the groups to which a user belongs without being connected to a CIFS server. The credential is specified by the user's:

Windows username and the domain to which the user belongs

or

Windows security identifier

or

UNIX name

or

UNIX identifier

```
[-build [-ldap][-admin <admin_name>]]
Rebuilds a user credential. If -build is not specified, the system searches the known user credentials in cache. If none are found, an error message is generated. The -ldap option retrieves the user's universal groups to be included in the credential. If none are found, no universal groups are incorporated into the credential. The -admin option indicates the name of an administrative user for creating the credential. The password of
```

the <admin\_name> is prompted when executing the command.

```
[-netbios <netbios_servername>|-compname <comp_name>|-standalone <netbios_name>]}
Indicates the CIFS server, as specified by its NetBIOS name or computer name to use when rebuilding the user credential.
```

The **-standalone** option specifies the stand-alone CIFS server to use when rebuilding a user credential.

**Note:** If no CIFS server is specified, the system uses the default CIFS server, which uses all interfaces not assigned to other CIFS servers on the Data Mover.

### **PINGDC OPTIONS**

```
-pingdc {-netbios <netbios_servername>|-compname
<comp_ name>}
```

Checks the network connectivity for the CIFS server as specified by its NetBIOS name or by its computer name with a domain controller. Once connectivity is established, it verifies that a CIFS server can access and use the domain controller services.

**Note:** An IP address can be used for the <netbios\_servername> and the <comp\_name>.

```
[-dc <netbios_Dcname>]
```

Indicates the domain controller to ping for network and resource connectivity with the CIFS server. If not specified, the command checks the domain controllers currently in use by the CIFS server.

**Note:** An IP address can be used for the <netbios\_Dcname>.

```
[-verbose]
```

Adds troubleshooting information to the command output.

#### **SECMAP OPTIONS**

#### -secmap -list

Lists the secmap mapping entries.

**-secmap -list -name** <name> **-domain** <domain\_name> Lists the secmap mapping entries with the specified name and domain name.

```
-secmap -list -domain <domain_name>
```

Lists the secmap mapping entries with the specified domain name.

```
-secmap -list -sid <SID>
```

Lists the secmap mapping entries with the specified SID.

```
-secmap -list -uid <user_id>
```

Lists the secmap mapping entries with the specified UID (reverse mapping).

-secmap -list -gid <group\_id>

Lists the secmap mapping entries with the specified GID (reverse mapping).

- **-secmap -create** {-name <name> [-domain <domain\_name>]} Creates the secmap mapping entry with the specified name and domain name.
- -secmap -create -sid <SID>

Creates the secmap mapping entry with the specified SID.

**-secmap -verify** {**-name** <name> [**-domain** <domain\_name>]} Checks the mapping entry stored in secmap with the specified name and optional domain name with what is currently available in the mapping sources. If a mapping has changed, it is marked.

-secmap -verify -sid <SID>

Checks the secmap mapping entry with the specified SID.

**-secmap -update** {-name <name> [-domain <domain\_name>]} Updates the specified mapping entry stored in secmap with the mappings currently available in the mapping sources. Once this option is performed, force an update of the Data Mover's filesystems ACLs so that the new mappings are recognized.

-secmap -update -sid <SID>

Updates the secmap mapping entry with the specified SID.

**-secmap -delete -name** <name> [**-domain** <domain\_name>] Deletes the secmap mapping entry with the specified name and domain name.

-secmap -delete -sid <SID>

Deletes the secmap mapping entry with the specified SID.

-secmap -export [-file <filename>]

Exports the secmap mapping entry to the specified file.

**Note:** If no filename is specified, the secmap database is displayed on the screen.

-secmap -import -file <filename>

Imports secmap mapping entries from the specified file.

-secmap -report

Displays current secmap status, including database state, domains handled by secmap, and resource usage.

#### -secmap -migration

Displays secmap database migration information like start date and end date of the operation, and migration status.

# **EXAMPLE #1** To display user access rights to a file for **user1**, type:

\$ server\_cifssupport server\_2 -accessright -name user1
-domain NASDOCS -path /ufs1/test/test.txt

server\_2 : done

#### ACCOUNT GENERAL INFORMATIONS

Name : user1
Domain : NASDOCS

Path : /ufs1/test/test.txt

Allowed mask : 0x200a9

Action : List Folder / Read data
Action : Read Extended Attributes
Action : Traverse Folder / Execute File

Action : Read Attributes Action : Read Permissions

# **EXAMPLE #2** To rebuild a credential for a user to a file using the SID, type:

\$ server\_cifssupport server\_2 -accessright -sid
S-1-5-15-b8e641e2-33f0942d-8f03a08f-1f4 -path
/ufs1/test/test.txt -build -compname dm102-cge0

server 2 : done

#### ACCOUNT GENERAL INFORMATIONS

Path : /ufs1/test/test.txt

Allowed mask : 0x301ff

Action : List Folder / Read data
Action : Create Files / Write data
Action : Create Folders / Append Data
Action : Read Extended Attributes
Action : Write Extended Attributes
Action : Traverse Folder / Execute File
Action : Delete Subfolders and Files

Action : Read Attributes
Action : Write Attributes

Action : Delete

Action : Read Permissions

# **EXAMPLE #3** To display user access rights to a file for **user1** with access-checking policy UNIX, type:

\$ server\_cifssupport server\_2 -accessright -name user1
-domain NASDOCS -path /ufs1/test/test.txt -policy unix

server\_2 : done

#### ACCOUNT GENERAL INFORMATIONS

Name : user1
Domain : NASDOCS

Path : /ufs1/test/test.txt

Allowed mask : 0x20089

Action : List Folder / Read data
Action : Read Extended Attributes

Action : Read Attributes
Action : Read Permissions

# **EXAMPLE #4** To rebuild a credential for **user1** to a file using an administrative

account, type:

\$ server\_cifssupport server\_2 -accessright -name user1
-domain NASDOCS -path /ufs1/test/test.txt -build -admin

administrator

server\_2 : Enter Password:\*\*\*\*\*\*

done

#### ACCOUNT GENERAL INFORMATIONS

Name : user1
Domain : NASDOCS

Path : /ufs1/test/test.txt

Allowed mask : 0x200a9

Action : List Folder / Read data
Action : Read Extended Attributes
Action : Traverse Folder / Execute File

Action : Read Attributes
Action : Read Permissions

# **EXAMPLE #5** To display the verbose ACL information of a file, type:

\$ server\_cifssupport server\_2 -acl -path
/ufs1/test/test.txt -verbose

server\_2 : done

#### ACL DUMP REPORT

Path : /ufs1/test/test.txt

UID : 32770
GID : 32797
Rights : rw-r--racl ID : 0x4
acl size : 174

owner SID : S-1-5-20-220

group SID : S-1-5-15-b8e641e2-33f0942d-8f03a08f-201

DACL

Owner : USER 32770 S-1-5-15-b8e641e2-33f0942d-8f03a08f-1f4

Access : ALLOWED 0x0 0x1f01ff RWXPDO Rights : List Folder / Read data

Create Files / Write data
Create Folders / Append Data
Read Extended Attributes
Write Extended Attributes
Traverse Folder / Execute File
Delete Subfolders and Files

Read Attributes Write Attributes

Delete

Read Permissions Change Permissions Take Ownership Synchronize

Owner : USER 32771 S-1-5-15-b8e641e2-33f0942d-8f03a08f-a59

Access : ALLOWED 0x0 0x1200a9 R-X--Rights : List Folder / Read data
Read Extended Attributes

Traverse Folder / Execute File

Read Attributes Read Permissions Synchronize

# **EXAMPLE #6** To display the access control level of a share, type:

\$ server\_cifssupport server\_2 -acl -share ufs1

server\_2 : done
ACL DUMP REPORT

Share : ufs1 UID : 0 GID : 1

Rights : rwxr-xr-x

# **EXAMPLE #7** To generate a credential for **user1**, type:

\$ server\_cifssupport server\_2 -cred -name user1 -domain
NASDOCS

server\_2 : done

ACCOUNT GENERAL INFORMATIONS

Name : user1
Domain : NASDOCS

Primary SID : S-1-5-15-b8e641e2-33f0942d-8f03a08f-a59

UID : 32771
GID : 32768
Authentification : KERBEROS
Credential capability : 0x2

Privileges : 0x8
System privileges : 0x2
Default Options : 0x2
NT administrator : False
Backup administrator : False
Backup : False
NT credential capability : 0x2

#### ACCOUNT GROUPS INFORMATIONS

Type UNIX ID Domain SID Name NT 32797 S-1-5-15-b8e641e2-33f0942d-8f03a08f-201 32798 S-1-5-15-b8e641e2-33f0942d-8f03a08f-e45 NТ 4294967294 S-1-1-0S-1-5-2 NT4294967294 NT 4294967294 S-1-5-b NT 2151678497 S-1-5-20-221 UNIX 32797 UNIX 32798 UNIX 4294967294

**EXAMPLE #8** To rebuild a user credential including the user's universal groups for a user using SID, type:

\$ server\_cifssupport server\_2 -cred -sid
S-1-5-15-b8e641e2-33f0942d-8f03a08f-1f4 -build -ldap
-compname dm102-cge0

server\_2 : done

UNIX 2151678497

#### ACCOUNT GENERAL INFORMATIONS

Name : NASDOCS

Server : dm102-cge0

Primary SID : S-1-5-15-b8e641e2-33f0942d-8f03a08f-1f4

UID : 32770
GID : 32768
Authentification : NTLM
Credential capability : 0x0
Privileges : 0x7f
System privileges : 0x1
Default Options : 0xe
NT administrator : True
Backup administrator : True
Backup : False
NT credential capability : 0x0

ACCOUNT GROUPS INFORMATIONS

```
Type UNIX ID
               Name
                                   Domain
                                                       SID
    32794
              Group Policy Cre NASDOCS S-1-5-15-b8e641e2-33f0942d-8f03a08f-208
NT
              Schema Admins NASDOCS S-1-5-15-b8e641e2-33f0942d-8f03a08f-206
NT
    32795
             Enterprise Admin
NT
    32796
                                NASDOCS S-1-5-15-b8e641e2-33f0942d-8f03a08f-207
              Domain Users
    32797
                                NASDOCS S-1-5-15-b8e641e2-33f0942d-8f03a08f-201
NТ
   32793
              Domain Admins
                               NASDOCS S-1-5-15-b8e641e2-33f0942d-8f03a08f-200
NT
    4294967294 Everyone
                                                      S-1-1-0
NT
    4294967294 NETWORK
                                                      S-1-5-2
NT
                                   NT AUTHORITY
NT
    4294967294 ANONYMOUS LOGON
                                  NT AUTHORITY
                                                      S-1-5-7
    2151678496 Administrators
NT
                                  BUILTIN
                                                      S-1-5-20-220
    2151678497 Users
                                                      S-1-5-20-221
NT
                                   BUILTIN
                                                      S-1-5-12-2-1
NT
    1
               UNIX GID=0x1 &ap
UNIX 32794
UNIX 32795
UNIX 32796
UNIX 32797
UNIX 32793
```

# **EXAMPLE #9** To check the network connectivity for the CIFS server with **netbios dm102-cge0**, type:

\$ server\_cifssupport server\_2 -pingdc -netbios dm102-cge0

server\_2 : done

PINGDC GENERAL INFORMATIONS

DC SERVER:

Netbios name : NASDOCSDC

CIFS SERVER :

Compname : dm102-cge0
Domain : nasdocs.emc.com

### **EXAMPLE #10**

To check the network connectivity between the domain controller and the CIFS server with **compname dm102-cge0**, type:

\$ server\_cifssupport server\_2 -pingdc -compname
dm102-cge0 -dc NASDOCSDC -verbose

server\_2 : done

PINGDC GENERAL INFORMATIONS

DC SERVER:

Netbios name : NASDOCSDC

CIFS SERVER :

Compname : dm102-cge0
Domain : nasdocs.emc.com

# **EXAMPLE #11** To display the secmap mapping entries, type:

\$ server\_cifssupport server\_2 -secmap -list

```
server_2 : done
SECMAP USER MAPPING TABLE
           Origin
                                                                             STD
UTU
                     Date
                                              Name
32772
           usermapper Tue Sep 18 19:08:40 2007 NASDOCS\user2
S-1-5-15-b8e641e2-33f0942d-8f03a08f-452
32771 usermapper Tue Sep 18 17:56:53 2007 NASDOCS\user1
S-1-5-15-b8e641e2-33f0942d-8f03a08f-a59
32770
           usermapper Sun Sep 16 07:50:39 2007 NASDOCS\Administrator
S-1-5-15-b8e641e2-33f0942d-8f03a08f-1f4
SECMAP GROUP MAPPING TABLE
GID
          Origin
                     Date
                                              Name
                                                                             SID
32793
         usermapper Wed Sep 12 14:16:18 2007 NASDOCS\Domain Admins
S-1-5-15-b8e641e2-33f0942d-8f03a08f-200
32797
          usermapper Sun Sep 16 07:50:40 2007 NASDOCS\Domain Users
S-1-5-15-b8e641e2-33f0942d-8f03a08f-201
32799
         usermapper Mon Sep 17 19:13:16 2007 NASDOCS\Domain Guests
S-1-5-15-b8e641e2-33f0942d-8f03a08f-202
32800 usermapper Mon Sep 17 19:13:22 2007 NASDOCS\Domain Computers
S-1-5-15-b8e641e2-33f0942d-8f03a08f-203
32795
          usermapper Sun Sep 16 07:50:40 2007 NASDOCS\Schema Admins
S-1-5-15-b8e641e2-33f0942d-8f03a08f-206
32796 usermapper Sun Sep 16 07:50:40 2007 NASDOCS\Enterprise Admins
S-1-5-15-b8e641e2-33f0942d-8f03a08f-207
32794
          usermapper Sun Sep 16 07:50:40 2007 NASDOCS\Group Policy Creator
Owners S-1-5-15-b8e641e2-33f0942d-8f03a08f-208
          usermapper Mon Sep 17 19:13:15 2007 NASDOCS\CERTSVC_DCOM_ACCESS
32798
S-1-5-15-b8e641e2-33f0942d-8f03a08f-e45
          usermapper Tue Sep 18 19:08:41 2007 NASDOCS\NASDOCS Group
S-1-5-15-b8e641e2-33f0942d-8f03a08f-45b
        EXAMPLE #12
                       To display the secmap mapping entry for a user user1 in a domain
                       NASDOCS, type:
                       $ server_cifssupport server_2 -secmap -list -name user1
                       -domain NASDOCS
server_2 : done
SECMAP USER MAPPING TABLE
UID
           Origin
                      Date
                                              Name
                                                                             SID
           usermapper Tue Sep 18 17:56:53 2007 NASDOCS\user1
S-1-5-15-b8e641e2-33f0942d-8f03a08f-a59
        EXAMPLE #13
                       To display the secmap mapping entry for a user with UID 32771,
                       type:
                       $ server_cifssupport server_2 -secmap -list -uid 32771
```

server\_2 : done

SECMAP USER MAPPING TABLE

UID Origin Date Name SID

32771 usermapper Tue Sep 18 17:56:53 2007 NASDOCS\user1

S-1-5-15-b8e641e2-33f0942d-8f03a08f-a59

**EXAMPLE #14** To create the secmap mapping entry for user3 in a domain

NASDOCS, type:

\$ server\_cifssupport server\_2 -secmap -create -name user3

-domain NASDOCS

server\_2 : done

SECMAP USER MAPPING TABLE

UID Origin Date Name SID

32773 usermapper Tue Sep 18 19:21:59 2007 NASDOCS\user3

S-1-5-15-b8e641e2-33f0942d-8f03a08f-a3d

**EXAMPLE #15** To check the secmap mapping for **user1** in a domain NASDOCS,

type:

\$ server\_cifssupport server\_2 -secmap -verify -name user1

-domain NASDOCS

server\_2 : done

**EXAMPLE #16** To update the secmap mapping entry for a user using SID, type:

\$ server\_cifssupport server\_2 -secmap -update -sid

S-1-5-15-b8e641e2-33f0942d-8f03a08f-a3d

server\_2 : done

**EXAMPLE #17** To delete the secmap mapping entry for **user3**, type:

\$ server\_cifssupport server\_2 -secmap -delete -name user3

-domain NASDOCS

server\_2 : done

**EXAMPLE #18** To display current secmap status, type:

\$ server\_cifssupport server\_2 -secmap -report

server\_2 : done

SECMAP GENERAL INFORMATIONS

Name : server\_2 State : Enabled

Fs : / Used nodes : 12 Used blocks : 8192

SECMAP MAPPED DOMAIN

```
Name
                        SID
                        S-1-5-15-b8e641e2-33f0942d-8f03a08f-ffffffff
NASDOCS
        EXAMPLE #19
                        To export the secmap mapping entries to the display, type:
                        $ server_cifssupport server_2 -secmap -export
server_2 : done
SECMAP MAPPING RECORDS
S-1-5-15-b8e641e2-33f0942d-8f03a08f-200:2:96:8019:8019:NASDOCS\Domain Admins
S-1-5-15-b8e641e2-33f0942d-8f03a08f-201:2:96:801d:801d:NASDOCS\Domain Users
S-1-5-15-b8e641e2-33f0942d-8f03a08f-202:2:96:801f:801f:NASDOCS\Domain Guests
S-1-5-15-b8e641e2-33f0942d-8f03a08f-203:2:96:8020:8020:NASDOCS\Domain Computers
S-1-5-15-b8e641e2-33f0942d-8f03a08f-206:2:96:801b:801b:NASDOCS\Schema Admins
S-1-5-15-b8e641e2-33f0942d-8f03a08f-207:2:96:801c:801c:NASDOCS\Enterprise Admins
S-1-5-15-b8e641e2-33f0942d-8f03a08f-208:2:96:801a:801a:NASDOCS\Group Policy
Creator Owners
S-1-5-15-b8e641e2-33f0942d-8f03a08f-e45:2:96:801e:801e:NASDOCS\CERTSVC_DCOM_ACCE
S-1-5-15-b8e641e2-33f0942d-8f03a08f-452:1:96:8004:8000:NASDOCS\user2
S-1-5-15-b8e641e2-33f0942d-8f03a08f-a59:1:96:8003:8000:NASDOCS\user1
S-1-5-15-b8e641e2-33f0942d-8f03a08f-45b:2:96:8021:8021:NASDOCS\NASDOCS Group
S-1-5-15-b8e641e2-33f0942d-8f03a08f-1f4:1:96:8002:8000:NASDOCS\Administrator
        EXAMPLE #20
                        To export the secmap mapping entries to a file, type:
                        $ server cifssupport server 2 -secmap -export -file
                        exportfile.txt
server_2 : done
        EXAMPLE #21
                        To import the secmap mapping entries from a file, type:
                        $ server_cifssupport server_2 -secmap -import -file
                        exportfile.txt
server 2 :
Secmap import in progress : #
done
```

# server\_cpu

Performs an orderly, timed, or immediate halt or reboot of a Data Mover.

### **SYNOPSIS**

```
server_cpu {<movername> | ALL}
{-halt | -reboot[cold | warm] } [-monitor] <time>
```

### **DESCRIPTION**

**server\_cpu** performs an orderly halt or reboot of the specified Data Mover.

The **ALL** option executes the command for all Data Movers.

## **OPTIONS**

#### -halt

Performs an orderly shutdown of a Data Mover for the Celerra Network Server. To restart a Data Mover, perform a **-reboot**. For the NS series, a **-halt** causes a system reboot.

#### -reboot

Performs an orderly shutdown, and restarts a Data Mover. The default parameter of the -reboot option will be the warm parameter. In case the warm rebooting fails, the -reboot option will use the cold parameter to reboot the Data Mover.

#### [cold]

A cold reboot or a hardware reset shuts down the Data Mover completely before restarting, including a Power on Self Test (POST).

#### [warm]

A warm reboot or a software reset performs a partial shutdown of the Data Mover, and skips the POST after restarting. A software reset is faster than the hardware reset.



#### **CAUTION**

Performing a reboot for ALL Data Movers can be time consuming relative to the size of the mounted filesystem configuration.

#### -monitor

Polls and displays the boot status until completion of the halt or reboot.

<time>

Specifies the time when the Data Mover is to be halted or rebooted. Time is specified as {now | +<min> | <hour>:<min>}. The now option is used for an immediate shutdown or reboot.

After a power fail and crash recovery, the system reboots itself at power-up unless previously halted.

**SEE ALSO** *Celerra Network Server System Operations.* 

**EXAMPLE #1** To monitor an immediate reboot of **server\_2**, type:

\$ server\_cpu server\_2 -reboot -monitor now

server\_2 : reboot in progress 0.0.0.0.0.0.0.0.0.0.3.3.3.3.3.3.4.done

# Where:

Value	Definition
0	Reset
1	DOS booted
2	SIB failed
3	Loaded
4	Configured
5	Contacted
7	Panicked
9	Reboot pending

# **EXAMPLE #2** To immediately halt **server\_2**, type:

\$ server\_cpu server\_2 -halt now

server\_2 : done

**EXAMPLE #3** To immediately reboot **server\_2**, type:

\$ server\_cpu server\_2 -reboot now

server\_2 : done

**EXAMPLE #4** To monitor a reboot of **server\_2**, that is set to take place in one minute, type:

\$ server\_cpu server\_2 -reboot -monitor +1

server\_2 : reboot in progress ......3.3.3.3.done

# server\_date

Displays or sets the date and time for a Data Mover, and synchronizes time between a Data Mover and an external time source.

#### **SYNOPSIS**

```
server_date {<movername> | ALL}
   [+<format>][<yymmddhhmm>[<ss>]]
| timesvc start ntp [-sync_delay][-interval <hh>[:<mm>]][<host>[<host>...]]
| timesvc update ntp
| timesvc stop ntp
| timesvc delete ntp
| timesvc set ntp
| timesvc stats ntp
| timesvc stats ntp
| timesvc timezone [<timezonestr>]
| timezone -name <timezonename>
```

### **DESCRIPTION**

**server\_date** sets and displays the current date and time for the specified Data Movers.

The **server\_date timesvc** commands control the synchronization of the Data Mover with external timing sources and gets and sets the time zone.

The **ALL** option executes the command for all Data Movers.

#### **OPTIONS**

# No arguments

Displays the current date and time for the specified Data Mover.

```
+<format>
```

Displays the date information in the format specified by each field descriptor. Each field descriptor is preceded by percent and is replaced in the output by its corresponding value. A single percent is encoded by double percent (%%).

If the argument contains embedded blanks, it must be quoted.

The complete listing of all field descriptors can be viewed using the Linux strftime (3C) man page.

```
<yymmddhhmm>[<ss>]
```

Sets a two-digit number for the year, month, day, hour, minutes, and seconds in this order where <yy> is the year; the first <mm> is the month; <dd> is the day; <hh> is the hour (in 24-hour system); and the second <mm> is the minute, and <ss> is the second.

#### timesvc start ntp <host> [<host>...]

Starts time synchronization immediately between a Data Mover and a host, which is the IP address of the time server hosts, and adds an entry to the database. The host must be running the NTP protocol. Only four host entries are allowed.

Other options include:

#### -sync\_delay

Indicates that the clock should not be synchronized when the time server is activated. Instead, when the first poll is taken, latency adjustments are handled slowly. This option is generally used if time service is started after the Data Mover has already started, or if synchronization is starting after other services have already started.

**Note:** If **-sync\_delay** is not typed, by default, the clock is set at Data Mover startup. The clock is synchronized after the first poll.

#### -interval <hh>[:<mm>]

Sets the delay in hours (or hours and minutes) between polls (default=1 hour which is entered 01 or 00:60). Interval is displayed in minutes.

#### timesvc update ntp

Immediately polls the external source and synchronizes the time on the Data Mover.

# timesvc stop ntp

Stops timing synchronization between the Data Mover and an external timing host for the NTP protocol, and does not remove the entry from the database.

**Note:** A stop of time services takes about 12 seconds. If time service is restarted within this time, a "busy" status message is returned.

#### timesvc delete ntp

Stops time synchronization and deletes the NTP protocol from the database.

#### timesvc set ntp

Immediately polls the external source and synchronizes the time on the Data Mover without slewing the clock.

#### timesvc stats ntp

Displays the statistical information of time synchronization for the

Network Time Protocol such as time differences between the Data Mover and the time server. Also provides information about the current state of NTP service on the Data Mover.

#### timesvc

Displays the current time service configuration.

#### timezone

Displays the current time zone on the specified Data Mover.

```
[<timezonestr>]
```

Sets the current time zone on the specified Data Mover. The <timezonestr> is a POSIX style time zone specification with the following formats:

```
<std><offset> (no daylight savings time)
<std><offset><dst>[offset],start[/time],end[/time] (adjusts
for daylight savings time)
```

**Note:** The Linux man page for tzset provides information about the format.

```
timezone -name <timezonename>
```

Sets the time zone on the Data Mover to the specified <timezonename>. The <timezonename> is in Linux style time zone specification. A list of valid Linux timezones is located in the /usr/share/zoneinfo directory. The third column in the table in this file lists the valid timezones.

**Note**: The **timezone -name** option does not reset time on the Data Mover to the specified <timezonename> time.

**SEE ALSO** *Configuring Celerra Time Services*, server\_dns, and server\_nis.

**EXAMPLE #1** To display the current date and time on a Data Mover, type:

```
$ server_date server_2
```

server\_2 : Thu Jan 6 16:55:09 EST 2005

**EXAMPLE #2** To customize the display of the date and time on a Data Mover, type:

```
$ server_date server_2 "+%Y-%m-%d %H:%M:%S"
```

server\_2 : 2005-01-06 16:55:58

EXAMPLE #3

To start time synchronization between a Data Mover and an external

source, type:

\$ server\_date server\_2 timesvc start ntp -interval 06:00

172.24.102.20

server\_2 : done

**EXAMPLE #4** 

To set the time service without slewing the clock, type:

\$ server\_date server\_2 timesvc set ntp

server\_2 : done

**EXAMPLE #5** 

To display statistical information, type:

\$ server\_date server\_2 timesvc stats ntp

server\_2 :

Time synchronization statistics since start: hits= 2, misses= 0, first poll hit= 2, miss= 0

Last offset: 0 secs, 0 usecs

Current State: Running, connected, interval=360

Time sync hosts:

0 1 172.24.102.20

#### Where:

Value	Definition
	When a client sends a request to the server requesting the current time, if there is a reply, that is a hit.
misses	No reply from any of the time servers.
first poll hit	First poll hit which sets the first official time for the Data Mover.
miss	First poll miss.
Last offset	Time difference between the time server and the Data Mover.
Current State	State of the time server.
Time sync hosts	IP address of the time server.

**EXAMPLE #6** To update time syn

To update time synchronization between a Data Mover and an external source, type:

\$ server\_date server\_2 timesvc update ntp

server\_2 : done

**EXAMPLE #7** To get the time zone on the specified Data Mover, type:

\$ server\_date server\_2 timezone

server\_2 : Local timezone: GMT

**EXAMPLE #8** 

To set the time zone to Central Time for a Data Mover when you do not have to adjust for daylight savings time, type:

\$ server\_date server\_2 timezone CST6

server\_2 : done

**EXAMPLE #9** 

To set the time zone to Central Time and adjust for daylight savings time for a Data Mover, type:

\$ server\_date server\_2 timezone CST6CDT5,M4.1.0,M10.5.0

server\_2 : done

**EXAMPLE #10** 

To set the time zone to Central Time and adjust the daylight savings time for a Data Mover using the Linux method, type:

\$ server\_date server\_2 timezone -name America/Chicago

server\_2 : done

**EXAMPLE #11** 

To display the time service configuration for a Data Mover, type:

\$ server\_date server\_2 timesvc

server\_2 :

Timeservice State

time: Thu Jan 6 17:04:28 EST 2005

type: ntp
sync delay: off
interval: 360

hosts: 172.24.102.20,

#### Where:

Value	Definition
time	Date and time known to the Data Mover.
type	Time service protocol configured on the Data Mover.
sync delay	Whether sync delay is on or off.
interval	Time interval between polls.
hosts	Specifies the IP address of the time server.

**EXAMPLE #12** To stop time services for a Data Mover, type:

\$ server\_date server\_2 timesvc stop

server\_2 : done

**EXAMPLE #13** 

To delete the time service configuration for a Data Mover, type:

\$ server\_date server\_2 timesvc delete ntp

server\_2 : done

**EXAMPLE #14** 

To set the timezone on a Data Mover to Los Angeles, type:

\$ server\_date server\_2 timezone -n America/Los\_Angeles

server\_2 : done

# server\_dbms

Enables backup and restore of databases, displays database environment statistics.

### **SYNOPSIS**

```
server_dbms {<movername> | ALL}
{-db
    {-list [<db_name>]
    | -delete <db_name>
| -check [<db_name>]
| -repair [<db_name>]
| -compact [<db_name>]
| -fullbackup -target <pathname>
| -incrbckup -previous <pathname> -target <pathname>
| -restore [<db_name>] -source <pathname>
| -stats [<db_name> [-table <name>]][-reset]}
| -service -stats [transaction|memory|log|lock|mutex][-reset]
```

### **DESCRIPTION**

**server\_dbms** provides recovery of media failure or application corruption, displays database information, checks application database consistency, and fixes inconsistencies.

The **ALL** option executes the command for all Data Movers.

#### **OPTIONS**

```
-db -list [<db_name>]
```

Gets the list of all application databases and their status. If <db\_name> is specified, displays the list of all tables belonging to that database.

```
-db -delete <db_name>
```

Deletes the target application database.

**Note**: This command fails if the target database is not closed.

```
-db -check [<db_name>]
```

Checks the consistency of the target database at application level.

```
-db -repair [<db_name>]
```

Fixes the application level inconsistencies in the database.

```
-db -compact [<db_name> [-table <name>]]
```

Frees up disc space by compacting the target environment or database.

#### -db -fullbackup -target <pathname>

Performs an online full backup of the VDM database environment. The target parameter specifies the location to copy the database files.

The <pathname> specifies the local path of the database environment on the Control Station.

-db -incrbackup -previous <pathname> -target <pathname>
Downloads the transactional logs from the VDM and replays them on
a copy of the previous VDM backup specified by previous
<pathname>.

The **-target** option specifies the location to which the database files are copied.

-db -restore [<db\_name>] -source <pathname>
Restores the environment or database specified by <db\_name>.
source <pathname> specifies the location for the backup of the environment/database to be restored.

**Note**: The database must be closed before the command is executed.

-db -stats [<db\_name> [-table <name>]][-reset]
Displays statistics related to the specified databases and tables. If
-reset is specified, resets the statistics.

```
-service -stats [transaction|memory|log|lock|
mutex][-reset]
```

Displays transaction, memory, logging, locking or mutex statistics of the VDM database environment. If **-reset** is specified, resets all or specified statistics.

**Note**: For this command to be executed, the VDM on which the target environment resides, must be up.

# **EXAMPLE #1**

To get the list of all application databases and their status, type:

\$ server\_dbms server\_3 -db -list

```
server_3 : done

BASE NAME : Secmap

Version : 1

Comment : CIFS Secure mapping database.

This is a cache of the sid to uid/gid mapping of the VDM.

This database is part of the CIFS application.

It can closed with the command server_setup

Size : 16384

Modification time : Fri May 25 09:58:21 2007

Creation time : Fri May 25 09:58:21 2007
```

```
TABLE NAME : Mapping
Version: 1
Comment: Sid to uid/gid mapping table with one secondary key on xid ((1, uid) & amp;
(2, gid))
Size : 16384
Modification time: Fri May 25 09:58:21 2007
Creation time: Fri May 25 09:58:21 2007
BASE NAME : V4NameSpace
Version: 1
Comment: NFSv4 namespace database, this represents the pseudofs and referrals.
Size : 8192
Modification time: Tue Jun 5 08:57:12 2007
Creation time: Tue Jun 5 08:57:12 2007
TABLE NAME : pseudofs
Version: 1
Comment: Pseudofs-table, this holds the export tree heirarchy
Size : 8192
Modification time: Mon Jun 11 11:06:23 2007
Creation time : Mon Jun 11 11:06:23 2007
BASE NAME : Usermapper
Version: 1
Comment: Usermapper database. It allows to assign a new uid or gid to a given SID.
Size : 57344
Modification time: Tue Jun 12 09:14:31 2007
Creation time: Tue Jun 12 09:14:31 2007
TABLE NAME : aliases
Version: 1
Comment: This table allows to retrieve a domain name from one of his aliases
Size : 8192
Modification time: Tue Jun 12 09:14:31 2007
Creation time: Tue Jun 12 09:14:31 2007
TABLE NAME : usrmapc
Version: 1
Comment: Store the uid & amp; gid ranges allocations for domains.
Size : 8192
Modification time: Tue Jun 12 09:14:31 2007
Creation time : Tue Jun 12 09:14:31 2007
TABLE NAME : idxname
Version: 1
Comment: Store the reverse mapping uid/gid to sid.
Modification time: Tue Jun 12 09:14:31 2007
Creation time : Tue Jun 12 09:14:31 2007
TABLE NAME : usrmapusrc
Version: 1
Comment: Store the mapping SID -> (uid, name).
Size : 8192
Modification time: Tue Jun 12 09:14:31 2007
Creation time: Tue Jun 12 09:14:31 2007
TABLE NAME: usrgrpmapnamesid
Version: 1
Comment: Store the mapping user.domain -> SID.
```

```
Size: 8192
Modification time: Tue Jun 12 09:14:31 2007
Creation time: Tue Jun 12 09:14:31 2007
TABLE NAME: usrmapgrpc
Version: 1
Comment: Store the mapping SID -> (gid, name).
Size: 8192
Modification time: Tue Jun 12 09:14:31 2007
Creation time: Tue Jun 12 09:14:31 2007
TABLE NAME: groupmapnamesid
Version: 1
Comment: Store the mapping group.domain -> SID.
Size: 8192
Modification time: Tue Jun 12 09:14:31 2007
Creation time: Tue Jun 12 09:14:31 2007
```

# **EXAMPLE #2** To display Secmap statistics, type:

#### \$ server\_dbms server\_3 -db -stats Secmap

```
server 3 : done
STATISTICS FOR DATABASE : Secmap
TABLE : Mapping
NAME VALUE COMMENT
magic 340322 Magic number.
version 9 Table version number.
metaflags 0 Metadata flags.
nkeys 14 Number of unique keys.
ndata 14 Number of data items.
pagesize 4096 Page size.
minkey 2 Minkey value.
re_len 0 Fixed-length record length.
re_pad 32 Fixed-length record pad.
levels 1 Tree levels.
int_pg 0 Internal pages.
leaf_pg 1 Leaf pages.
dup_pg 0 Duplicate pages.
over_pg 0 Overflow pages.
empty_pg 0 Empty pages.
free 0 Pages on the free list.
int_pgfree 0 Bytes free in internal pages.
leaf_pgfree 2982 Bytes free in leaf pages.
dup_pgfree 0 Bytes free in duplicate pages.
over_pgfree 0 Bytes free in overflow pages.
```

# **EXAMPLE #3** To display statistics of the VDM database environment, type:

#### \$ server\_dbms server\_3 -service -stats

STATISTICS FOR MODULE : LOG

NAME VALUE COMMENT
magic 264584 Log file magic number.
version 12 Log file version number.

mode	0	Log file mode.
lg_bsize	32768	Log buffer size.
lg_size	5242880	Log file size.
record	96	Records entered into the log.
w_bytes	16001	Bytes to log.
w_mbytes	0	Megabytes to log.
wc_bytes	0	Bytes to log since checkpoint.
wc_mbytes	0	Megabytes to log since checkpoint.
wcount	31	Total writes to the log.
wcount_fill	0	Overflow writes to the log.
rcount	137	Total I/O reads from the log.
scount	31	Total syncs to the log.
region_wait	0	Region lock granted after wait.
region_nowait	0	Region lock granted without wait.
cur_file	3	Current log file number.
cur_offset	16001	Current log file offset.
disk_file	3	Known on disk log file number.
disk_offset	16001	Known on disk log file offset.
regsize	98304	Region size.
maxcommitperflush	1	Max number of commits in a flush.
mincommitperflush	1	Min number of commits in a flush.

# STATISTICS FOR MODULE : LOCK

NAME	VALUE	COMMENT
last_id	91	Last allocated locker ID.
cur_maxid	2147483647	Current maximum unused ID.
maxlocks	1000	Maximum number of locks in table.
maxlockers	1000	Maximum num of lockers in table.
maxobjects	1000	Maximum num of objects in table.
nmodes	9	Number of lock modes.
nlocks	20	Current number of locks.
maxnlocks	21	Maximum number of locks so far.
nlockers	49	Current number of lockers.
maxnlockers	49	Maximum number of lockers so far.
nobjects	20	Current number of objects.
maxnobjects	21	Maximum number of objects so far.
nrequests	65711	Number of lock gets.
nreleases	65691	Number of lock puts.
nupgrade	0	Number of lock upgrades.
ndowngrade	20	Number of lock downgrades.
lock_wait	0	Lock conflicts w/ subsequent wait.
lock_nowait	0	Lock conflicts w/o subsequent wait.
ndeadlocks	0	Number of lock deadlocks.
locktimeout	0	Lock timeout.
nlocktimeouts	0	Number of lock timeouts.
txntimeout	0	Transaction timeout.
ntxntimeouts	0	Number of transaction timeouts.
region_wait	0	Region lock granted after wait.
region_nowait	0	Region lock granted without wait.
regsize	352256	Region size.

### STATISTICS FOR MODULE : TXN

NAME	VALUE	COMMENT
last_ckp	3/15945	lsn of the last checkpoint.
time_ckp	Fri Aug 3 09:38:36 2007	time of last checkpoint.
last_txnid	0x8000001a	last transaction id given out.
maxtxns	20	maximum txns possible.
naborts	0	number of aborted transactions.
nbegins	26	number of begun transactions.
ncommits	26	number of committed transactions.
nactive	0	number of active transactions.
nsnapshot	0	number of snapshot transactions.
nrestores	0	number of restored transactions
		after recovery.
maxnactive	2	maximum active transactions.
maxnsnapshot	0	maximum snapshot transactions.
region_wait	0	Region lock granted after wait.
region_nowait	0	Region lock granted without wait.
regsize	16384	Region size.

### STATISTICS FOR MODULE : MPOOL

NAME	VALUE	COMMENT
gbytes	0	Total cache size: GB.
bytes	10487684	Total cache size: B.
ncache	1	Number of caches.
regsize	10493952	Region size.
mmapsize	0	Maximum file size for mmap.
maxopenfd	0	Maximum number of open fd's.
maxwrite	0	Maximum buffers to write.
maxwrite_sleep	0	Sleep after writing max buffers.
map	0	Pages from mapped files.
cache_hit	65672	Pages found in the cache.
cache_miss	36	Pages not found in the cache.
page_create	0	Pages created in the cache.
page_in	36	Pages read in.
page_out	2	Pages written out.
ro_evict	0	Clean pages forced from the cache.
rw_evict	0	Dirty pages forced from the cache.
page_trickle	0	Pages written by memp_trickle.
pages	36	Total number of pages.
page_clean	36	Clean pages.
page_dirty	0	Dirty pages.
hash_buckets	1031	Number of hash buckets.
hash_searches	65744	Total hash chain searches.
hash_longest	1	Longest hash chain searched.
hash_examined	65672	Total hash entries searched.
hash_nowait	0	Hash lock granted with nowait.
hash_wait	0	Hash lock granted after wait.
hash_max_nowait	0	Max hash lock granted with nowait
hash_max_wait	0	Max hash lock granted after wait.
region_nowait	0	Region lock granted with nowait.

# **The server Commands**

region_wait	0	Region lock granted after wait.
mvcc_frozen	0	Buffers frozen.
mvcc_thawed	0	Buffers thawed.
mvcc_freed	0	Frozen buffers freed.
alloc	123	Number of page allocations.
alloc_buckets	0	Buckets checked during allocation.
alloc_max_buckets	0	Max checked during allocation.
alloc_pages	0	Pages checked during allocation.
alloc_max_pages	0	Max checked during allocation.
io_wait	0	Thread waited on buffer I/O.

### STATISTICS FOR MODULE : MUTEX

NAME	VALUE	COMMENT
mutex_align	4	Mutex alignment.
mutex_tas_spins	1	Mutex test-and-set spins.
mutex_cnt	3254	Mutex count.
mutex_free	1078	Available mutexes.
mutex_inuse	2176	Mutexes in use.
mutex_inuse_max	2176	Maximum mutexes ever in use.
region_wait	0	Region lock granted after wait.
region_nowait	0	Region lock granted without wait.
regsize	278528	Region size.

# server\_devconfig

Queries, saves, and displays the SCSI over Fibre Channel device configuration connected to the specified Data Movers.

#### **SYNOPSIS**

```
server_devconfig {<movername> | ALL}
| -create -scsi [<chain_number>] {-disks | -nondisks | -all}
| [-discovery {y|n}][-monitor {y|n}][-Force {y|n}]
| -list -scsi [<chain_number>] {-disks | -nondisks | -all}
| -probe -scsi [<chain_number>] {-disks | -nondisks | -all}
| -rename <old_name> <new_name>
```

#### DESCRIPTION

**server\_devconfig** queries the available storage system device and tape device configuration, and saves the device configuration into the Data Mover's database. **server\_devconfig** renames the device name, and lists SCSI devices.



#### **CAUTION**

It is recommended that all Data Movers have the same device configuration. When adding devices to the device table for a single Data Mover only, certain actions such as standby failover will not be successful unless the standby Data Mover has the same disk device configuration as the primary Data Mover.

The ALL option executes the command for all Data Movers.

#### **OPTIONS**

-create -scsi [<chain\_number>] {-disks|-nondisks| -a11}
Queries SCSI devices and saves them into the device table database
on the Data Mover. The <chain\_number> specifies a SCSI chain
number.

The **-disks** option limits operations to disks. The **-nondisks** option limits operations to non-disks such as tapes, juke boxes, and gatekeeper devices. The **-all** option permits all SCSI devices (disks and non-disks).

**Note:** The **-create** option modifies CLARiiON LUN names to the Celerra\_celerra-hostname>\_clun-id>\_celerra-dvol-name> format, if the LUNs use the default Unisphere Software name.



# **CAUTION**

The time taken to complete this command might be lengthy, dependent on the number and type of attached devices.

[-discovery {y|n}]

Enables or disables the storage discovery operation.



#### **CAUTION**

Disabling the -discovery option should only be done under the direction of an EMC Customer Service Engineer.

[-monitor {y | n}]

Displays the progress of the query and discovery operations.

[-Force {y | n}]

Overrides the health check failures and changes the storage configuration.



# **CAUTION**

High availability can be lost when changing the storage configuration. Changing the storage configuration should only be done under the direction of an EMC Customer Service Engineer.

-list -scsi [<chain\_number>] {-disks|-nondisks| -all}
Lists the SCSI device table database that has been saved on the Data
Mover. The <chain\_number> specifies a SCSI chain number.

**Note:** Fibre Channel devices appear as SCSI devices. Therefore, chain numbers might be different for Fibre Channel.

The **-disks** option limits operations to disks. The **-nondisks** option limits operations to non-disks such as tapes, juke boxes, and gatekeeper devices. The **-all** option permits all SCSI devices (disks and non-disks).

**-probe -scsi** [<chain\_number>] {**-disks**|**-nondisks**| **-all**} Queries and displays the SCSI devices without saving them into the database. The <chain\_number> specifies a SCSI chain number.

**Note:** Fibre Channel devices appear as SCSI devices, therefore, chain numbers may be different for Fibre Channel.

The **-disks** option limits operations to disks. The **-nondisks** option limits operations to non-disks such as tapes, juke boxes, and gatekeeper devices. The **-all** option permits all SCSI devices (disks and non-disks).

-rename <old\_name> <new\_name>

Renames the specified non-disk from the <old\_name> to <new\_name>. The **-rename** option is available for non-disks only.

# **SEE ALSO**

Celerra Network Server System Operations, nas\_disk, and nas\_storage.

# STORAGE SYSTEM OUTPUT

The number associated with the storage device is dependent on the attached storage system. CLARiiON storage systems display a prefix of APM before a set of integers, for example, APM00033900124-0019. For example, Symmetrix storage systems appear as 002804000190-003C.

### **EXAMPLE #1**

For the Celerra Network Server with a CLARiiON storage system, to list all devices, type:

# \$ server\_devconfig server\_2 -list -scsi -all

#### server\_2:

d8

d8

		Dire	ctor	Port			
name	addr	num	type	num	sts	stor_id	stor_dev
root_disk	c0t010					APM00043807043	0000
root_disk	c16t010					APM00043807043	0000
root_ldisk	c0t011					APM00043807043	0001
root_ldisk	c16t011					APM00043807043	0001
d3	c0t012					APM00043807043	0002
d3	c16t012					APM00043807043	0002
d4	c0t013					APM00043807043	0003
d4	c16t013					APM00043807043	0003
d5	c0t014					APM00043807043	0004
d5	c16t014					APM00043807043	0004
d6	c0t015					APM00043807043	0005
d6	c16t015					APM00043807043	0005
d7	c0t110					APM00043807043	0010
d7	c16t110					APM00043807043	0010

# Scsi Device Table

c16t111

c0t111

Scsi Disk Table

Hallie	auui	суре	11110
gk01	c0t010	disk	5 020700000000APM00043807043
ggk01	c0t110	disk	5 020710001000APM00043807043
gk161	c16t1l1	disk	5 020711001100APM00043807043

time info

APM00043807043 0011

APM00043807043 0011

For the Celerra Network Server with a Symmetrix storage system, to list all the devices in the SCSI table, type:

# \$ server\_devconfig server\_2 -list -scsi -all

\$ server_devconing server_2 -list -scsi -all								
server_2 :	~ .	1 - 1 -						
	SCS1	Disk Table						
			Direc		Port			
name		addr	num	type	num		stor_id	stor_dev
root_disk		c0t010	16C	FA	0	On	000187940268	0000
root_disk		c16t010	01C	FA	0	On	000187940268	0000
root_ldisk		c0t011	16C	FA	0	On	000187940268	0001
root_ldisk		c16t011	01C	FA	0	On	000187940268	0001
d3		c0t110	16C	FA	0	On	000187940268	0006
d3		c16t110	01C	FA	0	On	000187940268	0006
d4		c0t111	16C	FA	0	On	000187940268	0007
d4		c16t111	01C	FA	0	On	000187940268	0007
d5		c0t112	16C	FA	0	On	000187940268	8000
d5		c16t112	01C	FA	0	On	000187940268	8000
d6		c0t113	16C	FA	0	On	000187940268	0009
d6		c16t113	01C	FA	0	On	000187940268	0009
d7		c0t114	16C	FA	0	On	000187940268	A000
d7		c16t114	01C	FA	0	On	000187940268	000A
< removed	>							
d377		c1t816	16C	FA	0	On	000187940268	017C
d377		c17t816	01C	FA	0	On	000187940268	017C
rootd378		c1t817	16C	FA	0	On	000187940268	0180
rootd378		c17t817	01C	FA	0	On	000187940268	0180
rootd379		c1t818	16C	FA	0	On	000187940268	0181
rootd379		c17t818	01C	FA	0	On	000187940268	0181
rootd380		c1t819	16C	FA	0	On	000187940268	0182
rootd380		c17t819	01C	FA	0	On	000187940268	0182
rootd381		c1t8l10	16C	FA	0	On	000187940268	0183
rootd381		c17t8l10	01C	FA	0	On	000187940268	0183
	Scsi	Device Table						
name		addr	type	info	)			
gk01		c0t0115	disk		06817D	480	000187940268	
gk161		c16t0115	disk	5670	06817D	330	000187940268	

**Note:** This is a partial display due to the length of the output.

#### Where:

Value	Definition
name	A unique name for each device in the chain.
addr	SCSI chain, target, and LUN information.
Director num	Director number. This output is applicable for Symmetrix storage systems only.
type	Device type, as specified in the SCSI spec. for peripherals. This output is applicable for Symmetrix storage systems only.
Port num	Port number. This output is applicable for Symmetrix storage systems only.
sts	Indicates the port status. Possible values are: On, Off, WD (write disabled), and NA. This output is applicable for Symmetrix storage systems only.
stor_id	Storage system ID.
stor_dev	Storage system device ID.

# **EXAMPLE #2**

For the Celerra Network Server with a CLARiiON storage system, to list all SCSI-attached non-disk devices, type:

# \$ server\_devconfig server\_2 -list -scsi -nondisks

server\_2 :

Scsi Device Table

name	addr	type	ir	nfo
gk01	c0t010	disk	5	020700000000APM00043807043
ggk01	c0t110	disk	5	020710001000APM00043807043
gk161	c16t1l1	disk	5	020711001100APM00043807043

For the Celerra Network Server with a Symmetrix storage system, to list all SCSI-attached non-disk devices, type:

## \$ server\_devconfig server\_2 -list -scsi -nondisks

server\_2 :

Scsi Device Table

 name
 addr
 type
 info

 gk01
 c0t0115
 disk
 56706817D480
 000187940268

 gk161
 c16t0115
 disk
 56706817D330
 000187940268

For info=56706817D480, the following breakdown applies:

5670	68	17D	48	0
Symm code	Last 2 digits in the Symm S/N	Symm Device ID#	Symm SA #	SA Port # (0= <b>a</b> , 1= <b>b</b> )

# **EXAMPLE #3** To rename a device, type:

# \$ server\_devconfig server\_2 -rename gk161 gk201

server\_2 : done

## EXAMPLE #4

For the Celerra Network Server with a CLARiiON storage system, to discover SCSI disk devices, without saving them to the database table, type:

```
$ server_devconfig server_2 -probe -scsi -disks
server_2 :
SCSI disk devices :
chain= 0, scsi-0
 stor_id= APM00043807043 celerra_id= APM000438070430000
 tid/lun= 0/0 type= disk sz= 11263 val= 1 info= DGC RAID 5 02070000000000NI
 tid/lun= 0/1 type= disk sz= 11263 val= 2 info= DGC RAID 5 02070100010001NI
 tid/lun= 0/2 type= disk sz= 2047 val= 3 info= DGC RAID 5 02070200020002NI
 tid/lun= 0/3 type= disk sz= 2047 val= 4 info= DGC RAID 5 02070300030003NI
 tid/lun= 0/4 type= disk sz= 2047 val= 5 info= DGC RAID 5 02070400040004NI
 tid/lun= 0/5 type= disk sz= 2047 val= 6 info= DGC RAID 5 02070500050005NI
 tid/lun= 1/0 type= disk sz= 245625 val= 7 info= DGC RAID 5 02071000100010NI
 tid/lun= 1/1 type= disk sz= 0 val= -5 info= DGC RAID 5 02071100110011NI
 tid/lun= 1/2 type= disk sz= 273709 val= 9 info= DGC RAID 5 02071200120012NI
 tid/lun= 1/3 type= disk sz= 0 val= -5 info= DGC RAID 5 02071300130013NI
 tid/lun= 1/4 type= disk sz= 273709 val= 10 info= DGC RAID 5 02071400140014NI
 tid/lun= 1/5 type= disk sz= 0 val= -5 info= DGC RAID 5 02071500150015NI
 tid/lun= 1/6 type= disk sz= 273709 val= 11 info= DGC RAID 5 02071600160016NI
 tid/lun= 1/7 type= disk sz= 0 val= -5 info= DGC RAID 5 02071700170017NI
 tid/lun= 1/8 type= disk sz= 273709 val= 12 info= DGC RAID 5 02071800180018NI
 tid/lun= 1/9 type= disk sz= 0 val= -5 info= DGC RAID 5 02071900190019NI
chain= 1, scsi-1 : no devices on chain
chain= 2, scsi-2 : no devices on chain
chain= 3, scsi-3 : no devices on chain
chain= 4, scsi-4 : no devices on chain
chain= 5, scsi-5 : no devices on chain
chain= 6, scsi-6 : no devices on chain
chain= 7, scsi-7 : no devices on chain
chain= 8, scsi-8 : no devices on chain
chain= 9, scsi-9 : no devices on chain
chain= 10, scsi-10 : no devices on chain
chain= 11, scsi-11 : no devices on chain
chain= 12, scsi-12 : no devices on chain
chain= 13, scsi-13 : no devices on chain
chain= 14, scsi-14 : no devices on chain
chain= 15, scsi-15 : no devices on chain
```

For the Celerra Network Server with a Symmetrix storage system, to discover SCSI disk devices, without saving them to the database table, type:

```
$ server_devconfig server_2 -probe -scsi -disks
server_2 :
SCSI disk devices :
chain= 0, scsi-0 : no devices on chain
chain= 1, scsi-1 : no devices on chain
chain= 2, scsi-2
stor_id= 000190102173 celerra_id= 0001901021730041
```

```
tid/lun= 0/0 type= disk sz= 11507 val= 1 info= 577273041291SI00041
  tid/lun= 0/1 type= disk sz= 11507 val= 2 info= 577273042291SI00042
 tid/lun= 1/0 type= disk sz= 11501 val= 3 info= 57727304F291SI0004F
 tid/lun= 1/1 type= disk sz= 11501 val= 4 info= 577273050291SI00050
 tid/lun= 1/2 type= disk sz= 11501 val= 5 info= 577273051291SI00051
 tid/lun= 1/3 type= disk sz= 11501 val= 6 info= 577273052291SI00052
 tid/lun= 1/4 type= disk sz= 11501 val= 7 info= 577273053291SI00053
 tid/lun= 1/5 type= disk sz= 11501 val= 8 info= 577273054291SI00054
 tid/lun= 1/6 type= disk sz= 11501 val= 9 info= 577273055291SI00055
 tid/lun= 1/7 type= disk sz= 11501 val= 10 info= 577273056291SI00056
 tid/lun= 1/8 type= disk sz= 11501 val= 11 info= 577273057291SI00057
 tid/lun= 1/9 type= disk sz= 11501 val= 12 info= 577273058291SI00058
 tid/lun= 1/10 type= disk sz= 11501 val= 13 info= 577273059291SI00059
 tid/lun= 1/11 type= disk sz= 11501 val= 14 info= 57727305A291SI0005A
 tid/lun= 1/12 type= disk sz= 11501 val= 15 info= 57727305B291SI0005B
 tid/lun= 1/13 type= disk sz= 11501 val= 16 info= 57727305C291SI0005C
 tid/lun= 1/14 type= disk sz= 11501 val= 17 info= 57727305D291SI0005D
 tid/lun= 1/15 type= disk sz= 11501 val= 18 info= 57727305E291SI0005E
 tid/lun= 2/0 type= disk sz= 11501 val= 19 info= 57727305F291SI0005F
 tid/lun= 2/1 type= disk sz= 11501 val= 20 info= 577273060291SI00060
 tid/lun= 2/2 type= disk sz= 11501 val= 21 info= 577273061291SI00061
<... removed ...>
  tid/lun= 7/6 type= disk sz= 11501 val= 105 info= 577273517291SI00517
  tid/lun= 7/7 type= disk sz= 11501 val= 106 info= 577273518291SI00518
 tid/lun= 7/8 type= disk sz= 11501 val= 107 info= 577273519291SI00519
 tid/lun= 7/9 type= disk sz= 11501 val= 108 info= 57727351A291SI0051A
 tid/lun= 7/10 type= disk sz= 11501 val= 109 info= 57727351B291SI0051B
 tid/lun= 7/11 type= disk sz= 11501 val= 110 info= 57727351C291SI0051C
 tid/lun= 7/12 type= disk sz= 11501 val= 111 info= 57727351D291SI0051D
 tid/lun= 7/13 type= disk sz= 11501 val= 112 info= 57727351E291SI0051E
 tid/lun= 7/14 type= disk sz= 11501 val= 113 info= 57727351F291SI0051F
 tid/lun= 7/15 type= disk sz= 11501 val= 114 info= 577273520291SI00520
chain= 3, scsi-3 : no devices on chain
chain= 4, scsi-4 : no devices on chain
chain= 5, scsi-5 : no devices on chain
chain= 6, scsi-6 : no devices on chain
<... removed ...>
chain= 18, scsi-18
stor_id= 000190102173 celerra_id= 0001901021730041
tid/lun= 0/0 type= disk sz= 11507 val= 1 info= 577273041201SI00041
tid/lun= 0/1 type= disk sz= 11507 val= 2 info= 577273042201SI00042
tid/lun= 1/0 type= disk sz= 11501 val= 3 info= 57727304F201SI0004F
tid/lun= 1/1 type= disk sz= 11501 val= 4 info= 577273050201SI00050
tid/lun= 1/2 type= disk sz= 11501 val= 5 info= 577273051201SI00051
tid/lun= 1/3 type= disk sz= 11501 val= 6 info= 577273052201SI00052
tid/lun= 1/4 type= disk sz= 11501 val= 7 info= 577273053201SI00053
```

**Note:** This is a partial listing due to the length of the output.

# **EXAMPLE #5** To discover and save all SCSI devices, type:

```
$ server_devconfig server_2 -create -scsi -all
Discovering storage (may take several minutes)
server_2 : done
```

**EXAMPLE #6** To discover and save all non-disk devices, type:

```
$ server_devconfig server_2 -create -scsi -nondisks Discovering storage (may take several minutes) server_2 : done
```

#### **EXAMPLE #7**

To save all SCSI devices with the discovery operation disabled, and display information regarding the progress, type:

\$ server\_devconfig ALL -create -scsi -all -discovery n
-monitor y

```
server_2 :
server_2:
chain 0 .....
chain 16 ....
done
server_3 :
server_3:
chain 0 .....
chain 16 .....
done
server 4:
server_4:
chain 0 .....
chain 16 .....
done
server_5 :
server 5:
chain 0 .....
chain 16 .....
done
```

# server\_df

Reports free and used disk space and inodes for mounted filesystems on the specified Data Movers.

## **SYNOPSIS**

```
server_df {<movername>|ALL}
  [-inode][<pathname>|<fs_name>]
```

## DESCRIPTION

**server\_df** reports the amount of used and available disk space for a Data Mover or filesystem, how much of a filesystem's total capacity has been used, and the number of used and free inodes.

The ALL option executes the command for all Data Movers.

#### **OPTIONS**

# No arguments

Displays the amount of disk space in kilobytes used by filesystems.

#### -inode

Reports used and free inodes.

```
[<pathname>|<fs_name>]
```

Gets filesystem information. If <fs\_name> specified, gets information for filesystem only.

# **SEE ALSO**

Managing Celerra Volumes and File Systems Manually, nas\_disk, and nas\_fs.

## **EXAMPLE #1**

To display the amount of used and available disk space on a Data Mover, type:

## \$ server\_df server\_2

server_2 :					
Filesystem	kbytes	used	avail	capacity	Mounted on
ufs1	1075686032	477816	1075208216	0%	/ufs1
ufs4	101683184	584	101682600	0%	/nmfs1/ufs4
ufs2	206515184	600	206514584	0%	/nmfs1/ufs2
nmfs1	308198368	1184	308197184	0%	/nmfs1
root_fs_common	13624	5264	8360	39%	/.etc_common
root_fs_2	114592	760	113832	1%	/

#### Where:

Value	Definition
Filesystem	Name of the filesystem.
kbytes	Total amount of space in kilobytes for the filesystem.
used	Amount of kilobytes used by the filesystem.

Value	Definition
avail	Amount of space in kilobytes available for the filesystem.
capacity	Percentage capacity that is used.
Mounted on	Mount point of the filesystem.

# **EXAMPLE #2** To display the amount of disk space and the amount of free and unused inodes on a Data Mover, type:

# \$ server\_df server\_2 -inode

server_2 :					
Filesystem	inodes	used	avail	capacity	Mounted on
ufs1	131210494	140	131210354	0%	/ufs1
ufs4	25190398	10	25190388	0%	/nmfs1/ufs4
ufs2	25190398	11	25190387	0%	/nmfs1/ufs2
nmfs1	50380796	21	50380775	0%	/nmfs1
root_fs_common	21822	26	21796	0%	/.etc_common
root fs 2	130942	66	130876	0%	/

# **EXAMPLE #3** To display the amount of disk space and the amount of free and unused inodes on a filesystem, type:

# \$ server\_df server\_2 -inode ufs1

server_z :						
Filesystem	inodes	used	avail	capacity	Mounted	on
ufs1	131210494	140	131210354	0%	/ufs1	

# server\_dns

Manages the Domain Name System (DNS) lookup server configuration for the specified Data Movers.

## **SYNOPSIS**

```
server_dns {<movername> | ALL}
  [[-protocol {tcp|udp}] <domainname> {<ip_addr>,...}]
| [-delete <domainname>]
| [-option {start|stop|flush|dump}]
```

# **DESCRIPTION**

**server\_dns** provides connectivity to the DNS lookup servers for the specified Data Movers to convert hostnames and IP addresses. Up to three DNS lookup servers are supported for each domain on the Data Mover.

**server\_dns** also provides the ability to clear the cache that has been saved on the Data Mover as a result of the DNS lookup servers.

The ALL option executes the command for all Data Movers.

# **OPTIONS**

# No arguments

Displays the DNS configuration.

```
-protocol {tcp|udp} <domainname> {<ip_addr>,...}
Sets the protocol for the DNS lookup servers (udp is the default).
```

```
<domainname> {<ip_addr>,...}
```

Creates list of up to three IP addresses to be used as the DNS lookup servers for the specified <domainname>.

```
-delete <domainname>
```

Deletes the DNS lookup servers in the DNS domain name.

```
-option {start|stop|flush|dump}
```

The **start** option activates the link for the DNS lookup servers. The **stop** option halts access to the DNS lookup servers. After DNS service has been halted, the **flush** option can be used to clear the cache that has been saved on the Data Mover, and the **dump** option displays the DNS cache.

#### SEE ALSO

Configuring Celerra Naming Services and server\_nis.

#### **EXAMPLE #1**

To connect to a DNS lookup server, type:

```
$ server_dns server_2 prod.emc.com 172.10.20.10
```

server\_2 : done

```
EXAMPLE #2
                       To display the DNS configuration, type:
                       $ server_dns server_2
server_2 :
DNS is running.
prod.emc.com
proto:udp server(s):172.10.20.10
         EXAMPLE #3
                       To change the protocol to TCP from UDP, type:
                       $ server_dns server_2 -protocol tcp prod.emc.com
                       172.10.20.10
server_2 : done
         EXAMPLE #4
                       To halt access to the DNS lookup servers, type:
                       $ server_dns server_2 -option stop
server 2 : done
         EXAMPLE #5
                       To flush the cache on a Data Mover, type:
                       $ server_dns server_2 -option flush
server 2 : done
         EXAMPLE #6
                       To dump the DNS cache, type:
                       $ server_dns server_2 -option dump
server_2 :
DNS cache size for one record type: 64
DNS cache includes 6 item(s):
dm102-cge0.nasdocs.emc.com
                TTL=184 s dataCount:1
 172.24.102.202 (local subnet)
winserver1.nasdocs.emc.com
                TTL=3258 s dataCount:1
Type:A
  172.24.103.60
_ldap._tcp.Default-First-Site-Name._sites.dc._msdcs.nasdocs.emc.com
Type:SRV
          TTL=258 s dataCount:1
 priority:0
                weight:100
                                port:389
                                          server:winserver1.nasdocs.emc.com
_kerberos._tcp.Default-First-Site-Name._sites.dc._msdcs.nasdocs.emc.com
Type:SRV
               TTL=258 s dataCount:1
 priority:0
                weight:100 port:88
                                          server:winserver1.nasdocs.emc.com
Expired item(s): 2
         EXAMPLE #7
                       To delete the DNS lookup servers, type:
                       $ server dns server 2 -delete prod.emc.com
server_2 : done
```

# server\_export

Exports filesystems, and manages access on the specified Data Movers for NFS and CIFS clients.

#### **SYNOPSIS**

# **DESCRIPTION**

**server\_export** provides user access by exporting an NFS pathname, or creating a CIFS share. Allows specification of multiple clients identified by hostnames or network and subnet addresses separated by a colon.

**server\_export** removes access by unexporting an NFS pathname, deleting a CIFS share, and displaying the exported entries and available shares for the specified Data Mover.

The **ALL** option executes the command for all Data Movers.

Note: NFSv4 does not support the -alias option.

# GENERAL OPTIONS FOR CIFS AND NFS OPERATIONS

## No arguments

Displays all exported NFS entries and CIFS shares.

```
[-Protocol {cifs|nfs}] -list -all
```

Lists all exported entries as defined by the protocol. The default is NFS.

```
[-Protocol {cifs | nfs}] -all
```

Exports all entries on a Data Mover as defined by the protocol. The default is NFS.

[-Protocol {cifs|nfs}] -unexport [-perm] -all Unexports all entries as defined by the protocol. By default,

unexports all entries as defined by the protocol. By default, unexports are permanent for CIFS, and temporary for NFS, unless **-perm** is specified. If **-perm** is specified, removes all entries from the export table. When the entry is temporarily unexported, clients are denied access to the entry until it is re-exported or the system is rebooted, but the entries are not removed from the export table. The default is NFS.

## FOR NFS OPERATIONS

#### -list <pathname>

Lists a specific NFS entry. If there are extra spaces in the <pathname>, the entire pathname must be enclosed by quotes. By using the server\_export command, IPv6 addresses can be specified and the hosts configured with these addresses can mount and access filesystems over NFS.

**Note:** If you are configuring an IPv6 address for ro, rw, access, and root, it must be enclosed in [] or square brackets. This is to properly handle the colon used to separate entries. Link local addresses are not supported.

#### -Protocol nfs [-name <name>] <pathname>

Exports an NFS <pathname> by default as read-write for everyone. If specified, assigns an optional filesystem name for the <name>. Pathname length is limited to 1024 bytes (represented as 1024 ASCII characters or a variable number of Unicode multibyte characters), and must be enclosed by quotes, if spaces are used. Name length is limited to 255 bytes.

**Note:** In a nested mount filesystem hierarchy, users can export the mount point path of the component filesystem. Subdirectories of the component filesystem cannot be exported. In a multilevel filesystem hierarchy, users can export any part of a filesystem independent of existing exports.

[-ignore] <pathname>

Overwrites previous options and comments in the export table for the entry.

[-comment <comment>] <pathname>

Adds a comment for the specified NFS export entry. The comment is displayed when listing the exported entries.

[-option <options>] <pathname>

Specifies the following comma-separated options:

sec=[sys|krb5|krb5i|krb5p]:<mode> [,<mode>,...]
Specifies a user authentication or security method with an access
mode setting. The sys (default) security option specifies
AUTH\_SYS security. The access mode can be one, or a
combination of the following: ro, rw=, ro=, root=, access=, anon=,
webroot, public.

If the **sec** option is specified, it must always be the first option specified in the string.

**krb5** security specifies Kerberos user and data authentication.

**krb5i** checks for the integrity of the data by adding a signature to each NFS packet and **krb5p** encrypts the data before sending it over the network.

For **krb5**, **krb5i**, and **krb5p** security, the access mode can be one, or a combination of the following: **ro**, **rw**=, **ro**=, **root**=, **access**=.

#### rc

Exports the <pathname> for all NFS clients as read-only.

```
ro=<client>[:<client>]...
```

Exports the <pathname> for the specified NFS clients as read-only.

**Note:** If **<cli>ent>** is an IPv6 address, it must be enclosed in square brackets or [].

```
ro=<-client>[:<-client>]...
```

Excludes the specified NFS clients from ro privileges. Clients must be preceded with dash (-) to specify exclusion.

**Note:** If **<cli>client>** is an IPv6 address, it must be enclosed in square brackets or [].

```
rw=<client>[:<client>]...
```

Exports the <pathname> as read-mostly for the specified NFS clients. Read-mostly means exported read-only to most machines, but read-write to those specified. The default is read-write to all.

**Note:** If **<cli>client>** is an IPv6 address, it must be enclosed in square brackets or [].

```
rw=<-client>[:<-client>]...
```

Excludes the specified NFS clients from rw privileges. The description of read-mostly provides information. Clients must be preceded with - (dash) to specify exclusion.

**Note:** If **<cli>ent>** is an IPv6 address, it must be enclosed in square brackets or [].

```
root = < client > [: < client >] . . .
```

Provides **root** privileges for the specified NFS clients. By default, no host is granted **root** privilege.

**Note:** If **<cli>is** an IPv6 address, it must be enclosed in square brackets or [].

```
root = < -client > [: < -client >] . . .
```

Excludes the specified NFS clients from **root** privileges. Clients must be preceded with - (dash) to specify exclusion.

**Note:** If **<cli>is** an IPv6 address, it must be enclosed in square brackets or [].

anon=<uid>

If a request comes from an unknown user, the UID should be used as the effective user ID. Root users (UID =0) are considered "unknown" by the NFS server unless they are included in the root option. The default value for anon=<uid> is the user "nobody." If the user "nobody" does not exist, then the value 65534 is used.



# **CAUTION**

Using anon=0 is not recommended for security concerns.

access=<client>[:<client>]...

Provides mount access for the specified NFS clients.

**Note:** If **<cli>is** an IPv6 address, it must be enclosed in square brackets or [].

```
access = < - client > [: < - client > ] . . .
```

Excludes the specified NFS clients from access even if they are part of a subnet or netgroup that is allowed access. Clients must be preceded with - (dash) to specify exclusion.

**Note:** If **<cli>is** an IPv6 address, it must be enclosed in square brackets or [].

#### nfsv4only

Specifies that the NFS export can be accessed only when a client is using NFSv4.

```
vlan=<vlanid>[:<vlanid>]...
```

Specifies that all hosts belonging to the VLAN ID will have access to the exported filesystem. Hosts on other VLANs will be denied.

Clients can be a hostname, netgroup, subnet, or IP address and must be colon-separated, without spaces. A subnet is an IP address/netmask (for example, 168.159.50.0/255.255.255.0). A hostname is first checked for in the Data Mover's local hosts database, then on the NIS (host database) or DNS server (if enabled). A netgroup is searched in the local netgroup database and then on the NIS server's netgroup database. If the client name does not exist in any case, then access is denied.

**Note:** Netgroups are supported. The hosts and netgroup files can be created on the Control Station using your preferred method (for example, with an editor, or by copying from another node), then copied to the Data Mover.

```
-unexport [-perm] <pathname>
```

Temporarily unexports a <pathname> unless -perm is specified. If -perm is specified, removes the entry from the export table.

# FOR CIFS OPERATIONS

-list -name <sharename>

Displays the specified CIFS share.

```
[-option <options>]
```

Specifies the following comma-separated list of options:

```
[netbios = < netbios name > ]
```

When the share has an associated NetBIOS name that name is required to locate the entry. Multiple CIFS entries can have same <sharename> when belonging to a different NetBIOS name.

```
-name <sharename> [-ignore][-option <options>]
[-comment <comment>] <pathname>
```

Creates a CIFS share. Share name length is limited to 12 ASCII characters unless Unicode is enabled, in which case the limit is 80 multibyte characters. Share names cannot include the following characters: /, \, %, ", NUL (Null character), STX (start of header), SOT (start of text), and LF (line feed). Share names can contain spaces and other nonalphanumeric characters, but must be enclosed by quotes if spaces are used. Share names cannot begin with a - (hyphen). Share names are case-sensitive.

Comment length is limited to 256 bytes (represented as 256 ASCII characters or a variable number of Unicode multibyte characters). A comment cannot include the following characters: NUL (Null character), STX (start of header), and SOT (start of text). Comments can contain spaces and other nonalphanumeric characters, but must be enclosed by quotes if spaces are used. Pathname length is limited to 1024 bytes.

The **-ignore** option overwrites the previous options and comment in the export table.

```
[-option <options>]
```

Specifies the following comma-separated options:

#### rc

Exports the <pathname> for CIFS clients as read-only.

```
rw=<client>[:<client>]...
```

Creates the share for CIFS clients as read-mostly. Read-mostly means shared read-only to most clients, but read-write to those specified. By default, the <pathname> is shared read-write to all. A client may be either a <user\_name> or <group\_name>.

**Note:** If **<cli>is** an IPv6 address, it must be enclosed in [] or square brackets.

```
umask=<mask>
```

Specifies a user file-creation mask for the umask allowing NFS permissions to be determined for the share.

**Note:** The value of the umask is specified in octal and is XORed with the permissions of 666 for files and 777 for directories. Common values include 002, which gives complete access to the group, and read (and directory search) access to others or 022 (default) which gives read (and directory search), but not write permission to the group and others.

#### user=<default\_user>

When using share level access (server\_checkup provides information), specifies a <default\_user> which must be entered as a character string. The user must be defined in the Data Mover's password file. There is a 20 character limit for the username.

#### group=<default\_group>

When using share level access (server\_checkup provides information), indicates a <default\_group> which must be entered as a character string. There is a 256 character limit for group names.

#### ropasswd=<share\_passwd>

When using share level access (server\_checkup provides information), creates a read-only password to allow clients access to the share. Passwords can be viewed in the list of shared entries.

#### rwpasswd=<share\_rw\_passwd>

When using share level access (server\_checkup provides information), creates a read-write password to allow clients access to the share. Passwords are displayed in the list of shared entries.

**Note:** Users from any client machine who know the value of the ropasswd or rwpasswd can access the share for read-only and read-write operations.

#### maxusr=<maxusr>

Sets the maximum number of simultaneous users permitted for a share.

**netbios**=<netbiosName>[, **netbios**=<netbiosName>]...
Associates a share on a single domain with one or more NetBIOS names created with server\_checkup. By default, if a NetBIOS name is not specified for a share, the share is visible to all NetBIOS names.

#### -comment

Adds a comment for the specified CIFS share. The comment is displayed when listing the shared entries.

#### -unexport -name <sharename>

Permanently removes access to a share by removing the entry from the export table.

[-option <options>]

Specifies the following comma-separated options:

netbios = < netbios name >

When the share has an associated NetBIOS name the NetBIOS name is required to locate the entry. This is required because multiple CIFS entries can have same <sharename> when belonging to different NetBIOS name.

SEE ALSO

Configuring NFS on Celerra, Managing Celerra Volumes and File Systems Manually, Configuring CIFS on Celerra, server\_checkup, and server\_mount.

#### **EXAMPLE #1**

To export a specific NFS entry, type:

\$ server\_export server\_2 -Protocol nfs /ufs1

server\_2 : done

#### **EXAMPLE #2**

To export an NFS entry and overwrite existing settings, type:

\$ server\_export server\_2 -Protocol nfs -ignore -option
access=172.24.102.0/255.255.255.0,root=172.24.102.240
-comment 'NFS Export for ufs1' /ufs1

server 2 : done

# **EXAMPLE #3**

To export NFS entry **dir1**, a subdirectory of the exported entry **/ufs1** in a multilevel filesystem hierarchy, type:

\$ server\_export server\_2 -Protocol nfs /ufs1/dir1

server 2 : done

# **EXAMPLE #4**

To assign a name to a NFS export, type:

\$ server\_export server\_2 -Protocol nfs -name nasdocsfs
/ufs1

server\_2 : done

# **EXAMPLE #5**

To export an NFS entry using Kerberos authentication, type:

\$ server\_export server\_2 -Protocol nfs -option
sec=krb5:ro,root=172.24.102.240,access=172.24.102.0/255.
255.255.0 /ufs2

server\_2 : done

```
EXAMPLE #6
                        To export an NFS entry for NFSv4 only, type:
                         $ server_export server_2 -Protocol nfs -option nfsv4only
                         /ufs1
server_2 : done
         EXAMPLE #7
                        To list all NFS entries, type:
                        $ server_export server_2 -Protocol nfs -list -all
server_2 :
export "/ufs2" sec=krb5 ro root=172.24.102.240 access=172.24.102.0/255.255.255.0
export "/ufs1" name="/nasdocsfs" access=172.24.102.0/255.255.255.0
root=172.24.102.240 nfsv4only comment="NFS Export for ufs1"
export "/" anon=0
access=128.221.252.100:128.221.253.100:128.221.252.101:128.221.253.101
         EXAMPLE #8
                        To list NFS entries for the specified path, type:
                        $ server_export server_2 -list /ufs1
server_2 :
export "/ufs1" name="/nasdocsfs" access=172.24.102.0/255.255.255.0
root=172.24.102.240 nfsv4only comment="NFS Export for ufs1"
         EXAMPLE #9
                        To temporarily unexport an NFS entry, type:
                        $ server export server 2 -Protocol nfs -unexport /ufs2
server_2 : done
        EXAMPLE #10
                        To export all NFS entries, type:
                        $ server_export server_2 -Protocol nfs -all
server_2 : done
        EXAMPLE #11
                        To export a specific NFS entry in a language that uses multibyte
                        characters (such as Japanese), type:
                         $ server_export server_2 -Protocol nfs / オリビアさん
server_2 : done
        EXAMPLE #12
                        To permanently unexport an NFS entry, type:
                         $ server_export server_2 -unexport -perm /ufs1
server_2 : done
        EXAMPLE #13
                        To permanently unexport all NFS entries, type:
                         $ server_export server_2 -Protocol nfs -unexport -perm
                         -all
server_2 : done
        EXAMPLE #14
                        To provide access to a CIFS share, type:
                        $ server_export server_2 -name ufs1 /ufs1
server_2 : done
```

# **EXAMPLE #15** To create a CIFS share and overwrite existing settings, type:

\$ server\_export server\_2 -name ufs1 -ignore -option
ro,umask=027,maxusr=200,netbios=dm112-cge0 -comment
'CIFS share' /ufs1

server\_2 : done

# **EXAMPLE #16**

To create a CIFS share in a language that uses multibyte characters (such as Japanese), type:

\$ server\_export server\_2 -Protocol cifs -name 共有名 -comment まだかいな / オリビアさん

server\_2 : done

# EXAMPLE #17

To list all CIFS entries, type:

\$ server\_export server\_2 -Protocol cifs -list

server\_2 :

share "ufs1" "/ufs1" ro umask=027 maxusr=200 netbios=DM112-CGE0 comment="CIFS share"

share "ufs2" "/ufs2" umask=022 maxusr=4294967295

# **EXAMPLE #18** To display a specific CIFS share, type:

\$ server\_export server\_2 -list -name ufs1 -option

netbios=dm112-cge0

server\_2 :

share "ufs1" "/ufs1" ro umask=027 maxusr=200 netbios=DM112-CGE0 comment="CIFS share"

# **EXAMPLE #19** To export all CIFS entries, type:

\$ server\_export server\_2 -Protocol cifs -all

server\_2 : done

#### EXAMPLE #20

To list all NFS and CIFS entries, type:

\$ server\_export server\_2

server\_2 :

export "/ufs2" sec=krb5 ro root=172.24.102.240 access=172.24.102.0/255.255.255.0 export "/ufs1" nfsv4only

export "/" anon=0

access=128.221.252.100:128.221.253.100:128.221.252.101:128.221.253.101

share "ufs2" "/ufs2" umask=022 maxusr=4294967295

share "ufs1" "/ufs1" ro umask=027 maxusr=200 netbios=DM112-CGE0 comment="CIFS share"

#### Where:

Value	Definition
export	A filesystem entry to be exported.
sec	Security mode for the filesystem.
ro	Filesystem is to be exported as read-only.

root	IP address with root access.
access	Access is permitted for those IP addresses.
share	Entry to be shared.
ro	Filesystem is to be shared as read-only.
umask	User creation mask.
maxuser	Maximum number of simultaneous users.
netbios	NetBIOS name for the share.
comment	Comment specified for the share.

**EXAMPLE #21** To permanently unexport all CIFS and NFS entries, type:

\$ server\_export server\_2 -unexport -perm -all

server\_2 : done

**EXAMPLE #22** To delete a CIFS share, type:

\$ server\_export server\_2 -unexport -name ufs1 -option
netbios=dm112-cge0

server\_2 : done

**EXAMPLE #23** To delete all CIFS shares, type:

\$ server\_export server\_2 -Protocol cifs -unexport -all

server\_2 : done

**EXAMPLE #24** To export a filesystem for NFS that specifies an IPv4 and IPv6 address, type:

\$ server\_export server\_2 -Protocol nfs -option
access=172.24.108.10:[1080:0:0:0:8:800:200C:417A] /fs1

server\_2 : done

**EXAMPLE #25** To verify that the filesystem was exported, type:

\$ server\_export server\_2 -list /fs1

server\_2 :

export "/fs1" access=172.24.108.10:[1080:0:0:0:8:800:200C:417A]

**EXAMPLE #26** To export a filesystem for NFS that specifies two IPv6 addresses, type:

\$ server\_export server\_2 -Protocol nfs -option
rw=[1080:0:0:0:8:80:200C:417A]:[1080:0:0:0:8:800:200C:41
7B] /fs1

server\_2 : done

**EXAMPLE #27** To verify that the filesystem was exported, type:

# The server Commands

```
$ server_export server_2 -list /fs1
server_2 :
export "/fs1" rw=[1080:0:0:0:8:80:200C:417A]:[1080:0:0:0:8:800:200C:417B]
```

# server\_extlog

Manages the configuration of extending logging in the Data Mover.

## **SYNOPSIS**

# **DESCRIPTION**

Displays the current configuration, starts, stops, and modifies the logging for the specified service.

Extending logging are a set of log files located on the Data Mover that contains all the activities of the clients for a given service. Currently, only the FTP and SSH server supports these extending logs. That allows to track the client's activities for any FTP, FTPS, SFTP, and SCP connections, including the time of connection, the client IP parameters, the main requested commands, and responses.

The format of these log files follows the W3C format. See http://www.w3.org/TR/WD-logfile.html for more info about that format.

By default, these logs are disabled for performance reasons.

#### **OPTIONS**

#### -service <service name>

Gives the name of the service, such as ftp and ssh.

#### -info

Displays the current configuration, and the state of the logging for the specified service.

#### -start

Starts the logging for the specified service, that is, enable it. If the logging for the specified service is already active, an error is returned.

# -stop

Stops the logging for the specified service, that is, disable it. If the logging for the specified service is already stopped, an error is returned.

#### -modify

Allows to define some configuration parameters of the logging for the specified service.

#### -filesize <size>

This parameter defines the maximum size of the log files. The default value is 512KB; the minimum value is 8KB. The maximum value depends on the next parameter.

#### -filecount <count>

This parameter defines the maximum numbers of log files. When the size reaches the max log filesize, a new log file is created. If the number of log files reaches the <count>, the oldest log file is removed.

The default value is 5; the minimum is 2; the maximum is 10 and depends on the previous parameter: the total size (<size> \* <count>) must not exceed 4 GB on a user filesystem and 4 MB on the root filesystem.

## -dir {default | <path>}

This parameter defines the directory where the log files are stored. The default value is /.etc/log. The specified directory must exists. The default directory is always created by the data mover.

# **SEE ALSO** *server\_standby*

# **EXAMPLE #1**

To display the current configuration of the extending logs of the FTP service, type:

#### \$ server\_extlog server\_3 -service ftp -info

server\_3 : done
Service name
State
File size

File count

Directory

: ftp : Started : 524288 : 5 : /.etc/log

EXAMPLE #2

To enable the extending logs for the FTP service, type:

\$ server\_extlog server\_3 -service ftp -start

server\_3 : done

## **EXAMPLE #3**

To disable the extending logs for the FTP service, type:

\$ server\_extlog server\_3 -service ftp -stop

server\_3 : done

# **EXAMPLE #4**

To configure the DART to keep four logs files for the FTP service (i.e. 1 active log file and 3 old log files that have reached the max log filesize), type:

\$ server\_extlog server\_3 -service ftp -modify -filecount

server\_3 : done

## **EXAMPLE #5**

To display the supported services of the extending logs, type:

\$ server\_extlog server\_3

server\_3 : done

Services supporting extended log feature are : ssh and ftp.

## **EXAMPLE #5**

#Software: EMC-SNAS:T7.1.5.0

#Version: 1.0

#Date: 2011-07-19 14:07:05

#Fields: date time c-ip c-port s-ip s-port x-session x-proto cs-username cs-method cs-uri-stem sc-status time-taken sc-bytes cs-bytes

- 2011-07-19 14:07:05 10.32.45.123 54221 20.45.101.89 21 ftp0000-d031ff7a7725 ftp unknown ControlChannelOpened 0 0 0 0
- 2011-07-19 14:07:08 10.32.45.123 54221 20.45.101.89 21 ftp0000-d031ff7a7725 ftp luser2 USER luser2 331 0 35 13
- 2011-07-19 14:07:11 10.32.45.123 54221 20.45.101.89 21 ftp0000-d031ff7a7725 ftp luser2 PASS \*\*\* 230 32 104 13
- 2011-07-19 14:07:11 10.32.45.123 54221 20.45.101.89 21 ftp0000-d031ff7a7725 ftp luser2 SYST 215 0 19 6
- 2011-07-19 14:07:15 10.32.45.123 54221 20.45.101.89 21 ftp0000-d031ff7a7725 ftp luser2 PASV 227 0 51 6
- 2011-07-19 14:07:15 10.32.45.123 54221 20.45.101.89 21 ftp0000-d031ff7a7725 ftp luser2 LIST -1 150 0 57 9
- 2011-07-19 14:07:15 10.32.45.123 39795 20.45.101.89 60670 ftp0000-d031ff7a7725 ftp luser2 DataChannelOpened 0 0 0 0
- 2011-07-19 14:07:15 10.32.45.123 39795 20.45.101.89 60670 ftp0000-d031ff7a7725 ftp luser2 DataChannelClosed 0 80 1698 0
- 2011-07-19 14:07:20 10.32.45.123 54221 20.45.101.89 21 ftp0000-d031ff7a7725 ftp luser2 CWD dir0000 250 0 29 13
- 2011-07-19 14:07:28 10.32.45.123 54221 20.45.101.89 21 ftp0000-d031ff7a7725 ftp luser2 TYPE I 200 0 20 8
- 2011-07-19 14:07:28 10.32.45.123 54221 20.45.101.89 21 ftp0000-d031ff7a7725 ftp luser2 PASV 227 0 50 6
- 2011-07-19 14:07:28 10.32.45.123 54221 20.45.101.89 21 ftp0000-d031ff7a7725 ftp luser2 STOR test.txt.raw 150 21 61 19
- 2011-07-19 14:07:28 10.32.45.123 56147 20.45.101.89 64076 ftp0000-d031ff7a7725 ftp luser2 DataChannelOpened 0 0 0 0
- 2011-07-19 14:07:28 10.32.45.123 56147 20.45.101.89 64076 ftp0000-d031ff7a7725 ftp luser2 DataChannelClosed 0 49 0 1668
- 2011-07-19 14:07:34 10.32.45.123 54221 20.45.101.89 21 ftp0000-d031ff7a7725 ftp luser2 QUIT 221 0 14 6

# The server Commands

2011-07-19 14:07:34 10.32.45.123 54221 20.45.101.89 21 ftp0000-d031ff7a7725 ftp luser2 ControlChannelClosed - 0 28365 549 99

# server file

Copies files between the Control Station and the specified Data Movers.

#### **SYNOPSIS**

```
server_file {<movername> | ALL}
{-get | -put} <src_file> <dst_file>
```

# DESCRIPTION

**server\_file** copies the source file from the specified Data Mover (or Control Station) to the destination file on the Control Station (or specified Data Mover). The <src\_file> indicates the source file, and the name <dst\_file> indicates destination file. By default, if a directory is not specified on the Data Mover, the /.etc directory is used.

The ALL option executes the command for all Data Movers.

## **OPTIONS**

```
-get <src_file> <dst_file>
```

Copies the source file on Data Mover to destination file on Control Station. Both the <src\_file> and <dst\_file> may be full pathnames.

```
-put <src file> <dst file>
```

Copies source file on the Control Station to the destination file on the Data Mover. Both the <src\_file> and <dst\_file> must be full pathnames.



# **CAUTION**

This command overwrites existing files of the same name without notification. Use care when copying files.

## **EXAMPLE #1**

To copy a file from the Control Station to a Data Mover, type:

\$ server\_file server\_2 -put passwd passwd

server\_2 : done

#### **EXAMPLE #2**

To copy a file from the Data Mover to the Control Station, type:

\$ server\_file server\_2 -get passwd /home/nasadmin/passwd

server\_2 : done

# server\_ftp

Configures the FTP server configuration for the specified Data Movers.

#### **SYNOPSIS**

```
server_ftp {<movername> | ALL}
-service {-status|-start|-stop|{-stats [-all|-reset]}}
  -info
  -modify
    [-controlport <controlport>]
    [-dataport <dataport>]
    [-defaultdir <path>]
    [-homedir {enable | disable}]
    [-keepalive <keepalive>]
    [-highwatermark < highwatermark > ]
    [-lowwatermark <lowwatermark>]
    [-deniedusers [<path>]]
    [-welcome [<path>]]
    [-motd [<path>]]
    [-timeout <timeout>]
    [-maxtimeout <maxtimeout>]
    [-readsize <readsize>]
    [-writesize <writesize>]
    [-maxcnx <maxcnx>]
    [-umask <umask>]
    [-sslcontrol {no allow require requireforauth}]
    [-ssldata {allow|require|deny}]
    [-sslpersona {anonymous|default|<persona_name>}]
    [-sslprotocol {default|ssl3|tls1|all}]
    [-sslcipher {default | <cipherlist>}]
    [-sslcontrolport <sslcontrolport>]
    [-ssldataport <ssldataport>]
```

#### DESCRIPTION

**server\_ftp** configures the ftp daemon. Optional SSL security support is available. The modifications are performed when the ftp daemon is stopped and are reflected after restart of the ftp daemon. There is no need to reboot the Data Mover for the changes to be reflected.

## **OPTIONS**

```
server_ftp {<movername> | ALL}
```

Sends a request to the Data Mover to get all the parameters of the ftp daemon.

**ALL** option executes the command for all Data Movers.

```
-service {-status|-start|-stop|{-stats [-all| -reset]}}
```

#### -status

Retrieves the current status of the ftp daemon.

#### -start

Starts the ftp daemon. The **start** option persists after the daemon is rebooted.

# -stop

Stops the ftp daemon.

#### -stats [all|reset]

Displays the statistics of the ftp daemon. The reset option resets all the ftp server statistics. The **all** option displays detailed statistics.

#### -info

Retrieves all the parameters for the ftp daemon along with its current status.

#### -modify

Modifies the ftp daemon configuration. The ftp daemon has to be stopped to carry out the changes. The modifications are taken into account when the service is restarted.

#### -controlport <controlport>

Sets the local tcp port for control connections. By default, the port is 21. When control port is set to 0, it disables unsecure ftp usage and validates port for implicit secure connection on SSL port (default 990).

**Note:** This default port can be changed using the sslcontrolport option.

#### -dataport <dataport>

Sets the local tcp port for active data connections. By default, the port is 20. When <dataport> is set to 0, the port is allocated dynamically by the server in active mode.

# -defaultdir <path>

Sets the default user directory when the user home directory is not accessible. This option replaces "ftpd.defaultdir." By default, "/" is used.

## -homedir {enable|disable}

Restricts or allows user access to their home directory tree. When enabled the user is allowed access to their home directory only. If the user home directory is not accessible, access is denied. During connection the user is denied access to data outside of their home directory space. By default, this feature is disabled.

**Note:** *Using FTP on Celerra* provides more information about how the home directory of a user is managed.

#### -umask <umask>

Defines the mask to set the mode bits on file or directory creation. By default the mask is 027, which means that **xrw-r---** mode bits are assigned.

# -keepalive <keepalive>

Sets TCP keepalive value for the ftp daemon. This value is given in seconds. By default, the value is 60. The value 0 disables the TCP keepalive option. The maximum value is 15300 (255 minutes).

# -highwatermark <highwatermark>

Sets TCP high watermark value (amount of data stored without knowledge of the client) for the ftp daemon. By default, the value is 65536. The minimum value is 8192, and the maximum value is 1048576 (1 MB).



## **CAUTION**

Do not modify this parameter without a thorough knowledge of the impact on FTP client behavior.

#### -lowwatermark <lowwatermark>

Sets TCP low watermark value (amount of the data to be added, after the highwatermark has been reached and new data can be accepted from the client) for the ftp daemon. The minimum value is 8192, maximum value is 1048576 (1 MB), and default value is 32768.



# **CAUTION**

Do not modify this parameter without a thorough knowledge of the impact on FTP client behavior.

#### -deniedusers <deniedusers file>

Denies FTP access to specific users on a Data Mover. Specifies the path and text file containing the list of usernames to be denied access. Places each username on a separate line. By default, all users are allowed.

#### -welcome <welcome file>

Specifies path of the file to be displayed on the welcome screen. For example, this file can display a login banner before the user is requested for authentication data. By default, no welcome message is displayed.

## -motd <motd\_file>

Specifies path of the file to be displayed on the welcome screen. Users see a welcome screen ("message of the day") after they successfully log in. By default, no message of the day is displayed.

#### -timeout <timeout>

Specifies the default inactivity time-out period (when not set by the client). The value is given in seconds. After the specified time if there is no activity, the client is disconnected from the server and will have to re-open a connection. By default, the <timeout> value is 900 seconds. The minimum value is 10 seconds, and the maximum value is 7200.

#### -maxtimeout <maxtimeout>

Sets the maximum time-out period allowed by the client. The value is given in seconds and any value larger than maximum time-out period is not allowed. By default, the <maxtimeout> value is 7200 seconds. The minimum value is 10 seconds, and the maximum value is 7200.

#### -readsize <readsize>

Sets the size for reading files from the disk. The value must be greater than 8192, and a multiple of 8K. By default, the <readsize> is 8192 bytes. The minimum value is 8192, and the maximum value is 1048576 (1 MB).

#### -writesize <writesize>

Sets the size for writing files from the disk. The value must be greater than 8192, and a multiple of 8K. By default, the <writesize> is 49152 (48 KB). The minimum value is 8192, and the maximum value is 1048576 (1 MB).

#### -maxcnx <maxcnx>

Sets the maximum number of control connections the ftp daemon will support. By default, the <maxcnx> value is set to 65535 (64K-1). The minimum value is 1, and the maximum value is 65535 (64K-1).

#### -sslcontrol {no|allow|require|requireforauth}

Uses SSL for the ftp control connection depending on the attributes specified. By default, SSL is disabled. The **no** option disables SSL control. The **allow** option specifies that SSL is enabled, but the user can still connect without SSL. The **require** option specifies that SSL is required for the connection. The **requireforauth** option specifies that SSL is required for authentication. The control path goes back to unsecure after this authentication. When the client is behind a firewall, this helps the firewall to filter the ftp commands requiring new port access.

**Note:** Before the server can be configured with SSL, the Data Mover must be set up with a private key and a public certificate. This key and certificate are identified using a **persona**. In addition, the necessary Certificate Authority (CA) certificates used to identify trusted servers must be imported into the Data Mover. Use Celerra's PKI feature to manage the use of certificates prior to configuring SSL operation.

#### -ssldata {no|allow|require}

Uses SSL for the data connection depending on the attributes specified. The **no** option disables SSL. The **allow** option specifies that SSL is enabled, but the user can also transfer data without SSL. The **require** option specifies that SSL is required for data connection. The **ssldata** value cannot be set to **allow** or **require** if **sslcontrol** is set to **no**. By default, SSL is disabled.

**Note:** These options are set on the server but are dependent on ftp client capabilities. Some client capabilities may be incompatible with server settings. *Using FTP on Celerra* provides information on validating compatibility.

-sslpersona {anonymous|default|<persona\_name>} Specifies the persona associated with the Data Mover. Personas are used to identify the private key and public certificate used by SSL. The default value specified is default (each Data Mover is configured with a persona named default). The anonymous value

specifies that SSL can operate without using a certificate. This implies that the communication between client and server is encrypted and data integrity is guaranteed.

**Note:** Use **server\_certificate** to configure the persona before using **server\_ftp**.

# -sslprotocol {default|ssl3|tls1|all}

Specifies the SSL protocol version that the ftp daemon on the server accepts:

- ssl3 Only SSLv3 connections
- tls1 Only TLSv1 connections
- ◆ all Both SSLv3 and TLSv1 connections
- ◆ **default** Uses the value set in the ssl.protocol parameter which, by default, is 0 (SSLv3 and TLSv1)

# -sslcipher {default | <cipherlist>}

Specifies the SSL cipher suite. The value of default is the value set in the ssl.cipher parameter. This value means that all ciphers are supported by Celerra except the Anonymous Diffie-Hellman, NULL, and SSLv2 ciphers and that the supported ciphers are sorted by the size of the encryption key.

#### -sslcontrolport <sslcontrolport>

Sets the implicit control port for FTP connections over SSL. By default, the port is 990. To disable implicit FTP connections over SSL, the <sslcontrolport> must be set to 0.

#### -ssldataport <ssldataport>

Sets the local tcp port for active data connections using implicit FTP connections over SSL. By default, the port is 899. If the ssldataport is set to 0, the Data Mover will use a port allocated by the system.

#### SEE ALSO

server certificate.

#### **EXAMPLE #1**

To retrieve all the parameters for the ftp daemon and its status, type:

#### \$ server\_ftp server\_2 -info

FTP started ====== controlport dataport defaultdir / homedir umask

```
21
20
.etc/ftpd/pub
disable
027
```

Login Type =======

Anonymous

Unix

tcp keepalive	1 minute
tcp high watermark	65536 bytes
tcp low watermark	32768 bytes
readsize	8192 bytes
writesize	49152 bytes
denied users file path	/.etc/ftpd/conf/ftpusers
welcome file path	/.etc/ftpd/conf/welcome
motd file path	/.etc/ftpd/conf/motd
session timeout	900 seconds
max session timeoutQ	7200 seconds
Security Options	
==========	
sslpersona	default
sslprotocol	default
sslcipher	default
FTP over TLS explicit Options	
sslcontrol	 SSL require for authentication
ssldata	allow SSL
FTP over SSL implicit Options	
sslcontrolport	990
ssldataport	989

#### **EXAMPLE #2** To display the statistics of the ftp daemon, type:

========

10

3

# \$ server\_ftp server\_2 -service -stats Successful Failed ======

0 2

CIFS	7	1		
Data transfers	Count =====	min ====	Throughput (MBytes/sec) average	max ====
Write Bin Read Bin Write ASCII Read ASCII	10 0 2 0	10.00	19.00  1.50 	20.00
SSL Write Bin SSL Read Bin SSL Write ASCII SSL Read ASCII	5 15 0 0	5.00 7.00 	17.00 25.00 	18.00 35.00 

# Where:

Value	Definition
Throughput (MBytes/sec)	Throughput is calculated using the size of the file (Mbytes) divided by the duration of the transfer (in seconds).
average	Average is the average of the throughputs (sum of the throughputs divided by the number of transfers).
Data transfers	Defines the type of transfer.
Count	Number of operations for a transfer type.
min	Minimum time in milliseconds required to execute the operation (with regards to Data Mover).
max	Maximum time in milliseconds required to execute the operation (with regards to Data Mover).

# **EXAMPLE #3** To display the statistics of the ftp daemon with details, type:

		_
	<pre>\$ server_ftp se</pre>	erver_2 -service -stats -all
Commands	Count	
======	=====	
USER	23	
PASS	23	
QUIT	23	
PORT	45	
EPRT	10	
FEAT	23	
SITE Commands	Count	
=========	=====	
UMASK	0	
IDLE	10	
CHMOD	0	
HELP	0	
BANDWIDTH	0	
KEEPALIVE	10	
PASV	56	
OPTS Commands	Count	
=========	====	
UTF8	10	
0110	10	
Login Type	Successful	Failed
========	========	======
Anonymous	10	0
Unix	3	2
CIFS	7	1
Connections	Count	
========	=====	
Non secure		
Control	10	

## **The server Commands**

Data	44
Explicit SSL	
Control Auth Control Data	3 8 20
Implicit SSL	
Control Data	0

			Throughput (MBytes/sec)	
Data transfers	Count	min	average	max
=========	=====	======	=======================================	======
Write Bin	10	10.00	19.00	20.00
Read Bin	0			
Write ASCII	2	1.00	1.50	2.00
Read ASCII	0			
SSL Write Bin	5	5.00	17.00	18.00
SSL Read Bin	15	7.00	25.00	35.00
SSL Write ASCII	0			
SSL Read ASCII	0			

# Where:

Value	Definition	
Commands	FTP protocol command name.	
Count	Number of commands received by Data Mover.	
SITE Commands	Class of command in FTP protocol.	
POTS Commands	Class of command in FTP protocol.	

**EXAMPLE #4** To retrieve the status of the ftp daemon, type:

\$ server\_ftp server\_3 -service -status

server\_3 : done

State : running

**EXAMPLE #5** To start the ftp daemon, type:

\$ server\_ftp server\_2 -service -start

server\_2 : done

**EXAMPLE #6** To stop the ftp daemon, type:

\$ server\_ftp server\_2 -service -stop

server\_2 : done

# **EXAMPLE #7** To set the local tcp port for the control connections, type:

\$ server\_ftp server\_2 -modify -controlport 256

server\_2 :done

# FTPD CONFIGURATION

State : stopped Control Port : 256 Data Port : 20 Default dir : / : disable Home dir Keepalive : 1 High watermark : 65536 : 32768 Low watermark : 900 Timeout Max timeout : 7200 Read size : 8192 : 49152 Write size

> : 27 : 65535

SSL CONFIGURATION

Max connection

Umask

Control channel mode : disable
Data channel mode : disable
Persona : default
Protocol : default
Cipher : default
Control port : 990
Data port : 989

# **EXAMPLE #8** To set the local tcp port for active data connections, type:

\$ server\_ftp server\_2 -modify -dataport 257

server\_2 : done

# FTPD CONFIGURATION

State : stopped
Control Port : 256
Data Port : 257

Default dir : / Home dir : disable Keepalive : 1 : 65536 High watermark Low watermark : 32768 Timeout : 900 Max timeout : 7200 Read size : 8192

Write size : 49152 Umask : 27

#### **The server Commands**

Max connection : 65535

SSL CONFIGURATION

Control channel mode : disable : disable Data channel mode Persona : default Protocol Cipher : default : default Control port : 990 Data port : 989

## EXAMPLE #9

To change the default directory of a user when his home directory is not accessible, type:

\$ server\_ftp server\_2 -modify -defaultdir /big

server\_2 : done

FTPD CONFIGURATION

State : stopped Control Port : 256 : 257 Data Port Default dir : /big : disable Home dir Keepalive : 1

: 65536 : 32768 High watermark Low watermark Timeout : 900 Max timeout : 7200 Read Size Read size : 8192 : 49152 : 27 Umask Max connection : 65535

SSL CONFIGURATION ===========

Control channel mode : disable Data channel mode : disable Persona : default Protocol : default : default Control port : 990 : 989 Data port

**EXAMPLE #10** To allow users access to their home directory tree, type:

\$ server\_ftp server\_2 -modify -homedir enable

server\_2 : done

FTPD CONFIGURATION ============

State : stopped : 256 Control Port Data Port : 257 Default dir : /big : enable Home dir Keepalive : 1 : 65536 High watermark Low watermark : 32768 Timeout : 900 Max timeout : 7200 : 8192 Read size Write size : 49152 : 27 Umask : 65535 Max connection SSL CONFIGURATION Control channel mode : disable

Data channel mode : disable Persona : default Protocol : default Cipher : default Control port : 990

Control port : 990 Data port : 989

# **EXAMPLE #11** To restrict user access to their home directory tree, type:

#### \$ server\_ftp server\_2 -modify -homedir disable

#### server\_2 : done

# FTPD CONFIGURATION

: stopped State Control Port : 256 Data Port : 257 Default dir : /big Home dir : disable Keepalive : 1 : 65536 High watermark : 32768 Low watermark

Timeout : 900
Max timeout : 7200
Read size : 8192
Write size : 49152
Umask : 27
Max connection : 65535

# SSL CONFIGURATION

Control channel mode : disable
Data channel mode : disable
Persona : default

Protocol : default
Cipher : default
Control port : 990
Data port : 989

# EXAMPLE #12

To set the default umask for creating a file or a directory by means of the ftp daemon, type:

\$ server\_ftp server\_2 -modify -umask 077

server\_2 : done

# FTPD CONFIGURATION

State : stopped Control Port : 256
Data Port : 257
Default dir : /big
Home dir : disable Keepalive : 1
High watermark : 65536
Low watermark : 32768
Timeout : 900
Max timeout : 7200
Read size : 8192
Write size : 49152
Umask : 77
Max connection : 65535

# SSL CONFIGURATION

Control channel mode : disable
Data channel mode : disable
Persona : default
Protocol : default
Cipher : default
Control port : 990
Data port : 989

# **EXAMPLE #13** To set the TCP keepalive for the ftp daemon, type:

\$ server\_ftp server\_2 -modify -keepalive 120

server\_2 : done

# FTPD CONFIGURATION

State : stopped
Control Port : 256
Data Port : 257
Default dir : /big
Home dir : disable
Keepalive : 120
High watermark : 65536

```
Low watermark : 32768
Timeout : 900
Max timeout : 7200
Read size : 8192
Write size : 49152
Umask : 77
Max connection : 65535
```

# SSL CONFIGURATION

Control channel mode : disable
Data channel mode : disable
Persona : default
Protocol : default
Cipher : default
Control port : 990
Data port : 989

# **EXAMPLE #14** To set the TCP high water mark for the ftp daemon, type:

# \$ server\_ftp server\_2 -modify -highwatermark 90112

# server\_2 : done

# FTPD CONFIGURATION

State : stopped : 256 Control Port Data Port : 257 Default dir : /big Home dir : disable Keepalive : 120 High watermark : 90112 Low watermark : 32768 Timeout : 900 Max timeout : 7200 Read size : 8192 Write size : 49152 Umask : 77 Max connection : 65535

# SSL CONFIGURATION

===========

Control channel mode : disable
Data channel mode : disable
Persona : default
Protocol : default
Cipher : default
Control port : 990
Data port : 989

#### EXAMPLE #15 To set the TCP low water mark for the ftp daemon, type:

\$ server\_ftp server\_2 -modify -lowwatermark 32768

# server\_2 : done

# FTPD CONFIGURATION

: stopped State Control Port : 256 : 257 Data Port : /big : disable Default dir Home dir Keepalive : 120 High watermark : 90112 : 32768 Low watermark : 900 Timeout Max timeout : 7200 Read size Write size : 8192 : 49152 Umask : 77 Max connection : 65535

# SSL CONFIGURATION ===========

Control channel mode : disable Data channel mode : disable Persona : default : default Protocol Cipher : default Control port : 990 : 989 Data port

# **EXAMPLE #16** To restrict FTP server access to specific users, type:

\$ server\_ftp server\_2 -modify -deniedusers /.etc/mydeniedlist

#### server\_2 : done

# FTPD CONFIGURATION

==============

State : stopped : 256 Control Port : 257 Data Port Default dir : /big : disable Home dir Keepalive High watermark : 90112
Low watermark : 32700 Low watermark : 32768

Denied users conf file : /.etc/mydeniedlist
Timeout

Max timeout : 7200 : 8192 Read size

```
Write size : 49152
Umask : 77
Max connection : 65535
```

# SSL CONFIGURATION

Control channel mode : disable
Data channel mode : disable
Persona : default
Protocol : default
Cipher : default
Control port : 990
Data port : 989

# **EXAMPLE #17** To set the path of the file displayed before the user logs in, type:

\$ server\_ftp server\_2 -modify -welcome
/.etc/mywelcomefile

server\_2 : done

# FTPD CONFIGURATION

: stopped State Control Port : 256 : 257 Data Port Default dir : /big : disable Home dir : 120 Keepalive : 90112 High watermark Low watermark : 32768

Welcome file : /.etc/mywelcomefile

Timeout : 900
Max timeout : 7200
Read size : 8192
Write size : 49152
Umask : 77
Max connection : 65535

# SSL CONFIGURATION

===========

Control channel mode : disable
Data channel mode : disable
Persona : default
Protocol : default
Cipher : default
Control port : 990
Data port : 989

# server\_http

Configures the HTTP configuration file for independent services, such as Celerra FileMover, for the specified Data Movers.

# **SYNOPSIS**

```
server_http {<movername> | ALL }
  -info [<feature>]
  -service <feature> {-start | -stop}
  -service [<feature>] -stats [-reset]
  -modify <feature>
    [-threads <threads>]
    [-users {valid | <user>[, <user>, <user>...]}]
    [-hosts <ip>[,<ip>,<ip>...]]
    [-port <port number>]
    [-timeout <max_idle_time>]
    [-maxrequests <maxrequests>]
    [-authentication {none|basic|digest}]
    [-realm < realm_name>]
    [-ssl {required off}]
    [-sslpersona {anonymous|default|<persona_name>}]
    [-sslprotocol {default|ssl3|tls1|all}]
    [-sslcipher {default | <cipherlist>}]
-append <feature>
    [-users {valid | <user>[, <user>, <user>...]}]
    [-hosts <ip>[, <ip>, <ip>...]]
-remove <feature>
    [-users {valid | <user>[, <user>, <user>...]}]
    [-hosts <ip>[,<ip>,<ip>...]]
```

#### **DESCRIPTION**

**server\_http** manages user and host access to HTTP servers for independent services such as FileMover.

**ALL** option executes the command for all Data Movers.

# **OPTIONS**

```
-info [<feature>]
```

Displays information about the specified feature or all features including server status, port, threads, requests allowed, timeout, access control, and SSL configuration.

```
-service <feature> {-start|-stop}
Stops or starts the HTTP server for the specified feature.
```

```
-service [<feature>] -stats [-reset}
```

Lists the usage statistics of the HTTP server for the specified feature or all features. If **-reset** is specified, statistics are reset to zero.

# -modify <feature>

Displays the current HTTP protocol connection for the specified feature. When issued with options, **-modify** sets the HTTP protocol connection for the specified option. Any options previously set will be overwritten.

# [-threads <threads>]

Sets the number of threads (default=20) for incoming service requests. The minimum value is 4, the maximum 99. The HTTP threads are started on the Data Mover at boot time.

```
[-users [valid | <user>[, <user>, <user>...]]]
```

Allows the users who correctly authenticate as defined in the Data Mover passwd file (server\_user provides more information) to execute commands for the specified <feature>.

If **valid** is entered, all users in the passwd file are allowed to use digest authentication. A comma-separated list of users can also be given. If no users are given, digest authentication is turned off.

```
[-hosts <ip>[, <ip>, <ip>...]]
```

Specifies hosts by their IP addresses that are allowed to execute commands for the specified <feature>.

```
[-port <port_number>]
```

Specifies the port on which the HTTP server listens for incoming service requests. By default, the HTTP server instance for FileMover listens on port 5080.

```
[-timeout <max_idle_time>]
```

Specifies the maximum time the HTTP server waits for a request before disconnecting from the client. The default value is 60 seconds.

```
[-maxrequests <max_requests>]
```

Specifies the maximum number of requests allowed. The default value is 300 requests.

```
[-authentication {none|basic|digest}]
```

Specifies the authentication method. **none** disables user authentication, allowing for anonymous access (that is, no authentication). **basic** authentication uses a clear text password. **digest** authentication uses a scripted password. The default value is **digest** authentication.

```
[-realm < realm_name>]
```

Specifies the realm name. This information is required when authentication is enabled (that is, the **-authentication** option is set to basic or digest). The default realm name for FileMover is DHSM\_authorization.

# [-ssl {required|off}]

Specifies whether the HTTP server runs in secure mode, that is, only accepts data receive on encrypted SSL sessions. The default value is **off**.

**Note:** Before the HTTP server can be configured with SSL, the Data Mover must be set up with a private key and public certificate. This key and certificate are identified using a persona. In addition, the necessary Certificate Authority (CA) certificates to identify trusted servers must be imported into the Data Mover. Use Celerra's PKI feature to manage the use of certificates prior to configuring SSL operation.

[-sslpersona {default | anonymous | <persona\_name>}] Specifies the persona associated with the Data Mover. Personas are used to identify the private key and public certificate used by SSL. The default value is default (each Data Mover is currently configured with a single persona named default). anonymous specifies that SSL can operate without using a certificate.

[-sslprotocol {default|ssl3|tls1|all}]
Specifies the SSL protocol version the HTTPS server accepts:

- ssl3 Only SSLv3 connections
- tls1 Only TLSv1 connections
- all Both SSLv3 and TLSv1 connections
- default Uses the value set in the ssl.protocol parameter which, by default, is 0 (SSLv3 and TLSv1)

```
[-sslcipher {default | <cipherlist>}]
```

Specifies the SSL cipher suite. The value of **default** is the value set in the ssl.cipher parameter which, by default, is ALL:!ADH:!SSLv2:@STRENGTH. This value means that all ciphers are supported by Celerra except the Anonymous Diffie-Hellman, NULL, and SSLv2 ciphers and that the supported ciphers are sorted by the size of the encryption key.

```
-append <feature> [-users {valid|<user>[, <user>,
<user>...]}][-hosts <ip>[, <ip>...]}]
```

Adds the specified users or hosts to the list of those who can execute commands for the specified <feature> without having to re-enter the

existing list. The **users** and **hosts** descriptions provide information. If users or hosts are not specified, displays the current HTTP configuration.

```
-remove <feature> [-users {valid | <user>[, <user>,
<user>...]}][-hosts <ip>[,<ip>,<ip>...]]
```

Removes the specified users and hosts from the list of those who can execute commands for the specified <feature> without impacting others in the list. The **users** and **hosts** descriptions provide information. If users or hosts are not specified, displays the current HTTP configuration.

# **SEE ALSO**

Using Celerra FileMover, Celerra Security Configuration Guide, fs dhsm, server certificate, and nas ca certificate.

# **EXAMPLE #1**

To display information about the HTTP protocol connection for the FileMover service, type:

# \$ server\_http server\_2 -info dhsm

server\_2 : done

DHSM FACILITY CONFIGURATION

Service name : EMC File Mover service

Comment : Service facility for getting DHSM attributes

: False Active : 5080 Port Threads : 16
Max requests : 300
Timeout : 60 seconds

ACCESS CONTROL

Allowed IPs : any
Authentication : digest ,Realm : DHSM\_Authorization
Allowed user : nobody

SSL CONFIGURATION

Mode : OFF Persona : default Protocol : default Cipher : default

#### Where:

Value	Definition
Service name	Name of the FileMover service.
active	Whether Celerra FileMover is enabled or disabled on the filesystem.
port	TCP port of the FileMover service.
threads	Number of threads reserved for the FileMover service.
max requests	Maximum number of HTTP requests the service allows to keep the connection alive.
timeout	The time in seconds until which the service is kept alive after a period of no activity.
allowed IPs	List of client IP addresses that are allowed to connect to the service.

authentication	The HTTP authentication method used by the service.
allowed user	Users allowed to connect to the service.
mode	The SSL mode.
persona	Name of the persona associated with the Certificate for establishing a secure connection.
protocol	The level of SSL protocol used for the service.
cipher	The cipher suite the service is negotiating, for establishing a secure connection with the client.

**EXAMPLE #2** To display statistical information about the HTTP protocol connection for the FileMover service, type:

\$ server\_http server\_2 -service dhsm -stats

server\_2 : done Statistics report for HTTPD facility DHSM: Thread activity Maximum in use count : 0 Connection IP filtering rejection count : 0 Request Authentication failure count : 0 SSL

Handshake failure count

**EXAMPLE #3** To configure an HTTP protocol connection for FileMover using SSL, type:

: 0

\$ server\_http server\_2 -modify dhsm -ssl required

server\_2 : done

**EXAMPLE #4** To modify the threads option of the HTTP protocol connection for FileMover, type:

\$ server\_http server\_2 -modify dhsm -threads 40

server\_2 : done

DHSM FACILITY CONFIGURATION

Service name : EMC File Mover service

: Service facility for getting DHSM attributes Comment

Active : False : 5080 Port Threads : 40 : 300 Max requests

Timeout : 60 seconds

ACCESS CONTROL

Allowed IPs : any
Authentication : digest ,Realm : DHSM\_Authorization
Allowed user : nobody

SSL CONFIGURATION

Mode : OFF Persona : default Protocol : default Cipher : default

**EXAMPLE #5** To allow specific users to manage the HTTP protocol connection for

FileMover, type:

\$ server\_http server\_2 -modify dhsm -users valid -hosts

10.240.12.146

server\_2 : done

**EXAMPLE #6** To add specific users who can manage the existing HTTP protocol

connection for FileMover, type:

\$ server\_http server\_2 -append dhsm -users

user1, user2, user3

server\_2 : done

**EXAMPLE #7** To add a specific user who can manage the existing HTTP protocol

connection for FileMover, type:

\$ server\_http server\_2 -append dhsm -users user4 -hosts

172.24.102.20,172.24.102.21

server\_2 : done

**EXAMPLE #8** To remove the specified users and hosts so they can no longer

manage the HTTP connection for FileMover, type:

\$ server\_http server\_2 -remove dhsm -users user1,user2

-hosts 10.240.12.146

server\_2 : done

# server\_ifconfig

Manages the network interface configuration for the specified Data Movers.

# **SYNOPSIS**

# **DESCRIPTION**

**server\_ifconfig** creates a network interface, assigns an IP address to a network interface, enables and disables an interface, sets the MTU size and the VLAN ID, and displays network interface parameters for the specified Data Mover.

**server\_ifconfig** is used to define the network address of each interface existing on a machine, to delete and recreate an interface's address and operating parameters.

The ALL option executes the command for all Data Movers.

# **OPTIONS**

```
-all [ip4|ip6]
```

Displays parameters for all configured interfaces. The **-ip4** option displays all ipv4 only interfaces and the **-ip6** option displays all ipv6 only interfaces.

```
-delete <if_name>
```

Deletes a network interface configuration. However, the autogenerated link local interfaces cannot be deleted.

```
-create -Device <device_name> -name <if_name>
-protocol IP <ipv4_addr> <ipmask> <ipbroadcast> | IP6
<ipv6_addr> [/PrefixLength]}
```

Creates a network interface configuration on the specified device with the specified name and assigns a protocol to the interface. The <if\_name> must not contain a colon (:).

Available protocols are:

IP <ipv4\_addr> <ipmask> <ipbroadcast>

IP6 <ipv6\_addr> [/PrefixLength]}

IPv4 assigns the IP protocol with the specified IP address, mask, and broadcast address. The **IP address** is the address of a particular interface. Multiple interfaces are allowed for each device, each identified by a different IP address. The **IP mask** includes the network part of the local address and the subnet, which is taken from the host field of the address. For example, 255.255.255.0 would be a mask for a Class C network. The **IP broadcast** is a special destination address that specifies a broadcast message to a network. For example, x.x.x.255 is the broadcast address for a Class C network.

IP6 assigns the IPv6 address and prefix length. When prefix length is not specified, the default value of 64 is used.

When creating the first IPv6 interface with a global unicast address on a broadcast domain, the system automatically creates an associated IPv6 link-local interface. Similarly, when deleting the last remaining IPv6 interface on a broadcast domain, the system automatically deletes the associated IPv6 link-local interface.

For CIFS users, when an interface is created, deleted, or marked up or down, use the server\_setup command to stop and then restart the CIFS service in order to update the CIFS interface list.

<if name> up

Allows the interface to receive and transmit data, but does not enable the physical port. Interfaces are marked up automatically when initially setting up the IP address.

<if\_name> down

Stops data from being transmitted through that interface. If possible, the interface is reset to disable reception as well. This does not automatically disable routes using the interface.

<if\_name> ipsec | noipsec

**ipsec** sets the network interface to IPSec enabled, and ready to process IPSec packets. **noipsec** sets the network interface to IPSec disabled.

**Note:** Celerra Network Server 5.6 does not support the IPSec service.

<if\_name> mtu=<MTUbytes>

Resets the maximum transmission unit (MTU) size in bytes for the specified interface. By default, the MTU is automatically set depending on the type of network interface card installed.

Regardless of whether you have Ethernet or Gigabit Ethernet, the initial default MTU size is 1500 bytes. To take advantage of the capacity of Gigabit Ethernet, the MTU size can be increased up to 9000 bytes if your switch supports jumbo frames. Jumbo frames should be used only when the entire infrastructure, including client NICs, supports them.

For UDP, it is important that both the client and server use the same MTU size. TCP negotiates the MTU size when the connection is initialized. The switch's MTU must be greater than or equal to the host's MTU.

**Note:** The MTU size specified here is for the interface. The MTU size specified in server\_netstat applies to the device and is automatically set.

```
<if_name> vlan=<vlanID>
```

Sets the ID for the virtual LAN (VLAN). Valid inputs are 0 (default) to 4094. When a VLAN ID other than 0 is set, the interface only accepts packets tagged with that specified ID. Outbound packets are also tagged with the specified ID.

Note: IEEE 802.1Q VLAN tagging is supported. VLAN tagging is not supported on ana interfaces.

# SEE ALSO

Configuring and Managing Celerra Networking and Configuring and Managing Celerra Network High Availability, server\_netstat, server\_setup, and server\_sysconfig.

# FRONT-END OUTPUT

The network device name is dependent on the front end of the system (for example, NS series Data Mover, 514 Data Movers, 510 Data Movers, and so on) and the network device type. NS series and 514 Data Movers network device name display a prefix of **cge**, for example, cge0. 510 or earlier Data Movers display a prefix of ana or ace, for example, ana0, ace0. Internal network devices on a Data Mover are displayed as el30, el31.

# **EXAMPLE #1**

To display parameters of all interfaces on a Data Mover, type:

# \$ server\_ifconfig server\_2 -all

```
server_2 :
loop protocol=IP device=loop
         inet=127.0.0.1 netmask=255.0.0.0 broadcast=127.255.255.255
        UP, loopback, mtu=32768, vlan=0, macaddr=0:0:0:0:0:0 netname=localhost
cge0 protocol=IP device=cge0
         inet=172.24.102.238 netmask=255.255.255.0 broadcast=172.24.102.255
```

516

```
UP, ethernet, mtu=1500, vlan=0, macaddr=0:60:16:4:29:87
el31 protocol=IP device=cge6
         inet=128.221.253.2 netmask=255.255.255.0 broadcast=128.221.253.255
       UP, ethernet, mtu=1500, vlan=0, macaddr=0:60:16:4:11:a6 netname=localhost
el30 protocol=IP device=fxp0
         inet=128.221.252.2 netmask=255.255.255.0 broadcast=128.221.252.255
       UP, ethernet, mtu=1500, vlan=0, macaddr=8:0:1b:43:7e:b8 netname=localhost
         EXAMPLE #2
                        To create an IP interface for Gigabit Ethernet, type:
                         $ server_ifconfig server_2 -create -Device cge1 -name
                         cge1 -protocol IP 172.24.102.239 255.255.255.0
                         172.24.102.255
server_2 : done
         EXAMPLE #3
                        To create an interface for network device cge0 with an IPv6 address
                         with a nondefault prefix length on server_2, type:
                         $ server_ifconfig server_2 -create -Device cge0 -name
                         cge0 int1 -protocol IP6
                         3ffe:0000:3c4d:0015:0435:0200:0300:ED20/48
server_2 : done
         EXAMPLE #4
                        To create an interface for network device cge0 with an IPv6 address
                         on server_2, type:
                         $ server_ifconfig server_2 -create -Device cge0 -name
                         cge0 int1 -protocol IP6
                         3ffe:0000:3c4d:0015:0435:0200:0300:ED20
server 2 : done
         EXAMPLE #5
                        To verify that the settings for the cge0_int1 interface for server_2 are
                         correct, type:
                         $ server_ifconfig server_2 cge0_int1
server 2 :
cge0_int1 protocol=IP6 device=cge0
         inet=3ffe:0:3c4d:15:435:200:300:ed20 prefix=48
        UP, Ethernet, mtu=1500, vlan=0, macaddr=0:60:16:c:5:5
                         Note: The bold item in the output highlights the nondefault 48-bit prefix.
         EXAMPLE #6
                        To verify that the interface settings for server_2 are correct, type:
                         $ server_ifconfig server_2 -all
server 2 :
el30 protocol=IP device=mge0
         inet=128.221.252.2 netmask=255.255.255.0 broadcast=128.221.252.255
       UP, Ethernet, mtu=1500, vlan=0, macaddr=0:60:16:d:30:b1 netname=localhost
```

el31 protocol=IP device=mge1

```
inet=128.221.253.2 netmask=255.255.255.0 broadcast=128.221.253.255
    UP, Ethernet, mtu=1500, vlan=0, macaddr=0:60:16:d:30:b2 netname=localhost
loop6 protocol=IP6 device=loop
    inet=::1 prefix=128
    UP, Loopback, mtu=32768, vlan=0, macaddr=0:0:0:0:0:0 netname=localhost
loop protocol=IP device=loop
    inet=127.0.0.1 netmask=255.0.0.0 broadcast=127.255.255.255
    UP, Loopback, mtu=32768, vlan=0, macaddr=0:0:0:0:0:0 netname=localhost
cge0_int1 protocol=IP6 device=cge0
    inet=3ffe:0:3c4d:15:435:200:300:ed20 prefix=64
    UP, Ethernet, mtu=1500, vlan=0, macaddr=0:60:16:c:2:5
cge0_0000_11 protocol=IP6 device=cge0
    inet=fe80::260:16ff:fe0c:205 prefix=64
    UP, Ethernet, mtu=1500, vlan=0, macaddr=0:60:16:c:2:5
```

**Note:** The first bold item in the output highlights the default 64-bit prefix. The second and third bold items highlight the link-local name and address that are automatically generated when you configure a global address for cge0. The automatically created link local interface name is made by concatinating the device name with the four digit VLAN ID between 0 and 4094. Note that the interface you configured with the IPv6 address 3ffe:0:3c4d:15:435:200:300:ed20 and the address with the link-local address fe80::260:16ff:fe0c:205 share the same MAC address. The link-local address is derived from the MAC address.

# **EXAMPLE #7** To verify that the interface settings for server\_2 are correct, type:

```
$ server_ifconfig server_2 -all
server_2 :
cge0 int2 protocol=IP device=cge0
         inet=172.24.108.10 netmask=255.255.255.0 broadcast=172.24.108.255
        UP, Ethernet, mtu=1500, vlan=0, macaddr=0:60:16:c:2:5
cge0_int1 protocol=IP6 device=cge0
         inet=3ffe:0:3c4d:15:435:200:300:ed20 prefix=64
        UP, Ethernet, mtu=1500, vlan=0, macaddr=0:60:16:c:2:5
cge0_0000_ll protocol=IP6 device=cge0
         inet=fe80::260:16ff:fe0c:205 prefix=64
        UP, Ethernet, mtu=1500, vlan=0, macaddr=0:60:16:c:2:5
el30 protocol=IP device=mge0
         inet=128.221.252.2 netmask=255.255.255.0 broadcast=128.221.252.255
       UP, Ethernet, mtu=1500, vlan=0, macaddr=0:60:16:d:30:b1 netname=localhost
el31 protocol=IP device=mge1
         inet=128.221.253.2 netmask=255.255.255.0 broadcast=128.221.253.255
       UP, Ethernet, mtu=1500, vlan=0, macaddr=0:60:16:d:30:b2 netname=localhost
loop6 protocol=IP6 device=loop
         inet=::1 prefix=128
        UP, Loopback, mtu=32768, vlan=0, macaddr=0:0:0:0:0:0 netname=localhost
loop protocol=IP device=loop
         inet=127.0.0.1 netmask=255.0.0.0 broadcast=127.255.255.255
```

UP, Loopback, mtu=32768, vlan=0, macaddr=0:0:0:0:0:0 netname=localhost

Note: The bold items in the output highlight the IPv4 interface, cge0\_int2, and the IPv6 interface, cge0\_int1. **EXAMPLE #8** To disable an interface, type: \$ server ifconfig server 2 cge0 int2 down server\_2 : done **EXAMPLE #9** To enable an interface, type: \$ server ifconfig server 2 cge0 int2 up server\_2 : done EXAMPLE #10 To reset the MTU for Gigabit Ethernet, type: \$ server ifconfig server 2 cge0 int2 mtu=9000 server\_2 : done EXAMPLE #11 To set the ID for the Virtual LAN, type: \$ server\_ifconfig server\_2 cge0\_int1 vlan=40 server 2 : done EXAMPLE #12 To verify that the VLAN ID in the interface settings for server 2 are correct, type: \$ server\_ifconfig server\_2 -all server\_2 : cge0\_int1 protocol=IP6 device=cge0 inet=3ffe:0:3c4d:15:435:200:300:ed20 prefix=64 UP, Ethernet, mtu=1500, vlan=40, macaddr=0:60:16:c:2:5 cge0\_0040\_11 protocol=IP6 device=cge0 inet=fe80::260:16ff:fe0c:205 prefix=64 UP, Ethernet, mtu=1500, vlan=40, macaddr=0:60:16:c:2:5 cge0\_int2 protocol=IP device=cge0 inet=172.24.108.10 netmask=255.255.255.0 broadcast=172.24.108.255 UP, Ethernet, mtu=1500, vlan=20, macaddr=0:60:16:c:2:5 el30 protocol=IP device=mge0 inet=128.221.252.2 netmask=255.255.255.0 broadcast=128.221.252.255 UP, Ethernet, mtu=1500, vlan=0, macaddr=0:60:16:d:30:b1 netname=localhost el31 protocol=IP device=mge1 inet=128.221.253.2 netmask=255.255.255.0 broadcast=128.221.253.255 UP, Ethernet, mtu=1500, vlan=0, macaddr=0:60:16:d:30:b2 netname=localhost loop6 protocol=IP6 device=loop inet=::1 prefix=128 UP, Loopback, mtu=32768, vlan=0, macaddr=0:0:0:0:0:0 netname=localhost loop protocol=IP device=loop inet=127.0.0.1 netmask=255.0.0.0 broadcast=127.255.255.255 UP, Loopback, mtu=32768, vlan=0, macaddr=0:0:0:0:0:0 netname=localhost

**Note:** The bold items in the output highlight the VLAN tag. Note that the link-local address uses the VLAN tag as part of its name.

# **EXAMPLE #13**

To delete an IP interface, type:

\$ server\_ifconfig server\_2 -delete cge1\_int2

server\_2 : done

**Note:** The autogenerated link local interfaces cannot be deleted.

# server ip

Manages the IPv6 neighbor cache and route table for the Celerra Network Server.

# **SYNOPSIS**

```
server_ip {ALL | <mover>}
 -neighbor {
     -list [<v6addr> [-interface <ifname>]]
     -create <v6addr> -lladdress <macaddr> [-interface <ifname>]
     -delete {-all|<v6addr> [-interface <ifname>]}
-route {
     -list
   -create {
       -destination <destination> -interface <ifname>
      -default -gateway <v6qw> [-interface <ifname>]
    }
   -delete {
       -destination <destination>
      -default -gateway <v6gw> [-interface <ifname>]
      -all
  }
```

# DESCRIPTION

**server** ip creates, deletes, and lists the neighbor cache and route tables.

# **OPTIONS**

```
server_ip {<movername> | ALL}
```

Sends a request to the Data Mover to get IPv6 parameters related to the IPv6 routing table and neighbor cache.

**ALL** option executes the command for all Data Movers.

```
-neighbor {-list|-create|-delete}
```

Lists, creates, or deletes the neighbor cache entries from the neighbor cache table.

#### -list

Displays the neighbor cache entries.

#### -create

Creates a neighbor cache table entry with the specified details.

# -delete

Deletes the specified neighbor cache table entries or all entries.

```
-route {-list|-create|-delete}
```

Lists, creates, or deletes entries in the IPv6 route table.

521

# -list

Displays the IPv6 route table.

# -create

Creates a route table entry with the specified details.

#### -delete

Deletes the specified route table entries.

# **EXAMPLE #1** To view a list of neighbor cache entries on Data Mover **server\_2**, type:

\$ server\_ip server\_2 -neighbor -list

# server\_2:

Address	Link layer address	Interface	Туре	State
fe80::204:23ff:fead:4fd4	0:4:23:ad:4f:d4	cge1_0000_ll	host	STALE
fe80::216:9cff:fe15:c00	0:16:9c:15:c:0	cge1_0000_ll	router	STALE
fe80::216:9cff:fe15:c00	0:16:9c:15:c:0	cge4_0000_ll	router	STALE
fe80::216:9cff:fe15:c00	0:16:9c:15:c:0	cge3_2998_II	router	STALE
fe80::216:9cff:fe15:c00	0:16:9c:15:c:0	cge2_2442_ll	router	STALE
3ffe::1	0:16:9c:15:c:10	cge3_0000_II	router	REACHABLE

# Where:

Value	Definition
Address	The neighbor IPv6 address.
Link layer address	The link layer address of the neighbor.
Interface	Interface name of the interface connecting to the neighbor.
Туре	Type of neighbor. The neighbor can be either host or router.
State	The state of the neighbor such as REACHABLE, INCOMPLETE, STALE, DELAY, or PROBE.

# **EXAMPLE #2**

To view a list of neighbor cache entries for a specific IP address on Data Mover **server\_2**, type:

\$ server\_ip server\_2 -neighbor -list
fe80::216:9cff:fe15:c00

# server\_2:

Address	Link layer address	Interface	Туре	State
fe80::216:9cff:fe15:c00	0:16:9c:15:c:0	cge1_0000_ll	router	STALE
fe80::216:9cff:fe15:c00	0:16:9c:15:c:0	cge4_0000_ll	router	STALE
fe80::216:9cff:fe15:c00	0:16:9c:15:c:0	cge3_2998_ll	router	STALE
fe80::216:9cff:fe15:c00	0:16:9c:15:c:0	cge2_2442_ll	router	STALE

**EXAMPLE #3** To view a list of neighbor cache entries for a specific IP address and interface type, on Data Mover **server\_2**, type:

\$ server\_ip server\_2 -neighbor -list
fe80::216:9cff:fe15:c00 -interface cge1\_0000\_11

server\_2:

Address	Link layer address	Interface	Туре	State
fe80::216:9cff:fe15:c00	0:16:9c:15:c:0	cge1_0000_ll	router	STALE

**EXAMPLE #4** To add an entry to the neighbor cache for a global unicast IPv6 address, on Data Mover **server\_2**, type:

\$ server\_ip server\_2 -neighbor -create 2002:8c8:0:2310::2
-lladdress 0:16:9c:15:c:15

OK

**EXAMPLE #5** To add an entry to the neighbor cache for a link local IPv6 address, on Data Mover **server\_2**, type:

\$ server\_ip server\_2 -neighbor -create fe80::2 -lladdress
0:16:9c:15:c:12 -interface cge1v6

OK

**EXAMPLE #6** To delete an entry from the neighbor cache for a global unicast IPv6 address, on Data Mover **server\_2**, type:

\$ server\_ip server\_2 -neighbor -delete
2002:8c8:0:2310:0:2:ac18:f401

OK

**EXAMPLE #7** To delete an entry from the neighbor cache for a link local IPv6 address, on all Data Movers, type:

\$ server\_ip ALL -neighbor -delete fe80::1 -interface
cge1v6

OK

**EXAMPLE #8** To delete entries from the neighbor cache on Data Mover **server\_2** type:

\$ server\_ip server\_2 -neighbor -delete -all

OK

**EXAMPLE #9** To view a list of route table entries on Data Mover **server\_2**, type:

\$ server\_ip server\_2 -route -list

# server\_2:

Destination	Gateway	Interface	Expires (secs)
2002:8c8:0:2310::/64		cge1v6	0
2002:8c8:0:2311::/64		cge1v6	0
2002:8c8:0:2312::/64		cge1v6	0
2002:8c8:0:2313::/64		cge1v6	0
default	fe80::260:16ff:fe05:1bdd	cge1_0000_ll	1785
default	fe80::260:16ff:fe05:1bdc	cge1_0000_ll	1785
default	2002:8c8:0:2314::1	cge4v6	0
selected default	fe80::260:16ff:fe05:1bdd	cge1_0000_ll	1785

# Where:

Value	Definition
Destination	The prefix of the destination or the default route entry. There can be multiple default routes, but only one is active and shown as "selected default". The default sorting of the destination column displays the default routes at the bottom of the list and the selected default at the end of the list.
Gateway	The default gateway for default route entries. This value is blank for prefix destination entries.
Interface	Interface name of the interface used for the route.
Expires	The time until the route entry is valid. Zero denotes route is permanent and does not have an expiry.

# **EXAMPLE #10**

To add a default route table entry on the Data Mover **server\_2** to the destination network with the specified prefix, type:

\$ server\_ip server\_2 -route -create -destination
2002:8c8:0:2314::/64 -interface cge4v6

OK

# **EXAMPLE #11**

To add a default route table entry on the Data Mover **server\_2** through the specified gateway, type:

\$ server\_ip server\_2 -route -create -default -gateway
2002:8c8:0:2314::1

OK

# **EXAMPLE #12**

To add a default route table entry on the Data Mover **server\_2** through the specified gateway using the link-local interface, type:

\$ server\_ip server\_2 -route -create -default -gateway
fe80::1 -interface cge1v6

OK

	EXAMPLE #13	To delete an entry from the route table with an IPv6 prefix route destination for all Data Movers, type:
OK		<pre>\$ server_ip ALL -route -delete -destination 2002:8c8:0:2314::/64</pre>
	EXAMPLE #14	To delete an entry from the route table for a global unicast IPv6 address, on Data Mover <b>server_2</b> , type:
OK		<pre>\$ server_ip server_2 -route -delete -default -gateway 2002:8c8:0:2314::1</pre>
	EXAMPLE #15	To delete an entry from the route table for a link local IPv6 address, on Data Mover server_2, type:
OK		<pre>\$ server_ip server_2 -route -delete -default -gateway fe80::1 -interface cge1v6</pre>
	EXAMPLE #16	To delete all entries from the IPv6 route table on Data Mover <b>server_2</b> type:
OK		<pre>\$ server_ip server_2 -route -delete -all</pre>

# server\_iscsi

Manages the iSCSI configuration for the specified Data Movers.

# **SYNOPSIS**

```
server_iscsi {<movername> | ALL}
   -target <target_options>
| -lun <lun_options>
| -mask <lun_mask_options>
| -ns <name_service_options>
| -service {-start|-stop|-status}
| -snap <snap_options>
| -help
```

# **DESCRIPTION**

Manages iSCSI targets, logical unit numbers (LUNs), and snapshots; performs name service management (iSNS) and iSCSI service management.

To get usage for all options, type **-help**, or to get options for target, LUN, LUN mask, name service, or snapshot management, type **-target**, **-lun**, **-mask**, **-ns**, or **-snap**, respectively after the <movername>.

The **ALL** option executes the command for all Data Movers.

# **TARGET OPTIONS**

```
-target {
    -alias <alias_name> [-Q <iscsi_name>]
    -create [<pg_tag>:np=<np_list> [<pg_tag>:np=<np_list>]]
| -delete <alias_name>
| -rename <old_alias_name> <new_alias_name>
| -bind <alias_name> <pg_tag>:np=<np_list>
| -unbind <alias_name> {<pg_tag>|np=<np_list>}
| -info {<alias_name> | -all}
| -stat {<alias_name> | -all}
| -list}
```

To display target options, type **-target**.

```
-target -alias <alias_name> [-Q <iscsi_name>]
-create [<pg_tag>:np=<np_list> [<pg_tag> :np=<np_list> ]]
Creates an iSCSI target with the specified alias on the Data Mover
with an optional portal group tag and a list of network portals.
```

The **-alias** <alias\_name> option assigns a name to the new iSCSI target for local identification. The <alias\_name> is used for designating a specific iSCSI target in other commands. The <alias\_name> is not used for authentication. The **-Q** <iscsi\_name>

option specifies an iSCSI name string that must be entered as either an iSCSI Qualified Name ("iqn.") or in IEEE EUI-64 format ("eui."). If no name is entered, an iSCSI-compliant target name is generated automatically.

The iSCSI standard requires that an iSCSI name be globally unique and in a specific format. To avoid entering illegal names, do not use the **-Q** option; instead, let the Celerra Network Server generates a legal iSCSI name.

For international support, both the target name (given by **-alias**) and the iSCSI name (given by **-Q**) support strings of Unicode characters in UTF-8 encoding. The <pg\_tag>, which is the portal group tag, identifies the portal group within an iSCSI node. The <pg\_tag> is an integer within the range of 0 to 65535.

The <np\_list> is a comma-separated list of network portals. A network portal in a target is identified by its IP address and its listening TCP port. The format of a network portal is {<IPv4Addr>[:<port>] | <IPv6Addr> | <\[IPv6Addr\]:port> }. IPv6 addresses need to be enclosed in square brackets if a port is specified; the brackets do not signify optional content. If unspecified, port 3260 is used.

-target -delete <alias\_name>
Removes a target specified by its <alias\_name> from a Data Mover.



# **CAUTION**

If one or more initiators are logged in to the target, or if all logical units have not been removed, the -delete option will fail. The -delete option should be performed with caution when no initiators are logged in to the target. The preconditions listed above are provided as a safeguard to prevent a system administrator from deleting the wrong target.

<sup>-</sup>target -rename <old\_alias\_name> <new\_alias\_name>
Changes the <alias\_name> of the target.

**<sup>-</sup>target -bind** <alias\_name> <pg\_tag>:np=<np\_list> Adds target portal group tags or network portals to an existing target specified by the <alias\_name>. If the portal group tag number does not exist, a new portal group is created and bound to the target.

<sup>-</sup>target -unbind <alias\_name>{<pg\_tag>|np=<np\_list>}
Removes a portal group tag and network portals from an existing target.

**Note:** In case the portal group has only one network portal, then removing this network portal also removes the portal group and the associated sessions.

```
-target -info {<alias_name> | -all}
```

Displays attributes such as alias, iSCSI name, serial number, portal group tag, network portal, and details about logical units and connected initiators for all targets or the specified target.

```
-target -stat {<alias_name>|-all}
```

Displays statistical attributes such as initiator login/logout statistics and iSCSI session statistics for all targets or the specified target.

# -target -list

Displays a listing of all targets.

# **LUN OPTIONS**

To display LUN options, type **-lun**. The Celerra Network Server supports a maximum of 256 LUNs.

```
-lun -number <lun_number> -create <target_alias_name>
-size <size> -fs <fs name>
```

Creates a new logical unit number (LUN) and assigns it to an existing target specified by the <target\_alias\_name>. The <lun\_number> can be an integer within the range of 0 to 255, but must be unique on the target or the command fails. The <size> specifies the size of the LUN in megabytes (M, the default), gigabytes (G), or terabytes (T). Maximum LUN size is 2 TB minus 1 MB. The <fs\_name> specifies the name of a mounted filesystem that will serve as the storage pool for the LUN. When creating regular LUNs (the default LUN type), the mounted filesystem must be able to support the size specified.

# [-vp {yes | no}]

The **-vp yes** option creates a virtually provisioned iSCSI LUN instead of a regular LUN. Regular LUNs reserve space on the filesystem for the entire LUN at the time of creation, but virtually provisioned LUNs do not reserve space. Virtually provisioned LUNs can be created larger than the underlying filesystem; however, when data is written to the virtually provisioned LUN, the filesystem must have sufficient free space to contain the data. The default is **no**.



# **CAUTION**

When using virtually provisioned LUNs, monitoring the filesystem space usage is very important. Attempting to write data to a virtually provisioned LUN without sufficient free space in the underlying filesystem can result in client filesystem corruption on the iSCSI host system.

# [-readonly {yes|no}]

The **-readonly yes** option creates a protected, read-only LUN that is available as a destination in an iSCSI LUN replication relationship. By default, the iSCSI LUN is created as read-write. The default is **no**.

-lun -modify <lun\_number> -target <target\_alias\_name>
-readonly {yes [-Force] | no}

Modifies a LUN by changing it to read-only or read/write. The LUN cannot be a promoted snapshot. The **-readonly no** option converts the specified read-only LUN into a read/write LUN (for example, to change a replication destination LUN into a copy of the source LUN). The **-readonly yes** option converts the specified read/write LUN to read-only (for example, to prepare a replication source LUN for failback). If the LUN's mask grants access to an iSCSI initiator, and that initiator is logged in to the target that contains the LUN, the **-readonly yes** option fails unless the **-Force** option is included.

- **-lun -extend** <target\_alias\_name> **-size** <size> [M|G|T]Extends an existing LUN by the specified number of megabytes (M, the default), gigabytes (G), or terabytes (T).
- -lun -delete <lun\_number> -target <target\_alias\_name>
  Deletes a LUN from a target as specified by the <target\_alias\_name>.



# **CAUTION**

The LUN must not be in use when a delete is performed. Deleting a LUN deletes all of the data stored on the LUN.

```
-lun -info {<lun_number>|-all} [-target
<target_alias_name>]
```

Displays the size, space usage, and filesystem information where the LUN resides, for the specified LUN on the specified target, or on all LUNs.

```
-lun -stat {<lun_number>|-all} [-target
<target_alias_name>]
```

Displays statistical information such as size, LUN type, filesystem location, replication relationship (if any), maximum extension size, and SCSI CDB statistics for the specified LUN or all LUNs on the specified target.

-lun -list [-target <target\_alias\_name>]
Displays information on all LUNs, or to those belonging to a
specified target.

# **MASK OPTIONS**

To display mask options, type **-mask**.

```
-mask -list [<target_alias_name>]
```

Displays a list of existing LUN masks on the existing target on the specified Data Mover.

```
-mask -info <target_alias_name> {-initiator
<initiator_name>|-all}
```

Returns LUN mask information for the specified target, or all targets including the alias name, iSCSI name, portal group information, default LUN mask information, and list of existing LUNs. The <initiator\_name> is the iSCSI name of the initiator to which the mask command is applied.

```
-mask -set <target_alias_name> -initiator
<initiator_name>
```

Sets the initiator access options (grant or deny or both) for the LUN mask on the specified <target\_alias\_name>. The <initiator\_name> is the iSCSI name of the initiator to which the mask is applied. Outside of a clustered environment, do not grant multiple initiators access to the same LUN. Conflicts can occur if more than one initiator tries writing to the LUN. If your environment requires multiple initiators to access the same LUN, create a mask for each initiator through a separate command.

# -grant <access\_list>

Allows the initiator access to all LUNs specified in <access\_list>. The <access\_list> is a LUN number or a comma-separated list of LUN numbers. Access-list examples:

```
grant 2, 4, 6 grant 6-8
```

```
-deny <access_list>
```

Denies the initiator access to all LUNs specified in <access\_list>. Access-list example: deny 1, 3–5, 9.

```
-mask -clear <target_alias_name> -initiator
<initiator_name>
```

Clears the existing LUN mask for the initiator.

# NAME SERVICE OPTIONS

Performs name service client management. Before modifying the **-ns** server or port, iSCSI service must be stopped, then restarted once the changes are made. If the administrator fails to stop the service, attempted modifications will have no effect even though the command reports "done." Available options are:

```
-ns isns {
| -info
| -set {-server {<IPv4Addr>[:<port>] | <IPv6Addr>| <\[IPv6Addr\]:port>}
| -clear {-server | -esiport | -all}}
```

To display name service options, type -ns.

#### -ns isns

Selects the naming service to be configured.

**Note:** iSNS is the only naming service supported.

#### -ns isns -info

Displays status information for the iSNS service such as state, iSNS server address, entity, source, Entity Status Inquiry (ESI) information, and timeout value.

```
-ns isns -set -server {<IPv4Addr>[:<port>] |
<IPv6Addr>|<\[IPv6Addr\]:port>}
```

Set the IPv4 address or the IPv6 address and, optionally, the port of the iSNS server. IPv6 addresses need to be enclosed in square brackets if a port is specified; the brackets do not signify optional content. The default port is 3205.

# -ns isns -set -esiport <port>

Sets the Entity Status Inquiry port of iSNS client. This port is a TCP port that by default, when iSCSI service is started up, is dynamically bound at the iSNS client. If this option is not entered at all, the application will assign a port when the service is started.

```
-clear {-server|-esiport|-all}
```

Clear current settings for iSNS server, for the ESI port, or for both the iSNS server and ESI port.

When a **-clear** option is issued, the system returns the message "done," however, settings are not cleared until iSCSI service is restarted.

# iSCSI SERVICE OPTIONS

```
-service {-start|-stop|-status}
```

Starts, stops, or displays the status of the iSCSI service.

# **SNAP OPTIONS**

Performs snap management functions. Available options are:

```
-snap {
    -list [-target <target_alias_name> -lun <lun_number>]
| -info <snap_name>
| -create -target <target_alias_name> -lun <lun_number>
        [-data <app_data>]
| -modify <snap_name> -data <app_data>
| -delete {<snap_name> | -target <target_alias_name> -lun <lun_number>}
        [-Force]
| -restore <snap_name> -initiator <initiator_name>
| -pemote <snap_name> [-Force] }
```

To display snap management options, type **-snap**.

```
-snap -list [-target <target_alias_name> -lun
<lun_number>]
```

Lists all snapshots on the Data Mover or all snapshots of the specified iSCSI LUN (regardless of which application created the snapshots).

```
-snap -info <snap_name>
```

Displays information about the specified snapshot, including any application data (if the data is UTF-8 compliant).

```
-snap -create -target <target_alias_name> -lun
<lun_number>[-data <app_data>]
```

Creates a snapshot of the specified iSCSI LUN. The snapshot is crash consistent, after a power loss, and not necessarily consistent from the point of view of the host applications. If the **-data** option is included, the provided comment text (limited to 4096 bytes) is associated with the snapshot as application data and saved on persistent storage provided by the Data Mover. If it contains a space character, the string must be enclosed in double quotation marks. To clear existing data, provide a null string (" ").

**Note:** Although the snapshot supports 4096 bytes, the command line interface might limit the number of characters you can type.

```
-snap -modify <snap_name> -data <app_data>
```

Replaces the application data (text or binary data) associated with the snapshot with the provided comment text (limited to 4096 bytes) or a null string. If it contains a space character, the string must be enclosed in double quotation marks. To clear existing data, provide a null string ("").

**Note:** Although the snapshot supports 4096 bytes, the command line interface might limit the number of characters you can type.

```
-snap -delete {<snap_name> | -target
```

<target\_alias\_name>-lun <lun\_number>} [-Force]

Deletes either the specified snapshot or all snapshots of the specified iSCSI LUN. The snapshot to delete cannot be promoted. To delete a snapshot created by an application other than the Control Station CLI, include **-Force**.

```
-snap -restore <snap_name>
```

Restores the content of a production iSCSI LUN from a snapshot of that LUN on the same target. The snapshot used for the LUN restoration is preserved, but all newer snapshots of the LUN are deleted. The snapshot must be the same size as the production LUN (that is, the restore fails if the production LUN was extended after the snapshot was taken). The snapshot (or a more recent snapshot of the same LUN) cannot be promoted.

-snap -promote -snap\_name> -initiator <initiator\_name> Promotes the specified snapshot so it can be used as an iSCSI LUN. The new LUN has the same data content as the snapshot. A LUN mask is created to allow access to the LUN by the specified iSCSI initiator.

-snap -Demote <snap\_name> [-Force]

Demotes the specified snapshot LUN. To demote a snapshot created by another application (not the Control Station CLI) or a snapshot that is in use, include **-Force**. The demoted LUN and its contents (including data written to the LUN while promoted) are irretrievable. The read-only snapshot is preserved.

SEE ALSO Configuring iSCSI Targets on Celerra, Installing Celerra iSCSI Host Components, Using Celerra Replicator (V2), Using Celerra Replicator for iSCSI (V1), and nas\_replicate.

To create a target with an alias of t1, and a portal group of 100, and a network portal of np=172.24.102.242, type:

\$ server\_iscsi server\_2 -target -alias t1 -create
100:np=172.24.102.242

server\_2 : done

**EXAMPLE #1** 

**EXAMPLE #2** 

To create a target with an alias of t1, and a portal group of 100, and a network portal of np=172.24.102.242 [2002:c8c::24:172:65], type:

\$ server\_iscsi server\_2 -target -alias t1 -create
100:np=172.24.102.242,[2002:c8c::24:172:65]

server\_2 : done

**EXAMPLE #3** To create a target with an alias of **t1**, and a portal group of **100**, with a port of 3261, and a network portal of **np=172.24.102.242:3261,[2002** :c8c::24:172:65]:3261, type:

\$ server\_iscsi server\_2 -target -alias t1 -create
100:np=172.24.102.242:3261,[2002:c8c::24:172:65]:3261

server\_2 : done

**EXAMPLE #4** To rename a target alias, type:

\$ server iscsi server 2 -target -rename t1 t2

server\_2 : done

**EXAMPLE #5** To bind target portal groups as specified by the alias **t1** to an existing target, **200:np=172.24.102.243**, type:

\$ server\_iscsi server\_2 -target -bind t1
200:np=172.24.102.243

server\_2 : done

# **EXAMPLE #6**

To bind target portal groups as specified by the alias **t1** to an existing target with a port of 63, **200:np=2002:c8c::24:172:63**, type:

\$ server\_iscsi server\_2 -target -bind t1
200:np=2002:c8c::24:172:63

server\_2 : done

# **EXAMPLE #7**

To configure server\_2 to use the iSNS server at IP address

**200:c8c::24:172:63** with the default port, type:

\$ server\_iscsi server\_2 -ns isns -set -server
2002:c8c::24:172:63

server\_2 : done

# **EXAMPLE #8**

To list all targets, type:

\$ server\_iscsi server\_2 -target -list

server\_2 :

alias\_name por

portal\_groups iscsi\_name

t1 100,200

iqn.1992-05.com.emc:apm000438070430000-1

# Where:

Value	Definition
alias_name	Alias for the iSCSI target's qualified name.
portal_groups	Portal groups used by the target.
iscsi_name	iSCSI qualified name of the target.

# **EXAMPLE #9**

To create a new standard iSCSI logical unit number (LUN) of **0** and assign it to an existing target, **t1**, with a specified size of **102400** MB on filesystem, **ufs1**, type:

\$ server\_iscsi server\_2 -lun -number 0 -create t1 -size
102400 -fs ufs1

server\_2 : done

# EXAMPLE #10

To create a new virtually provisioned iSCSI logical unit number (LUN) of 1 and assign it to an existing target, t1, with a specified size of 10 GB on filesystem, ufs1, type:

\$ server\_iscsi server\_2 -lun -number 1 -create t1 -size
10G -fs ufs1 -vp yes

server\_2 : done

# EXAMPLE #11

To create a new, protected, read-only iSCSI LUN of **2** and assign it to an existing target **t1**, with a specified size of **10** GB on filesystem, **ufs1**, type:

\$ server\_iscsi server\_2 -lun -number 2 -create t1 -size
10G -fs ufs1 -readonly yes

server\_2 : done

# **EXAMPLE #12** To extend LUN **0** on target, **t1**, by **4096** MB, type:

\$ server\_iscsi server\_2 -lun -extend 0 -target t1 -size
4096M

server\_2 : done

# EXAMPLE #13

To display a list of all LUNs on a Data Mover, type:

\$ server\_iscsi server\_2 -lun -list

server\_2 :
target: t1

lun size(MB) filesystem
0 106496 ufs1 (id=22)
1 10240 ufs1 (id=22)
2 10240 ufs1 (id=22)

# Where:

Value	Definition
target	Alias for the iSCSI target's qualified name.
lun	ID for the LUN.
size (MB)	Size of the LUN in MBs.
filesystem	Name of the filesystem containing the LUN.

# **EXAMPLE #14** To display information for LUN **0**, on target **t1**, type:

\$ server\_iscsi server\_2 -lun -info 0 -target t1

server\_2 :

Logical Unit 0 on target t1:

(Production) fsid=22 size=106496MB alloc=0MB dense

path=/ufs1/fs22\_T1\_LUN0\_APM00043807043\_0000/fs22\_T1\_LUN0\_APM00043807043\_0000
(extended)

replication=none

max\_extension\_size=84494MB

# Where:

Value	Definition
fsid	ID of the filesystem containing the LUN.
size	Reserved size of the LUN in MBs.
alloc	Actual space allocated for the LUN.
dense	Storage method used by the LUN.

path	Filesystem path to the LUN.
replication	Status of the LUN for iSCSI replication. Possible statuses are:  none — The LUN is not in a replication relationship.  source — The LUN is the source LUN in a replication relationship.
	<ul> <li>destination — The LUN is the destination LUN in a replication relationship.</li> <li>available destination — The LUN is configured as a destination LUN (read-only) but not currently in a replication relationship.</li> </ul>
max_extension_size	Maximum amount of space by which the LUN can be extended.

# **EXAMPLE #15** To display statistical information for all LUNs on a target t1, type:

```
$ server_iscsi server_2 -lun -stat -all -target t1
server_2 :
Logical Unit 0 on target t1:
  (Production) fsid=22 size=106496MB alloc=218MB dense
path=/ufs1/fs22_T1_LUN0_APM00043807043_0000/fs22_T1_LUN0_APM00043807043_0000
(extended)
              replication=none
              max_extension_size=84441MB
Statistics:
  Total SCSI CDBs: 6837
  Read: 2381 Failed: 0
                               Blks read:
                                              147769
  Write: 4314
               Failed: 0
                               Blks written: 449833
Logical Unit 1 on target t1:
  (Production) fsid=22 size=10240MB alloc=53MB virtually_provisioned
path=/ufs1/fs22_T1_LUN1_APM00043807043_0000/fs22_T1_LUN1_APM00043807043_0000
              replication=none
              max_extension_size=2086911MB
Statistics:
  Total SCSI CDBs: 2272
  Read: 1179 Failed: 0
                               Blks read:
                                              111831
  Write: 973
                Failed: 0
                               Blks written: 110617
Logical Unit 2 on target t1:
  (Production) fsid=22 size=10240MB alloc=0MB dense
path=/ufs1/fs22_T1_LUN2_APM00043807043_0000/fs22_T1_LUN2_APM00043807043_0000
              replication=available destination
              max_extension_size=84441MB
Statistics:
  Total SCSI CDBs: 0
  Read: 0 Failed: 0
                               Blks read:
  Write: 0
                Failed: 0
                                Blks written: 0
```

# **EXAMPLE #16** To set a LUN mask on target **t1**, for initiator **iqn.1991-05.com.microsoft:winserver1.nasdocs.emc.com**, with the access rights to LUNs **0** and **1**, type:

\$ server\_iscsi server\_2 -mask -set t1 -initiator iqn.1991-05.com.microsoft:winserver1.nasdocs.emc.com -grant 0,1

server\_2 : done

# EXAMPLE #17

To display a list of LUN masks, type:

\$ server\_iscsi server\_2 -mask -list

server\_2 : target: t1

initiator\_name grant LUNs 0 - 1

ign.1991-05.com.microsoft:winserver1.nasdocs.emc.com

#### Where:

Value	Definition
target	Alias name of the target.
initiator_name	Qualified name of the iSCSI initiator on the host.
grant LUNs	Number of the LUNs to which the initiator is granted access.

# **EXAMPLE #18**

To clear the LUN mask for the initiator,

iqn.1991-05.com.microsoft:winserver1.nasdocs.emc.com, type:

\$ server\_iscsi server\_2 -mask -clear t1 -initiator ign.1991-05.com.microsoft:winserver1.nasdocs.emc.com

server\_2 : done

# **EXAMPLE #19**

To configure an iSNS client on the Data Mover for iSNS server with IP address of 172.24.102.66, type:

\$ server\_iscsi server\_2 -ns isns -set -server 172.24.102.66

server\_2 : done

**Note:** Before modifying the **-ns** server or port, iSCSI service must be stopped, then restarted once the changes are made. If the administrator fails to stop the service, attempted modifications will have no effect until the service is started even though the command reports "done."

# **EXAMPLE #20**

To start the iSCSI service on the Data Mover, type:

\$ server\_iscsi server\_2 -service -start

server\_2 : done

# **EXAMPLE #21**

To display the iSNS client service status, type:

\$ server\_iscsi server\_2 -ns isns -info

server\_2 :

iSNS Client Service Status

service : started server : 172.24.1 : 172.24.102.66:3205[tcp]

: SYNC state

entity : isns:00000002

source : iqn.1992-05.com.emc:apm000438070430000-1

ESI support : TRUE ESI port : 52427

ESI Time : 1131484090 (Tue Nov 8 21:08:10 2005)

timeout : 3000 trace : FALSE PG support : TRUE

# Where:

Value	Definition
service	Whether the iSNS service is started or stopped.
server	IP address of the iSNS server and port.
state	<ul> <li>State of the iSNS server. Possible states are:</li> <li>SYNC — The iSNS client is OK.</li> <li>VIRGIN — The iSNS client is not in use—there are no targets or portals configured.</li> </ul>
	<ul> <li>INDOUBT — The iSNS client cannot currently contact the iSNS server and is retrying.</li> <li>NOTSYNC — The iSNS client has experienced multiple failures attempting to contact the iSNS server and is retrying.</li> </ul>
	• DEAD — The iSNS client cannot contact the iSNS server. Still retrying at a lower frequency.
entity	Name of the network entity that serves as a container for iSCSI targets and portal objects on the Data Mover.
source	Name of the iSCSI target used as the source attribute in iSNS requests that require source identification.
ESI support	Whether Entity Status Inquiry (ESI) is supported as the means for refreshing the entity registration.
ESI port	TCP port that the iSNS client service listens to for incoming ESI messages.
ESI Time	Last time the ESI was received from the iSNS server or UNKNOWN if never.
timeout	Network timeout value (in milliseconds) for communication with the iSNS server.
trace	Whether extensive tracing output of iSNS messages, both incoming and outgoing, is enabled (TRUE) or disabled (FALSE).
PG support	Whether portal groups are supported; if not, all portals are implicitly placed in one portal group.

#### **EXAMPLE #22** To display status information for the target **t1**, type:

# \$ server\_iscsi server\_2 -target -info t1

server\_2 :

Target Alias Name: t1

Target iSCSI Name: iqn.1992-05.com.emc:apm000438070430000-1

Serial Number: 1

Portal Group Tag: 100

Portal: 172.24.102.242:3260

Portal Group Tag: 200

Portal: 172.24.102.243:3260

```
Logical Units:
    0 : (Production) fsid=22 size=106496MB alloc=0MB dense
path=/ufs1/fs22_T1_LUN0_APM00043807043_0000/fs22_T1_LUN0_APM00043807043_0000
    1 : (Production) fsid=22 size=10240MB alloc=0MB virtually_provisioned
path=/ufs1/fs22_T1_LUN1_APM00043807043_0000/fs22_T1_LUN1_APM00043807043_0000
    2 : (Production) fsid=22 size=10240MB alloc=0MB dense
path=/ufs1/fs22_T1_LUN2_APM00043807043_0000/fs22_T1_LUN2_APM00043807043_0000
Connected Initiators:
```

ign.1991-05.com.microsoft:winserver1.nasdocs.emc.com

#### Where:

Value	Definition
Target Alias Name	Alias for the iSCSI target's qualified name.
Target iSCSI Name	iSCSI qualified name of the target.
Serial Number	Unique, Celerra-wide serial number for the target.
Portal Group Tag	Portal groups used by the target.
Portal	IP address and port of the portals in the portal group.
Logical Units	Logical units (LUNs) assigned to the target.
fsid	ID the filesystem containing the LUN.
size	Reserved size of the LUN in MBs.
alloc	Amount of data currently stored in the LUN.
dense	Storage method used by the LUN.
path	Filesystem path to the LUN.
Connected Initiators	Qualified name of the initiators logged in to the target.

# **EXAMPLE #23** To display statistics for the target **t1**, and the current sessions on the target, type:

```
$ server_iscsi server_2 -target -stat t1
server_2 :
Target Alias Name: t1
Target iSCSI Name: ign.1992-05.com.emc:apm000438070430000-1
[Login/Logout Stats]
Login Accepted: 1
Login Failed: 0
  Redirect: 0
  Authorize: 0
  Authenticate: 0
  Negotiate: 0
  Other: 0
Logout Normal: 0
Logout Other: 0
Last Login Failed: N/A
[Session Stats]
Initiator: iqn.1991-05.com.microsoft:winserver1.nasdocs.emc.com TSIH: 1 ISID:
400001370000
Command PDUs: 5886
  Read Command PDUs: 3604
```

Write Command PDUs: 2256

Response PDUs: 5886 Data-In PDUs: 1936 Data-Out PDUs: 0

R2Ts: 0

Tx Data Octets: 132134422 Rx Data Octets: 132573696

Digest Errors: 0

Connection Timeout Errors: 0

# Where:

Value	Definition
Target Alias Name	Alias for the iSCSI target's qualified name.
Target iSCSI Name	iSCSI qualified name of the target.
Login Accepts	Number of accepted login requests.
Login Failed	Number of rejected login requests.
Redirect	Number of logins that are redirected.
Authorize	Number of logins that failed due to authorization failure.
Authenticate	Number of login failures due to failed authentication.
Negotiate	Number of login failures due to errors in iSCSI parameter negotiation.
Other	Number of login failures due to other causes.
Logout Normal	Number of logouts requesting session closing.
Logout Other	Number of other logouts.
Initiator	ID of initiator logged into target.
TSIH	iSCSI session handle.
ISID	iSCSI session ID.
Command PDUs	Number of SCSI command protocol data units (PDUs) received by target.
Read Command PDUs	Number of read command PDUs received by target.
Write Command PDUs	Number of write command PDUs received by target.
Response PDUs	Number of iSCSI response PDUs sent by target.
Data-In PDUs	Number of data-in PDUs.
Data-Out PDUs	Number of data-out PDUs.
R2Ts	Number of request to transfers issued during the session.
Tx Data Octets	Amount of transmitted data (in bytes).
Rx Data Octets	Amount of received data (in bytes).
Digest Errors	Number or errors in data or header digests or both.
Connection Timeout Errors	Number of errors caused by the connection between target and initiator timing out.

# **EXAMPLE #24** To unbind the network portal **172.24.102.243:3260** on a target **t1**, type:

\$ server\_iscsi server\_2 -target -unbind t1
np=172.24.102.243:3260

server\_2 : done

# **EXAMPLE #25**

To display the status of the iSCSI service, iSCSI session statistics and statistics about login/logout attempts, type:

```
$ server_iscsi server_2 -service -status
```

```
server_2 :
**** ISCSI Service ****
Status: Running
Iscsi is configured to listen on ports:
TCP Port: 3260
Header Digests are preferred
Data Digests are not preferred
[Session Stats]
Session Failures: 0
Last Session Failed: N/A
[Login/Logout Stats]
Login Accepted: 0
Login Failed: 0
  Redirect: 0
  Authorize: 0
  Authenticate: 0
  Negotiate: 0
  Other: 0
Logout Normal: 0
Logout Other: 0
Last Login Failed: N/A
```

## Where:

Value	Definition
Status	Status of the iSCSI service on the Data Mover.
TCP Port	Listening port used by the iSCSI service.
Header Digests	Whether header digests are enabled or disabled.
Data Digests	Whether data digests are enabled or disabled.
Session Failures	Number of session failures.
Last Session Failed	ID of the last failed session.
Login Accepts	Number of accepted login requests.
Login Failed	Number of rejected login requests.
Redirect	Number of logins that are redirected.
Authorize	Number of logins that failed due to authorization failure.
Authenticate	Number of login failures due to failed authentication.
Negotiate	Number of login failures due to errors in iSCSI parameter negotiation.
Other	Number of login failures due to other causes.
Logout Normal	Number of logouts requesting session closing.
Logout Other	Number of other logouts.
Last Failure Type	Type of last login failure.

Last Failure Time	Time of last login failure.
Last Failure Initiator	Initiator ID from which failed login came.
Last Failure IP	IP from which failed login request came.

**EXAMPLE #26** To stop iSCSI service, type:

\$ server\_iscsi server\_2 -service -stop

server\_2 : done

**EXAMPLE #27** To delete an iSCSI LUN 1 on the target **t1**, type:

\$ server\_iscsi server\_2 -lun -delete 1 -target t1

server\_2 : done

**EXAMPLE #28** To clear iSNS client service configuration, type:

\$ server\_iscsi server\_2 -ns isns -clear -server

server\_2 : done

When a **-clear** is issued, the system returns the message "done", however, settings are not cleared until iSNS service is restarted.

**EXAMPLE #29** To delete the target **t1**, type:

\$ server\_iscsi server\_2 -target -delete t1

server\_2 : done

**EXAMPLE #30** To change LUN **24** (a read-only replication destination LUN) on

target t1 to a read/write LUN, type:

\$ server\_iscsi server\_2 -lun -modify 24 -target t1
-readonly no

server 2 : done

**EXAMPLE #31** To change LUN 11 (a read/write LUN) on target t1 to a read-only

LUN, type:

\$ server\_iscsi server\_2 -lun -modify 11 -target t1

-readonly yes

server\_2 : done

**EXAMPLE #32** To list the snapshots of LUN 22 on target t2 on server\_2, type:

\$ server\_iscsi server\_2 -snap -list -target t2 -lun 22

server\_2:
Snap Name
Lun Target Create Time
Num.

fs14\_T11\_LUN22\_FCNBD054507021\_003C.ckpt006 24 t2 Thu Mar 06 14:33:09 EST 2008 fs14\_T11\_LUN22\_FCNBD054507021\_003C.ckpt005 24 t2 Thu Mar 06 14:29:22 EST 2008

## Where:

Value	Description	
Snap Name	Name of the snapshot.	
Lun Number	Number of the LUN used to create the snapshot.	
Target	Alias of the target to which the LUN belongs.	
Create Time	Date and time that the snapshot was created.	

# **EXAMPLE #33** To view information about the specified snapshot, type:

\$ server\_iscsi server\_2 -snap-info
fs14\_T11\_LUN18\_000172900116\_0004.ckpt008

server\_2:

Snap Name =  $fs14_T11_LUN18_000172900116_0004.ckpt008$ 

Lun Number = 18
Target = t2
Promoted Number =
Promoted Initiator =

Application Label = CelerraCS LUN Size MB = 10240

Create Time = Tue Mar 04 10:14:06 EST 2008

Application Data = windows\_app\_data

### Where:

Value	Description	
Snap Name	Name of the snapshot.	
Lun Number	Number of the LUN used to create the snapshot.	
Target	Alias of the target to which the LUN belongs.	
Create Time	Date and time that the snapshot was created.	
Application Data	Comment text or other data associated with the snapshot.	

# **EXAMPLE #34** To view information about the specified promoted snapshot, type:

\$ server\_iscsi server\_2 -snap -info
fs14\_T8\_LUN24\_000183500235\_0006.ckpt461

server\_2:

Snap Name =  $fs14_T8_LUN24_000183500235_0006.ckpt461$ 

Lun Number = 24Target = t2

Create Time = Wed Mar 05 11:30:52 EST 2008

Application Data = windows\_app\_data

Promoted Number = 132

Promoted Initiator = iqn.1991-05.com.microsoft:winserver1.nasdoc.emc.com

Application Label = CelerraCS

LUN size MB = 256

## Where:

Value	Description
Promoted Number	LUN number of the promoted snapshot.
Promoted Initiator	iSCSI initiator granted access to the promoted LUN.
Application Label	Label identifying the creator of the snapshot.
LUN size MB	Size of the promoted LUN in megabytes.

# **EXAMPLE #35**

To create a snapshot of LUN 24 on target **t1** on **server\_2** and attach the specified comment text, type:

\$ server\_iscsi server\_2 -snap -create -target t1 -lun 24
-data "Q3 analysis snap Target t1 LUN 24"

server 2 :

Info 26306936878: OK.id=fs23\_t1\_LUN24\_HK190807300021\_0000.ckpt000

### **EXAMPLE #36**

To modify the comment text associated with the specified snapshot on **server\_2**, type:

\$ server\_iscsi server\_2 -snap -modify
fs14\_T8\_LUN24\_000183500235\_0006.ckpt461 -data "Q4
analysis snap Target t1 LUN 24"

server\_2 :
OK

# **EXAMPLE #37**

To delete the specified snapshot on **server\_2**, type:

\$ server\_iscsi server\_2 -snap -delete
fs14 T8 LUN24\_000183500235\_0006.ckpt461

server\_2 :
OK

# **EXAMPLE #38**

To delete all snapshots of LUN 24 on target t1 on server\_2, type:

\$ server iscsi server 2 -snap -delete -target t1 -lun 24

server\_2 :

## **EXAMPLE #39**

To restore a production LUN using the specified snapshot on **server\_2**, type:

\$ server\_iscsi server\_2 -snap -restore
fs14 T8 LUN24 000183500235 0006.ckpt461

server\_2 :
OK

# **EXAMPLE #40**

To promote the specified snapshot and grant access to the specified iSCSI initiator, type:

\$ server\_iscsi server\_2 -snap -promote
fs14\_T8\_LUN24\_000183500235\_0006.ckpt461 -initiator
iqn.1991-05.com.microsoft:winserver1.nasdocs.emc.com

server\_2 :
OK

# **EXAMPLE #41**

To demote the specified snapshot regardless of the application that created it, type:

\$ server\_iscsi server\_2 -snap -Demote
fs14\_T8\_LUN24\_000183500235\_0006.ckpt461 -Force

server\_2 :
OK

# server\_kerberos

Manages the Kerberos configuration within the specified Data Movers.

## **SYNOPSIS**

```
server_kerberos {<movername> | ALL}
  -add realm=<realm_name>, kdc=<fqdn_kdc_name>[:<port>]
  [,kdc=<fqdn_kdc_name>[:<port>]...]
  [,kpasswd=<fqdn_kpasswd_server_name>]
  [,kadmin=<kadmin_server>]
  [,domain=<domain_name>][,defaultrealm]
| -delete realm=<realm_name>
| -keytab
| -ccache [-flush]
| -list
| -kadmin [<kadmin_options>]
```

# **DESCRIPTION**

**server\_kerberos** adds, deletes, lists the realms within the Kerberos configuration of a Data Mover, and manages the Data Movers service principals and keys.

**server\_kerberos** displays the key table content, and specifies a **kadmin** server.

### **OPTIONS**

-add realm=<realm\_name>, kdc=<fqdn\_kdc\_name>

Adds the specified realm to the Kerberos configuration on the specified Data Mover. The <realm\_name> is the fully qualified domain name of the Kerberos realm to be added to the key distribution center (KDC) configuration. The <fqdn\_kdc\_name> is the fully qualified domain name of the KDC for the specified realm.

**Note:** The **-add** option is only relevant if you are using a UNIX/Linux Kerberos KDC.

```
[:<port>]
Specifies a port that the KDC listens on.

[,kdc=<fqdn_kdc_name[:<port>]...]
Specifies additional KDCs with ports that KDCs listen on.

[,kpasswd=<fqdn_kpasswd_server_name>]
Specifies a password server for the KDC. The
<fqdn_kpasswd_server_name> must be a fully qualified domain name for the server.
```

### [, kadmin=<kadmin\_server>]

Specifies the kadmin server.

#### [,domain=<domain\_name>]

The <domain\_name> is the full name of the DNS domain for the realm.

### [,defaultrealm]

Indicates that the default realm is to be used.

#### -delete realm=<realm\_name>

Deletes the specified realm from the Kerberos configuration for the specified Data Mover.

**Note:** The **-delete** option is only relevant if you are using a UNIX/Linux Kerberos KDC.

## -keytab

Displays the principal names for the keys stored in the keytab file.

#### -ccache

Displays the entries in the Data Movers Kerberos credential cache.

**Note:** The **-ccache** option can also be used to provide EMC Customer Support with information for troubleshooting user access problems.

### [-flush]

Flushes the Kerberos credential cache removing all entries. Credential cache entries are automatically flushed when they expire or during a Data Mover reboot.

Once the cache is flushed, Kerberos obtains new credentials when needed. The repopulation of credentials may take place immediately, over several hours, or be put off indefinitely if no Kerberos activity occurs.

#### -list

Displays a listing of all configured realms on a specified Data Mover or on all Data Movers.

#### -kadmin [<kadmin\_options>]

Invokes the kadmin tool with the following specified options:

```
[-r <realm>]
```

Specifies a realm as the default database realm.

```
[-p <pri>principal>]
```

Specifies the principal for authentication. Otherwise, **kadmin** will append "/admin" to the primary principal name of the default cache, the value of the USER environment variable, or the username as obtained with **getpwuid**, in order of preference.

```
[-q <query>]
```

Runs **kadmin** in non-interactive mode. This passes the query directly to kadmin, which performs the query, then exits.

```
[-w <password>]
```

Uses a specified password instead of prompting for a password.

```
[-s <admin_server> [:<port>]]
```

Specifies the **kadmin** server with its associated port.

**Note:** The **kadmin** tool is only relevant if you are using a UNIX/Linux Kerberos KDC. You must be root to execute the **-kadmin** option.

## **SEE ALSO**

Configuring NFS on Celerra, server\_checkup, and server\_nfs.

# OUTPUT

Dates appearing in output are in UTC format.

# **EXAMPLE #1**

To add a realm to the Kerberos configuration of a Data Mover, type:

```
$ server_kerberos server_2 -add
realm=nasdocs.emc.com, kdc=winserver1.nasdocs.emc.com, dom
ain=nasdocs.emc.com
```

server\_2 : done

# **EXAMPLE #2**

To list the keytabs, type:

```
$ server_kerberos server_2 -keytab
server_2:
Dumping keytab file

keytab file major version = 0, minor version 0

-- Entry number 1 --
principal: DM102-CGE0$@NASDOCS.EMC.COM
realm: NASDOCS.EMC.COM
encryption type: rc4-hmac-md5
principal type 1, key version: 332
    key length: 16, key: blc199a6ac11cd529df172e270326d5e
    key flags:(0x0), Dynamic Key, Not Cached
    key cache hits: 0
```

```
-- Entry number 2 --
```

principal: DM102-CGE0\$@NASDOCS.EMC.COM

```
realm: NASDOCS.EMC.COM
encryption type: des-cbc-md5
principal type 1, key version: 332
     key length: 8, key: ced9a23183619267
     key flags: (0x0), Dynamic Key, Not Cached
     key cache hits: 0
-- Entry number 3 --
principal: DM102-CGE0$@NASDOCS.EMC.COM
realm: NASDOCS.EMC.COM
encryption type: des-cbc-crc
principal type 1, key version: 332
     key length: 8, key: ced9a23183619267
     key flags: (0x0), Dynamic Key, Not Cached
     key cache hits: 0
-- Entry number 4 --
principal: host/dm102-cge0@NASDOCS.EMC.COM
realm: NASDOCS.EMC.COM
encryption type: rc4-hmac-md5
principal type 1, key version: 332
     key length: 16, key: b1c199a6ac11cd529df172e270326d5e
     key flags:(0x0), Dynamic Key, Not Cached
     key cache hits: 0
<... removed ...>
-- Entry number 30 --
principal: cifs/dm102-cge0.nasdocs.emc.com@NASDOCS.EMC.COM
realm: NASDOCS.EMC.COM
encryption type: des-cbc-crc
principal type 1, key version: 333
     key length: 8, key: d95e1940b910ec61
     key flags: (0x0), Dynamic Key, Not Cached
     key cache hits: 0
End of keytab entries. 30 entries found.
```

#### Where:

Value	Definition
principal type	Type of the principal as defined in the GSS-API. Reference to RFC 2743.
key version	Every time a key is regenerated its version changes.

# **EXAMPLE #3** To list all the realms on a Data Mover, type:

# \$ server\_kerberos server\_2 -list

server\_2 :

Kerberos common attributes section:

Supported TGS encryption types: rc4-hmac-md5 des-cbc-md5 des-cbc-crc Supported TKT encryption types: rc4-hmac-md5 des-cbc-md5 des-cbc-crc

Use DNS locator: yes

End of Kerberos common attributes.

Kerberos realm configuration:

realm name: NASDOCS.EMC.COM

kdc: winserver1.nasdocs.emc.com admin server: winserver1.nasdocs.emc.com kpasswd server: winserver1.nasdocs.emc.com

default domain: nasdocs.emc.com

End of Kerberos realm configuration.

Kerberos domain\_realm section:

DNS domain = Kerberos realm

.nasdocs.emc.com = NASDOCS.EMC.COM

End of Krb5.conf domain realm section.

# **EXAMPLE #4** To specify a kadmin server, type:

\$ server\_kerberos server\_2 -add

realm=eng.nasdocs.emc.com,kdc=winserver1.nasdocs.emc.com

,kadmin=172.24.102.67

server\_2 : done

**Note:** You must be root to execute the -kadmin option. Replace \$ with # as the root login is a requirement.

# **EXAMPLE #5** To delete a realm on a Data Mover, type:

\$ server\_kerberos server\_2 -delete

realm=eng.nasdocs.emc.com

server\_2 : done

# **EXAMPLE #6** To display the credential cache on a Data Mover, type:

\$ server\_kerberos server\_2 -ccache

server\_2 :

Dumping credential cache

Names:

Client: DM102-CGE0\$@NASDOCS.EMC.COM

Service: WINSERVER1.NASDOCS.EMC.COM

Target: HOST/WINSERVER1.NASDOCS.EMC.COM@NASDOCS.EMC.COM

Times:

Auth: 09/12/2005 07:15:04 GMT Start: 09/12/2005 07:15:04 GMT End: 09/12/2005 17:15:04 GMT

Flags: PRE\_AUTH, OK\_AS\_DELEGATE

Encryption Types:

Key: rc4-hmac-md5
Ticket: rc4-hmac-md5

Names:

Client: DM102-CGE0\$@NASDOCS.EMC.COM
Service: winserver1.nasdocs.emc.com

Target: ldap/winserver1.nasdocs.emc.com@NASDOCS.EMC.COM

Times:

Auth: 09/12/2005 07:15:04 GMT Start: 09/12/2005 07:15:04 GMT End: 09/12/2005 17:15:04 GMT Flags: PRE AUTH,OK AS DELEGATE

Encryption Types:

Key: rc4-hmac-md5
Ticket: rc4-hmac-md5

Names:

Client: DM102-CGE0\$@NASDOCS.EMC.COM

Service: NASDOCS.EMC.COM

Target: krbtgt/NASDOCS.EMC.COM@NASDOCS.EMC.COM

Times:

Auth: 09/12/2005 07:15:04 GMT Start: 09/12/2005 07:15:04 GMT End: 09/12/2005 17:15:04 GMT

Flags: INITIAL, PRE\_AUTH

Encryption Types:

Key: rc4-hmac-md5
Ticket: rc4-hmac-md5

End of credential cache entries.

#### Where:

Value	Definition	
client	Client name and its realm.	
service	Domain controller and its realm.	
target	Target name and its realm.	
auth	Time of the initial authentication for the named principal.	
start	Time after which the ticket is valid.	
end	Time after which the ticket will not be honored (its expiration time).	

Value	Definition
flags	Options used or requested when the ticket was issued.
key	Key encryption type.
ticket	Ticket encryption type.

# **EXAMPLE #7** To flush the credential cache on a Data Mover, type:

\$ server\_kerberos server\_2 -ccache flush

server\_2 :
Purging credential cache.
Credential cache flushed.

# server\_ldap

Manages the LDAP-based directory client configuration and LDAP over SSL for the specified Data Movers.

#### **SYNOPSIS**

```
server_ldap {<movername> | ALL}
  {-set|-add} [-p] -domain <FODN>|-basedn
  <attribute_name>=<attribute_value>[,...]
   [-servers {<IPv4_addr>[:<port>]|<IPv6_addr>|<\[IPv6_addr\]:port>}[,...]]
   [-profile <profile>]
   [-nisdomain <NIS_domain>]
   [-binddn <bind_DN> | {-kerberos -kaccount <account_name> [-realm
  <realm_name>] } ]
   [-sslenabled {y | n}]
   [-sslpersona {none | <persona_name>}]
   [-sslcipher {default | <cipher_list>}]
  -clear
  -info [-verbose]
  -service {-start|-stop|-status}
  -lookup {-user <username>
      -group <groupname>
      -uid <uid>
      -gid <qid>
      -hostbyname <hostname>
      -netgroup <qroupname> }
```

#### **DESCRIPTION**

**server\_ldap** configures, starts, stops, deletes, and displays the status of the LDAP-based directory client configuration, and queries the LDAP-based directory server.

## **OPTIONS**

```
-set [-p] -domain <FQDN>|-basedn
<attribute_name>=<attribute_value>[,...]
```

Specifies the LDAP-based directory client domain for the specified Data Mover and starts the service. The **-p** option requests a prompt for the password. A password is required in conjunction with a bind distinguished name in order to specify the use of simple authentication. The **-basedn** option specifies the Distinguished Name (DN) of the directory base, an x509 formatted name that uniquely identifies the directory base. For example: ou=abc,o=def,c=ghi. If a base distinguished name contains space characters, enclose the entire string within double quotation marks and enclose the name with a backslash and double quotation mark. For example, "\"cn=abc,cn=def ghi,dc=com\"".

server\_ldap

It is recommended configuring an LDAP-based directory client by using the **-basedn** option instead of the **-domain** option. The DN provides the root position for:

- Searching for iPlanet profiles
- Defining default search containers for users, groups, hosts, and netgroups according to RFC 2307. An iPlanet profile and OpenLDAP or Active Directory with SFU or IdMU ldap.conf file are only required for customized setups.

**Note:** In the case in which the DN of the directory base contains dots and the client is configured using the domain name, the default containers may not be set up correctly. For example, if the name is dc=my.company,dc=com and it is specified as domain name my.company.com, Celerra incorrectly defines the default containers as dc=my,dc=company,dc=com.

```
[-servers {<IPv4_addr>[:<port>] | <IPv6_addr> |
<\[IPv6_addr\] : port> } [, . . . ]]
```

Specifies the IP addresses of the LDAP-based directory client servers. <IPv4\_addr> or <IPv6\_addr> indicates the IP address of the LDAP-based directory servers. IPv6 addresses need to be enclosed in square brackets if a port is specified; the brackets do not signify optional content. The <port> option specifies the LDAP-based directory server TCP port number. If the port is not specified, the default port is 389 for LDAP and 636 for SSL-based LDAP. It is recommended that at least two LDAP servers are defined, so that can switch to the second server in case the first cannot be reached.

**Note:** IP addresses of the LDAP-based directory servers do not have to be included every time with the **server\_ldap** command once you have indicated the configuration server, and if configuring the same LDAP-based directory service.

```
[-profile cprofile>]
```

Specifies the profile name or the profile distinguished name which provides the iPlanet client with configuration information about the directory service. For example, both the following values are allowed: -profile celerra\_profile and -profile cn=celerra\_profile,ou=admin,dc=mycompany,dc=com.

**Note:** It is recommended that unique profile names be used in the Directory Information Tree (DIT). The specified profile is searched for by scanning the entire tree and if it is present in multiple locations, the first available profile is used unless the profile distinguished name is specified.

### [-nisdomain <NIS domain>]

Specifies the NIS domain of which the Data Mover is a member since an LDAP-based directory domain can host more than one NIS domain.

```
[-binddn <bind_DN> | {-kerberos -kaccount
<account_name> [-realm <realm_name>]}]
```

Specifies the distinguished name (DN) or Kerberos account of the identity used to bind to the service. Active Directory with SFU or IdMU requires an authentication method that uses simple authentication, SSL, or Kerberos.

Simple authentication requires that a DN be specified along with a password. For SSL-based client authentication to succeed, the Data Mover certificate Subject must match the distinguished name for an existing user (account) at the directory server.

**Note:** To configure a LDAP-based directory service for authentication, **-binddn** is not required if the **-sslpersona** option is specified. In this case, SSL-based client authentication will be used.

The Kerberos account name must be the CIFS server computer name known by the KDC. The account name must terminate with a \$ symbol.

By default, the Data Mover assumes that the realm is the same as the LDAP domain provided in the **-domain** or **-basedn** options. But a different realm name can be specified if necessary.

```
[-sslenabled \{y | n\}]
```

Enables (y) or disables (n) SSL. SSL is disabled by default.

```
[-sslpersona {none | <persona_name>}]
```

Specifies the key and certificate of the directory server. If a persona has been previously configured, **none** disables the user of a client key and certificate. The **-sslpersona** option without the **-binddn** option indicates that the user wants to authenticate using the client (persona) certificate. To authenticate using the client certificate, the LDAP server must be configured to **always** 

**request** (or require) the persona certificate during the SSL transaction, or the authentication will fail. If authentication using the client certificate is not desired, then the **-binddn** option must be used. The configuration rules are explained in Table 1 on page 558.

**Note:** The **-sslpersona** option does not automatically enable SSL, but configures the specified value. The value remains persistent and is used whenever SSL is enabled.

# Table 1 Configuration rules

Description	Data Mover configuration
SSL enabled on Data Mover, LDAP server should accept SSL, anonymous authentication is used.	server_ldap -sslenabled y
SSL enabled, password-based authentication is used.	server_ldap -p -binddn cn=foo -sslenabled y
SSL enabled , SSL certificate authentication is used, LDAP server should be configured to request client certificate.	server_ldap -sslenabled y -sslpersona default (use server_certificate to verify that the certificate for the Data Mover's default persona exists)

**Note:** The user should refer to the LDAP server documentation for information about configuring the server to request the client certificate.

[-sslcipher {default|<cipher\_list>}] Specifies default or the cipher list.

**Note:** The **-sslcipher** option does not automatically enable SSL, but configures the specified value. The value remains persistent and is used whenever SSL is enabled.

#### -clear

Deletes the LDAP-based directory client configuration for the specified Data Mover and stops the service.

#### -info

Displays the service status and the static and dynamic configuration.

# [-verbose]

Adds troubleshooting information to the output.

# -service {-start|-stop|-status}

The **-start** option enables the LDAP-based directory client service.

The LDAP-based directory client service is also restarted when the Celerra Network Service is rebooted. The **-stop** option disables the LDAP-based directory client service, and the **-status** option displays the status of the LDAP-based directory service.

#### -lookup

{user=<username>|group=<groupname>|uid=<uid>|gid=<gid>|hostbyname=<hostname>|netgroup=<groupname>} Provides lookup information about the specified resource for troubleshooting purposes.

## SEE ALSO

Configuring Celerra Naming Services.

# **EXAMPLE #1**

To configure the use of an LDAP-based directory by a Data Mover, type:

\$ server\_1dap server\_4 -set -domain nasdocs.emc.com
-servers 172.24.102.62

server\_4 : done

## **EXAMPLE #2**

To configure the use of an LDAP-based directory by a Data Mover using the Distinguished Name of the server at IPv4 address 172.24.102.62 with the default port, type:

\$ server\_ldap server\_2 -set -basedn
dc=nasdocs,dc=emc,dc=com -servers 172.24.102.62

server 2 : done

## **EXAMPLE #3**

To configure the use of an LDAP-based directory by a Data Mover using the Distinguished Name of the server at IPv6 address 2002:c8c::24:172:63 with the default port, type:

\$ server\_ldap server\_2 -set -basedn
dc=nasdocs,dc=emc,dc=com -servers 2002:c8c::24:172:63

server\_2 : done

### **EXAMPLE #4**

To configure the use of an LDAP-based directory by a Data Mover and specify the use of the client profile **celerra\_profile**, type:

\$ server\_ldap server\_4 -set -domain nasdocs.emc.com
-servers 172.24.102.62 -profile celerra\_profile
-nisdomain nasdocs -sslenabled y

server\_4 : done

## **EXAMPLE #5**

To configure the use of an LDAP-based directory by a Data Mover and specify the use of the client profile using its distinguished name, type:

```
$ server_ldap server_4 -set -domain nasdocs.emc.com
                         -servers 172.24.102.62 -profile
                         cn=celerra_profile,dc=nasdocs,dc=emc,dc=com -nisdomain
                         nasdocs -sslenabled y
server_4 : done
          EXAMPLE #6
                         To specify the NIS domain to which the Data Mover is a member,
                         type:
                         $ server_ldap server_2 -set -domain nasdocs.emc.com
                         -servers 172.24.102.62 -nisdomain nasdocs
server_2 : done
          EXAMPLE #7
                         To configure the use of simple authentication by specifying a bind
                         Distinguished Name (DN) and password, type:
                         $ server_ldap server_2 -set -p -domain nasdocs.emc.com
                         -servers 172.24.102.10 -binddn
                         "cn=admin, cn=users, dc=nasdocs, dc=emc"
server_2 : Enter Password: ******
done
          EXAMPLE #8
                         To configure the use of an LDAP-based directory by a Data Mover
                         using SSL, type:
                         $ server_ldap server_4 -set -basedn
                         dc=nasdocs,dc=emc,dc=com -servers 172.24.102.62
                         -sslenabled y
server_4 : done
          EXAMPLE #9
                         To configure the use of an LDAP-based directory by a Data Mover
                         using SSL and user key and certificate, type:
                         $ server ldap server 4 -set -basedn
                         dc=nasdocs,dc=emc,dc=com -servers 172.24.102.62
                         -sslenabled y -sslpersona default
server_4 : done
        EXAMPLE #10
                         To configure the use of an LDAP-based directory by a Data Mover
                         using SSL and using specified ciphers, type:
                         $ server_ldap server_4 -set -basedn
                         dc=nasdocs,dc=emc,dc=com -servers 172.24.102.62
                         -sslenabled y -sslcipher "RC4-MD5, RC4-SHA"
server_4 : done
        EXAMPLE #11
                         To display information about the LDAP-based directory configuration
                         on a Data Mover, type:
                         $ server_ldap server_4 -info
server_4 :
LDAP domain: nasdocs.emc.com
```

```
base DN: dc=nasdocs,dc=emc,dc=com
     State: Configured - Connected
NIS domain: nasdocs.emc.com
     No client profile nor config. file provided (using default setup)
Connected to LDAP server address: 172.24.102.62 - port 636
SSL enabled/disabled by Command line, cipher suites configured by Command line
        EXAMPLE #12
                        To configure the use of Kerberos authentication by specifying a
                        Kerberos account, type:
                        $ server_ldap server_2 -set -basedn
                        dc=nasdocs,dc=emc,dc=com -servers 172.24.102.62 -kerberos
                        -kaccount cifs_compname$
server_2 : done
        EXAMPLE #13
                        To display detailed information about the LDAP-based directory
                        configuration on a Data Mover, type:
                        $ server_ldap server_2 -info -verbose
server_ldap server_2 -i -v
server_2 :
LDAP domain: devldapdom1.lcsc
     base DN: dc=devldapdom1,dc=lcsc
     State: Configured - Connected
NIS domain: devldapdom1.lcsc
     Configuration file - TTL: 1200 seconds
     Next configuration update in 1107 seconds
LDAP configuration servers:
     Server 10.64.220.148 port 389 : Active, connected
      SSL not enabled, Persona: none specified, Cipher Suite List: none specified
     Server 10.64.220.114 port 389 : Spare , disconnected
      SSL not enabled, Persona: none specified, Cipher Suite List: none specified
Domain naming contexts:
     dc=devldapdom1,dc=lcsc
Domain supported LDAP controls:
     1.3.6.1.4.1.4203.1.9.1.1
     2.16.840.1.113730.3.4.18
     2.16.840.1.113730.3.4.2
     1.3.6.1.4.1.4203.1.10.1
     1.2.840.113556.1.4.319
     1.2.826.0.1.334810.2.3
     1.2.826.0.1.3344810.2.3
     1.3.6.1.1.13.2
     1.3.6.1.1.13.1
     1.3.6.1.1.12
Domain supported authentication mechanisms:
Default search base: dc=devldapdom1,dc=lcsc
Domain default search Scope: single-level
     passwd base DN:
          ou=people,dc=devldapdom1,dc=lcsc - search scope single-level
          passwd object class: posixAccount
```

```
passwd attributes: cn, uid, uidNumber, gidNumber, userPassword,
loginShell, gecos, description
     group base DN:
          ou=group,dc=devldapdom1,dc=lcsc - search scope single-level
          group object class: posixGroup
          group attributes: cn, gidNumber, userPassword, memberUid, description
     hosts base DN:
          ou=hosts,dc=devldapdom1,dc=lcsc - search scope single-level
          host object class: ipHost
          host attributes: cn, ipHostNumber, description
     netgroup base DN:
          ou=netgroup,dc=devldapdom1,dc=lcsc - search scope single-level
          netgroup object class: nisNetgroup
          host attributes: cn, nisNetgroupTriple, memberNisNetgroup description
        EXAMPLE #14
                        To display lookup information about the user nasadmin, type:
                        $ server_ldap server_4 -lookup -user nasadmin
server_4 :
user: nasadmin, uid: 1, gid: 201, gecos: nasadmin, home dir: /home/nasadmin,
shell: /bin/csh
        EXAMPLE #15
                        To display the status of the LDAP-based directory service, type:
                        $ server_ldap server_4 -service -status
server_4 :
LDAP service active
        EXAMPLE #16
                        To stop the LDAP-based directory service, type:
                        $ server_ldap server_4 -service -stop
server_4 : done
        EXAMPLE #17
                        To delete the LDAP configuration for the specified Data Mover and
                        stop the service, type:
                        $ server_ldap server_4 -clear
server_4 : done
```

# server\_log

Displays the log generated by the specified Data Mover.

# **SYNOPSIS**

```
server_log <movername>
  [-a][-f][-n][-s][-v|-t]
```

## **DESCRIPTION**

**server\_log** reads and displays the log generated by the Data Mover. Information in the log file is read from oldest to newest. To view that most recent log activity, add | tail to the end of your command line.

# **OPTIONS**

# No arguments

Displays the contents of the log added since the last reboot.

#### -a

Displays the complete log.

#### -f

Monitors the growth of the log by entering into an endless loop, pausing, reading the log being generated. The output is updated every second. To exit, press **Ctrl-C** together.

#### $-\mathbf{n}$

Displays the log without the time stamp.

#### -s

Displays the time in yyyy-mm-dd format when each command in the log was executed.

#### -v | -t

Displays the log files in verbose form or terse form.

## **EXAMPLE #1**

To monitor the growth of the current log, type:

#### \$ server\_log server\_2 -f

```
1200229390: UFS: 6: gid map file processing is completed.
1200229390: DPSVC: 6: DpRequest::done() BEGIN reqType:DpRequest_VersionInt_SchSr
cRefresh reqCaller:DpRequest_Caller_Scheduler status:0
1200229390: DPSVC: 6: SchedulerSrc=118_APM00062400708_0000_253_APM00062400708_00
00, curState=active, input=refreshDone
1200229390: DPSVC: 6: DpVersion::getTotalBlocksVolume enter
1200229390: DPSVC: 6: DpVersion::getTotalBlocksVolume found newV 118.ckpt003, bl
ocks 17534
1200229390: DPSVC: 6: DpVersion::getTotalBlocksVolume 0 blocks for vnumber 1038
totalB 0
1200229390: DPSVC: 6: DpVersion::getTotalBlocksVolume found oldV 118.ckpt004
1200229390: DPSVC: 6: DpVersion::getTotalBlocksVolume exit
1200229390: DPSVC: 6: DpVersion::getTotalBytes 0 blocks 0 bytes
1200229390: DPSVC: 6: SchedulerSrc=118_APM00062400708_0000_253_APM00062400708_00
00, newState=active
1200229390: SVFS: 6: D113118_736: After Merge err:4 full:0 mD:0
1200229390: SVFS: 6: D113118_736: prev !full release ch:82944 newPrev:99328
1200229390: SVFS: 6: D113118_737: Chunk:0 hdrAdd:50176 ==> prevChunk:82944 befor
e changePrevChunk
1200229390: SVFS: 6: D113118_737: Ch:0 hdr:50176 : prevCh:99328 after changePrev
1200229510: DPSVC: 6: refreshSnap: cur=1200229510, dl=1200229520, kbytes=0, setu
p=0, rate=1000
1200229510: DPSVC: 6: SchedulerSrc=199_APM00062400708_0000_258_APM00062400708_00
00, curState=active, input=refresh
1200229510: DPSVC: 6: DpRequest::execute() BEGIN reqType:DpRequest_VersionInt_Sc
hSrcRefresh reqCaller:DpRequest_Caller_Scheduler reqMode:0
1200229510: DPSVC: 6: DpRequest::execute() END regType:DpRequest_VersionInt_SchS
rcRefresh reqCaller:DpRequest_Caller_Scheduler status:0 reqMode:0
1200229510: DPSVC: 6: SchedulerSrc=199_APM00062400708_0000_258_APM00062400708_00
00, newState=active
--More--
```

# **EXAMPLE #2** To display the current log, type:

## \$ server\_log server\_2

```
2008-01-13 08:03:10: DPSVC: 6: DpRequest::done() BEGIN reqType:DpRequest_Version
Int_SchSrcRefresh reqCaller:DpRequest_Caller_Scheduler status:0
2008-01-13 08:03:10: DPSVC: 6: SchedulerSrc=118_APM00062400708_0000_253_APM00062
400708_0000, curState=active, input=refreshDone
2008-01-13 08:03:10: DPSVC: 6: DpVersion::getTotalBlocksVolume enter
2008-01-13 08:03:10: DPSVC: 6: DpVersion::getTotalBlocksVolume found newV 118.ck
pt003, blocks 17534
2008-01-13 08:03:10: DPSVC: 6: DpVersion::getTotalBlocksVolume 0 blocks for vnum
ber 1038 totalB 0
2008-01-13 08:03:10: DPSVC: 6: DpVersion::getTotalBlocksVolume found oldV 118.ck
pt004
2008-01-13 08:03:10: DPSVC: 6: DpVersion::getTotalBlocksVolume exit
2008-01-13 08:03:10: DPSVC: 6: DpVersion::getTotalBytes 0 blocks 0 bytes
2008-01-13 08:03:10: DPSVC: 6: SchedulerSrc=118_APM00062400708_0000_253_APM00062
400708_0000, newState=active
2008-01-13 08:03:10: SVFS: 6: D113118_736: After Merge err:4 full:0 mD:0
2008-01-13 08:03:10: SVFS: 6: D113118_736: prev !full release ch:82944 newPrev:9
9328
2008-01-13 08:03:10: SVFS: 6: D113118_737: Chunk:0 hdrAdd:50176 ==> prevChunk:82
944 before changePrevChunk
2008-01-13 08:03:10: SVFS: 6: D113118_737: Ch:0 hdr:50176 : prevCh:99328 after c
hangePrev
2008-01-13 08:05:10: DPSVC: 6: refreshSnap: cur=1200229510, dl=1200229520, kbyte
s=0, setup=0, rate=1000
2008-01-13 08:05:10: DPSVC: 6: SchedulerSrc=199_APM00062400708_0000_258_APM00062
400708_0000, curState=active, input=refresh
2008-01-13 08:05:10: DPSVC: 6: DpRequest::execute() BEGIN reqType:DpRequest_Vers
ionInt_SchSrcRefresh regCaller:DpRequest_Caller_Scheduler regMode:0
--More--
```

# **EXAMPLE #3** To display the log file without the time stamp, type:

```
DPSVC: 6: DpVersion::getTotalBlocksVolume enter
DPSVC: 6: DpVersion::getTotalBlocksVolume found newV 118.ckpt003, blocks 17534
DPSVC: 6: DpVersion::getTotalBlocksVolume 0 blocks for vnumber 1038 totalB 0
DPSVC: 6: DpVersion::getTotalBlocksVolume found oldV 118.ckpt004
DPSVC: 6: DpVersion::getTotalBlocksVolume exit
DPSVC: 6: DpVersion::getTotalBytes 0 blocks 0 bytes
DPSVC: 6: SchedulerSrc=118_APM00062400708_0000_253_APM00062400708_0000, newState
=active
SVFS: 6: D113118_736: After Merge err:4 full:0 mD:0
SVFS: 6: D113118_736: prev !full release ch:82944 newPrev:99328
SVFS: 6: D113118_737: Chunk:0 hdrAdd:50176 ==> prevChunk:82944 before changePrev
SVFS: 6: D113118_737: Ch:0 hdr:50176 : prevCh:99328 after changePrev
DPSVC: 6: refreshSnap: cur=1200229510, dl=1200229520, kbytes=0, setup=0, rate=10
DPSVC: 6: SchedulerSrc=199_APM00062400708_0000_258_APM00062400708_0000, curState
=active, input=refresh
DPSVC: 6: DpRequest::execute() BEGIN reqType:DpRequest_VersionInt_SchSrcRefresh
reqCaller:DpRequest_Caller_Scheduler reqMode:0
DPSVC: 6: DpRequest::execute() END reqType:DpRequest_VersionInt_SchSrcRefresh re
qCaller:DpRequest_Caller_Scheduler status:0 reqMode:0
DPSVC: 6: SchedulerSrc=199_APM00062400708_0000_258_APM00062400708_0000, newState
=active
VBB: 6: VBB session list empty
CFS: 6: fs 0x78 type = dhfs being unmounted. Waiting for quiesce ...
CFS: 6: fs 0x78 type = dhfs unmounted
--More--
```

# **EXAMPLE #4** To display all the current logs available, type:

## \$ server\_log server\_2 -a

```
NAS LOG for slot 2:
1200152690: SVFS: 6: D113118_606: prev !full release ch:82944 newPrev:99328
1200152690: SVFS: 6: D113118_607: Chunk:0 hdrAdd:50176 ==> prevChunk:82944 befor
e changePrevChunk
1200152690: SVFS: 6: D113118_607: Ch:0 hdr:50176 : prevCh:99328 after changePrev
1200152950: DPSVC: 6: refreshSnap: cur=1200152950, dl=1200152960, kbytes=0, setu
p=0, rate=666
1200152950: DPSVC: 6: SchedulerSrc=199 APM00062400708 0000 258 APM00062400708 00
00, curState=active, input=refresh
1200152950: DPSVC: 6: DpRequest::execute() BEGIN reqType:DpRequest_VersionInt_Sc
hSrcRefresh reqCaller:DpRequest_Caller_Scheduler reqMode:0
1200152950: DPSVC: 6: DpRequest::execute() END regType:DpRequest_VersionInt_SchS
rcRefresh reqCaller:DpRequest_Caller_Scheduler status:0 reqMode:0
1200152950: DPSVC: 6: SchedulerSrc=199_APM00062400708_0000_258_APM00062400708_00
00, newState=active
1200152950: VBB: 6: VBB session list empty
1200152950: CFS: 6: fs 0x78 type = dhfs being unmounted. Waiting for quiesce ...
```

```
1200152950: CFS: 6: fs 0x78 type = dhfs unmounted
1200152950: SVFS: 6: pause() requested on fsid:78
1200152950: SVFS: 6: pause done on fsid:78
1200152950: SVFS: 6: Cascaded Delete...
1200152950: SVFS: 6: D120199_1131: createBlockMap PBM root=0 keys=0 h=0 nc=0
1200152950: VRPL: 6: 217: Allocating chunk: 4 Add: 66560 Chunks: 15
1200152950: SVFS: 6: Merge Start FsVol:199 event:0x0
1200152950: SVFS: 6: D120199_1130: hdr:99328 currInd:6, Destpmdv:D119199_1124
1200152950: CFS: 6: Resuming fs 78
1200152950: SVFS: 6: 199:D120199_1130:Merge hdr=99328 prev=82944 id=120 chunk=0
stableEntry=7
1200152950: UFS: 6: Volume name:Sh217120
1200152950: UFS: 6: starting gid map file processing.
1200152950: SVFS: 6: D120199_1130: After Merge err:4 full:0 mD:0
1200152950: SVFS: 6: D120199_1130: prev !full release ch:99328 newPrev:82944
1200152950: SVFS: 6: D120199_1131: Chunk:0 hdrAdd:66560 ==> prevChunk:99328 befo
re changePrevChunk
1200152950: SVFS: 6: D120199_1131: Ch:0 hdr:66560 : prevCh:82944 after changePre
1200152950: UFS: 6: gid map file processing is completed.
1200152950: DPSVC: 6: DpRequest::done() BEGIN reqType:DpRequest_VersionInt_SchSr
cRefresh regCaller:DpRequest_Caller_Scheduler status:0
1200152950: DPSVC: 6: SchedulerSrc=199_APM00062400708_0000_258_APM00062400708_00
00, curState=active, input=refreshDone
--More--
```

# **EXAMPLE #5** To display the current log in terse form, type:

## \$ server\_log server\_2 -t

```
NAS LOG for slot 2:
0 keys=0 h=0 nc=0
1200229390: 26043285504: 122: Allocating chunk: 3 Add: 50176 Chunks: 24
1200229390: 26042826752: Merge Start FsVol:118 event:0x0
1200229390: 26042826752: D113118_736: hdr:82944 currInd:6, Destpmdv:D114118_503
1200229390: 26040008704: Resuming fs 24
1200229390: 26042826752: 118:D113118_736:Merge hdr=82944 prev=99328 id=113 chunk
=0 stableEntry=7
1200229390: 26042433536: Volume name: Sh122113
1200229390: 26042433536: starting gid map file processing.
1200229390: 26042433536: gid map file processing is completed.
1200229390: 26045513728: DpRequest::done() BEGIN reqType:DpRequest_VersionInt_Sc
hSrcRefresh regCaller:DpRequest_Caller_Scheduler status:0
1200229390: 26045513728: SchedulerSrc=118_APM00062400708_0000_253_APM00062400708
_0000, curState=active, input=refreshDone
1200229390: 26045513728: DpVersion::getTotalBlocksVolume enter
1200229390: 26045513728: DpVersion::getTotalBlocksVolume found newV 118.ckpt003,
blocks 17534
```

```
1200229390: 26045513728: DpVersion::getTotalBlocksVolume 0 blocks for vnumber 10
38 totalB 0
1200229390: 26045513728: DpVersion::getTotalBlocksVolume found oldV 118.ckpt004
1200229390: 26045513728: DpVersion::getTotalBlocksVolume exit
1200229390: 26045513728: DpVersion::getTotalBytes 0 blocks 0 bytes
1200229390: 26045513728: SchedulerSrc=118_APM00062400708_0000_253_APM00062400708
_0000, newState=active
1200229390: 26042826752: D113118_736: After Merge err:4 full:0 mD:0
1200229390: 26042826752: D113118_736: prev !full release ch:82944 newPrev:99328
1200229390: 26042826752: D113118_737: Chunk:0 hdrAdd:50176 ==> prevChunk:82944 b
efore changePrevChunk
1200229390: 26042826752: D113118_737: Ch:0 hdr:50176 : prevCh:99328 after change
1200229510: 26045513728: refreshSnap: cur=1200229510, dl=1200229520, kbytes=0, s
etup=0, rate=1000
1200229510: 26045513728: SchedulerSrc=199_APM00062400708_0000_258_APM00062400708
_0000, curState=active, input=refresh
1200229510: 26045513728: DpRequest::execute() BEGIN reqType:DpRequest_VersionInt
_SchSrcRefresh reqCaller:DpRequest_Caller_Scheduler reqMode:0
1200229510: 26045513728: DpRequest::execute() END reqType:DpRequest_VersionInt_S
--More--
```

#### **EXAMPLE #6** To display the current log in verbose form, type:

```
$ server_log server_2 -v
```

Work Partition Layout found @ LBA 0x43000 (134MB boundary)

```
slot 2) About to dump log @ LBA 0xc7800
NAS LOG for slot 2:
About to print log from LBA c8825 to c97ff
0 keys=0 h=0 nc=0
logged time = 2008-01-13 \ 08:03:10
id
                  = 26043285504
                  = INFO
severity
component
                  =
                  = VRPL
facility
                  = 0
baseid
type
                  = STATUS
argument name = arg0 = 122: Allocating chunk: 3 Add: 50176 Chunks: 24
```

= string (8) argument type

brief description = 122: Allocating chunk: 3 Add: 50176 Chunks: 24

full description = No additional information is available.

recommended action = No recommended action is available. Use the text from the error message's brief description to search the Knowledgebase on Powerlink. Afte r logging in to Powerlink, go to Support > Knowledgebase Search > Support Soluti ons Search.

logged time = 2008-01-13 08:03:10

id = 26042826752

severity = INFO component =

argument value = Merge Start FsVol:118 event:0x0

argument type = string (8)

brief description = Merge Start FsVol:118 event:0x0

full description = No additional information is available.

recommended action = No recommended action is available. Use the text from the error message's brief description to search the Knowledgebase on Powerlink. Afte r logging in to Powerlink, go to Support > Knowledgebase Search > Support Soluti ons Search.

--More--

**Note:** This is a partial listing due to the length of the output.

# server\_mount

Mounts filesystems and manages mount options for the specified Data Movers.

## **SYNOPSIS**

```
server_mount {<movername> | ALL}
  [-all]
[-Force][-option < options>] < fs_name> [< mount_point>]
<options>:
[ro | rw] [primary=<movername>]
[ceppcifs]
[ceppnfs]
[nonotify] [nooplock]
[notifyonaccess] [notifyonwrite]
[accesspolicy={NT|UNIX|SECURE|NATIVE|MIXED|MIXED_COMPAT}]
[nolock|wlock|rwlock]
[cvfsname=<newname>]
[noscan]
[noprefetch]
[uncached]
[cifssyncwrite]
[triggerlevel=<value>]
[ntcredential]
[renamepolicy={CIFS|FULL|NO}]
[cifsnanoroundup]
[nfsv4delegation={NONE|READ|RW}]
```

## **DESCRIPTION**

**server\_mount** attaches a filesystem to the specified <mount\_point> with the specified options, and displays a listing of mounted filesystems. **server\_umount** unmounts the filesystem.

The **ALL** option executes the command for all Data Movers.

Note: The primary=<movername> option is not used.

# **OPTIONS** No arguments

Displays a listing of all mounted, and temporarily unmounted filesystems.

### -all

Mounts all filesystems in the mount table.

```
-Force -option rw <fs_name> <mount_point>
```

Forces a mount of a filesystem copy (created using fs\_timefinder) as read-write. By default, all filesystem copies are mounted as read-only.

<fs\_name> <mount\_point>

Mounts a filesystem to the specified <mount\_point>. When a filesystem is initially mounted, the <mount\_point> is required; however, remounting a filesystem after a temporary unmount does not require the use of a <mount\_point>.

[-option <options>]

Specifies the following comma-separated options:

[ro|rw]

Specifies the mount as read-write (default), or read-only which is the default for checkpoints and TimeFinder/FS.

**Note:** MPFS clients do not acknowledge filesystems that are mounted read-only and allow their clients to write to the filesystem.

[ceppcifs]

It enables the CEPA events for CIFS on a filesystem. This option is enabled by default.

[ceppnfs]

It enables the CEPA events for NFS on a filesystem.

**Note:** If ceppnfs is used without the ceppcifs option, the CEPA events for CIFS are disabled. To enable CEPA events for NFS and CIFS on a filesystem, make sure that you add both these options in the command.

Indicates the access control policy as defined in the table.

**Note:** When accessed from a Windows client, ACLs are only checked if the CIFS user authentication method is set to the recommended default, NT. This is set using the **-add** security option in the **server\_cifs** command.

Access policy	CIFS clients	NFS clients
NATIVE (default)	ACL is checked.	UNIX rights are checked.
UNIX	ACL and UNIX rights are checked.	UNIX rights are checked.
NT	ACL is checked.	ACL and UNIX rights are checked.

Access policy	CIFS clients	NFS clients
SECURE	ACL and UNIX rights are checked.	ACL and UNIX rights are checked.
MIXED	ACL is checked. If there is not an ACL, one is created based on the UNIX mode bits. Access is also determined by the ACL.  NFSv4 clients can manage the ACL.  An ACL modification rebuilds the UNIX mode bits but the UNIX rights are not checked.	ACL is checked. If there is not an ACL, one is created based on the UNIX mode bits. Access is also determined by the ACL.  NFSv4 clients can manage the ACL.  A modification to the UNIX mode bits rebuilds the ACL permissions but the UNIX rights are not checked.
MIXED_COMPAT	If the permissions of a file or directory were last set or changed by a CIFS client, the ACL is checked and the UNIX rights are rebuilt but are not checked.  If the permissions of a file or directory were last set or changed by an NFS client, the UNIX rights are checked and the ACL is rebuilt but is not checked.  NFSv4 clients can manage the ACL.	If the permissions of a file or directory were last set or changed by an NFS client, the UNIX rights are checked and the ACL is rebuilt but is not checked.  If the permissions of a file or directory were last set or changed by a CIFS client, the ACL is checked and the UNIX rights are rebuilt but are not checked.  NFSv4 clients can manage the ACL.

**Note:** The **MIXED** policy translates the UNIX ownership mode bits into three ACEs: Owner, Group, and Everyone, which can result in different permissions for the Group ACE and the Everyone ACE. The **MIXED\_COMPAT** policy does not translate a UNIX Group into a Group ACE. The Everyone ACE is generated from the UNIX Group.

#### [cvfsname=<newname>]

Changes the default name of the checkpoint in each of the .ckpt directories. The default name is the timestamp of when the checkpoint was taken.

# [noprefetch]

Turns prefetch processing off. When on (default), performs read ahead processing for filesystems.



# **CAUTION**

Turning the prefetch option to off may affect performance.

# For CIFS clients only

When mounting a filesystem, if the default options are not manually entered, the options are active but not displayed in the listing of mounted filesystems. Available options are:

#### [nonotify]

Turns notify off. When on (default), the notify option informs the client of changes made to the directory file structure.

#### [nooplock]

Turns opportunistic locks (oplocks) off. When oplocks are on (default), they reduce network traffic by enabling clients to cache the file and make changes locally. To turn Windows oplocks off, unmount the filesystem, then remount with **nooplock**.

### [notifyonaccess]

Provides a notification when a filesystem is accessed. By default, **notifyonaccess** is disabled.

## [notifyonwrite]

Provides a notification of write access to a filesystem. By default, the **notifyonwrite** option is disabled.

### [noscan]

Disables the Virus Checker protocol for a filesystem. The Virus Checker protocol is enabled using server\_setup and managed by server\_viruschk.

#### [uncached]

Allows well-formed writes (that is, multiple of a disk block and disk block aligned) to be sent directly to the disk without being cached on the server.

#### [cifssyncwrite]

Performs an immediate synchronous write on disk independently of CIFS write protocol option. This can impact write performance.

## [triggerlevel=<value>]

Specifies the deepest directory level at which notification occurs. The default is 512. The value -1 disables the notification feature.

## [ntcredential]

Enables the Celerra Network Server to take full account of a user's Windows group memberships when checking an ACL for access through NFS. When a UNIX user intiates a full request for a filesystem object, the UNIX UID is mapped to the Windows SID, then merges the user's UNIX and Windows groups together to generate a Windows NT Credential. This applies to NT, SECURE, MIXED, and MIXED\_COMPAT access-checking policies.

# [renamepolicy={CIFS|FULL|NO}]

Enables or disables control if any file or directory is opened on the current directory or any subdirectory, before the current directory is renamed. **CIFS** (default) stops the renaming of CIFS directories when in use by CIFS clients. **FULL** denies permission for the renaming of CIFS and NFS directories when in use by CIFS or NFS clients. **NO** automatically performs the directory rename without checking if a CIFS or NFS client is opening the directory.

**Note:** The **renamepolicy** is not supported by NFSv4.

# cifsnanoroundup

Rounds up to the next second any date set by a CIFS client.

# For NFS clients only

# [nolock|wlock|rwlock]

Indicates the impact of locking behavior on NFSv2 and NFSv3 clients against NFSv4 and CIFS file locking. In NFSv2 and NFSv3, locking rules are cooperative, so a client is not prevented from accessing a file locked by another client if it does not use the lock procedure. NFSv2 and NFSv3 locks as advisory. An advisory lock does not affect read and write access to the file, but informs other users that the file is already in use.

**Note:** NFSv4 and CIFS clients have mandatory locking schemes and do not require a locking policy.

Locking policy	NFS clients
nolock	This (default) can open and write to a file when it is locked by CIFS or NFSv4 clients.
wlock	This can read but cannot write data to a file locked by CIFS or NFSv4 clients.
rwlock	This (recommended) cannot read or write data to files locked by CIFS or NFSv4 clients.

# nfsv4delegation={NONE | READ | RW}

Indicates that specific actions on a file are delegated to the NFSv4 client. **NONE** indicates that no file delegation is granted. **READ** indicates only read delegation is granted. **RW** (default) indicates write delegation is granted.

## SEE ALSO

Managing Celerra Volumes and File Systems with Automatic Volume Management, Managing Celerra Volumes and File Systems Manually, Configuring NFS on Celerra, Configuring CIFS on Celerra, Managing Celerra for the Windows Environment, Managing Celerra for a Multiprotocol Environment, Using SnapSure on Celerra, nas\_fs, server\_checkup, server\_export, server\_mountpoint, server\_nfs, server\_setup, server\_umount, and server\_viruschk.

# **EXAMPLE #1** To display all mounted filesystems on **server\_2**, type:

\$ server\_mount server\_2

server\_2 :
root\_fs\_2 on / uxfs,perm,rw
root\_fs\_common on /.etc\_common uxfs,perm,ro
ufs2 on /ufs2 uxfs,perm,rw

**EXAMPLE #2** To mount all filesystems temporarily umounted from the mount table of **server\_2**, type:

\$ server\_mount server\_2 -all

server\_2 : done

**EXAMPLE #3** To mount ufs1, on mount point /ufs1, and enable CEPP for both CIFS and NFS, type::

\$ server\_mount server\_2 -o ceppcifs,ceppnfs ufs1 /ufs1

server\_2 : done

**EXAMPLE #4** To mount **ufs1**, on mount point **/ufs1**, with **nonotify**, **nolock**, and **cifssyncwrite** turned on, type:

\$ server\_mount server\_2 -option
nonotify,nolock,cifssyncwrite ufs1 /ufs1

server\_2 : done

**EXAMPLE #5** To mount **ufs1**, on mount point **/ufs1**, with the access policy set to **NATIVE**, and **nooplock** turned on, type:

\$ server\_mount server\_2 -option
accesspolicy=NATIVE,nooplock ufs1 /ufs1

server\_2 : done

**EXAMPLE #6** To mount ufs1, on mount point /ufs1, with noscan and noprefetch

set to on, type:

\$ server\_mount server\_2 -option noscan,noprefetch ufs1
/ufs1

server\_2 : done

**EXAMPLE #7** To mount ufs1, on mount point /ufs1, with notifyonaccess, notifyonwrite set to on, type:

\$ server\_mount server\_2 -option
notifyonaccess,notifyonwrite ufs1 /ufs1

server\_2 : done

**EXAMPLE #8** To mount a copy of a filesystem **ufs1\_snap1** on mount point /**ufs1\_snap1** with read-write access, type:

\$ server\_mount server\_2 -Force -option rw ufs1\_snap1
/ufs1\_snap1

server\_2 : done

**EXAMPLE #9** To mount **ufs1**, on mount point **/ufs1**, with uncached writes turned on, type:

\$ server\_mount server\_2 -option uncached ufs1 /ufs1

server\_2 : done

**EXAMPLE #10** To mount **ufs1**, on mount point **/ufs1**, with the trigger level of notification change set to **256**, type:

\$ server\_mount server\_2 -option triggerlevel=256 ufs1
/ufs1

server\_2 : done

**EXAMPLE #11** To mount **ufs1**, on mount point **/ufs1**, change the default name of the checkpoint in the .ckpt directory, and specify a mount point, type:

\$ server\_mount server\_2 -option cvfsname=test ufs1 /ufs1

server\_2 : done

**EXAMPLE #12** To mount **ufs1**, on mount point **/ufs1**, with the access policy set to **MIXED**, type:

\$ server\_mount server\_2 -option accesspolicy=MIXED ufs1
/ufs1

server\_2 : done

**EXAMPLE #13** To mount **ufs1**, on mount point **/ufs1**, with the access policy set to **MIXED\_COMPAT**, type:

\$ server\_mount server\_2 -option accesspolicy=MIXED\_COMPAT
ufs1 /ufs1

server\_2 : done

**EXAMPLE #14** To mount **ufs1**, as a part of the nested filesystem **nmfs1**, type:

\$ server\_mount server\_2 ufs1 /nmfs1/ufs1

server\_2 : done

**EXAMPLE #15** To mount **ufs1**, specifying that no file is granted to the NFSv4 client,

type:

\$ server\_mount server\_2 ufs1 nfsv4delegation=NONE

server\_2 : done

# server\_mountpoint

Manages mount points for the specified Data Movers.

# **SYNOPSIS**

```
server_mountpoint {<movername> | ALL}
   -list
   | {-create | -delete | -exist} < pathname>
```

## **DESCRIPTION**

**server\_mountpoint** creates, deletes, lists, or queries a mount point for the specified Data Mover or all Data Movers.

The ALL option executes the command for all Data Movers.

# OPTIONS -list

Lists all mount points for the specified Data Movers.

-create <pathname>

Creates a mount point. A <pathname> must begin with a slash (/).

-delete <pathname>
Deletes a mount point.

-exist <pathname>

Displays whether or not a mount point exists.

## SEE ALSO

Managing Celerra Volumes and File Systems with Automatic Volume Management, Managing Celerra Volumes and File Systems Manually, nas\_fs, server\_export, and server\_mountpoint.

## **EXAMPLE #1**

To create a mount point on **server\_2**, type:

\$ server\_mountpoint server\_2 -create /ufs1

server\_2 : done

## **EXAMPLE #2**

To list all mount points on a **server\_2**, type:

\$ server\_mountpoint server\_2 -list

```
server_2 :
/.etc_common
/ufs1
/ufs1_ckpt1
/ufs2
/ufs3
```

# **EXAMPLE #3** To verify that the mount point /ufs1, exists on all Data Movers, type:

\$ server\_mountpoint ALL -exist /ufs1

server\_2 : /ufs1 : exists

server\_3 : /ufs1 : does not exist

# **EXAMPLE #4** To delete the mount point /ufs1, on server\_2, type:

\$ server\_mountpoint server\_2 -delete /ufs1

server\_2 : done

# server\_mpfs

Sets up and configures MPFS protocol.

# **SYNOPSIS**

```
server_mpfs {<movername> | ALL}
   -set <var>=<value>
   -add <number_of_threads>
   -delete <number_of_threads>
   -Stats
   -Default [<var>|
   -mountstatus
```

# DESCRIPTION

**server\_mpfs** sets up the MPFS protocol. The configuration values entered with this command are saved into a configuration file on the Data Mover. MPFS is not supported on the NS series.

server\_setup provides information to start and stop MPFS for a Data Mover.

The **ALL** option executes the command for all Data Movers.

# **OPTIONS**

# No arguments

Lists the current MPFS configuration.

```
-set <var>=<value>
```

Sets the specified value for the specified variable. Currently, the only valid <var> is **threads**.

If this command is executed before the **server\_setup -P mpfs -o start** command is issued, the system sets the number of threads that will be started with the **server\_setup -o start** command, thereby overriding the default number of threads. If this command is executed after MPFS service is started, threads are to be added and removed dynamically.

```
-add <number_of_threads>
```

Increases the previously specified number of MPFS threads (default=16) by <number\_of\_threads> for the specified Data Movers.

```
-delete <number_of_threads>
```

Decreases the number of threads by the <number\_of\_threads> indicated for the specified Data Movers.

#### -Stats

Displays the current MPFS server statistics.

#### -mountstatus

Displays the mountability of filesystems for MPFS.

Certain filesystems cannot be mounted using MPFS, therefore before attempting to mount a filesystem on an MPFS client, compatibility should be determined. Filesystems that are not supported are running quotas, have checkpoints, or are using TimeFinder/FS.

# -Default [<var>]

Without a <var> entry, resets all variables to their factory-default values. Currently the only valid <var> is **threads**.

If a <var> is specified, only the specified value is reset to its factory-default value.

Note: Variable names are case-sensitive.

# **SEE ALSO**

*Using MPFS on Celerra*, server\_setup, and server\_mt.

# **EXAMPLE #1**

To set a value for a specified MPFS variable, type:

\$ server\_mpfs server\_2 -set threads=32

server\_2 :done

## **EXAMPLE #2**

To display the MPFS stats for **server\_2**, type:

\$ server\_mpfs server\_2 -Stats

server\_2 :

Server ID=server\_2 FMP Threads=32 Max Threads Used=2 FMP Open Files=0 FMP Port=4656

HeartBeat Time Interval=30

# **EXAMPLE #3**

To reset all variables back to their factory default value, type:

\$ server\_mpfs server\_2 -Default

server\_2 :done

# **EXAMPLE #4**

To check the mount status of a Data Mover, type:

\$ server\_mpfs server\_2 -mountstatus

server\_2 :

fs	mpfs compatible?	reason
	no	not a ufs file system
testing_renaming	no	volume structure not FMP compatible
	no	not a ufs file system
server2_fs1_ckpt	no	volume structure not FMP compatible

# **The server Commands**

```
mpfs_fs2_lockdb_ckpt_5
                                       volume structure not FMP compatible
mpfs_fs2_lockdb_ckpt_4
                                       volume structure not FMP compatible
mpfs_fs2_lockdb_ckpt_3
                                       volume structure not FMP compatible
mpfs_fs2_lockdb_ckpt_2
                                       volume structure not FMP compatible
mpfs_fs2_lockdb_ckpt_1
                                       volume structure not FMP compatible
mpfs_fs2_lockdb_ckpt_10
                                       volume structure not FMP compatible
mpfs_fs2_lockdb_ckpt_9
                                       volume structure not FMP compatible
mpfs_fs2_lockdb_ckpt_8
                                       volume structure not FMP compatible
mpfs_fs2_lockdb_ckpt_7
                                       volume structure not FMP compatible
                                       not a ufs file system
mpfs_fs2_lockdb_ckpt_6
                                       volume structure not FMP compatible
root_fs_common
                   yes
mpfs_fs2
                   yes
mpfs_fs1
                   mounted
server2_fs1
                   yes
root_fs_2
                   yes
```

**EXAMPLE #5** To add 16 threads to **server\_2**, type:

\$ server\_mpfs server\_2 -add 16

server\_2 : done

**EXAMPLE #6** To delete 16 threads from **server\_2**, type:

\$ server\_mpfs server\_2 -delete 16

server\_2 : done

# server\_mt

Manages the magnetic tape drive for the specified Data Mover.

# **SYNOPSIS**

# server\_mt <movername> [-f <device>][-N] <command> [<count>]

# DESCRIPTION

**server\_mt** sends commands to a magnetic tape drive located on the specified Data Mover.

By default, **server\_mt** performs the requested operation once. Specify <count> to perform multiple operations.

## **OPTIONS**

```
<command> <count>
```

Operations for the command are as follows:

#### eof, weof

Write <count> EOF marks at the current position on the tape.

#### fsf

Forward space over <count> EOF marks. The tape is positioned on the first block of the file.

## bsf

Back space over <count> EOF marks. The tape is positioned on the beginning-of-tape side of the EOF mark.

#### asf

Absolute space to <count> file number. This is equivalent to a rewind followed by a fsf <count>.

For the following commands, <count> is ignored:

#### eom

Space to the end of recorded media on the tape. This is useful for appending files onto previously written tapes.

## rewind

Rewind the tape.

#### offline, rewoffl

Rewind the tape and, if appropriate, take the drive unit offline by unloading the tape.

#### status

Print status information about the tape unit.

#### erase

Erase the entire tape.

# -f <device>

Specifies the magnetic tape device to which the command is applied, and the <device> which is the name of the tape drive.

# -N

Indicates no rewind at the end of the tape.

The **server\_devconfig** <movername> **-list -scsi -nondisks** command displays a listing of devices.

# **EXAMPLE**

To send the **rewind** command to **tape1** (magnetic tape drive) on a Data Mover, type:

\$ server\_mt server\_2 -f tape1 rewind

server\_2: done

# server\_name

Manages the name for the specified Data Movers.

You must delete all user-defined interconnects configured for a Data Mover before you can rename it using this command. After you rename the Data Mover, you must re-create the source and peer interconnects with the new Data Mover name and then restart any associated replication sessions.

# **SYNOPSIS**

```
server_name {<movername>|ALL}
  [<new_name>]
```

# **DESCRIPTION**

**server\_name** displays and renames the current movername for the specified Data Mover.

The **ALL** option executes the command for all Data Movers.

# **OPTIONS**

# No arguments

Displays movernames for all Data Movers.

<new\_name>

Changes the current movername to the newly specified movername. The movername is case-sensitive and supports the following characters: a through z, A through Z, 0 through 9, \_(underscore), - (hyphen), though names may not start with a hyphen.

The **ALL** option is not available when renaming Data Movers.

## SEE ALSO

nas\_server.

# **EXAMPLE #1**

To change the current movername for a Data Mover, type:

```
$ server_name server_2 my_srv2
```

server\_2 : my\_srv2

# **EXAMPLE #2**

To view all movernames, type:

\$ server\_name ALL

server\_2 : my\_srv2
server\_3 : my\_srv3
server\_4 : my\_srv4
server\_5 : my\_srv5

# server netstat

Displays the network statistics for the specified Data Mover.

# **SYNOPSIS**

```
server_netstat {<movername>}
[-a][-i][-r][-s][-p {tcp|udp|icmp|ip}]
```

# **DESCRIPTION**

**server\_netstat** displays network statistics for the specified Data Mover.

# **OPTIONS**

# No arguments

Displays the active TCP connections.

#### -a

Displays the state of all the IP, ICMP, TCP, and UDP sockets.

#### -i

Displays a summary of the state of all physical interfaces including the total amount of data transferred both ways and the number of errors during this process as well as the MTU size and MAC address.

#### -r

Displays the routing table.

#### -s

Displays the statistics of each protocol.

# -p {tcp|udp|icmp|ip}

Limits the display of the sockets, or the statistics of each protocol to the specified protocol. **tcp** is default.

## SEE ALSO

server\_ifconfig and server\_route.

# **EXAMPLE #1**

To display all active IP, ICMP, TCP, and UDP sockets, type:

## \$ server\_netstat server\_2

Proto	Local Address	Foreign Address	(state)
****	********	*******	*****
tcp	172.24.102.237.2483	172.24.102.17.445	ESTABLISHED
tcp	172.24.102.237.netbios-ssn	172.24.103.63.3962	ESTABLISHED

# **EXAMPLE #2** To display the states of the IP, ICMP, TCP, and UDP sockets, type:

## \$ server\_netstat server\_2 -a

Proto	Local Address	Foreign Address	(state)
****	* * * * * * * * * * * * * * * * * * * *	* * * * * * * * * * * * * * * * * * * *	*****
tcp	172.24.102.237.2483	172.24.102.17.445	ESTABLISHED

tcp	172.24.102.237.netbios-ssn	172.24.103.63.3962	ESTABLISHED
tcp	*.microsoft-ds	* • *	LISTEN
tcp	*.netbios-ssn	* * *	LISTEN
tcp	*.12345	* • *	LISTEN
tcp	*.5080	* • *	LISTEN
tcp	*.2272	* • *	LISTEN
tcp	*.2271	* • *	LISTEN
tcp	*.2270	*.*	LISTEN
tcp	*.ftp	*.*	LISTEN
tcp	*.10000	* • *	LISTEN
tcp	*.4658	*.*	LISTEN
tcp	*.2269	*.*	LISTEN
tcp	*.2268	*.*	LISTEN
tcp	*.nfs	* • *	LISTEN
tcp	*.1234	*.*	LISTEN
tcp	*.5033	*.*	LISTEN
tcp	*.8888	*.*	LISTEN
tcp	*.sunrpc	* • *	LISTEN
Proto	Local Address		
****	*****		
udp	*.sunrpc		
_	*.netbios-ns		
	* nethiog-dam		

udp \*.netbios-dgm

udp \*.snmp

udp \*.router udp \*.1024

udp \*.1036

\*.1037

udp

\*.1038 udp

udp \*.1046

udp \*.1054

\*.1065

udp \*.1234 udp

udp \*.nfs

\*.2268 udp

\*.4646 udp

udp \*.4647

\*.4658 udp

\*.9999 udp

udp \*.12345

udp \*.31491

udp \*.38914

#### EXAMPLE #3 To display a summary of the state of all physical interfaces, type:

# \$ server\_netstat server\_2 -i

Name	Mtu	Ibytes	Ierror	Obytes	0error	PhysAddr
******	*****	******	*****	******	*****	******
fxp0	1500	758568220	0	534867239	0	8:0:1b:43:49:9a
cge0	9000	18014329	0	7195540	0	8:0:1b:42:46:3
cge1	9000	306495706	0	9984	0	8:0:1b:42:46:4

cge2	9000	0	0	0	0	8:0:1b:42:46:2
cge3	9000	0	0	0	0	8:0:1b:42:46:7
cge4	9000	0	0	0	0	8:0:1b:42:46:5
cae5	9000	0	0	0	0	8:0:1b:42:46:6

#### **EXAMPLE #4** To display routing table statistics, type:

## \$ server netstat server 2 -r

Destination	Gateway *******		_ Type ******	Proto ******	Interface ****
0.0.0.0	172.24.102.254	255.255.255.0	DIRECT	RIP	cge0
128.221.253.0	128.221.253.2	255.255.255.0	DIRECT	RIP	fxp0
172.24.102.0	172.24.102.237	255.255.255.0	DIRECT	RIP	cge0
128.221.252.0	128.221.252.2	255.255.255.0	DIRECT	RIP	fxp0

#### **EXAMPLE #5** To display the statistics of each protocol, type:

\$ server\_netstat server\_2 -s

```
ip:
* * *
2315636 total packets received
0 bad header checksums
0 with unknown protocol
4 fragments received
0 fragments dropped (dup or out of space)
O fragments dropped after timeout
4 packets reassembled
2 packets forwarded
13046 packets not forwardable
13046 no routes
2302596 packets delivered
2267772 total packets sent
3 packets fragmented
0 packets not fragmentable
6 fragments created
icmp:
****
162 calls to icmp_error
Output histogram:
        echo reply: 1079145
        destination unreachable: 90
        echo: 1996
Input histogram:
        echo reply: 1993
        destination unreachable: 162
        routing redirect: 0
        echo: 1079145
        time exceeded: 0
        address mask request: 0
1081300 messages received
```

```
1081231 messages sent
tcp:
****
437648 packets sent
2 data packets retransmitted
0 resets
434138 packets received
212 connection requests
19 connections lingered
udp:
***
0 incomplete headers
27048 bad ports
760361 input packets delivered
744999 packets sent
         EXAMPLE #6
                        To display TCP protocol statistics, type:
                        $ server_netstat server_2 -s -p tcp
tcp:
****
437690 packets sent
2 data packets retransmitted
0 resets
434195 packets received
212 connection requests
19 connections lingered
```

# server\_nfs

Manages the NFS service, including secure NFS and NVSv4, for the specified Data Movers.

## SYNOPSIS

```
server_nfs {<movername> | ALL}
  -secnfs
     [-service {-start|-stop}]
   | -user {-list|-info {principal=<user_principal>|handle=<handle_number>}
        -release {principal=<user_principal>|handle=<handle_number>}}
    -principal {-create <service@server>|-delete <service@server>}
    -mapper
      {-info | -set {-source {auto | nis | file [path=<file_path>]}}
           -passwddb {nis|file}}
       -mapping {-list [name=<user_name>|uid=<UID>]
            -create {name=<user_name> | uid=<UID>}
            -delete {name=<user_name> | uid=<UID>}}}]
    -v4 {-service {-start|-stop}}
    -client {-list
       -info {index=<index>|hostname=<host>|ipaddress=<ipaddr>}
       -release {index=<index>|hostname=<host>|ipaddress=<ipaddr>}}
      -pnfs -service {-start | -stop}
  -stats [-zero][-rpc|-nfs]
```

## **DESCRIPTION**

**server\_nfs** manages the NFS service, including secure NFS and NFSv4.

For secure NFS, **server\_nfs** displays the status of the secure NFS service, stops and starts Kerberos authentication for secure NFS, and manages users authenticated through secure NFS. It also manages secure NFS security principals, releases the security context of a particular user, and manages the secure NFS user mapping service. Secure NFS supports UNIX-based, Linux-based, and Windows-based Kerberos KDCs.

For NFSv4, **server\_nfs** stops and starts the NFSv4 service, lists NFSv4 clients, and displays information about those clients.

**Note:** NFSv2 and v3 services are not started or stopped by this command.

The **ALL** option executes the command for all Data Movers.

# **OPTIONS**

-secnfs

Displays the status of the secure NFS service.

# -service {-start|-stop}

Enables or disables the secure NFS service on the Data Mover.

## -user -list

Lists all the NFS users authenticated by secure NFS.

```
-user -info {principal=<user_principal>|handle=
<handle_number>}
```

Displays the attributes of an authenticated user as specified by the <user\_principal> or <handle\_number>. The <user\_principal> is the Kerberos representation of a user. It is the username followed by the Kerberos realm name. The <handle\_number> is the identifier of a secure NFS user authentication context and is displayed using -user -list.

```
-user -release {principal=<user_principal>|
handle=<handle_number>}}
```

Releases the authentication context of the specified user.

```
-principal {-create <service@server>|-delete
<service@server>}
```

Creates or deletes a secure NFS service instance.

# -mapper -info

Displays the secure NFS mapping service configurations for the local file.

```
-mapper -set {-source {auto|nis|file [path=
<file_path>]}
```

Sets the type of mapping provider and the file path if it is a local file (the default file is gsscred\_db). These options are only relevant when using a Solaris UNIX Kerberos KDC. The default and recommended value is **-source auto**.

```
-mapper -set -passwddb {nis|file}
Sets the location of the password database.
```

-mapper -mapping -list [name=<user\_name>|uid= <UID>] Lists one or more user-mapping records as specified by <user\_name> or <UID>. This option is only relevant if you are using a local mapping file and a Solaris UNIX Kerberos KDC.

-mapper -mapping -create {name=<user\_name> | uid=<UID>} Creates a new mapping record. This option is only relevant if you are using a local mapping file and a Solaris UNIX Kerberos KDC.

```
-mapper -mapping -delete {name=<user_name>|uid=
<UID>}]
```

Deletes the mapping record from the database. This option is only relevant if you are using a local mapping file and a Solaris UNIX Kerberos KDC.

#### -v4

Displays the status of the NFSv4 service.

# -service {-start | -stop}

Enables or disables the NFSv4 service on the Data Mover.

To start the NFSv4 service, the /nas/server/slot\_<x>/config file, where <x> is the slot number of the Data Mover, must be configured so that the hivers argument is set to 4. If the hivers is not set to 4, append the nfs start line and ensure that the hivers value is set to 4. Restart the Data Mover to apply the change. Other versions of NFS can continue to run even if NFSv4 is disabled.

#### -client -list

Lists all NFSv4 client systems that have established state with the Celerra NFSv4 server.

```
-client -info {index | hostname = <host> | ipaddress = 
  <ipaddr>}
```

Displays the attributes of the client (clientID) as specified by the index, <host>, or <ipaddr>. The index value is obtained from the client list.

```
-client -release {index|hostname=<host>|
ipaddress=<ipaddr>}
```

Releases the state of the client (clientID) as specified by the index, <host>, or <ipaddr>. The index value is obtained from the client list.

# -pnfs -service {-start | -stop}

The pNFS is part of the NFS version 4.1 that allows clients to access storage devices directly and in parallel.

The pNFS service starts automatically once the NFSv4 starts. The pNFS service is stopped temporarily by using the -stop argument. The service can be restarted using the -start argument.

## -stats [-zero]

Displays all statistics or resets to zero all NFS and RPC statistics.

```
Displays RPC statistics or displays NFS statistics.
                             [-zero][-rpc]
                             Resets to zero all RPC statistics.
                             [-zero][-nfs]
                             Resets to zero all NFS statistics.
             SEE ALSO
                         Configuring NFS on Celerra and server kerberos.
          EXAMPLE #1
                         To display the status of the secure NFS service, type:
                         $ server nfs server 2 -secnfs
server_2 :
RPCSEC_GSS server stats
Credential count: 2
  principal: nfs@dm112-cge0.nasdocs.emc.com
  principal: nfs@dm112-cge0
  Total number of user contexts: 1
  Current context handle: 3
          EXAMPLE #2
                         To enable secure NFS service on server 2, type:
                         $ server_nfs server_2 -secnfs -service -start
server 2 : done
          EXAMPLE #3
                         To disable secure NFS service on server 2, type:
                         $ server_nfs server_2 -secnfs -service -stop
server 2 : done
          EXAMPLE #4
                         To display all secure NFS service instances, type:
                         $ server_nfs server_2 -secnfs -user -list
server_2 :
RPCSEC_GSS server stats
Credential count: 2
  principal: nfs@dm112-cge0.nasdocs.emc.com
  principal: nfs@dm112-cge0
  Total number of user contexts: 1
  Current context handle: 3
  PARTIAL user contexts:
  Total PARTIAL user contexts: 0
  USED user contexts:
```

[-rpc|-nfs]

```
principal=nfsuser1@NASDOCS.EMC.COM, service=nfs@dm112-cge0.nasdocs.emc.com,
handle=3, validity=35914s
  Total USED user contexts: 1
  EXPIRED user contexts:
  Total EXPIRED user contexts: 0
          EXAMPLE #5
                         To display the attributes of an authenticated server as specified by the
                         handle, type:
                         $ server_nfs server_2 -secnfs -user -info handle=3
server 2 :
  principal: nfsuser1@NASDOCS.EMC.COM
  service: nfs@dm112-cge0.nasdocs.emc.com
  handle: 3
  validity: 35844s
  GSS flags: mutl conf intg redy tran
  credential: uid=1010, inuid=1010, gid=1000
          EXAMPLE #6
                         To release the authentication context of the user specified by the
                         handle, type:
                         $ server_nfs server_2 -secnfs -user -release handle=3
server_2 : done
          EXAMPLE #7
                         To create a secure NFS service instance, type:
                         $ server_nfs server_2 -secnfs -principal -create
                         nfs1@dm112-cge0.nasdocs.emc.com
server_2 : done
          EXAMPLE #8
                         To delete a secure NFS service instance, type:
                         $ server nfs server 2 -secnfs -principal -delete
                         nfs1@dm112-cge0.nasdocs.emc.com
server_2 : done
          EXAMPLE #9
                         To set the mapping provider for the file, type:
                         $ server_nfs server_2 -secnfs -mapper -set -source file
server_2 : done
         EXAMPLE #10
                         To set the location of the password database, type:
                         $ server_nfs server_2 -secnfs -mapper -set -passwddb file
server_2 : done
         EXAMPLE #11
                         To display the secure NFS mapping service configurations for the
                         local file, type:
                         $ server_nfs server_2 -secnfs -mapper -info
server_2 :
Current NFS user mapping configuration is:
```

```
gsscred db = File /.etc/gsscred_db
gsscred db version = _v1
passwd db = File
```

# **EXAMPLE #12** To create a new mapping record, type:

\$ server\_nfs server\_2 -secnfs -mapper -mapping -create
name=nfsuser1

server\_2 : done

# **EXAMPLE #13** To display a list of the mapping records, type:

\$ server\_nfs server\_2 -secnfs -mapper -mapping -list

server\_2 :

 $0401000B06092A864886F7120102020000001A7365636E66737573657231407374617465732E656D\\632E636F6D\\1000\\nfsuser1, kerberos\_v5$ 

# **EXAMPLE #14** To delete a mapping record, type:

\$ server\_nfs server\_2 -secnfs -mapper -mapping -delete
name=nfsuser1

server\_2 : done

# **EXAMPLE #15** To

To enable the NFSv4 service on **server\_2**, type:

\$ server\_nfs server\_2 -v4 -service -start

server\_2 : done

# **EXAMPLE #16**

To start the NFSv4 service, type:

\$ server\_nfs {<movername>|ALL} -v4 -service -start

server\_2 : done

**Note:** The pNFS service starts automatically once the NFSv4 starts. To start the NFSv4 service, ensure that the /nas/server/slot\_<x>/config file has the hivers set to 4, where <x> is the slot number, for example slot\_2, slot\_3, and so on. The nfs entry in the above file should look like: nfs config hivers=4.

# **EXAMPLE #17**

To stop the pNFS service temporarily, type:

\$ server\_nfs {<movername>|ALL} -pnfs -service -stop

server\_2 : done

**Note:** When the pNFS service is stopped, all granted layouts are recalled and requests for new layouts are denied.

## **EXAMPLE #18**

To restart the pNFS service while the NFSv4 service is running, type:

 $\$ \ \ \, \texttt{server\_nfs} \ \, \{\texttt{<movername}\texttt{>} \, \big| \, \texttt{ALL} \} \ \ \, \texttt{-pnfs} \ \, \texttt{-service} \ \, \texttt{-start}$ 

server\_2 : done

# **EXAMPLE #19** To display the status of the NFSv4 service and the pNFS service, type:

```
$ server_nfs server_2 -v4
server_2 :
----- nfsv4 server status ------
            Service Started *
            pNFS service Started *
          * (yet operating) *
----- NFSv4 Clients -----
Confirmed Clients : 1
UnConfirmed Clients : 0
Number of users : 0
Number of lock owners : 0
Longest List
                    : 0
Shortest List : 0
Greatest depth to date : 0
Average List length : 0.00
Domain Name
                    : Not Defined
----- NFSv4 State -----
Opens : 4
Locks : 0
Delegations: 4
Layouts : 0
Free
        : 524280
```

# Where:

Value	Definition
Confirmed Clients	Active client (ready to work).
UnConfirmed Clients	Client in the process to establishing context.
Number of users	To be removed in non-debug images.
Longest List	To be removed in non-debug images.
Shortest List	To be removed in non-debug images.
Greatest depth to date	To be removed in non-debug images.
Average List length	To be removed in non-debug images.
Opens	Number of open files.
Locks	Number of locks being held.
Delegations	Number of granted delegations.
Free	To be removed in non-debug images.

# **EXAMPLE #20** To display all NFSv4 clients, type:

596

```
$ server_nfs server_2 -v4 -client -list
server_2 :
----- nfsv4 server client list -----
 hostname/ip : Index
NFSCLIENT1.nasdocs.emc.com : 0xa5400000
        EXAMPLE #21
                       To display the attributes of the NFSv4 client as specified by the index,
                       type:
                        $ server_nfs server_2 -v4 -client -info index=0xa5400000
server 2 :
 NFSCLIENT1.nasdocs.emc.com : 0xa5400000
     user: nfsuser1 : inode# 81
        EXAMPLE #22
                       To release the client ID of the client specified by the index, type:
                        $ server_nfs server_2 -v4 -client -release
                        index=0xa5400000
server_2 : done
        EXAMPLE #23
                       To disable the NFSv4 service on server_2, type:
                        $ server_nfs server_2 -v4 -service -stop
server_2 : done
        EXAMPLE #24
                       To display all NFS statistics, type:
                        $ server_nfs {<movername> | ALL} -stats
server_2 :
Server nfs (v2):
proc
              ncalls
                                %totcalls
                                                ms/call
                                                                failures
                                                0.0
               10
                                100.0
null
               0
                                                0.0
getattr
                                0.0
setattr
               Ο
                                0.0
                                                0.0
                                                                 Ω
                                0.0
                                                0.0
root.
                0
                                0.0
                                                 0.0
                                                                 0
lookup
readlink
                Ω
                                0.0
                                                 0.0
                                                                 0
                                0.0
                                                 0.0
read
                0
                                0.0
                                                 0.0
                                                                 0
wrcache
write
                0
                                0.0
                                                 0.0
                                                                 0
create
               0
                                0.0
                                                 0.0
                0
                                0.0
                                                 0.0
                                                                 0
remove
rename
                0
                                0.0
                                                 0.0
                0
                                                                 0
link
                                0.0
                                                 0.0
               0
                                0.0
                                                 0.0
                                                                 0
symlink
mkdir
                0
                                0.0
                                                 0.0
rmdir
               0
                                0.0
                                                0.0
readdir
               0
                                0.0
                                                0.0
                                                                 0
fsstat
                                0.0
                                                 0.0
```

Server nfs (v3):

proc	ncalls	%totcalls	ms/call	failures
v3null	0	0.0	0.0	0
v3getattr	233843	78.8	0.0	0
v3setattr	0	0.0	0.0	0
v3lookup	37130	12.5	0.0	0
v3access	11828	4.0	0.0	0
v3readlink	0	0.0	0.0	0
v3read	120	0.0	0.0	0
v3write	0	0.0	0.0	0
v3create	0	0.0	0.0	0
v3mkdir	0	0.0	0.0	0
v3symlink	0	0.0	0.0	0
v3mknod	0	0.0	0.0	0
v3remove	0	0.0	0.0	0
v3rmdir	0	0.0	0.0	0
v3rename	0	0.0	0.0	0
v3link	0	0.0	0.0	0
v3readdir	4005	1.3	0.0	0
v3readdirplus	0	0.0	0.0	0
v3fsstat	4878	1.6	0.0	0
v3fsinfo	4878	1.6	0.0	0
v3pathconf	0	0.0	0.0	0
v3commit	0	0.0	0.0	0
Server nfs (v4	):			
proc	ncalls	%totcalls	ms/call	failures
proc v4null	ncalls 4	%totcalls 0.0	ms/call 3.0	failures 0
-				
v4null	4	0.0	3.0	0
v4null v4compound	4 33645	0.0 48.8 0.0 0.3	3.0 0.1 0.0 0.0	0
v4null v4compound v4reserved	4 33645 0	0.0 48.8 0.0	3.0 0.1 0.0	0 0 0
v4null v4compound v4reserved v4access	4 33645 0 217	0.0 48.8 0.0 0.3	3.0 0.1 0.0 0.0	0 0 0 0
v4null v4compound v4reserved v4access v4close	4 33645 0 217 44	0.0 48.8 0.0 0.3	3.0 0.1 0.0 0.0 0.0	0 0 0 0
v4null v4compound v4reserved v4access v4close v4commit v4create v4delegPrg	4 33645 0 217 44 0	0.0 48.8 0.0 0.3 0.1	3.0 0.1 0.0 0.0 0.0 0.0	0 0 0 0 0
v4null v4compound v4reserved v4access v4close v4commit v4create v4delegPrg v4delegRet	4 33645 0 217 44 0	0.0 48.8 0.0 0.3 0.1 0.0 0.0	3.0 0.1 0.0 0.0 0.0 0.0 0.0	0 0 0 0 0 0
v4null v4compound v4reserved v4access v4close v4commit v4create v4delegPrg v4delegRet v4getAttr	4 33645 0 217 44 0 0 0 0 30 858	0.0 48.8 0.0 0.3 0.1 0.0 0.0 0.0 0.0	3.0 0.1 0.0 0.0 0.0 0.0 0.0 0.0 0	0 0 0 0 0 0 0 0
v4null v4compound v4reserved v4access v4close v4commit v4create v4delegPrg v4delegRet v4getAttr v4getFh	4 33645 0 217 44 0 0 0 30 858 220	0.0 48.8 0.0 0.3 0.1 0.0 0.0 0.0 0.0 1.2 0.3	3.0 0.1 0.0 0.0 0.0 0.0 0.0 0.0 0	0 0 0 0 0 0 0 0
v4null v4compound v4reserved v4access v4close v4commit v4create v4delegPrg v4delegRet v4getAttr v4getFh v4link	4 33645 0 217 44 0 0 0 30 858 220 0	0.0 48.8 0.0 0.3 0.1 0.0 0.0 0.0 0.0 1.2 0.3 0.0	3.0 0.1 0.0 0.0 0.0 0.0 0.0 0.0 0	0 0 0 0 0 0 0 0 0
v4null v4compound v4reserved v4access v4close v4commit v4create v4delegPrg v4delegRet v4getAttr v4getFh v4link v4lock	4 33645 0 217 44 0 0 0 30 858 220 0	0.0 48.8 0.0 0.3 0.1 0.0 0.0 0.0 0.0 1.2 0.3 0.0 0.0	3.0 0.1 0.0 0.0 0.0 0.0 0.0 0.0 0	0 0 0 0 0 0 0 0 0
v4null v4compound v4reserved v4access v4close v4commit v4create v4delegPrg v4delegRet v4getAttr v4getFh v4link v4lock v4lockT	4 33645 0 217 44 0 0 0 30 858 220 0 0	0.0 48.8 0.0 0.3 0.1 0.0 0.0 0.0 0.0 1.2 0.3 0.0 0.0 0.0	3.0 0.1 0.0 0.0 0.0 0.0 0.0 0.0 0	0 0 0 0 0 0 0 0 0 0
v4null v4compound v4reserved v4access v4close v4commit v4create v4delegPrg v4delegRet v4getAttr v4getFh v4link v4lock v4lockT v4lockU	4 33645 0 217 44 0 0 0 0 30 858 220 0 0	0.0 48.8 0.0 0.3 0.1 0.0 0.0 0.0 0.0 1.2 0.3 0.0 0.0 0.0	3.0 0.1 0.0 0.0 0.0 0.0 0.0 0.0 0	0 0 0 0 0 0 0 0 0 0
v4null v4compound v4reserved v4access v4close v4commit v4create v4delegPrg v4delegRet v4getAttr v4getFh v4link v4lock v4lockT v4lockU v4lookup	4 33645 0 217 44 0 0 0 30 858 220 0 0 0	0.0 48.8 0.0 0.3 0.1 0.0 0.0 0.0 0.0 1.2 0.3 0.0 0.0 0.0 0.0	3.0 0.1 0.0 0.0 0.0 0.0 0.0 0.0 0	0 0 0 0 0 0 0 0 0 0 0 0 0
v4null v4compound v4reserved v4access v4close v4commit v4create v4delegPrg v4delegRet v4getAttr v4getFh v4link v4lock v4lockT v4lockU v4lookup v4lookup	4 33645 0 217 44 0 0 0 0 30 858 220 0 0 0	0.0 48.8 0.0 0.3 0.1 0.0 0.0 0.0 0.0 1.2 0.3 0.0 0.0 0.0 0.0	3.0 0.1 0.0 0.0 0.0 0.0 0.0 0.0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
v4null v4compound v4reserved v4access v4close v4commit v4create v4delegPrg v4delegRet v4getAttr v4getFh v4link v4lock v4lockT v4lockU v4lookup v4lookupp v4nVerify	4 33645 0 217 44 0 0 0 0 30 858 220 0 0 0 0	0.0 48.8 0.0 0.3 0.1 0.0 0.0 0.0 0.0 1.2 0.3 0.0 0.0 0.0 0.0 0.0	3.0 0.1 0.0 0.0 0.0 0.0 0.0 0.0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
v4null v4compound v4reserved v4access v4close v4commit v4create v4delegPrg v4delegRet v4getAttr v4getFh v4link v4lock v4lockT v4lockU v4lookup v4lookup v4noverify v4open	4 33645 0 217 44 0 0 0 0 30 858 220 0 0 0 0 171 0	0.0 48.8 0.0 0.3 0.1 0.0 0.0 0.0 0.0 1.2 0.3 0.0 0.0 0.0 0.0 0.0	3.0 0.1 0.0 0.0 0.0 0.0 0.0 0.0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
v4null v4compound v4reserved v4access v4close v4commit v4create v4delegPrg v4delegRet v4getAttr v4getFh v4link v4lock v4lockT v4lockU v4lookup v4lookup v4nVerify v4open v4openAttr	4 33645 0 217 44 0 0 0 0 30 858 220 0 0 0 0 171 0 0 48	0.0 48.8 0.0 0.3 0.1 0.0 0.0 0.0 0.0 1.2 0.3 0.0 0.0 0.0 0.0 0.0 0.0	3.0 0.1 0.0 0.0 0.0 0.0 0.0 0.0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
v4null v4compound v4reserved v4access v4close v4commit v4create v4delegPrg v4delegRet v4getAttr v4getFh v4link v4lock v4lockT v4lockU v4lookup v4lookup v4lookup v4nVerify v4open v4openAttr	4 33645 0 217 44 0 0 0 0 30 858 220 0 0 0 0 171 0 0 48 0 5	0.0 48.8 0.0 0.3 0.1 0.0 0.0 0.0 0.0 1.2 0.3 0.0 0.0 0.0 0.0 0.0 0.0	3.0 0.1 0.0 0.0 0.0 0.0 0.0 0.0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
v4null v4compound v4reserved v4access v4close v4commit v4create v4delegPrg v4delegRet v4getAttr v4getFh v4link v4lock v4lockT v4lockU v4lookup v4lookup v4lookup v4nVerify v4open v4openAttr v4open_Conf v4open_DG	4 33645 0 217 44 0 0 0 0 30 858 220 0 0 0 0 171 0 0 48 0 5	0.0 48.8 0.0 0.3 0.1 0.0 0.0 0.0 0.0 1.2 0.3 0.0 0.0 0.0 0.0 0.0 0.0 0.0	3.0 0.1 0.0 0.0 0.0 0.0 0.0 0.0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
v4null v4compound v4reserved v4access v4close v4commit v4create v4delegPrg v4delegRet v4getAttr v4getFh v4link v4lock v4lockT v4lockU v4lookup v4lookup v4lookup v4nVerify v4open v4openAttr v4open_Conf v4open_DG v4putFh	4 33645 0 217 44 0 0 0 0 30 858 220 0 0 0 0 0 171 0 0 48 0 5 0	0.0 48.8 0.0 0.3 0.1 0.0 0.0 0.0 0.0 1.2 0.3 0.0 0.0 0.0 0.0 0.0 0.0 0.0	3.0 0.1 0.0 0.0 0.0 0.0 0.0 0.0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
v4null v4compound v4reserved v4access v4close v4commit v4create v4delegPrg v4delegRet v4getAttr v4getFh v4link v4lock v4lockT v4lockU v4lookup v4lookup v4lookup v4nVerify v4open v4openAttr v4open_Conf v4open_DG	4 33645 0 217 44 0 0 0 0 30 858 220 0 0 0 0 171 0 0 48 0 5	0.0 48.8 0.0 0.3 0.1 0.0 0.0 0.0 0.0 1.2 0.3 0.0 0.0 0.0 0.0 0.0 0.0 0.0	3.0 0.1 0.0 0.0 0.0 0.0 0.0 0.0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

v4read	1	0.0	0.0	0
v4readDir	21	0.0	0.6	0
v4readLink	0	0.0	0.0	0
v4remove	30	0.0	2.9	0
v4rename	2	0.0	0.0	0
v4renew	32335	46.9	0.0	2
v4restoreFh	0	0.0	0.0	0
v4saveFh	2	0.0	0.0	0
v4secInfo	0	0.0	0.0	0
v4setAttr	39	0.1	0.7	0
v4setClntid	2	0.0	0.0	0
v4clntid_Conf	2	0.0	0.0	0
v4verify	0	0.0	0.0	0
v4write	24	0.0	5.7	0
v4rel_Lockown	0	0.0	0.0	0
v4backChanCtl	0	0.0	0.0	0
v4bindConn	0	0.0	0.0	0
v4exchangeId	0	0.0	0.0	0
v4createSess	0	0.0	0.0	0
v4destroySess	0	0.0	0.0	0
v4freeStateid	0	0.0	0.0	0
v4getDirDeleg	0	0.0	0.0	0
v4getDevInfo	0	0.0	0.0	0
v4getDevList	0	0.0	0.0	0
v4layoutCmmt	0	0.0	0.0	0
v4layoutGet	0	0.0	0.0	0
v4layoutRet	0	0.0	0.0	0
v4secinfoNoName		0.0	0.0	0
v4sequence	0	0.0	0.0	0
v4setSsv	0	0.0	0.0	0
v4testStateid	0	0.0	0.0	0
v4wantDeleg	0	0.0	0.0	0
v4destroyClid	0	0.0	0.0	0
v4reclaimCmpl	0	0.0	0.0	0
v4illegal	0	0.0	0.0	0

# Server lookupcache:

nHit	nFind	nNegadd	nChecked
39459	46408	21	39459

# Server rpc:

ncalls	nBadRpcData	nDuplicates	nResends	nBadAuths
822126	0	0	0	0

# Where:

Value	Definition	
ncalls	Number of calls for each NFS operation.	
%totcalls	Percentage of calls per operation out of total NFS calls received.	
ms/call	Average time taken for the NFS operations.	

failures	Number of NFS failures for each NFS operation.
nHit	'Directory name lookup cache' hits.
nFind	'Directory name lookup cache' operations.
nNegadd	Number of negative entries added to the 'Directory name lookup cache'.
nChecked	'Directory name lookup cache' entries searched.
nBadRpcData	Calls with bad RPC header.
nDuplicates	Calls with duplicate XID.
nResends	Number of RPC replies resent.
nBadAuths	Number of replies failing RPC authentication.

# **EXAMPLE #25** To display RPC statistics, type:

\$ server\_nfs server\_2 -stats -rpc

server\_2 :
Server rpc:

ncalls nBadRpcData nDuplicates nResends nBadAuths 822155 0 0 0 0 0 0

# **EXAMPLE #26** To reset statistics counters, type:

\$ server\_nfs {<movername> | ALL} -stats -zero

server\_2 : done
server\_3 : done
server\_4 : done
server\_5 : done

# server\_nis

Manages the Network Information Service (NIS) configuration for the specified Data Movers.

# **SYNOPSIS**

```
server_nis {<movername> | ALL}
   [-delete | <domainname> { <ip_addr>,...}]
   | {<hostname> | <ip_addr>} [, {<hostname> | <ip_addr>}...]
   | -status
```

## DESCRIPTION

**server\_nis** provides connectivity to the NIS lookup servers.

The **ALL** option executes the command for all Data Movers.

# **OPTIONS**

# No arguments

Displays the NIS configuration.

#### -delete

Deletes all the NIS lookup servers.

```
<domainname> {<ip_addr>,<ip_addr>,... }
```

Creates a list of up to 10 IP addresses to be used as NIS lookup servers for the specified <domainname>. Up to 10 NIS servers in a single NIS domain are supported for each Data Mover. A Data Mover can support only one NIS domain.

Each time an NIS domain is configured, the previous configuration is overwritten.

To provide high availability, you must configure your environment with multiple NIS servers. If you configure only a single NIS server, a problem with this NIS server could result in data unavailability.

```
{<hostname>|<ip_addr>} [, {<hostname>|<ip_addr>}...] Queries the NIS lookup servers using either a <hostname> or IP address. Multiple hostnames or IP addresses can be queried. Prior to performing a query, the NIS server must be activated and operational.
```

#### -status

Displays the status of the NIS lookup server even when the NIS server is not operational.

# **SEE ALSO**

Configuring Celerra Naming Services and server\_dns.

**EXAMPLE #1** To provide connectivity to the NIS lookup server for the specified

domain, type:

\$ server\_nis server\_2 nasdocs 172.24.102.30

server\_2 : done

**EXAMPLE #2** To query NIS lookup servers using both a hostname and IP address,

type:

\$ server\_nis server\_2 test40,172.24.102.36,test44

server\_2 :

test40 = 172.24.102.30 test46 = 172.24.102.36 test44 = 172.24.102.34

**EXAMPLE #3** To display the NIS configuration, type:

\$ server\_nis server\_2

server\_2 : yp domain=nasdocs server=172.24.102.30

**EXAMPLE #4** To display the status of the NIS lookup servers, type:

\$ server\_nis server\_2 -status

server\_2 :

NIS default domain: nasdocs NIS server 172.24.102.30

If NIS was not started, the output of this command will appear as:

\$ server\_nis server\_2 -status

server\_2 :

NIS not started

**EXAMPLE #5** To delete all of the NIS lookup servers for a Data Mover, type:

\$ server\_nis server\_2 -delete

server\_2 : done

# server\_param

Manages parameter information for the specified Data Movers.

## **SYNOPSIS**

#### DESCRIPTION

**server\_param** modifies and displays information on parameters for the Celerra Network Server.

The **ALL** option executes the command for all Data Movers.

# OPTIONS -info

Displays the facilities that can be modified.

```
-facility {<facility_name>|-all} -list
```

Lists all of the parameters with their default, current, and configured values for the specified facility or facilities.

```
-facility {<facility_name>|-all} -info{<param_name>
[-verbose]|-all}
```

Displays information on the specified parameter or parameters for the specified facility or facilities. The **-verbose** option displays a more detailed description, if one is available, for the specified <param\_name>.

```
-facility <facility_name> -modify <param_name> -value
<new_value>
```

Modifies the value of the specified <param\_name> and changes the value to <new\_value>. The <facility\_name> is the facility for <param\_name>.

# **SEE ALSO** *Celerra Network Server Parameters Guide.*

# **EXAMPLE #1** To view all facilities that can be modified on **server\_2**, type:

## \$ server\_param server\_2 -info

```
server_2 :
facility description
FLRCompliance File Level Retention
NDMP Network Data Management Protocol
PAX Data Backup/Restore facility for NDMP/server_archive
RCP Replication Control Protocol
```

VRPL Volume replication control ana Adaptec Network Adapter arp Address Resolution Protocol

cfs Common File System

cifs Common internet FS (Microsoft Network File Protocol)

config Misc. configuration parameters for CIFS

cvfs Celerra Virtual File System dedupe File-level Data Deduplication

dns Domain Name Service

ds Domain Controller service for CIFS fcTach Agilent Fibre Channel Controller file Overall file system parameters filesystem File system

ftpd File Transfer Protocol Daemon http Hypertext Transfer Protocol

ip Internet protocol iscsi Internet Scsi Protocol

ldap Lightweight Directory Access Protocol

lockd Network Lock Manager mount NFS Mount Protocol

nbs Network Block Service Protocol

nfs Network File System
nfsv4 NFS version 4 protocol
quota File system quota management
replication Data replication facility
shadow Cross Protocol naming support
ssl SSL security network protocol

statd Host status demon

streamio Streaming tape I/O support svtl Software-Based Tape Library tcp Transmission Control Protocol

tftp Trivial File Transfer Protocol Daemon

trunk Network trunking support

vbb Volume Based Backup

vdevice Virtual IP Device Parameters

viruschk Virus checking service

## Where:

Value	Definition
facility	Facility for the parameter.
description	Description of the facility.

# **EXAMPLE #2** To view the PAX parameters that can be modified, type:

## \$ server\_param server\_2 -facility PAX -list

server_2 :	_		_	
param_name	facility	default	current	configured
checkUtf8Filenames	PAX	1	1	
dump	PAX	0	0	
nPrefetch	PAX	8	8	
nThread	PAX	64	64	

writeToArch	PAX	1	1
paxReadBuff	PAX	64	64
writeToTape	PAX	1	1
filter.numDirFilter	PAX	5	5
paxWriteBuff	PAX	64	64
filter.numFileFilter	PAX	5	5
filter.dialect	PAX	1.1	1.1
nFTSThreads	PAX	8	8
paxStatBuff	PAX	128	128
readWriteBlockSizeInKB	PAX	64	64
nRestore	PAX	8	8
filter.caseSensitive	PAX	1	1
scanOnRestore	PAX	1	1
noFileStreams	PAX	0	0
allowVLCRestoreToUFS	PAX	0	0

# Where:

Value	Definition
param_name	Name of the parameters with the specified facility that can be modified.
facility	Facility for the parameters.
default	Default value for the parameter.
current	Current value used by the Data Mover.
configured	Value set by the user. If some user action is pending (such as a Data Mover reboot), it might not have taken effect.

If the values for current and configured differ, refer to the user\_action field of the **-info** option.

# **EXAMPLE #3** To view information on the nThread parameter, type:

# \$ server\_param server\_2 -facility PAX -info nThread

server\_2 : name = nThread facility\_name = PAX default\_value = 64 current\_value = 64 configured\_value user\_action = none change\_effective = immediate = (1, 128)range

description = Number of worker threads per backup session

# Where:

Value	Definition	
facility_name	Facility for the parameter.	
default_value	Default value set for the parameter.	
current_value	Value set on the Data Mover.	

Value	Definition
configured_value	Value set by the user. If some user action is pending (such as a Data Mover reboot), it might not have taken effect.
user_action	Action necessary for the parameter to take effect.
change_effective	States when the change will be effective.
range	Range of possible parameter values.
description	Description of what the parameter does.

# **EXAMPLE #4** To modify the configured nThread parameter, type:

\$ server\_param server\_2 -facility PAX -modify nThread
-value 32

server\_2 : done

# **EXAMPLE #5** To modify the configured cipher parameter, type:

\$ server\_param server\_2 -facility PAX -modify cipher
-value foobar

server\_2 : done
Warning 17716815750: server\_2 : You must reboot server\_2 for paxReadBuff changes
to take effect.

To verify the configured cipher parameter, restart the Data Mover and type:

\$ server\_param server\_2 -facility ssl -info cipher

configured\_value = foobar

user\_action = reboot DataMover
change\_effective = reboot DataMover

range = \*

description = Keyword specifying the default supported SSL cipher suites (e.g: ALL:!LOW:@STRENGTH)

**Note:** If the current\_value and configured\_value parameters differ and if the user\_action and change\_effective parameters display the text reboot Data Mover, restart the Data Mover. After restarting the Data Mover, if the current\_value and configured\_value parameters continue to differ, it indicates that the Data Mover encountered an error after it was restarted. Check the server\_log output to view the error reported.

To view the server\_log command output file, type:

```
$ server_log server_2 | grep param
```

```
... 2009-08-25 12:20:59: ADMIN: 3: Command failed: param ssl cipher=foobar ...
```

# **EXAMPLE #6** To view the values of the NDMP port ranges on the Data Mover **server\_2**, type:

\$ server\_param server\_2 -facility NDMP -info portRange

server\_2 :

default\_value = 1024-65535 current\_value = 1024-65535

configured\_value

description = Port range for NDMP data connection listening

# **EXAMPLE #7**

To set the values of the NDMP port ranges on the Data Mover **server\_2**, type:

\$ server\_param server\_2 -facility NDMP -modify portRange
-value 50000-50100

server\_2 : done

# **EXAMPLE #8**

To display the parameters for the **SSL** facility, type:

\$ server\_param server\_2 -facility ssl -info -all

server\_2 :

name = trace facility\_name = ssl

default\_value = 0x00000000
current\_value = 0x00000000

configured\_value =
user\_action = none
change\_effective = immediate

range = (0x00000000, 0xffffffff)

description = Define SSL traces displayed in the server log

user\_action = reboot DataMover
change\_effective = reboot DataMover

range = (1,120)

description = Timeout (in seconds) used to receive SSL packets

from network during SSL handshake

```
default_value
                 = 0
current_value
                 = 0
configured_value =
user_action
                 = reboot DataMover
change_effective = reboot DataMover
                = (0,2)
range
                 = Set the default ssl protocol. Possible values are:
description
0=all ssl/tls protocol are allowed, 1=only sslv3 is allowed, 2=only tlsv1
is allowed
                 = threads
name
facility name
                = ssl
default_value
                = 10
current_value
                = 10
configured_value =
user_action
              = reboot DataMover
change_effective = reboot DataMover
                = (4,30)
range
description
                = Number of SSL threads
                 = cipher
name
facility_name
                 = ssl
                 = 'ALL:!ADH:!SSLv2:@STRENGTH'
default_value
current_value
                 = 'ALL:!ADH:!SSLv2:@STRENGTH'
configured_value =
user_action
               = none
change_effective = reboot DataMover
range
                 = Keyword specifying the default supported SSL cipher
description
suites (e.g: ALL:!LOW:@STRENGTH)
         EXAMPLE #9
                        To display the default SSL parameters on server_2, type:
                        $ server_param server_2 -facility ssl -list
server_2 :
param_name
                                             default
                                                        current
                                                                  configured
                                      ssl 0x00000000 0x00000000
trace
timeout
                                      ssl
                                                   5
                                                              5
                                                              0
                                                  0
protocol
                                      ssl
threads
                                      ssl
                                                  10
                                                             10
                                      ssl 'ALL:!ADH:!SSLv2:@STRENGTH'
cipher
'ALL: !ADH: !SSLv2:@STRENGTH'
        EXAMPLE #10
                        To modify the SSL dedicated threads to 20, type:
                        $ server_param server_2 -facility ssl -modify threads
                        -value 20
server_2 : done
Warning 17716815750: server_2: You must reboot server_2 for threads changes to
take effect.
```

# **EXAMPLE #11**

To modify the default cipher suite to all (except low-security algorithms and MD5), type:

\$ server\_param server\_2 -facility ssl -modify cipher
-value 'ALL:!LOW:!MD5:@STRENGTH'

server 2 : done

Warning 17716815750: server\_2 : You must reboot server\_2 for cipher changes to take effect.

# **EXAMPLE #12** To display the default ftpd parameters, type:

# # server\_param server\_2 -facility ftpd -list

server\_2 :

facility default current configured param\_name shortpathdir 0 ftpd 0 defaultdir ftpd ftpd 0 wildcharsInDir 0 bounceAttackChk ftpd 1

# **EXAMPLE #13** To display the parameters for the **ftpd** facility, type:

# \$ server\_param server\_2 -facility ftpd -info -all

server\_2 :
name = shortpathdir

facility\_name = ftpd
default\_value = 0
current value = 0

configured\_value =
user\_action = none
change\_effective = immediate

range = (0,1)

description = Enable return file name instead of full pathname in DIR

command

user\_action = none
change\_effective = immediate

range = \*

description = Sets the default working directory for FTP

name = wildcharsInDir

facility\_name = ftpd
default\_value = 0
current\_value = 0
configured\_value =

user\_action = none
change\_effective = immediate

range = (0,1)description = Enable wild characters for directory names name = bounceAttackChk facility\_name = ftpd default\_value = 1 current\_value = 1 configured\_value = user\_action = none

change\_effective = immediate = (0,1)range

description = Enable bounce attack check

# **EXAMPLE #14**

To display the detailed description of the shortpathdir parameter for the **ftpd** facility, type:

\$ server param server 2 -facility ftpd -info shortpathdir -verbose

server\_2 :

name = shortpathdir

facility\_name = ftpd default\_value = 0 current\_value = 0 configured\_value = user\_action = none change\_effective = immediate range = (0,1)

description = Enable return file name instead of full pathname in DIR

command

detailed\_description

Enable (1) or disable (0) return file name instead of full pathname in the commands dir or ls. if wild char are used this parameter is inefficient.

# server\_pax

Displays and resets backup and restore statistics and filesystem information for a backup session already in progress.

## **SYNOPSIS**

```
server_pax {<movername>|ALL}
-stats [-reset|-verbose]
```

## DESCRIPTION

**server\_pax** displays and resets backup and restore statistics and filesystem information for a backup session already in progress.

# **OPTIONS**

-stats

Displays backup and restore statistics for a backup or restore session in progress.

```
[-reset | -verbose]
```

The **-reset** option resets the statistics for a backup or restore session already in progress. The **-verbose** option displays backup performance statistics and file size distribution data for a backup or restore session in progress.

## SEE ALSO

Celerra Network Server Parameters Guide.

## EXAMPLE #1

To reset the statistics for a backup session already in progress, type:

```
$ server_pax server_2 -stats -reset
```

server 2 : done

# **EXAMPLE #2**

To view the verbose statistics for an active NDMP backup session on server\_2, type:

```
$ server_pax server_2 -stat -verbose
```

```
******* SUMMARY PAX STATS **********
     ---- NASS STATS ----
     nass00 is not doing backup
     nass01 is not doing backup
     nass02 is not doing backup
     nass03 is not doing backup
     ---- NASA STATS ----
     ** nasa thid 0 (RESTORE) **
     Backup root directory: /16m_ok_1_0
     Total bytes processed: 12140605440
             file
                      processed: 738
     Total
     throughput: 68 MB/sec
     average file size: 16065KB
     Total nasa wait nass count: 0
     Total nasa wait nass time: 0 msec
```

```
Total time since last reset: 170 sec
Tape device name: /dev/c80t010
             size file processed: 13
     1 -- 8KB size file processed: 1
 8KB+1 -- 16KB size file processed: 0
16KB+1 -- 32KB size file processed: 0
32KB+1 -- 64KB size file processed: 0
64KB+1 -- 1MB size file processed: 0
1MB+1 -- 32MB size file processed: 724
32MB+1 -- 1GB size file processed: 0
     1G more size file processed: 0
fs /16m ok 1 0 size is: 120855445504 Bytes
Estimated time remain is 1524 sec
nasa01 is not doing backup/restore
nasa02 is not doing backup/restore
nasa03 is not doing backup/restore
---- NASW STATS ----
nasw00 RESTORE (in progress)
Session Total Time: 00:02:50 (h:min:sec)
Session Idle Time: 00:00:56 (h:min:sec)
KB Tranferred: 11858820 Block Size: 61440 (60 KB)
Average Transfer Rate: 68 MB/Sec 239 GB/Hour
Average Burst Transfer: 101 MB/Sec 357 GB/Hour
__Point-in-Time__ (over the last 10 seconds):
Rate=69 MB/Sec Burst=96 MB/Sec Idle=283 msec/sec
Get Pool: 17 buffers Put Pool: 29 buffers
Compression Page not available
ReadC=0.00 WriteC=0.00 Read=0 KB Written=0 KB
nasw01 BACKUP
              (terminated)
nasw02 BACKUP (terminated)
nasw03 BACKUP (terminated)
```

## Where:

Value	Definition
NASS STATS	Thread responsible for traversing the filesystem and providing metadata for each directory or file or both.
Total file processed	Total number of files or directories or both for which metadata was processed.
Total NASS wait NASA count	The number of times NASS waited for NASA.
Total NASS wait NASA time	Amount of time NASS waited for NASA.
Total time since last reset	Time since the last reset; a reset occurs automatically when a backup completes.
fts_build time	Time spent building the filesystem or directory tree.
getstatpool	If the value is consistently 0, then NASA may be slowing down the backup.
putstatpool	If the value is consistently 0, then NASS may be slowing down the backup.
NASA STATS	Thread responsible for writing file header information, reading file data, and writing to the buffer.
Backup root directory	Directory being backed up.

Value	Definition
Total bytes processed	Bytes backed up since the last reset or start of the current backup.
Total file processed	Number of files backed up since the start or reset of the current backup.
Throughput	How fast NASA processed data.
Average file size	Average file size for the current backup.
Total nasa wait nass count time	Number of times NASA waited for NASS.
Total nasa wait nass time	Amount of time NASA waited for NASS.
Total time since last reset	Amount of time since the backup statistics were reset; a reset occurs automatically when a backup completes.
Tape device name	Target device for the backup data.
File size statistics	Statistics on the size of files backed up since the start or reset of the current backup.
NASW STATS	Thread responsible for getting data from the buffer pool, writing it to tape or sending it to a remote Data Mover.
Session total time	Total time of the current session.
Session idle time	Idle time for the current session.
KB transferred	Total KB transferred.
Average transfer rate	Average transfer rate for the current session's data per second and per hour.
Average burst transfer	Burst transfer rate in MB/s and GB/s.
Write block counters (List/Direct)	Scatter/gather write count.
_Point-in-time_ (over the last 10 seconds)	Information on data processed during a 10 second interval.
Rate	Transfer rate in MB/s.
Burst	Burst transfer rate in MB/s.
Idle	Amount of time NASW was idle in msec.
Get pool	Number of buffers in get pool; if value is consistently 0, then NASA and NASS may be slowing down the backup.
Put pool	Number of buffers in put pool; if value is consistently 0, then the tape may be slowing down the backup.
Compression rate retrieved	Compression rate.
ReadC	Read compression rate at the tape device.
WriteC	Write compression rate at the tape device.
Read	Amount of data read in KB.
Written	Amount of data written in KB.

# **EXAMPLE #3** To view the verbose statistics for an active NDMP restore session on **server\_2**, type:

```
$ server_pax server_2 -stats -verbose
server_2 :
******** SUMMARY PAX STATS **********
---- NASS STATS ----
nass00 is not doing backup
nass01 is not doing backup
nass02 is not doing backup
nass03 is not doing backup
---- NASA STATS ----
** nasa thid 0 (non-DAR RESTORE) **
The first five entries of restore name list are:
original name: /filt, destination name /ufsvbbr/r_filter_pax
Total bytes processed: 172326912
Total
        file
                 processed: 42
throughput: 7 MB/sec
average file size: 4006KB
Total nasa wait nass count: 0
Total nasa wait nass time: 0 msec
Total time since last reset: 21 sec
Tape device name: c0t0l1
dir
      or
           0 size file processed: 17
    1 -- 8KB size file processed: 6
8KB+1 -- 16KB size file processed: 18
16KB+1 -- 32KB size file processed: 0
32KB+1 -- 64KB size file processed: 0
64KB+1 -- 1MB size file processed: 1
1MB+1 -- 32MB size file processed: 0
32MB+1 -- 1GB size file processed: 0
    1G more size file processed: 0
nasa01 is not doing backup/restore
nasa02 is not doing backup/restore
nasa03 is not doing backup/restore
---- NASW STATS ----
nasw00 RESTORE (in progress)
Session Total Time: 00:00:21 (h:min:sec)
Session Idle Time: 00:00:00 (h:min:sec)
KB Tranferred: 168384 Block Size: 32768 (32 KB)
Average Transfer Rate: 7 MB/Sec 27 GB/Hour
Average Burst Transfer: 7 MB/Sec 27 GB/Hour
__Point-in-Time__ (over the last 10 seconds):
Rate=6 MB/Sec Burst=7 MB/Sec Idle=0 msec/sec
Get Pool: 61 buffers Put Pool: 0 buffers
nasw01 No session found
nasw02 No session found
nasw03 No session found
```

# server\_ping

Checks the network connectivity for the specified Data Movers.

## **SYNOPSIS**

```
server_ping {<movername>|ALL}
[-send|-interface <interface>] {<hostname>|<ip_addr>}
```

### **DESCRIPTION**

**server\_ping** sends ICMP ECHO\_REQUEST messages to a remote host and receives ICMP ECHO\_REPLY messages back that verify and determine network connectivity.

As the transmission time is stamped into the ECHO\_REQUEST message, **server\_ping** displays the transmission delay between the Data Mover and the remote host.

The **ALL** option executes the command for all Data Movers.

## **OPTIONS**

```
<hostname>|<ip_addr>
```

Provides a one-shot test to determine if the specified <hostname> or <ip\_addr> is accessible. The IP address may be used if NIS or DNS or both are not operational.

```
-send {<hostname>|<ip_addr>}
```

Repeats sending/receiving ECHO\_REQUEST messages to a specified <hostname> or <ip\_addr> for a remote host. The IP address may be used if NIS or DNS or both are not operational.

Press **Ctrl+C** to quit.

When invoking **server\_ping** with a hostname, the **/.etc/hosts** file is checked first, then the NIS and the DNS servers (if operational on the Data Mover) are checked. If the name does not exist in either case, an error message is displayed.

```
-interface {<interface> <hostname>|<ip_addr>}
```

Provides a one-shot test to determine if the specified <hostname> or <ip\_addr> is accessible through the specified <interface>. The IP address may be used if NIS or DNS or both are not operational.

## **SEE ALSO**

Configuring and Managing Celerra Networking, server\_dns, server\_ifconfig, and server\_nis.

## **EXAMPLE #1** To display connectivity from a Data Mover to the outside world, type:

```
$ server_ping server_2 172.24.102.2
server_2 : 172.24.102.2 is alive, time= 0 ms

or

$ server_ping server_2 172.24.102.5
server_2 :
Error 6: server_2 : No such device or address
no answer from 172.24.102.5
```

# **EXAMPLE #2** To display connectivity for a Data Mover to the outside world while sending continuous ECHO\_REQUEST messages, type:

```
$ server_ping server_2 -send 172.24.102.2

server_2:

172.24.102.2 is alive, time= 0 ms

172.24.102.2 is alive, time= 0 ms
```

# **EXAMPLE #3** To display connectivity from a Data Mover to the outside world using the specified interface, type:

```
$ server_ping server_2 -interface cge0 172.24.102.2 server_2 : 172.24.102.2 is alive, time= 0 ms
```

# server\_ping6

Checks the IPv6 network connectivity for the specified Data Movers.

### **SYNOPSIS**

```
server_ping6 {<movername>|ALL}
[-send|-interface <interface>] {<hostname>|<ip6_addr>}
```

### **DESCRIPTION**

**server\_ping6** sends ICMP6\_ECHO\_REQUEST messages to a remote host and receives ICMP6\_ECHO\_REPLIES messages back that verify and determine network connectivity.

As the transmission time is stamped into the ECHO\_REQUEST message, **server\_ping6** displays the transmission delay between the Data Mover and the remote host.

The **ALL** option executes the command for all Data Movers.

## **OPTIONS**

```
<hostname>|<ip6_addr>
```

Provides a one-shot test to determine if the specified <hostname> or <ip6\_addr> is accessible. The IP address may be used if NIS or DNS or both are not operational.

```
-send {<hostname>|<ip6_addr>}
```

Repeats sending/receiving ECHO\_REQUEST messages to a specified <hostname> or <ip6\_addr> for a remote host. The IP address may be used if NIS or DNS or both are not operational. Press **Ctrl C** to quit.

When invoking **server\_ping6** with a hostname, the **/etc/hosts** file is checked first, then the NIS and the DNS servers (if operational on the Data Mover) are checked. If the name does not exist in either case, an error message is displayed.

**Note:** server\_ping6 requests sent to a link-local or multicast destination address require the specification of a '%<intf>' scope identifier.

**<sup>-</sup>interface** {<interface> <hostname> | <ip6\_addr>} Provides a one-shot test to determine if the specified <hostname> or <ip6\_addr> is accessible through the specified <interface>. The IP address may be used if NIS or DNS or both are not operational.

```
EXAMPLE #1
                         To verify Data Mover connectivity from cge0_int1 on server_2 to a client at
                         3ffe:0000:3c4d:0015:0435:0200:0300:00aa, type:
                         $ server_ping6 server_2 -interface cge0_int1
                         3ffe:0000:3c4d:0015:0435:0200:0300:00aa
server_2 : 3ffe:0000:3c4d:0015:0435:0200:0300:00aa is alive, time= 0 ms
                         or
                         $ server_ping6 server_2 -interface cge0_int1
                         3ffe:0000:3c4d:0015:0435:0200:0300:00aa
server_2 :
Error 6: server_2 : No such device or address
no answer from client
          EXAMPLE #2
                         To ping link-local address fe80::260:16ff:fe0c:205%cge0_0000_ll, type:
                         $ server ping6 server 2
                         fe80::260:16ff:fe0c:205%cge0_0000_11
server_2 : fe80::260:16ff:fe0c:205%cge0_0000_11 is alive, time= 0 ms
                         or
                         $ server ping6 server 2
                         fe80::260:16ff:fe0c:205%cge0_0000_11
server_2 :
Error 6: server_2 : No such device or address
no answer from client
          EXAMPLE #3
                         To ping multicast address ff02::1%cge0_0000_ll, type:
                         $ server_ping6 server_2 ff02::1%cge0_0000_11
server_2 : ff02::1%cge0_0000_11 is alive, time= 0 ms
                         or
                         $ server_ping6 server_2 ff02::1%cge0_0000_11
server_2 :
Error 6: server 2: No such device or address
no answer from client
```

# server rip

Manages the Routing Information Protocol (RIP) configuration for the specified Data Movers.

#### **SYNOPSIS**

```
server_rip {<movername> | ALL}
  status
  {{ripin | noripin}} {<interface_name>[,...]}}
```

## DESCRIPTION

server\_rip disables and enables RIP processing for the specified interfaces on a specified Data Mover. **server\_rip** displays the status of the RIP configuration for the specified Data Mover. By default, all the interfaces process RIP messages. server\_setup provides information to start and stop RIP for a Data Mover.

The **ALL** option executes the command for all Data Movers.

#### **OPTIONS** status

Displays the status of the **routed** daemon, RIP processing, and the interfaces with RIP processing disabled.

ripin | noripin { <interface\_name > [, ...] }

ripin enables RIP processing and noripin disables RIP processing for the specified interfaces.

#### **SEE ALSO** server\_setup.

**EXAMPLE #1** To disable an interface for RIP processing, type:

\$ server\_rip server\_2 noripin cge0

server\_2 : done

server\_2 :

**EXAMPLE #2** To display the status of RIP processing, type:

```
$ server_rip server_2 status
routed started, RIP processing is on
    cge0 (172.24.102.237), ifp = 0x8d6a178, disabled
```

**EXAMPLE #3** To enable RIP processing for a specified interface, type:

```
$ server_rip server_2 ripin cge0
```

server\_2 : done

# server\_route

Manages the routing table for the specified Data Movers.

### **SYNOPSIS**

```
server_route {<movername> | ALL}
   -flush
| -DeleteAll
| -list
| {-add | -delete} | default <gateway>
| {-add | -delete} | [host | net] | <dest> <gateway> [<netmask>]
```

#### DESCRIPTION

**server\_route** creates, displays, and deletes routing entries for the specified Data Movers.

The **ALL** option executes the command for all Data Movers.

#### OPTIONS -flush

Temporarily removes all manually configured routing entries for as long as the Data Mover remains up. System generated entries are not removed. A reboot of the Data Mover re-establishes the routing entries that were temporarily removed.

#### -DeleteAll

Permanently removes all the routing entries.

#### -list

Lists all routing entries including the subnet mask for each route.

To display the full routing table for a Data Mover, type: \$ server netstat <movername> -r

```
{-add|-delete} default <gateway>
```

Adds or deletes a default gateway for all unspecified destinations. The Data Mover passes all packets not qualifying against a specific route entry to the specified gateway. The keyword **default** implies a network destination address of 0.0.0.0. A default route uses a fixed mask of 0.0.0.0.

```
{-add|-delete} net <dest> <gateway> [<netmask>] Adds or deletes a routing entry to a network host as specified by the optional keyword net (default) to the specified destination and gateway address. The <dest> argument represents the network for the routing entry, while the <gateway> argument indicates the network gateway to which packets should be addressed.
```

The <netmask> is an optional field for network addresses. In case a netmask is not specified, a netmask will be assigned based on the class of the address and addresses assigned on the system. A <netmask> of 0.0.0.0 or 255.255.255.255 is invalid in the case of **net** routes. The default for the route is **net**.

```
{-add | -delete} host <dest> <gateway>
```

Adds or deletes a routing entry to a particular host as specified by the optional keyword **host** (**net** is the default) to the specified destination and gateway address. The <dest> argument represents the specific host for the routing entry, while the <gateway> argument indicates the network gateway to which packets should be addressed.

By default, a netmask of 255.255.255.255 is assigned to **host** routes.

**SEE ALSO** Configuring and Managing Celerra Networking, server\_netstat, and server\_ifconfig.

**EXAMPLE #1** To list the routing table for **server\_2**, type:

```
$ server route server 2 -list
```

```
server_2 :
net 128.221.253.0 128.221.253.2 255.255.255.0 el31
net 128.221.252.0 128.221.252.2 255.255.255.0 el30
net 172.24.102.0 172.24.102.238 255.255.255.0 cge0
host 127.0.0.1 127.0.0.1 255.255.255.255 loop
```

**Where:** The output displayed is as follows:

<routing type> <desintation> <gateway> <mask> <network interface>

```
EXAMPLE #2 To create a host routing entry, type:
```

```
$ server_route server_2 -add host 10.10.1.2 172.24.102.77
```

server\_2 : done

**EXAMPLE #3** To create a default gateway for all unspecified destinations, type:

\$ server\_route server\_2 -add default 172.24.102.254

server\_2 : done

**EXAMPLE #4** To add a network routing entry with a netmask, type:

\$ server\_route server\_2 -add net 192.162.0.0
172.24.102.238 255.255.0.0

server\_2 : done

#### **The server Commands**

**EXAMPLE #5** To delete a routing entry, type:

\$ server\_route server\_2 -delete host 10.10.1.2

172.24.102.77

server\_2 : done

**EXAMPLE #6** To flush the routing table for a Data Mover, type:

\$ server\_route server\_2 -flush

server\_2 : done

**EXAMPLE #7** To delete all routing entries, type:

\$ server\_route server\_2 -DeleteAll

server\_2 : done

# server\_security

Manages the security policy settings for the specified Data Movers. The Celerra Network Server provides support for Group Policy Objects (GPOs) by retrieving and storing a copy of the GPO settings for each CIFS server joined to a Windows domain.

## **SYNOPSIS**

### **DESCRIPTION**

**server\_security** queries or updates the security policy settings on the Data Movers. The Celerra Network Server stores the GPO settings in a GPO cache on the Data Mover. When you start the CIFS service on a Data Mover, the Celerra Network Server reads the settings stored in the GPO cache, and then retrieves the most recent GPO settings from the Windows domain controller.

The **ALL** option executes the command for all Data Movers.

#### **OPTIONS**

```
-update -policy gpo
```

Forces an update of the specified GPO settings.

```
-info -policy gpo
```

Queries and displays the GPO settings.

```
[server=<server_name>|domain=<domain_name>]
Limits operations to the specified server or domain. The server
refers to the configured CIFS servers on the Data Movers. If no
server or domain has been specified, the operations are
performed to all configured servers.
```

-add -policy chap -name <client\_name>
Adds a new Challenge Handshake Authentication Protocol (CHAP)
security for the specified <client\_name>.

```
-delete -policy chap -name <client_name>
Deletes a CHAP security for the specified <client_name>.
```

```
-modify -policy chap -name <client_name>
Modifies a CHAP security setting for the specified <client_name>.
```

-info -policy chap [-name <client\_name>]

Displays current CHAP security settings for the Data Mover, or the specified <cli>client\_name>.

SEE ALSO

Configuring iSCSI targets on Celerra, Managing Celerra for the Windows Environment, server\_iscsi, and server\_checkup.

**EXAMPLE #1** To query the GPO settings for all configured CIFS servers, type:

\$ server\_security server\_2 -info -policy gpo

```
server 2:
Server compname: dm102-cge0
Server NetBIOS: DM102-CGE0
Domain: nasdocs.emc.com
Kerberos Max Clock Skew (minutes): 5
LAN Manager Auth Level: Not defined
Digitally sign client communications (always): Not defined
Digitally sign client communications (if server agrees): Not defined
Digitally sign server communications (always): Not defined
Digitally sign server communications (if client agrees): Not defined
Send unencrypted password to connect to third-party SMB servers: Not defined
Disable machine account password changes: Not defined
Maximum machine account password age: Not defined
Audit account logon events: Not defined
Audit account management: Not defined
Audit directory service access: Not defined
Audit logon events: Not defined
Audit object access: Not defined
Audit policy change: Not defined
Audit privilege use: Not defined
Audit process tracking: Not defined
Audit system events: Not defined
Back up files and directories: Not defined
Restore files and directories: Not defined
Bypass traverse checking: Not defined
Generate security audits: Not defined
Manage auditing and security log: Not defined
Access this computer from the network: Not defined
Deny access to this computer from the network: Not defined
Take ownership of files or other objects: Not defined
EMC Virus Checking: Not defined
Maximum security log size: Not defined
Restrict quest access to security log: Not defined
Retention period for security log: Not defined
Retention method for security log: Not defined
Maximum system log size: Not defined
Restrict guest access to system log: Not defined
Retention period for system log: Not defined
Retention method for system log: Not defined
Maximum application log size: Not defined
Restrict quest access to application log: Not defined
```

```
Retention period for application log: Not defined
Retention method for application log: Not defined
Disable background refresh of Group Policy: Not defined
Group Policy Refresh interval (minutes): 90
Refresh interval offset (minutes): Not defined
GPO Last Update time (local): Thu Dec 1 13:49:08 EST 2005
GPO Next Update time (local): Thu Dec 1 15:19:08 EST 2005
         EXAMPLE #2
                        To add a new CHAP security for client1, type:
                        $ server_security server_2 -add -policy chap -name
server_2 : Enter Secret: ****
done
         EXAMPLE #3
                        To display CHAP information for client1, type:
                        $ server_security server_2 -info -policy chap -name
                        client1
server_2 :
chapdb name=client1 pass=******
         EXAMPLE #4
                        To update the GPO settings for the CIFS server, type:
                        $ server_security server_2 -update -policy gpo
                        server=dm32-cge0
server_2 : done
         EXAMPLE #5
                        To modify a password for client1, type:
                        $ server_security server_2 -modify -policy chap -name
                        client1
server_2 : Enter New Secret: ****
done
         EXAMPLE #6
                        To delete CHAP security for client1, type:
```

client1

server 2 : done

\$ server\_security server\_2 -delete -policy chap -name

server\_security

# server\_setup

Manages the type and protocol component for the specified Data Movers.

#### **SYNOPSIS**

```
server_setup {<movername>|ALL}
   -type {nas|standby}
| -Load <image_name>
| -Protocol {cifs|mpfs|viruschk|rip|cdms}
   -option {start[=<n>]|stop[,comment=<message>]delete]}
```

#### DESCRIPTION

**server\_setup** sets up the type and protocol component for the specified Data Mover.

The ALL option executes the command for all Data Movers.

#### **OPTIONS**

```
-type {nas|standby}
```

Reboots the Data Mover as the type specified. When setting up a standby relationship between a primary Data Mover and a standby, make sure that the standby Data Mover has the same disk configuration as the primary.

```
-Load <image_name>
```

Loads an image onto a Data Mover. The image may be an emergency patch or debug image provided by EMC. Do not use this option unless directed by EMC customer service.

```
-Protocol {cifs | mpfs | viruschk | rip | cdms}
```

Selects the protocol configuration to be managed. Multiple protocol configurations can be enabled for a Data Mover.

**Note:** NFS and RIP processing are enabled by default. NFS cannot be disabled.

#### -option

Specifies the following options for the specified protocol:

#### start

Starts the specified protocol configuration. To verify whether the protocol configuration has started, use the respective server command for the protocol.

```
[=< n>]
```

Indicates the number of threads for users of CIFS, virus checker, and CDMS. The recommended number of CIFS

threads for each user is based on the amount of memory on the Data Mover. If there is less than 1 GB memory, the default thread number is 96, but if the memory is greater than 1 GB, the default thread number is 256.

**Note:** The [=<n>] option is not available for RIP protocol configurations.

#### stop

Stops the protocol configuration.

```
[,comment=<message>]
```

Allows a message enclosed with quotes to be entered. The message is delivered directly to CIFS clients.

**Note:** The [,comment=<message>] option is not available for RIP or virus checker protocol configurations.

#### delete

Deletes the protocol configuration.

# **SEE ALSO** Configuring Standbys on Celerra, server\_cdms, server\_checkup,

server\_mpfs, server\_rip, server\_standby, and server\_viruschk.

**EXAMPLE #1** To set up a Data Mover as a standby, type:

```
$ server_setup server_3 -type standby
```

server\_3 : reboot in progress 0.0.0.0.0.0.0.0.0.1.3.3.3.3.3.4. done done

**EXAMPLE #2** To set up a Data Mover as nas, type:

```
$ server_setup server_3 -type nas
```

server\_3 : reboot in progress 0.0.0.0.0.0.0.1.1.3.3.3.3.3.4. done
server\_3 : checking root filesystem ... done
done

**EXAMPLE #3** To start the CIFS protocol for a Data Mover, type:

\$ server\_setup server\_2 -Protocol cifs -option start

server\_2 : done

**EXAMPLE #4** To start the virus checker protocol for a Data Mover, type:

\$ server\_setup server\_2 -Protocol viruschk -option start

server\_2 : done

**EXAMPLE #5** To stop the CIFS protocol for a Data Mover, type:

\$ server\_setup server\_2 -Protocol cifs -option

stop,comment="CIFS is going down"
server\_2 : done

**EXAMPLE #6** To delete the CIFS protocol configuration for a Data Mover, type:

\$ server\_setup server\_2 -Protocol cifs -option delete

server\_2 : done

Warning 4037:  $server_2$ : CIFS configurations for  $server_2$  and VDM(s) currently loaded on it

**EXAMPLE #7** To load a new image onto a Data Mover, type:

\$ server\_setup server\_2 -Load nas.exe

server\_2: will load : nas.exe

# server\_snmpd

Manages the Simple Network Management Protocol (SNMP) configuration values for the specified Data Movers.

#### **SYNOPSIS**

```
server_snmpd {<movername> | ALL }
      -info
      -service {
          -status
          -stop
          -start
      }
      -modify [-location <sys_location>]
               [-contact <sys_contact>]
               [-community {-clear | < community>}]
      -user {
          -list
         -create <name> -authpw -privpw
          -delete <name>
          -modify <name> -authpw -privpw
      }
```

#### **DESCRIPTION**

**server\_snmpd** controls SNMP server agent and SNMP configuration for the specified Data Mover. The service can be started/stopped depending on user preferences; the location and contact can be configured. SNMPv1 community strings can be created and cleared. SNMPv3 users can be created or deleted, and modified. The **ALL** option executes the command for all Data Movers.

## OPTIONS -info

Displays information about the SNMP server agent and configuration such as location, contact, and community.

```
-service {-status| -stop| -start} Starts, stops, or displays the current SNMP service.
```

#### -modify

Modifies the configuration of the SNMP service.

```
-location <sys_location>
Modifies the location value.
-contact <sys_contact>
```

Modifies the syscontact value.

server\_2:

server\_2: enabled = yeslocation = RTP, NC contact = John Doe community = public

server\_3: enabled = no

server\_2:

OK

contact = Pandu

enabled = yeslocation = RTP, NC contact = John Doe community = public

```
-community {-clear | <community>}
                             Modifies the community name being used by the server SNMPD
                             agent and enables the SNMPv1/SNMPv2c access to the agent.
                             The -clear option allows the adminstrator to disable the
                             community.
                          -user {-list}
                          Lists the SNMPv3 users on the Data Mover.
                             -create <name> -authpw -privpw
                             Creates an SNMPv3 user on the Data Mover.
                             -delete <name>
                             Deletes an SNMPv3 user on the Data Mover.
                             -modify <name> -authpw -privpw
                             Modifies the SNMPv3 passwords of a user on the Data Mover.
                             Both the passwords should be changed together.
                          To view the SNMP service properties of the Data Mover server_2,
                          type:
                          $ server_snmpd server_2 -info
                          To view the SNMP service properties of all Data Movers, type:
                          $ server snmpd ALL -info
location = Bangalore, India
community = testcommunity
                         To start the SNMP service on the Data Mover server_2, type:
                          $ server_snmpd server_2 -service -start
```

**EXAMPLE #4** To stop the SNMP service on the Data Mover **server 3**, type:

**EXAMPLE #3** 

**EXAMPLE #1** 

**EXAMPLE #2** 

```
$ server_snmpd server_2 -service -stop
server_3:
OK
          EXAMPLE #5
                         To view the SNMP service status for all Data Movers, type:
                         $ server_snmpd ALL -service -status
server_2:
SNMP Running
server_3:
SNMP Stopped
          EXAMPLE #6
                         To modify the location, contact, and community values on Data
                         Mover server_2, type:
                         $ server_snmpd server_2 -modify -location "RTP, NC"
                         -contact "Anamika Kadian" -community public
server_2:
OK
          EXAMPLE #7
                         To clear the community value on Data Mover server_2, type:
                         $ server_snmpd server_2 -modify -community -clear
server_2:
OK
          EXAMPLE #8
                         To display the list of SNMPv3 users on all Data Movers, type:
                         $ server_snmpd ALL -user -list
server_2:
user
gsmith
smith
server 3:
user
clowe
          EXAMPLE #9
                         To create a new user John, on Data Mover server_2, type:
                         $ server_snmpd server_2 -user -create john -authpw
                         -privpw
Enter the authentication password:*******
Confirm the authentication password: *******
Enter the privacy password: *******
Confirm the privacy password: *******
server_2:
OK
        EXAMPLE #10
                         To delete the user John, on Data Mover server_2, type:
                         $ server_snmpd server_2 -user -delete John
```

#### **The server Commands**

```
server_2:
```

# **EXAMPLE #11**

To modify the passwords of the user John, on Data Mover **server\_2**, type:

```
$ server_snmpd server_2 -user -modify John -authpw
-privpw
```

Enter the authentication password:\*\*\*\*\*\*\*
Confirm the authentication password:\*\*\*\*\*\*\*
Enter the privacy password:\*\*\*\*\*\*\*
Confirm the privacy password:\*\*\*\*\*\*
server\_2:
OK

# server\_ssh

Manages and configures the SSH server on the specified Data Mover.

#### **SYNOPSIS**

```
server_ssh serverX
      -info
     -start
      -stop
      -modify {
       -banner <banner file>
       -port <port number>
       -threads <number>
       -defaultdir <path> |
       -restricthomedir {enable|disable} |
       -application <application_name>
       -cipher <keyword>
       -compression {yes|delayed|no} |
       -mac <keyword> |
       -maxauthtries <number> |
       -passwordauthentication {enable | disable}
       -permitemptypassword {enable|disable}
       -pubkeyauthentication {enable|disable} |
       -timeout <seconds> }
     -append {
       -allowhosts <ip>[,<ip>...]
       -allowgroups <group>[, <group>...]
       -allowusers <user>[,<user>...]
       -denyhosts <ip>[,<ip>...]
       -denygroups <group>[, <group>...]
       -denyusers <user>[, <user>...] }
     -remove {
       -allowhosts <ip>[,<ip>...]
       -allowgroups <group>[, <group>...]
       -allowusers <user>[,<user>...]
       -denyhosts <ip>[,<ip>...]
       -denygroups <group>[,<group>...]
       -denyusers <user>[, <user>...] }
      -generate [-type {rsa | dsa} [-keysize <bits>]]
      -delete [-type {rsa|dsa}]
```

#### OPTIONS -info

Displays the current configuration and state of the SSH server.

#### -start

Starts the SSH server. Allocates the pools of SSH threads and applications threads, and listens to incoming client requests from the network.

If the SSH server is already active, an error is returned.

#### -stop

Stops the SSH server. Stops listening to incoming client requests, and releases all the SSH threads.

If the SSH server is inactive, an error is returned.

#### -modify

Modifies some configuration parameters of the SSH server. The arguments are:

#### [-banner <banner file>]

The file name specified should contain an absolute path on the Data Mover. The file contains an unformatted text, which is sent to the SSH clients during the authentication step of the connection. It is a welcome message. The file name is saved in the SSH configuration file on the Data Mover. If the file name is " " or an empty string, then the current file name is removed from the configuration. If the file does not exist or is unreadable, an error message is returned.

#### [-port <number>]

Changes the TCP lists port of the SSH server. The server must be stopped before executing this command. The SSH server listens on only one TCP port. This port number is saved in the SSH configuration file on the Data Mover. If this port is invalid, or already used by another active service of the Data Mover, an error message is returned.

#### [-threads <number>]

Allocates the number of threads dedicated to the SSH server. If the server is started, the threads are restarted by this action, and all the established SSH connections are disconnected. Ten default threads are started when the server starts. The supported values are from 4 to 128 threads.

#### [-defaultdir <path>]

Specifies the default directory of an authenticated user when the home directory is undefined or inaccessible. If the given path does not exist or is inaccessible, then an error message is returned. The default value is /or forward slash.

## [-restricthomedir {enable|disable}]

Restricts authenticated users to their home directory tree, to prevent them from accessing other user's directories. If their home directory is undefined or unreachable, the login is denied. This feature is enabled, by default, for security reasons.

#### [-application <application\_name>]

Enables or disables the specified SSH application. The names of applications are case insensitive. If an SSH client tries to execute a disabled application, the Data Mover returns an error. The applications supported by the SSH server in VNX version 7.0 are SFTP and SCP. The default applications are disabled. To disable an enabled application, use the same option with the name no<application>. For instance, use the **-application nosftp** option to disable SFTP.

### [-cipher <keyword>]

Specifies the symmetric ciphers allowed to encrypt an SSH connection. The openSSH documentation or http://www.openssh.com provides more information on the supported ciphers algorithms. The default option is undefined, which means all these ciphers are allowed.

#### -compression {yes|delayed|no}

Specifies whether the compression of data inside an SSH connection is allowed, disallowed, or delayed.

The argument must be:

- yes: always compressed, if possible
- delayed: compressed after the user has authenticated successfully, if possible
- no: compression allowed

The default is delayed.

#### -mac <keyword>

Specifies the available Message Authentication Code or MAC algorithms to guarantee the integrity of the SSH packets on the network. The default value is undefined, which means all these algorithms are allowed.

#### -maxauthtries <number>

Specifies the maximum number of authentication attempts permitted per SSH connection. The supported values are between 1 and 99. The default value is 6.

#### -passwordauthentication {enable|disable}

Specifies whether password authentication is allowed during the establishment of an SSH connection. The parameters passwordAuthentication and pubkeyAuthentication cannot be disabled simultaneously. The default value is enabled.

#### -permitemptypassword {enable|disable}

When password authentication is allowed, it specifies whether the server allows log in to accounts with empty password strings. The supported values are: disabled or not permitted, and enabled or permitted. The default value is disabled.

#### -pubkeyauthentication {enable | disable}

Specifies whether public key authentication is allowed during the establishment of an SSH connection. The options passwordAuthentication and pubkeyAuthentication cannot be disabled simultaneously.

The default value is enabled.

#### -timeout <seconds>

Specifies the time out in seconds before a network disconnection in case of inactivity. The default value is 900 seconds or 15 minutes. The range is from 10 to 3600 seconds.

#### -append

Appends to the current configuration new allowed or denied host/group/user. The arguments are:

#### -allowhosts <ip>[,<ip>...]

Defines new client IPs or client subnets that are allowed to connect through SSH to the Data Mover. IP is the IPv4 or IPv6 address or subnet. In case of subnets, the CIDR notation must be used (for instance: 192.168.0.0./16 or 2001:DB8::/48). Network names are ignored.

If specified, SSH connections are allowed only for clients if their IP matches one of the defined allowed hosts, that is, any client host not declared on that list receives systematically an access denied error. If a subnet is defined on the allowed hosts list, then the client IP must belong to this subnet to be allowed to connect to the Data Mover.

If no allowed host is defined, then all client hosts are allowed to connect by default.

The list of the allowed host is saved in the SSH configuration file on the Data Mover. Duplicate items of the allowed host are prohibited. The Data Mover can save up to 256 different allowed hosts.

#### -allowgroups <group>[,<group>...]

Defines new groups of users that are allowed to connect through SSH to the Data Mover. The group should be the name of the allowed group and numerical group ID are ignored.

If specified, SSH connections are allowed only for users whose primary or supplementary group list matches one of the defined allowed groups, which means any user not a member of one of these groups receives an access denied error.

If no allowed group is defined, then all users of the groups are allowed to connect by default.

The list of allowed groups is saved in the SSH configuration file on the Data Mover.

Duplicate names of an allowed group are prohibited.

The Data Mover can save up to 256 different allowed groups.

```
-allowusers <user>[, <user>...]
```

Defines new users that are allowed to connect through an SSH server to the Data Mover. User ID should be the name of a user; numerical user IDs are ignored. If the user is a CIFS user, the format should be user@domain or domain\user.

If specified, SSH connections are allowed only for user names that match one of the patterns, which means any user not listed as an allowed user receives an access denied error.

If no allowed user is defined, then all users are allowed to connect by default.

The list of the allowed users is saved in the SSH configuration file on the Data Mover.

Duplicate names of allowed users are prohibited.

The Data Mover can save up to 256 different allowed users.

```
-denyhosts <ip>[,<ip>...]
```

Defines new client IPs or client subnets that are disallowed to connect through th SSH server to the Data Mover. IP is the IPv4 or IPv6 address or subnet (CIDR notation) of the client host. Network names are ignored.

If specified, SSH connections are disallowed for clients whose IP address matches one of the defined denied host or IP addresses belonging to a defined subnet, which means any client host declared in that list receives systematically an access denied error.

If no denied host is defined, then all client hosts are allowed to connect. This is the default.

The list of the denied hosts is saved in the SSH configuration file on the Data Mover.

Duplicate items of denied hosts are prohibited.

The Data Mover can save up to 256 different denied hosts.

## -denygroups <group>[,<group>...]

Defines new groups of users that are disallowed to connect through SSH to the Data Mover. 'group' should be the name of the group and numerical group IDs are ignored.

If specified, SSH connections are disallowed for users whose primary group or supplementary group list matches one of the defined "denied group", i.e. any user member of one of these groups receives systematically an "Access denied" error.

If no "denied group" is defined, then all users of all groups are allowed to connect. This is the default.

The list of the denied groups is saved in the SSH configuration file on the Data Mover.

Duplicate names of "denied group" are prohibited.

The Data Mover can save up to 256 different denied groups.

#### -denyusers <user>

Defines a new user that is disallowed to connect through SSH to the Data Mover. "user" should be the name of the user and numerical user IDs are ignored.

If the user is a CIFS user, his format should be user@domain or domain\user.

If specified, SSH connections are disallowed for user names that match one of the patterns, that is any user listed as a denied user receives systematically an access denied error.

If no denied user is defined, then all users are allowed to connect. This is the default.

The list of the denied users is saved in the SSH configuration file on the DART.

Duplicate names of denied users are prohibited.

The Data Mover can save up to 256 different denied users.

#### -remove

Removes from the current configuration an allowed/denied group/user. The arguments are:

```
-allowhosts <ip>[,<ip>...]
```

Removes the specified IPs or subnets from the allowed hosts list.

```
-allowgroups <group>[,<group>...]
```

Removes the specified groups from the allowed groups list.

```
-allowusers <user>[, <user>...]
```

Removes the specified users from the allowed users list.

```
-denyhosts <ip>[,<ip>...]
```

Removes the specified IPs or subnets from the denied hosts list.

```
-denygroups <group>[, <group>...]
```

Removes the specified groups from the denied groups list.

```
-denyusers <user>[,<user>...]
```

Removes the specified users from the denied users list.

#### -generate

Generates new public and private keys and saves it as the host key of the Data Mover. This operation must be the first step during the setup procedure of SSH. It could take some minutes to complete.

By default, two host keys are generated: a 2048-bit RSA key and a 1024-bit DSA key.

If the host keys of the Data Mover do not exist when the SSH server is started the first time, the keys are automatically generated with the default parameters.

The host keys are stored in the /.etc directory of the Data Mover, in an inaccessible directory.

The options are:

-type {rsa|dsa}: Specifies the type of key to generate. The two valid types are: rsa and dsa. If not specified, both keys types are generated.

**-keysize** <br/> <br/> <br/> Specifies the number of bits of the host key. Default is 2048 bits for RSA keys and 1024 bits for DSA keys.

Valid range for RSA keys is between 2048 and 8192 bits.

The length of a DSA key must be 1024 bits exactly, as specified by FIPS 186-2.

**-delete**: Deletes all the generated host keys of the Data Mover. This command is useful if the administrator needs to generate new host keys.

The options are:

**-type** {rsa | dsa }: This argument specifies the type of key to delete. The two valid types are: rsa and dsa. If not specified, both key types are deleted.

#### **GENERAL NOTES**

- The allow or deny directives are processed in the following order: denyhost, allowhost, denyusers, allowusers, denygroups and finally allowgroups.
- The allow or deny directives can specify multiple items separated by a , or comma without spaces.
- It is recommended to encapsulate IPv6 address in [] or square brackets.
- The host keys are generated automatically when the SSH server is started for the very first time if no host key exists. In that case, both keys are generated with their default size (RSA and DSA keys each).
- VDM are not supported by the server\_ssh command.

#### EXAMPLE #1

To display the current configuration, type:

\$ server\_ssh server\_2 -info

```
server_2 : done
```

#### SERVICE CONFIGURATION

Port : 22
State : running
Thread count : 4
Ranner : /server

Banner : /server2fs1/banner.txt

Default home directory : /

Restrict home directory : disabled Application : sftp,scp Cipher : <all>
Compression : delayed

```
Mac : <all>
Max authentication tries : 5
Password authentication : enabled
Permit empty password : disabled
Public key authentication : enabled
Timeout : 30 seconds
```

**EXAMPLE #2** To display the current configuration including allow and deny directives, type:

\$ server\_ssh server\_3 -info

```
server_3 : done
```

SERVICE CONFIGURATION

Port : 22 State : running Thread count : 16

Banner : /fs40/banner.txt

Default home directory : /

Restrict home directory : disabled Application : sftp,scp Cipher : <all> : no Compression : <all> Mac Max authentication tries : 10 Password authentication : enabled : disabled Permit empty password Public key authentication : enabled Timeout : 900 seconds

ACCESS CONTROL

Allowed IPs : <all>

Denied IPs : 192.168.34.23,192.168.34.10

Allowed users : jack,tony,angela,root,sheila,Monica

Denied users : <none>
Allowed groups : <all>
Denied groups : <none>

**EXAMPLE #3** To start the SSH server, type:

\$ server\_ssh server\_3 -start

server\_3 : done

**EXAMPLE #4** To stop the SSH server, type:

\$ server\_ssh server\_3 -stop

server\_3 : done

**EXAMPLE #5** To update the number of parameter threads, type:

\$ server\_ssh server\_3 -modify -threads 16

server\_3 : done

**EXAMPLE #6** To remove an allowed group, type: \$ server\_ssh server\_3 -remove -allowgroups guests server\_3 : done **EXAMPLE #7** To modify the banner file, type: \$ server\_ssh server\_2 -modify -banner /server2fs1/banner.txt server\_2 : done Note: The file /server2fs1/banner.txt must exist for the Data Mover point of view. **EXAMPLE #8** To modify the timeout to five minutes, type: \$ server\_ssh server\_2 -modify -timeout 300 server 2 : done **EXAMPLE #9** To add a new allowed user defined in NIS or LDAP, type: \$ server\_ssh server\_2 -append -allowusers john server\_2 : done **EXAMPLE #10** To add a new allowed user defined in the **dom10** Windows domain, type: \$ server\_ssh server\_2 -append -allowusers dom10\\cindy server 2 : done **EXAMPLE #11** To add a new allowed group of users, type: \$ server\_ssh server\_2 -append -allowgroups admin server\_2 : done **EXAMPLE #12** To add a new allowed client IP, type: \$ server\_ssh server\_2 -append -allowhosts 110.171.1.10 server\_2 : done **EXAMPLE #13** To add new allowed client hosts using their subnet, type: \$ server\_ssh server\_2 -append -allowhosts 110.121.0.0/16 server\_2 : done EXAMPLE #14 To add a new denied user, type: \$ server\_ssh server\_2 -append -denyusers john server\_2 : done **EXAMPLE #15** To add a new denied group of users, type:

\$ server\_ssh server\_2 -append -denygroups guest

server\_2 : done

**EXAMPLE #16** To add a new denied client IP, type:

\$ server\_ssh server\_2 -append -denyhosts 110.171.1.54

server\_2 : done

**EXAMPLE #17** To generate a new host key for the Data Mover, type:

\$ server\_ssh server\_2 -generate

server\_2 : done

**Note:** This operation may take a long time. The SSH server must be stopped as the Data Mover does not have existing host keys.

**EXAMPLE #22** To delete the existing host keys of the Data Mover, type:

\$ server\_ssh server\_2 -delete

server\_2 : done

**Note:** The SSH server must be stopped.

**ERROR CASE #1** To change the banner file to a non-existing file, type:

\$ server\_ssh server\_2 -modify -banner foo

server\_2 :

Error 13163823109: server\_2 : Invalid SSH configuration: Invalid banner file name.

**ERROR CASE #2** To enable an unknown application on top of SSH, type:

\$ server\_ssh server\_2 -modify -application foo

server\_2 :

Error 13163823110: server\_2 : Invalid configuration value for the SSH server: Unknown application 'foo'...

**ERROR CASE #3** To change the number of SSHD threads to an unauthorized value, type:

\$ server\_ssh server\_2 -modify -threads 256

server 2 :

Error 13163823110: server\_2 : Invalid configuration value for the SSH server: Bad threads value specified, allowed range is (4-128).

**ERROR CASE #4** To change the port of the SSH server to an already used port, type:

\$ server\_ssh server\_2 -modify -port 445

server\_2 :

Error 13163823111: server\_2: The SSH server cannot bind the TCP port 445.

Note: 445 is used by the CIFS server.

## **ERROR CASE #5**

To regenerate the Data Mover host keys while the SSH server is active, type:

\$ server\_ssh server\_2 -generate

#### server\_2 :

Error 13163823112: server\_2 : The SSH server must be stopped before executing this command.

## **ERROR CASE #6**

To generate the Data Mover host key, and specify an invalid key type, type:

\$ server\_ssh server\_2 -generate -type foo

#### server\_2 :

Error 13163823109: server\_2 : Invalid SSH configuration: Bad KEYTYPE value attribute.

#### **ERROR CASE #7**

To generate the Data Mover host key, and specify an RSA key with an invalid size, type:

\$ server\_ssh server\_2 -generate -type RSA -keysize 23

#### server\_2 :

Error 13163823110: server\_2 : Invalid configuration value for the SSH server: Bad keysize value specified.

### **ERROR CASE #8**

To generate the Data Mover host key as a DSA key when a key of this type already exists, type:

\$ server\_ssh server\_2 -generate -type dsa

#### server 2 :

Error 13163823123: server\_2 : The command failed as the DSA host key is already defined.

# server\_standby

Manages the standby and RDF relationships for the specified Data Movers.

#### **SYNOPSIS**

```
server_standby {<movername>|ALL}
-activate {mover|rdf}
-restore {mover|rdf}
-delete {mover[=<movername>]|rdf}
-verify {mover}
-create {mover=<source_movername> [-policy <policy_type>]|
rdf[=<slot number>]}
```

#### DESCRIPTION

**server\_standby** manages the standby and RDF relationship for the specified Data Mover.

The **ALL** option executes the command for all Data Movers.



#### **CAUTION**

The ALL option should not be used when creating, restoring, or activating standby or RDF Data Movers.

## **OPTIONS**

```
-activate {mover|rdf}
```

Activates the standby Data Mover or the RDF Data Mover for the primary Data Mover. Before activating the standby, the reboot caused by creating the standby relationship must be complete.

```
-delete {mover[=<movername>] | rdf}
```

Deletes relationship between the standby and primary Data Mover or deletes the RDF Data Mover for the primary Data Mover. If you have more than one standby for a primary, you can specify which standby is to be deleted by entering a <movername>. server\_setup provides information to change the type defined for a Data Mover.

#### -restore {mover|rdf}

Restores the standby Data Mover or the RDF Data Mover for the primary Data Mover once it has been activated.

```
-verify {mover}
```

Verifies the standby status of the Data Mover after a hardware replacement. If the Data Mover hardware has been replaced, the reboot cycle may not be displayed.

```
-create {mover=<source_movername> [-policy
<policy_type>]}
```

Designates the <source\_movername> as the standby. The following rules apply when selecting a standby Data Mover:

- The primary Data Mover cannot be a standby Data Mover.
- A standby cannot be created for a standby.
- No filesystems can be mounted on the standby.
- The standby must have the same disk configuration as the primary.
- One standby Data Mover can be created for multiple Data Movers. To do this, execute server\_standby -create for each primary Data Mover.

```
-policy <policy_type>
```

When a fault is detected on the primary Data Mover, the following **-policy** options apply:

#### auto

Initiates immediate activation of the standby Data Mover.

#### retry

Attempts to recover the primary Data Mover first, then if recovery fails, initiates activation of the standby.

```
manual (default)
```

Reboots the primary Data Mover. No action on the standby is initiated.

```
-create rdf[=<slot_number>]]
```

Designates a remote RDF standby Data Mover as indicated by its slot number.

**SEE ALSO** *Configuring Standbys on Celerra* and server\_setup.

#### **EXAMPLE #1**

To create a standby relationship between **server\_2** (primary) and **server\_3** (standby), type:

```
$ server_standby server_2 -create mover=server_3
```

server\_2 : server\_3 is rebooting as standby

**Note:** Before any other actions can take place, the reboot must be complete.

```
type:
                         $ server_standby server_2 -activate mover
server_2 :
server_2 : going offline
server_3 : going active
replace in progress ...done
 failover activity complete
commit in progress (not interruptible)...done
server_2 : renamed as server_2.faulted.server_3
server_3 : renamed as server_2
         EXAMPLE #3
                        To restore server 3 as the standby Data Mover and
                        server_2.faulted.server_3 as the primary, type:
                         $ server_standby server_2 -restore mover
server_2 :
server_2 : going standby
server_2.faulted.server_3 : going active
replace in progress ...done
 failover activity complete
commit in progress (not interruptible)...done
server_2 : renamed as server_3
server_2.faulted.server_3 : renamed as server_2
         EXAMPLE #4
                        To verify readiness of the standby Data Mover, type:
                         $ server_standby server_2 -verify mover
server_2 : ok
         EXAMPLE #5
                        To delete the standby relationship for server_2, type:
                         $ server_standby server_2 -delete mover
server_2 : done
         EXAMPLE #6
                        To create a standby relationship for three Data Movers, type:
                         $ server_standby server_2 -create mover=server_3
server_2 : server_3 is rebooting as standby
                         $ server_standby server_4 -create mover=server_3
server 4 : done
                         $ server_standby server_5 -create mover=server_3
server_5 : done
                        Note: Before any other actions can take place, the reboot must be complete.
```

To activate the **server\_3** (standby) to take over for **server\_2** (primary),

**EXAMPLE #2** 

# server\_stats

Displays sets of statistics that are running on the specified Data Mover.

#### **SYNOPSIS**

```
server_stats <movername>
      -list
      -info [-all|<statpath_name>[,...]]
      -service { -start [-port <port_number>]
                  -stop
                  -delete
                  -status }
      -monitor -action {status|enable|disable}
         [{ -monitor {statpath_name|statgroup_name}[,...]
            -monitor {statpath_name | statgroup_name}
            [-sort <field_name>]
            [-order {asc|desc}]
            [-lines <lines_of_output>]
         [-count <count>]
         [-interval <seconds>]
         [-terminationsummary {no|yes|only}]
         [-format {text [-titles {never|once|repeat}]|csv}]
         [-type {rate | diff | accu}]
         [-file <output_filepath> [-overwrite]]
```

#### **DESCRIPTION**

server\_stats provides real-time statistics for the specified Data Mover. Statistics are displayed in a time-series fashion at the end of each interval. The statistics are organized as a hierarchy of families that replaces the summary and table collections. Previously defined summary and table collections have been converted to system-defined statistics groups that can be used as arguments with the new -monitor option:

System-defined statistics group name	server_stats collection name
basic-std	-summary basic
caches-std	-summary caches
cifs-std	-summary cifs
nfs-std	-summary nfs
cifsOps-std	-table cifs
diskVolumes-std	-table dvol

System-defined statistics group name	server_stats collection name
metaVolumes-std	-table fsvol
netDevices-std	-table net
nfsOps-std	-table nfs

**Note:** The -summary and -table options are deprecated and will not be supported in future releases.

In addition, server\_stats manages the Statistics Monitoring service (statmonService) running on Data Movers including the ability to disable and enable statistics.

### **OPTIONS** No arguments

Displays a basic summary of statistics for the specified Data Mover as defined by the basic-std Statistics Group.

#### -list

Displays all defined statistics starting with the statgroup names followed by statpaths and their types.

#### -info

Displays the statgroup and statpath information.

#### -service

Specifies whether to start, stop, delete, or query the status of the statmonService. The statmonService runs on the Data Mover and listens or the server\_stats requests.

#### [-start]

Starts the statmonService on the Data Mover. If the -port argument is specified, it is used by statmonManager service. These settings are persistent and execute as part of the Data Mover's boot-up configurations.

#### [-stop]

Shuts down the statmonService on the specified Data Mover.

#### [-delete]

Deletes the statmonService persistent configurations so it does not execute as part of the Data Mover's boot-up settings. If -delete is executed while the statmonService is running, the service stops and its configuration is deleted.

#### [-status]

Checks the status of the statmonService on the specified Data Mover.

#### -monitor [-action]

Enables, disables, or queries the state of the stats collection.

#### -monitor {statpath\_name|statgroup\_name}

Takes a comma-separated list of statpath and statgroup names. In cases where stats are available for multiple elements, the user can specify an element name or use 'ALL-ELEMENTS' to refer to all elements at once.

Any duplicate statpath or statgroup names is consolidated and reported once. The options below are only applicable to Set and Correlated Set statpath names:

```
[-sort <field_name>]
```

Sorts a Set based on <field name> values.

```
[-order {asc|desc}]
```

Sorted Sets can be ordered in ascending or descending order.

```
[-lines <lines_of_output>]
```

Sets output can be limited to display the Set statistics lines\_of\_output> lines.

Set statistics is not sorted if none of the above three options are specified. If some of these options are specified with a Set statpath name, the omitted options default as follows:

- -sort defaults to the element name
- -lines defaults to all
- -order default value depends on the -sort field. If the Set is being sorted on a numeric field, -order defaults to descending order; otherwise, it defaults to ascending order. For sorting based on the Set element name, -order always defaults to ascending order.

Each Correlated Set are defined with a default sort field and order. If some of these options are specified with a Set statpath name, the omitted options default as follows:

- -sort default field is unique for each Correlated Set
- -lines always defaults to 20 for all Correlated Sets
- -order default value depends on the -sort field. If the Correlated Set is being sorted on a numeric field, -order defaults to descending order; otherwise, it defaults to ascending order. Correlated Sets cannot be sorted on non numeric fields, including the Correlated Set element ID.

#### [-count <count>]

Specifies the number of reports that are generated. Reports are produced after each set interval of time. The <count> must be a positive integer. By default, reports are generated indefinitely till the user presses Ctrl-C.

#### [-interval <seconds>]

Specifies the number of seconds between samples. The <seconds> must be an integer, 1 through 300 or higher. The default is 15.

#### [-terminationsummary {no|yes|only}]

Enables, disables, or manages the termination summary that appears after the <count> lines have completed or a Ctrl-C interrupt is executed. All items, both active and inactive, are reported. The summary reports average, minimum, and maximum values for numeric fields collected over the complete run. The **no** and **yes** choices control the summary message. The **only** option suppresses the reporting of the time series and reports only the summary. The default value is **yes**.

#### [-format {text | csv}]

Sets the format for printing statistics. The text (default) format prints statistics separated by a sequence of blank characters when rendered in a fixed-sized font. The time stamp that appears on the front of each snapshot is local time in the form hh:mm:ss.

The **csv** format supports the display of elements in dynamic Set statistic. A dynamic collection is a Set statistic where new elements are reported in any sample. In order to support this dynamic behavior for the csv format and keep the csv output consistent between the title line and the sample data lines, the title line is reprinted with samples where new elements are detected.

To avoid seeing multiple title lines in the **csv** output, use the **-file** option that saves the output to a specified file instead of outputting to the standard output.

The advantage of using the **-file** option is that server\_stats prints samples to the file and does not print the title line except once at the end of the session when the title line is complete. In this case, the title line appears once in the output at the top of the file.

The **csv** format produces output in a comma-separated list and renders the local time in a yyyy/mm/dd hh:mm:ss format.

The csv output and the -file option:

- Without the -file option: The output to stdout may show multiple title lines where new elements were detected.
- With the -file option: The output is printed to the specified file
  so at the end of the session, the final titles line is printed once
  at the top of the file. The result is a cleaner output than
  multiple titles lines where new elements were detected.

#### [-titles {never|once|repeat}]

Controls the generation of titles. The **never** option indicates that titles are not printed, the **once** option indicates that they are only printed at the top of the report, and the **repeat** option indicates that they are repeated for each time-series snapshot. The default setting is **once**.

The -titles option cannot be combined with the **-format csv** option since titles could be dynamically updated based on the dynamic nature of some statistics. The **-format** option provides more information.

#### [-type {rate|diff|accu}]

Specifies the display type of value for statistics with monotonically increasing values. The display type applies to statistics that increase monotonically, for example, network in-bound bytes. Other statistics that represent a point-in-time value, for example, current CIFS connections, are not affected by this option. The **rate** value displays the rate of change since the previous sample, the **diff** value displays the change in value since the previous sample, and the **accu** value displays the change in value since the initial sample. The default display type is **rate**.

#### [-file <output\_filepath> [-overwrite]]

Allows the users to specify a filename to save the output of session instead of printing to standard output (stdout). This is necessary to provide a cleaner csv output. Since the titles line continues to change each time a new element is detected, it is printed a number of times to stdout. However, with the **-file** option, the titles line appears once at the top of the output.

In case <output\_filepath> exists, the results will be appended to it. The **-overwrite** argument can be used to change this behavior so that an existing file is deleted before the results are written to it.

The argument of the **-file** option is a valid file name and does not have a default argument.

**SEE ALSO** *Managing Celerra Statistics.* 

**EXAMPLE #1** To display the basic-std group by indicating the rate of change since the previous sample, type:

# \$ server\_stats server\_2 -monitor basic-std -interval 5 -count 5 -type rate

server_2	CPU	Network	Network	dVol	dVol	
Timestamp	Util	In	Out	Read	Write	
	8	KiB/s	KiB/s	KiB/s	KiB/s	
02:53:09	33	88618	729	9988	89730	
02:53:11	61	142057	1087	18632	167076	
02:53:13	29	98110	762	8976	78599	
02:53:15	46	109783	836	14288	125717	
02:53:17	34	111847	886	10652	95719	
server_2	CPU	Network	Network	dVol	dVol	
Summary	Util	In	Out	Read	Write	
	용	KiB/s	KiB/s	KiB/s	KiB/s	
Minimum	29	88618	729	8976	78599	
Average	41	110083	860	12507	111368	
Maximum	61	142057	1087	18632	167076	

#### Where:

Value	Definition	
Timestamp	Time the poll was taken.	
CPU Util	Disk utilized in percentage in this interval.	
Network In KiB/s	Network kibibytes received over all network interfaces.	
Network Out KiB/s	Network kibibytes sent over all network interfaces.	
dVol Read KiB/s	Storage kibibytes received from all server-storage interfaces.	
dVol Write Kib/s	Storage kibibytes sent to all server-storage interfaces.	

# **EXAMPLE #2** To display the basic-std group by indicating the change in value since the previous sample, type:

# \$ server\_stats server\_2 -monitor basic-std -interval 5 -count 5 -type diff

CPU	Network	Network	dVol	dVol
Util	In KiB	Out KiB	Read KiB	Write
૪	diff	diff	diff	KiB diff
46	267660	2136	26128	232654
38	200668	1543	23144	211182
46	226761	1749	26488	230558
48	246921	1876	28720	255957
40	212353	1673	23016	210573
CPU	Network	Network	dVol	dVol
Util	In KiB	Out KiB	Read KiB	Write
용	diff	diff	diff	KiB diff
38	200668	1543	23016	210573
44	230873	1795	25499	228185
	Util % 46 38 46 48 40 CPU Util % 38	Util In KiB % diff 46 267660 38 200668 46 226761 48 246921 40 212353  CPU Network Util In KiB % diff 38 200668	Util         In KiB         Out KiB           %         diff         diff           46         267660         2136           38         200668         1543           46         226761         1749           48         246921         1876           40         212353         1673           CPU         Network         Network           Util         In KiB         Out KiB           %         diff         diff           38         200668         1543	Util         In KiB         Out KiB         Read KiB           %         diff         diff         diff           46         267660         2136         26128           38         200668         1543         23144           46         226761         1749         26488           48         246921         1876         28720           40         212353         1673         23016           CPU         Network         Network         dVol           Util         In KiB         Out KiB         Read KiB           %         diff         diff         diff           38         200668         1543         23016

Maximum

48

267660

2136 28720 255957

#### Where:

Value	Definition
Timestamp	Time the poll was taken.
CPU Util %	Disk utilized in percentage in this interval.
Network In KiB diff	Network kibibytes received over all network interfaces per differential value.
Network Out KiB diff	Network kibibytes sent over all network interfaces per differential value.
dVol Read KiB diff	Storage kibibytes received from all server-storage interfaces per differential value.
dVol Write KiB diff	Storage kibibytes sent to all server-storage interfaces per differential value.

#### **EXAMPLE #3** To display the basic-std group by indicating the change in value since the first sample, type:

## \$ server\_stats server\_2 -monitor basic-std -interval 5

		-count s	-type accu		
server_2	CPU	Network In KiB	Network Out KiB	dVol Read KiB	dVol Write KiB
Timestamp	Util				
	용				
02:53:48	42	236257	1880	25504	224832
02:53:50	54	505640	3983	55760	500538
02:53:52	29	686282	5377	74096	662494
02:53:54	46	922765	7183	101704	908813
02:53:56	41	1125518	8777	126640	1134362
server_2	CPU	Network In KiB	Network Out KiB	dVol Read KiB	dVol Write KiB
Summary	Util				
	8				
Minimum	29	236257	1880	25504	224832
Average	42	695293	5440	76741	686208
Maximum	54	1125518	8777	126640	1134362

#### Where:

Value	Definition
Timestamp	Time the poll was taken.
CPU Util	Disk utilized in percentage in this interval.
Network In KiB	Network kibibytes received over all network interfaces per accumulated value.
Network Out KiB	Network kibibytes sent over all network interfaces per accumulated value.
dVol Read KiB	Storage kibibytes received from all server-storage interfaces per accumulated value.
dVol Write KiB	Storage kibibytes sent to all server-storage interfaces per accumulated value.

#### **EXAMPLE #4** To display a list of statistics group names followed by statpaths and their types, type:

\$ server\_stats server\_2 -list

server\_2:

Type Stat Name

```
System
                  basic-std
System
                  caches-std
                  cifs-std
System
System
                  cifsOps-std
                  diskVolumes-std
System
User
                  front-end
System
                  metaVolumes-std
                  netDevices-std
System
System
                  nfs-std
System
                  nfsOps-std
                  volumes
User
. . .
Family
                  fs
Family
                  fs.asyncio
                  fs.asyncio.deletesDeferred
Counter
Fact
                  fs.asyncio.deletesPending
Counter
                  fs.asyncio.writesDeferred
                  fs.asyncio.writesPending
Fact
Counter
                  fs.asyncio.writesSaved
                  fs.dnlc
Family
Counter
                  fs.dnlc.entriesChecked
Counter
                  fs.dnlc.findRequests
                fs.dnlc.hitRatio
Computed Fact
Counter
                  fs.dnlc.hits
Counter
                  fs.dnlc.negAdds
```

### **EXAMPLE #5** To display the statgroup information, type:

```
$ server_stats server_2 -info statsb
```

# **EXAMPLE #6** To display information of statistics group names followed by statpaths, type:

#### \$ server\_stats server\_2 -info

```
server_2 :
                = statsA
name
description
                = My group # 1
                = Group - user-defined
type
member_stats
                = statsB
member_elements =
member_of
. . .
                = cifs
name
description
               = The CIFS-protocol service
```

**EXAMPLE #7** To start the statmonService, type:

\$ server\_stats server\_2 -service -start -port 7777 statmonService started on port: 7777.

**EXAMPLE #8** To stop the statmonService, type:

\$ server\_stats server\_2 -service -stop

server\_2: done.

**EXAMPLE #9** To delete the statmonService configurations from the boot-up settings of the Data Mover, type:

\$ server\_stats server\_2 -service -delete

server\_2: done.

**EXAMPLE #10** To query the status of the statmonService, type:

\$ server\_stats server\_2 -service -status

server\_2: The statmonService has started.

Interface=INTERNAL

Port=7777

Allow=128.221.252.100:128.221.252.101:128.221.253.100:128.221.253.101
The statmonService is listing for incoming network connections
Max Connections: 32, Current: 0

**EXAMPLE #11** To enable stats collection, type:

\$ server\_stats server\_2 -monitor -action enable

server\_2: done.

**EXAMPLE #12** To query the state of the stats collection, type:

\$ server\_stats server\_2 -monitor -action status

server\_2 : Statistics are enabled.

**EXAMPLE #13** To display five iterations of the cifs-std statistics group with a three second interval, type:

\$ server\_stats server\_2 -monitor cifs-std-i5-c5

server_2	CIFS	CIFS	CIFS	CIFS Av	g CIFS	CIFS	CIFS Avg	CIFS	CIFS
Timestam	p Total	Read	Read	Rea	d Write	Write	Write	Share	Open
	Ops/s	Ops/s	KiB/s	Size KiE	Ops/s	KiB/s	Size KiB	Connections	Files
02:54:31	2133	0	0		- 1947	110600	57	96	587
02:54:34	1895	0	0		- 1737	99057	57	96	631
02:54:37	2327	0	0		- 2104	119556	57	96	649
02:54:40	2109	0	0		- 1864	106081	57	96	653

02:54:43	2439	0	0	-	2172	123578	57	96	639
server_2	CIFS	CIFS	CIFS	CIFS Avg			CIFS Avg	CIFS	CIFS
Summary	Total	Read	Read	Read	Write	Write	Write	Share	Open
	Ops/s	Ops/s	KiB/s	Size KiB	Ops/s	KiB/s	Size KiB	Connections	Files
Minimum	1895	0	0	_	1737	99057	57	96	587
Average	2180	0	0	_	1965	111775	57	96	632
Maximum	2439	0	0	_	2172	123578	57	96	653

### Where:

Value	Definition
Timestamp	Time the poll was taken.
CIFS Total Ops/s	Total operations per second.
CIFS Read Ops/s	CIFS read operations per second in the interval.
CIFS Read KiB/s	CIFS read data response in kibibytes per second.
CIFS Avg Size Read KiB	Average read data response.
CIFS Write Ops/s	CIFS write operations per second.
CIFS Write KiB/s	CIFS write data response in kibibytes per second.
CIFS Avg Size Write KiB	Average write data size.
CIFS Share Connections	Number of CIFS protocol connections.
CIFS Open Files	Number of open CIFS files.

# **EXAMPLE #14** To display five iterations of the nfs-std statistics group with a three second interval, type:

server_2	Total	NFS		rver_stats NFS Avg		_	onitor nfs- NFS Avg	std -i 5-c	5
Timestamp							Write Size		
	Ops/s	Ops/s	K1B/s	Bytes	Ops/s	K1B/s	Bytes	Threads	
13:44:53	20650	4121	67506	16774	2214	29737	13754	648	
13:44:54	11663	2318	37140	16407	1238	17307	14316	648	
13:44:55	8678	1790	30761	17597	945	12511	13557	648	
13:44:56	17655	3543	56382	16296	1967	27077	14096	648	
13:44:57	20302	4033	63822	16205	2271	31469	14189	648	
server_2	Total	NFS	NFS	NFS Avg	NFS	NFS	NFS Avg	NFS	
Summary	NFS	Read	Read	Read Size	Write	Write	Write Size	Active	
	Ops/s	Ops/s	KiB/s	Bytes	Ops/s	KiB/s	Bytes	Threads	
Minimum	8678	1790	30761	16205	945	12511	13557	648	
Average	15790	3161	51122	16656	1727	23620	13982	648	
Maximum	20650	4121	67506	17597	2271	31469	14316	648	

### Where:

Value	Definition
Timestamp	Time the poll was taken.
Total NFS Ops/s	Total number of operations per second.

NFS	Read Ops/s	NFS read operations per second in the interval.
NFS	Read KiB/s	NFS read data response in kibibytes per second.
NFS	Avg Read Size Bytes	Average read data response.
NFS	Write Ops/s	NFS write operations per second.
NFS	Write KiB/s	NFS write data response in kibibytes per second.
NFS	Avg Write Size Bytes	Average write data size.
NFS	Active Threads	Number of NFS active threads.

# **EXAMPLE #15** To display five iterations of the caches-std statistics group with a three second interval, type:

		\$ serve:	r_stats	server_2	-monitor	caches-std	-i 3	-c 5
server_2	DNLC	OF Cache	Buffer					
Timestamp	Hit	Hit	Cache					
	Ratio %	Ratio %	Hit %					
02:55:26	-	100	71					
02:55:29	_	100	72					
02:55:32	_	100	73					
02:55:35	_	100	73					
02:55:38	_	100	72					
server_2	DNLC	OF Cache	Buffer					
Summary	Hit	Hit	Cache					
	Ratio %	Ratio %	Hit %					
Minimum	_	100	71					
Average	_	100	72					
Maximum	_	100	73					

#### Where:

Value	Definition
Timestamp	Time the poll was taken.
DNLC Hit Ratio %	Directory Name Lookup Cache (DNLC) hit ratio.
OF Cache Hit Ratio %	Open file cache hit ratio.
Buffer Cache Hit %	Kernel buffer cache hit ratio.

# **EXAMPLE #16** To display the netDevices-std statistics group with a three second interval, type:

		\$ \$	server_stats	server_2	-monitor	netDevices-	std -i 3 -c 3
server_2	device	Network	Network	Network	Network	Network	Network
Timestamp		In	In	In	Out	Out	Out
		Pkts/s	Errors/s	KiB/s	Pkts/s	Errors/s	KiB/s
02:55:52	mge0	2	0	0	1	0	0
	mge1	17	0	23	9	0	1
	cge0	3593	0	26566	2289	0	203
	cge1	6912	0	50206	4444	0	378
	cge2	3637	0	25570	2342	0	209
02:55:55	mge0	0	0	0	0	0	0

	mge1	7	0	9	4	0	0
	cge0	3444	0	24744	2252	0	204
	cge1	7415	0	53354	4721	0	400
	cge2	3913	0	27796	2502	0	222
02:55:58	mge0	2	0	0	2	0	0
	mge1	32	0	39	19	0	2
	cge0	4029	0	29334	2594	0	230
	cge1	7461	0	54030	4791	0	406
	cge2	3902	0	27319	2505	0	223
server_2	device	Network	Network	Network	Network	Network	Network
Summary		In	In	In	Out	Out	Out
		Pkts/s	Errors/s	KiB/s	Pkts/s	Errors/s	KiB/s
Minimum	mge0	0	0	0	0	0	0
	mge1	7	0	9	4	0	0
	cge0	3444	0	24744	2252	0	203
	cge1	6912	0	50206	4444	0	378
	cge2	3637	0	25570	2342	0	209
	cge3	0	0	0	0	0	0
Average	mge0	1	0	0	1	0	0
	mge1	19	0	24	11	0	1
	cge0	3689	0	26882	2378	0	213
	cge1	7263	0	52530	4652	0	395
	cge2	3817	0	26895	2450	0	218
	cge3	0	0	0	0	0	0
Maximum	mge0	2	0	0	2	0	0
	mge1	32	0	39	19	0	2
	cge0	4029	0	29334	2594	0	230
	cge1	7461	0	54030	4791	0	406
	cge2	3913	0	27796	2505	0	223
	cge3	0	0	0	0	0	0

#### Where:

Value	Definition
Timestamp	Time the poll was taken.
Device	Name of the network device.
Network In Pkts/s	Network packets received per second.
Network In Errors/s	Network input errors encountered per second.
Network In KiB/s	Network kibibytes received per second.
Network Out Pkts/s	Network packets sent per second.
Network Out Errors/s	Network output errors encountered per second.
Network Out KiB/s	Network kibibytes sent per second.

# **EXAMPLE #17** To display the netDevices-std statistics group without the summary and with a three second interval, type:

\$ server\_stats server\_2 -monitor netDevices-std -i 3 -c 3
-terminationsummary no

server\_2 device Network Network Network Network Network Network

Timestamp		In	In	In	Out	Out	Out
		Pkts/s	Errors/s	KiB/s	Pkts/s	Errors/s	KiB/s
02:56:11	mge0	16	0	1	19	0	23
	mge1	43	0	60	24	0	2
	cge0	3960	0	29053	2547	0	226
	cge1	6709	0	48414	4296	0	366
	cge2	4829	0	33996	3125	0	281
02:56:14	mge0	0	0	0	0	0	0
	mge1	3	0	3	2	0	0
	cge0	3580	0	25905	2335	0	211
	cge1	6663	0	48212	4273	0	364
	cge2	3970	0	28113	2523	0	222
02:56:17	mge0	2	0	0	2	0	0
	mge1	5	0	6	2	0	0
	cge0	3561	0	25891	2296	0	206
	cge1	7091	0	51721	4564	0	389
	cge2	3931	0	27703	2514	0	223
	cge3	0	0	0	0	0	0

## **EXAMPLE #18** To display the cifsOps-std statistics with a five second interval, type:

	\$	server_stats	server_2	-monitor	cifsops-std	-i
server_2	SMB Operation	Op	Min	Max	Avg	
Timestamp		Calls/s	uSec	uSec	uSec/call	
02:57:00	SMB1_Close	89	45	406775	10273	
	SMB1_WriteX	1837	30	1618776	144030	
	SMB1_CreateNTX		51	458090	379	
02:57:03	SMB1_Close	122		406775	10057	
	SMB1_WriteX	1867	30	1618776	133180	
	SMB1_CreateNTX		51	458090	1826	
02:57:06	SMB1_Close	105	45	406775	14663	
	SMB1_WriteX	2119	30	1618776	121976	
	SMB1_CreateNTX	103	51	458090	1801	
0					_	
server_2	SMB Operation	Op	Min	Max	Avg	
Summary		Calls/s	uSec		uSec/call	
Minimum	SMB1_Mkdir	0	0	0	_	
	SMB1_Rmdir	0	0	0	_	
	SMB1_Open	0	0	0	_	
	SMB1_Create	0	0	0	_	
	SMB1_Close	89	45	406775	10057	
	SMB1_Flush	0	0	0	_	
	SMB1_Unlink	0	0	0	-	
	SMB1_Rename	0	0	0	_	
	SMB1_GetAttr	0	0	0	_	
	SMB1_SetAttr	0	0	0	-	
	SMB1_Read	0	0	0	_	
	SMB1_Write	0	0	0	_	
	SMB1_Lock	0	0	0	_	
	SMB1_Unlock	0	0	0	_	
	SMB1_CreateTmp	0	0	0	_	

SMB1_MkNew	0	0	0	-
SMB1_ChkPath	0	0	0	-
SMB1_Exit	0	0	0	-
SMB1_Lseek	0	0	0	-
SMB1_LockRead	0	0	0	-
SMB1_WriteUnlock	0	0	0	-
SMB1_ReadBlockRaw	0	0	0	-
SMB1_WriteBlockRaw	0	0	0	-
SMB1_SetAttrExp	0	0	0	-
SMB1_GetAttrExp	0	0	0	_
SMB1_LockingX	0	0	0	_
SMB1_Trans	0	0	0	_
SMB1_TransSec	0	0	0	-
SMB1_Copy	0	0	0	-
SMB1_Move	0	0	0	-
SMB1_Echo	0	0	0	_
SMB1_WriteClose	0	0	0	_
SMB1_OpenX	0	0	0	_
SMB1_ReadX	0	0	0	-
SMB1_WriteX	1837	30	1618776	121976
SMB1_CloseTreeDisco	0	0	0	_
SMB1_Trans2Prim	0	0	0	_
SMB1_Trans2Secd	0	0	0	_
SMB1_FindClose2	0	0	0	_
SMB1_FindNotifyClose	0	0	0	_
SMB1_TreeConnect	0	0	0	_
SMB1_TreeDisco	0	0	0	_
SMB1_NegProt	0	44	85	_
SMB1_SessSetupX	0	1088	12058	_
SMB1_UserLogoffX	0	0	0	_
SMB1_TreeConnectX	0	82	499	_
SMB1_DiskAttr	0	0	0	_
SMB1_Search	0	0	0	_
SMB1_FindFirst	0	0	0	_
SMB1_FindUnique	0	0	0	_
SMB1_FindClose	0	0	0	_
	0	0	0	_
SMB1_TransNTSecd	0	0	0	_
SMB1_CreateNTX	84	51	458090	379
SMB1 CancelNT	0	0	0	_
SMB1_SendMessage	0	0	0	_
SMB1_BeginMessage	0	0	0	_
SMB1 EndMessage	0	0	0	_
SMB1_MessageText	0	0	0	_
SMB2_Negotiate	0	0	0	_
SMB2_SessionSetup	0	0	0	_
SMB2_Logoff	0	0	0	_
SMB2 TreeConnect	0	0	0	_
SMB2_TreeDisConnect	0	0	0	_
SMB2_Create	0	0	0	_
SMB2 Close	0	0	0	_
SMB2_Flush	0	0	0	_
D11121_1 1 4 D11	U	J	0	_

Average

SMB2_Read	0	0	0	_
SMB2_Write	0	0	0	_
SMB2_Lock	0	0	0	_
SMB2_Ioctl	0	0	0	_
SMB2_Cancel	0	0	0	_
SMB2_Echo	0	0	0	_
SMB2_QueryDirectory	0	0	0	_
SMB2_ChangeNotify	0	0	0	_
SMB2_QueryInfo	0	0	0	_
SMB2_SetInfo	0	0	0	_
SMB2_OplockBreak	0	0	0	_
SMB1_Mkdir	0	0	0	_
SMB1_Rmdir	0	0	0	_
SMB1_MMGII SMB1_Open	0	0	0	_
SMB1_Open SMB1_Create	0	0	0	
				11664
SMB1_Close	105	45	406775	11664
SMB1_Flush	0	0	0	_
SMB1_Unlink	0	0	0	_
SMB1_Rename	0	0	0	_
SMB1_GetAttr	0	0	0	_
SMB1_SetAttr	0	0	0	_
SMB1_Read	0	0	0	-
SMB1_Write	0	0	0	-
SMB1_Lock	0	0	0	_
SMB1_Unlock	0	0	0	_
SMB1_CreateTmp	0	0	0	_
SMB1_MkNew	0	0	0	-
SMB1_ChkPath	0	0	0	_
SMB1_Exit	0	0	0	_
SMB1_Lseek	0	0	0	_
SMB1_LockRead	0	0	0	_
SMB1_WriteUnlock	0	0	0	_
SMB1_ReadBlockRaw	0	0	0	_
SMB1_WriteBlockRaw	0	0	0	_
SMB1_SetAttrExp	0	0	0	_
SMB1_GetAttrExp	0	0	0	_
SMB1_LockingX	0	0	0	_
SMB1_Trans	0	0	0	_
SMB1_TransSec	0	0	0	_
SMB1_Copy	0	0	0	_
SMB1_Copy SMB1_Move	0	0	0	_
SMB1_Move SMB1_Echo	0	0	0	_
SMB1_Echo SMB1_WriteClose	0	0	0	_
SMB1_WIITECTOSE SMB1_OpenX	0	0	0	_
——————————————————————————————————————				_
SMB1_ReadX	1041	0	1610776	122062
SMB1_WriteX	1941	30	1618776	133062
SMB1_CloseTreeDisco	0	0	0	_
SMB1_Trans2Prim	0	0	0	_
SMB1_Trans2Secd	0	0	0	_
SMB1_FindClose2	0	0	0	_
SMB1_FindNotifyClose	0	0	0	_
SMB1_TreeConnect	0	0	0	-

SMB1_TreeDisco	0	0	0	_
SMB1_NegProt	0	44	85	-
SMB1_SessSetupX	0	1088	12058	-
SMB1_UserLogoffX	0	0	0	-
SMB1_TreeConnectX	0	82	499	-
SMB1_DiskAttr	0	0	0	-
SMB1_Search	0	0	0	_
SMB1_FindFirst	0	0	0	_
SMB1_FindUnique	0	0	0	_
SMB1_FindClose	0	0	0	_
SMB1_TransNT	0	0	0	_
SMB1_TransNTSecd	0	0	0	_
SMB1_CreateNTX	104	51	458090	1335
SMB1_CancelNT	0	0	0	_
SMB1_SendMessage	0	0	0	_
SMB1_BeginMessage	0	0	0	-
SMB1_EndMessage	0	0	0	_
SMB1_MessageText	0	0	0	-
SMB2_Negotiate	0	0	0	_
SMB2_SessionSetup	0	0	0	-
SMB2_Logoff	0	0	0	_
SMB2_TreeConnect	0	0	0	_
SMB2_TreeDisConnect	0	0	0	-
SMB2_Create	0	0	0	_
SMB2_Close	0	0	0	_
SMB2_Flush	0	0	0	_
SMB2_Read	0	0	0	_
SMB2_Write	0	0	0	_
SMB2_Lock	0	0	0	_
SMB2_loctl	0	0	0	_
SMB2_Cancel	0	0	0	_
SMB2_Echo	0	0	0	_
SMB2_QueryDirectory	0	0	0	_
SMB2_ChangeNotify	0	0	0	_
SMB2_QueryInfo	0	0	0	_
SMB2_SetInfo	0	0	0	_
SMB2_OplockBreak	0	0	0	_
SMB1_Mkdir	0	0	0	_
SMB1_Rmdir	0	0	0	_
SMB1_Open	0	0	0	_
SMB1_Create	0	0	0	_
SMB1_Close	122	45	406775	14663
SMB1_Flush	0	0	0	_
 SMB1_Unlink	0	0	0	_
SMB1_Rename	0	0	0	_
SMB1 GetAttr	0	0	0	_
SMB1_SetAttr	0	0	0	_
SMB1_Read	0	0	0	_
SMB1_Write	0	0	0	_
SMB1_Lock	0	0	0	_
SMB1_Unlock	0	0	0	_
SMB1_CreateTmp	0	0	0	_
	-	-	-	

Maximum

SMB1_MkNew	0	0	0	_
SMB1_ChkPath	0	0	0	_
SMB1_Exit	0	0	0	_
SMB1_Lseek	0	0	0	_
SMB1_LockRead	0	0	0	_
SMB1_WriteUnlock	0	0	0	_
SMB1_ReadBlockRaw	0	0	0	-
SMB1_WriteBlockRaw	0	0	0	_
SMB1_SetAttrExp	0	0	0	-
SMB1_GetAttrExp	0	0	0	_
SMB1_LockingX	0	0	0	-
SMB1_Trans	0	0	0	_
SMB1_TransSec	0	0	0	_
SMB1_Copy	0	0	0	_
SMB1_Move	0	0	0	_
SMB1_Echo	0	0	0	_
SMB1_WriteClose	0	0	0	_
SMB1_OpenX	0	0	0	_
SMB1_ReadX	0	0	0	_
SMB1_WriteX	2119	30	1618776	144030
SMB1_CloseTreeDisco	0	0	0	_
SMB1_Trans2Prim	0	0	0	_
SMB1_Trans2Secd	0	0	0	_
SMB1_FindClose2	0	0	0	_
SMB1_FindNotifyClose	0	0	0	_
SMB1_TreeConnect	0	0	0	_
SMB1_TreeDisco	0	0	0	_
SMB1_NegProt	0	44	85	_
SMB1_SessSetupX	0	1088	12058	_
SMB1_UserLogoffX	0	0	0	_
SMB1_TreeConnectX	0	82	499	_
 SMB1_DiskAttr	0	0	0	_
SMB1 Search	0	0	0	_
SMB1_FindFirst	0	0	0	_
SMB1_FindUnique	0	0	0	_
SMB1_FindClose	0	0	0	_
SMB1 TransNT	0	0	0	_
SMB1_TransNTSecd	0	0	0	_
SMB1_CreateNTX	126	51	458090	1826
SMB1 CancelNT	0	0	0	_
SMB1_SendMessage	0	0	0	_
SMB1_BeginMessage	0	0	0	_
SMB1_EndMessage	0	0	0	_
SMB1_MessageText	0	0	0	_
SMB2_Negotiate	0	0	0	_
SMB2_SessionSetup	0	0	0	_
SMB2_Logoff	0	0	0	_
SMB2_TreeConnect	0	0	0	_
SMB2_TreeDisConnect	0	0	0	_
SMB2_Create	0	0	0	_
SMB2_Close	0	0	0	_
SMB2_Flush	0	0	0	_
DITTO T TUBIT	U	U	Ü	_

SMB2_Read	0	0	0	-
SMB2_Write	0	0	0	-
SMB2_Lock	0	0	0	-
SMB2_Ioctl	0	0	0	_
SMB2_Cancel	0	0	0	_
SMB2_Echo	0	0	0	_
SMB2_QueryDirectory	0	0	0	_
SMB2_ChangeNotify	0	0	0	_
SMB2_QueryInfo	0	0	0	_
SMB2_SetInfo	0	0	0	_
SMB2_OplockBreak	0	0	0	_

#### Where:

Value	Definition
Timestamp	Time the poll was taken.
SMB Operation	Name of the SMB operation.
Op Calls/s	Number of calls to this SMB operation per second.
Min uSec	Minimum time in microseconds per call.
Max uSec	Maximum time in microseconds per call.
Avg uSec/Call	Average time in microseconds consumed per call.

# **EXAMPLE #19** To display the cifsOps-std statistics group without the summary and with a five second interval, type:

#### \$ server\_stats server\_2 -m cifsops-std -i 5-c 3 -te no server\_2 SMB Operation Οp Min Max Avg Timestamp Calls/s uSec uSec uSec/call 02:57:24 SMB1\_Close 56 45 552768 25299 29 1618776 SMB1\_WriteX 1360 161125 SMB1\_CreateNTX 46 51 458090 971 02:57:27 SMB1\_Close 130 45 568291 16814 SMB1\_WriteX 1627 29 1618776 182622 SMB1\_CreateNTX 147 51 458090 276 02:57:30 SMB1\_Close 29992 50 45 568291 SMB1\_WriteX 1615 29 1618776 151924 SMB1\_CreateNTX 37 51 458090 2850

# **EXAMPLE #20** To display the nfsOps-std statistics group without the summary and with a five second interval, type:

		\$ ser	ver_stats	server_2 -m	nfsops-std -i	5 -c 3 -te no
server_2	NFS Op	NFS	NFS	NFS	NFS Op %	
Timestamp		qO	Op	qO		
		Calls/s	Errors/s	uSec/call		
03:18:21	v3Read	23442	0	63846	50	
	v3Write	23372	0	99156	50	
03:18:24	v3Read	23260	0	65756	50	
	v3Write	23243	0	101135	50	
03:18:27	v3Read	23385	0	66808	50	

v3Write 23323 0 102201 50

#### Where:

Value	Definition
Timestamp	Time the poll was taken.
NFS Op	Name of the NFS operation.
NFS Op Calls/s	Number of calls to this NFS operation per second.
NFS Op Errors/s	Number of times the NFS operation failed per second.
NFS Op uSec/Call	Average time in microseconds consumed per call.
NFS Op %	Percent of total NFS calls attributed to this operation.

# **EXAMPLE #21** To display the diskVolumes-std statistics group without the summary and with a five second interval, type:

## $\$ server\_stats server\_2 -m diskVolumes-std -i 5-c 3 -te

			no							
server_2	dVol	Q.	ueue	Read	Read	Avg Read	Write	Write	Avg Write	Util %
Timestamp		D	epth	Ops/s	KiB/s	Size	Ops/s	KiB/s	Size	
						Bytes/s			Bytes/s	
02:58:09	NBS1		0	0	3	8192	1	7	6827	0
	root_	_ldisk	0	0	0	_	461	490	1090	47
	d7		0	113	904	8192	530	19619	37881	83
	d11		0	249	1995	8192	431	11640	27634	91
	d8		0	68	547	8192	372	11472	31607	79
	d12		33	424	3389	8192	609	20045	33705	99
	d9		0	36	291	8192	592	20339	35161	67
	d13		0	333	2664	8192	347	11925	35158	93
	d10		0	24	189	8192	385	11896	31668	63
	d14		36	573	4581	8192	454	20173	45468	100
02:58:12	root_	_ldisk	0	0	0	_	401	462	1182	44
	NBS6		0	0	0	_	1	3	3072	0
	d7		0	78	624	8192	388	13851	36523	70
	d11		0	216	1728	8192	470	11147	24268	84
	d8		0	51	411	8192	333	10672	32850	85
	d12		0	301	2408	8192	483	14411	30531	98
	d9		0	24	192	8192	422	14285	34691	50
	d13		0	290	2317	8192	340	10920	32856	87
	d10		0	19	152	8192	346	10944	32389	70
	d14		47	407	3259	8192	342	14288	42822	100
02:58:15	NBS1		0	0	0	_	3	1	512	0
	root_	_ldisk	0	0	0	_	409	454	1135	43
	NBS5		0	0	0	_	9	83	9070	1
	d7		0	122	976	8192	471	20179	43839	90
	d11		1	144	1149	8192	225	6608	30118	94
	d8		2	33	261	8192	229	6515	29131	48
	d12		41	424	3395	8192	666	20632	31722	93
	d9		0	44	355	8192	577	20848	36999	82
	d13		2	185	1483	8192	201	6768	34423	93
	d10		0	13	101	8192	238	6789	29252	36

d14 0 583 4667 8192 521 21131 41505 95

### Where:

Value	Definition
Timestamp	Time the poll was taken.
dVol	Name of the disk volume.
Queue Depth	Queue depth of the disk volume.
Read Ops/s	Number of read operations per second.
Read KiB/s	Kibibytes read per second.
Avg Read Size Bytes	Average size in bytes of read requests per second.
Write Ops/s	Number of write operations per second.
Write KiB/s	Kibibytes written per second.
Avg Write Size Bytes	Average size in bytes for write requests per second.
Util %	Disk utilized in percetage.

# **EXAMPLE #22** To display the metaVolumes-std statistics group without the summary and with a five second interval, type:

# $\mbox{\$}$ server\_stats server\_2 -m metaVolumes-std -i 5 -c 3 -te no

		_		110						
server_2	MetaVol	Read		Avg Read				Avg Write	Write	
Timestamp	Reques	sts/s	KiB/s	Size	Ops/s	Requests/s	KiB/s	Size	Ops/s	
				Bytes				Bytes		
02:58:37	SNBS6	0	0	-	0	1	3	3072	1	
	ufs_4	0	0	-	0	160	1285	8209	161	
	ufs_5	0	0	_	0	163	1299	8175	162	
	ufs_3	0	0	_	0	11	2155	200580	11	
	ufs_2	347	2776	8192	347	140	23544	172208	140	
	ufs_0	315	2517	8192	315	148	21427	147916	148	
	ufs_1	654	5229	8192	654	313	45512	148895	313	
	root_fs_3	3 1	11	8192	1	0	0	_	0	
02:58:40	SNBS5	0	0	_	0	3	37	12743	3	
	SNBS1	0	0	_	0	3	1	512	3	
	ufs_4	0	0	_	0	159	1257	8089	157	
	ufs_5	0	0	_	0	160	1273	8158	159	
	ufs_3	0	0	-	0	2	511	224695	2	
	ufs_2	396	3166	8192	396	195	27326	143200	195	
	ufs_0	431	3446	8192	431	187	29574	162161	187	
	ufs_1	408	3262	8192	408	159	27782	178784	159	
	root_fs_3	3 1	5	8192	1	0	0	_	0	
02:58:43	SNBS5	0	0	_	0	1	5	5461	1	
	SNBS6	0	0	_	0	1	3	4608	1	
	ufs_4	0	0	_	0	146	1159	8136	145	
	ufs_5	0	0	_	0	148	1183	8174	148	
	ufs_3	0	0	_	0	8	1965	262144	8	
	ufs_2	522	4174	8192	522	219	35546	166238	219	
	ufs_0	492	3933	8192	492	222	33356	153886	222	
	ufs_1	467	3736	8192	467	188	31955	173819	188	

#### Where:

Value	Definition
MetaVol	Name of the meta volume associated with the filesystem.
Read Request/s	Number of read requests per second to this volume.
Read KiB/s	Kibibytes read per second.
Avg Read Size Bytes	Average size for read requests to this volume.
Read Ops/s	Number of read operations per second.
Write Requests/s	Number of write requests per second.
Write KiB/s	Number of kibibytes written per second to this volume.
Avg Write Size Bytes	Average size in bytes for write requests.
Write Ops/s	Number of write operations per second.

# **EXAMPLE #23** To display the nfsOps-std statistics group sorted by the percentage of all the NFS operations for the five second interval, type:

## \$ server\_stats server\_2 -monitor nfsOps-std -sort opPct -i 5 -c 3 -te

		J -C J -C	-	
NFS Op	NFS	NFS	NFS	NFS Op %
	Op	qO	QO	
	Calls/s	Errors/s	uSec/call	
v3Read	23263	0	81632	50
v3Write	23352	0	116645	50
v3Read	23431	0	82443	50
v3Write	23345	0	118088	50
v3Read	23176	0	84759	50
v3Write	23326	0	119608	50
	v3Read v3Write v3Read v3Write v3Read	Op Calls/s v3Read 23263 v3Write 23352 v3Read 23431 v3Write 23345 v3Read 23176	Op Op Calls/s Errors/s v3Read 23263 0 v3Write 23352 0 v3Read 23431 0 v3Write 23345 0 v3Read 23176 0	Op         Op         Op           Calls/s         Errors/s         uSec/call           v3Read         23263         0         81632           v3Write         23352         0         116645           v3Read         23431         0         82443           v3Write         23345         0         118088           v3Read         23176         0         84759

# **EXAMPLE #24** To display the nfsOps-std statistics group sorted by the average time in microseconds used for the five second interval, in ascending order, type:

## \$ server\_stats server\_2 -m nfsops-std -sort avgTime -order asc -i 5 -c 3 -te no

server_2	NFS Op	NFS	NFS	NFS	NFS Op %
Timestamp		Op	Op	QO	
		Calls/s	Errors/s	uSec/call	
04:05:27	v3Write	605	0	8022318	100
	v3Create	2	0	25304786	0
04:05:30	v3Create	8	0	7722823	1
	v3Write	579	0	8435543	99
04:05:33	v3Create	41	0	1468883	7
	v3Write	567	0	8690860	93

#### **EXAMPLE #25**

To display the nfsOps-std statistics group sorted by the average time in microseconds used for the five second interval, in descending order, and including the three counts of data output, type:

## \$ server\_stats server\_2 -m nfsops-std -sort avgTime -order desc -lines 3 -i 5 -c 3 -te no

server_2 Timestamp	NFS Op	NFS Op	NFS Op	NFS Op	NFS Op %
		Calls/s	Errors/s	uSec/call	
04:09:39	v3Create	1	0	31657550	0
	v3Write	610	0	6223366	100
04:09:44	v3Write	607	0	6275942	98
	v3Create	11	0	3978054	2
04:09:49	v3Write	574	0	6691264	93
	v3Create	42	0	1073819	7

### EXAMPLE # 26

To display the information about minInterval and defaultInterval, type:

\$ server\_param server\_2 -f statmon -info minInterval

```
server_2 :
                       = minInterval
name
facility_name
                       = statmon
default_value
                       = 5
current_value
                       = 5
configured_value
user_action
                       = none
change_effective
                       = immediate
range
                       = (1,300)
```

description = Minimum Interval time used by the statmon session

\$ server\_param server\_2 -f statmon -info defaultInterval

user\_action = none change\_effective = immediate range = (1,300)

description = Default Interval time used by statmon session

#### EXAMPLE #27

To change the value of minInterval and defaultInterval, type:

\$ server\_param server\_2 -f statmon -modify minInterval
-value 5

server\_2: done.

\$ server\_param server\_2 -f statmon -modify
defaultInterval -value 10

server\_2: done.

#### **EXAMPLE #28**

To verify that server\_stats displays an error message if the value specified to modify for minInterval and defaultInterval variables is not in range, type:

\$ server\_param server\_2 -facility statmon -modify
minInterval -value 1000

#### server 2:

Error 4418: server\_2: 1000 is not in range (1,300)

\$ server\_param server\_2 -facility statmon -modify
defaultInterval -value 400

#### server\_2:

Error 4418: server\_3: 400 is not in range (1,300)

#### **EXAMPLE #29**

To check whether server\_stats displays an error message when a specified interval is smaller than the minInterval specified by server\_param, type:

\$ server\_param server\_2 -facility statmon -modify minInterval -value 5

server\_2: done

\$ server\_stats server\_2 -m -i 1 nfs.basic

ERROR (13421969453): The requested interval (1) is less than the minInterval value (5).

#### **EXAMPLE #30**

To verify that server\_stats displays an error message while it requests the stats data by using a defaultInterval value that is less than the minInterval value, type:

\$ server\_param server\_2 -facility statmon -modify
minInterval -value 5

server\_2: done

\$ server\_param server\_2 -facility statmon -modify
defaultInterval -value 1

server\_2: done

\$ server stats server 2 -m nfs.basic

ERROR (13421969452): The defaultInterval parameter value (1) is less than the minInterval parameter value (5).

## server\_sysconfig

Manages the hardware configuration for the specified or all Data Movers.

#### **SYNOPSIS**

#### **DESCRIPTION**

**server\_sysconfig** displays and modifies the hardware configuration of the Data Movers.

To use Ethernet channels and link aggregations, a complimentary configuration on the switch is required. The switch must support:

- IEEE standard Ethernet, Fast Ethernet, or Gigabit Ethernet
- ◆ IEEE 802.3ad Link Aggregation protocol

**server\_sysconfig** implements a virtual device which combines one or more devices on the same Data Mover into a single, logical device addressable as a unit. Fail-safe networks, Ethernet channels, and link aggregations are virtual devices. Both Ethernet channels and link aggregations are trunking devices.

The ALL option executes the command for all Data Movers.

#### **OPTIONS**

#### -Platform

Displays the hardware configuration of the Data Mover, including processor type and speed, the bus speed in MHz, the main memory in MB, and the motherboard type.

```
-pci [<device>]
```

Displays the attributes of the specified network adapter card or for all devices.

```
-pci <device> -option <options>
```

Sets the attributes of the specified network adapter card installed in the Data Mover. Options must be comma-separated and double quoted.

#### **Ethernet**

#### speed={10 | 100 | 1000 | auto}

Sets the speed for the port. **auto** (default) turns on autonegotiation; setting a fixed speed disables autonegotiation.

#### duplex={full|half|auto}

Sets the duplex to **full**, **half**, or **auto**. The **auto** (default) turns autonegotiation on; setting a fixed duplex disables autonegotiation.



#### **CAUTION**

The speed and duplex settings on both sides of the physical connection must be the same. Mismatched speed and duplex settings can cause errors and impact network performance. For example, if the duplex is set to half on one end and full on the other, there might be network errors and performance issues.

#### **Gigabit Ethernet Fiber**

For Gigabit Ethernet Fiber connections, the speed is automatically set to 1000, and since it must remain at that setting, no speed setting is required.

#### linkneg={enable|disable}

Disables autonegotiation on the network adapter card if it is not supported by the network Gigabit switch. The default is **enable**.

#### rxflowctl={enable|disable}

Enables the ability to accept and process pause frames. The default is **disable**.

#### txflowctl={enable|disable}

Enables pause frames to be transmited. The default is **disable**.

### Gigabit Ethernet Copper

#### speed={10 | 100 | 1000 | auto}

Sets the speed for the port. The **auto** (default) turns on autonegotiation; setting a fixed speed disables autonegotiation.

#### duplex={full|half|auto}

Sets the duplex to **full**, **half**, or **auto**. The **auto** (default) turns autonegotiation on; setting a fixed duplex disables autonegotiation.



#### **CAUTION**

The speed and duplex settings on both sides of the physical connection must be the same. Mismatched speed and duplex settings can cause errors and impact network performance. For example, if the duplex is set to half on one end and full on the other, there might be network errors and performance issues.

#### rxflowctl={enable|disable}

Enables the ability to accept and process pause frames. The default is **disable**.

#### txflowctl={enable|disable}

Enables pause frames to be transmited. The default is **disable**.

#### -virtual -delete [-Force] <device>

Deletes an Ethernet channel, link aggregation, or fail-safe network device. Valid entries for a <device> are **trk** or **fsn**. The **-Force** option is required if an IP address has been assigned.



#### **CAUTION**

All of the IP interfaces configured over a virtual device are deleted when the -Force option is used. As a result, any applications on the Celerra Network Server that use these interfaces might get disrupted. Among these applications are CIFS and Celerra Replicator V2. If all of the interfaces used by a specific CIFS server get deleted by the server\_sysconfig -Force command, the server does not communicate with clients any further, and Windows users lose access to their data.

#### -virtual

Information for all devices is displayed.

#### -virtual -info <device>

Displays information for the specified Ethernet channel, link aggregation, or fail-safe network device.

```
-virtual -name <device> -create trk -option
device=<device,...>
```

Creates a virtual device for **trk** (Ethernet channel or link aggregation). An Ethernet channel can combine two, four, or eight (must be a multiple of two) ports into a single virtual device. A maximum of eight devices for Fast Ethernet is allowed:

◆ For Fast Ethernet trunking, use ana0 through ana7.

- For Gigabit Ethernet trunking, use **ace0** and **ace1**.
- For Gigabit Ethernet using the NS series, use cge0 through cge5.

The **-option** string defines one or more devices for the Ethernet channel or link aggregation. The <device,...> specifies the device or devices separated by commas. The **-option** string must have the options separated with a space, and must be enclosed with double quotes. If not, the command appears to execute, but does not pick up any options not contained within double quotes.

A gigabit trunk can be created on more than two gigabit devices. The Celerra Network Server has only two gigabit devices (ace0, ace1), so two ports are created. The NS series has a six gigabit port, so a four port gigabit trunk can be created.

For link aggregation there can be one or more number of ports, up to a maximum of 12.



#### **CAUTION**

The virtual device must be created before the IP interfaces can be configured for a trunk network device.

#### [protocol=lacp]

Initializes link aggregation control protocol for Ethernet devices which allows a non-exponential number of devices. For example, one, three, or five entries.

#### [lb=mac|ip|tcp]

Specifies the statistical load balancing method that applies to the aggregator/trunk. The **mac** option indicates that the output link on the trunk/aggregator is chosen based on source and destination MAC addresses. The **ip** (default) option indicates that the output link on the trunk/aggregator is chosen based on source and destination IP addresses. This is useful in the situation that the Data Mover is connected to a router or to a layer 3 switch. The **tcp** option indicates that the output link on the trunk/aggregator is chosen based on source and destination IP addresses as well as source and destination TCP ports.

```
-virtual -name <device> -create fsn -option
primary=<device> device=<device,...>
```

Creates a virtual device for **fsn** (fail-safe network). The fail-safe network device is a high-availability feature that provides switch-level redundancy. Configuring a failover channel is optional. Specifies the device name of the virtual device.

The **-option** string indicates a primary device for the fail-safe network device and a second device to be defined as the standby in case the primary device fails. The <device,...> specifies the device or devices separated by commas. The **-option** string must be defined in sequence, separated with a space, and enclosed with double quotes. If not, the command appears to execute, but does not pick up any options not contained within double quotes.



#### **CAUTION**

The virtual device must be created before the IP interfaces can be configured for the fail-safe network device.

#### **SEE ALSO**

Configuring and Managing Celerra Networking, Configuring Managing Celerra Network High Availability, server\_dbms, and server\_ifconfig.

#### **INTERFACE OUTPUTS**

The network interface cards available are dependent on the type of system used.

For the Celerra Network Server, the following NICs are available: loop, ace, ana, cge, el30, el31, fpa, and fa2. Note that loop, el30, and el31 are for internal use only.

For the NS series, the following NICs are available: loop, cge, el30, el31, and fge.

VDMs are included in both the CNS and NS series.

#### **EXAMPLE #1**

For the NS series, to view the system configuration for a Data Mover, type:

#### \$ server\_sysconfig server\_2 -Platform

```
server_2 :
Processor = Intel Pentium 4
Processor speed (MHz) = 3100
Total main memory (MB) = 4023
Mother board = Barracuda XP
Bus speed (MHz) = 533
Bios Version = 3.30
Post Version = Rev. 02.14
```

For the CNS series, to view the system configuration for a Data Mover, type:

\$ server\_sysconfig server\_2 -Platform

server\_2 :

```
Processor = Intel Pentium 4
Processor speed (MHz) = 1600
Total main memory (MB) = 3967
Mother board = CMB-400
Bus speed (MHz) = 400
Bios Version = No Ver Info
Post Version = No Ver Info
```

**EXAMPLE #2** For the NS series, to view the installed PCI configuration for a Data Mover, type:

\$ server\_sysconfig server\_2 -pci server\_2 : PCI DEVICES: On Board: Agilent Fibre Channel Controller 0: fcp-0 IRQ: 22 addr: 50060160006004f0 0: fcp-1 IRQ: 21 addr: 50060161006004f0 0: fcp-2 IRQ: 18 addr: 50060162006004f0 0: fcp-3 IRQ: 20 addr: 50060163006004f0 Broadcom Gigabit Ethernet Controller 0: fge0 IRQ: 24 linkneg=enable txflowctl=disable rxflowctl=disable 0: fge1 IRQ: 23 linkneg=enable txflowctl=disable rxflowctl=disable 0: cge0 IRQ: 24 speed=auto duplex=auto txflowctl=disable rxflowctl=disable 0: cge1 IRQ: 23 speed=auto duplex=auto txflowctl=disable rxflowctl=disable 0: cge2 IRQ: 26 speed=auto duplex=auto txflowctl=disable rxflowctl=disable 0: cge3 IRO: 25 speed=auto duplex=auto txflowctl=disable rxflowctl=disable 0: cge4 IRQ: 28 speed=auto duplex=auto txflowctl=disable rxflowctl=disable 0: cge5 IRQ: 27 speed=auto duplex=auto txflowctl=disable rxflowctl=disable

For the CNS series, to view the installed PCI configuration for a Data Mover, type:

```
$ server_sysconfig server_2 -pci
server_2 : PCI DEVICES:
Slot: 1
 Emulex LP9000 Fibre Channel Controller
    0: fcp-0 IRO: 23 addr: 10000000c92b5a10
    1: fcp-1 IRQ: 24 addr: 10000000c92b5a11
Slot: 2
 Emulex LP9000 Fibre Channel Controller
   0: fcp-2 IRQ: 22 addr: 10000000c92b514e
Slot: 4
 Intel 10/100/1K Ethernet Controller
    0: cge0 IRQ: 18
    speed=auto duplex=auto rxflowctl=disable txflowctl=disable
    1: cge1 IRQ: 19
    speed=auto duplex=auto rxflowctl=disable txflowctl=disable
   2: cge2 IRQ: 20
    speed=auto duplex=auto rxflowctl=disable txflowctl=disable
    3: cge3 IRQ: 21
    speed=auto duplex=auto rxflowctl=disable txflowctl=disable
Slot: 5
 Alteon Tigon-2 Gigabit Ethernet Controller
    0: ace0 IRQ: 25
    linkneg=enable rxflowctl=disable txflowctl=disable
    0: ace0 IRQ: 25
    linkneq=enable rxflowctl=disable txflowctl=disable
```

#### Where:

Value	Definition
On Board	The names of each PCI card installed.
0	Port number inside the slot the card is plugged into. If the card put in the slot has 4 ports, the first port is marked as 0, second port as 1, third port as 2, and fourth port as 3.
IRQ	Interrupt vector.
speed	Speed configured. Possible values are: auto, 10, 100, 1000.
duplex	Duplex setting configured. Possible values are: auto, half, full.
txflowctl	Transmit MAC flow control. Possible values are: disable, enable.
rxflowctl	Receive MAC flow control. Possible values are: disable, enable.

# **EXAMPLE #3** To set the **100** Mbits speed and **full** duplex setting for **cge0** interface, type:

\$ server\_sysconfig server\_2 -pci cge0 -option
speed=100,duplex=full

server\_2 : done

**EXAMPLE #4** To display the hardware configuration for network device, **cge0**, type:

\$ server\_sysconfig server\_2 -pci cge0

```
server_2 :
On Board:
Broadcom Gigabit Ethernet Controller
    0: cge0 IRQ: 24
    speed=100 duplex=full txflowctl=disable rxflowctl=disable
```

**EXAMPLE #5** To create an Ethernet channel as a virtual device, type:

\$ server\_sysconfig server\_2 -virtual -name trk0\_ec
-create trk -option "device=cge2,cge3"

server\_2 : done

server\_2 :

**EXAMPLE #6** To display all virtual devices on **server\_2**, type:

\$ server\_sysconfig server\_2 -virtual

```
Virtual devices:

trk0_ec devices=cge2 cge3
fsn failsafe nic devices:
trk trunking devices: trk0_ec
```

#### Where:

Value	Definition
Virtual Devices	All the configured virtual devices ( trunking or fail safe) on the server.
devices	Lists the virtual or physical device names that in the <virtual device="">.</virtual>
failsafe nic devices	Name of FSN virtual devices configured on the server.
trunking devices	Name of trunking virtual devices configured on the server.

### **EXAMPLE #7** To view information for the Ethernet channel, **trk0\_ec**, type:

\$ server\_sysconfig server\_2 -virtual -info trk0\_ec

```
server_2:

*** Trunk trk0_ec: Link is Up ***

*** Trunk trk0_ec: Statistical Load Balancing is IP ***

Device Link Duplex Speed

cge2 Up Full 1000 Mbs

cge3 Up Full 1000 Mbs
```

#### Where:

Value	Definition
Devices	Name of devices participating in this trunk.
Link	Status of the link on the specified device (can be up or down).
Duplex	Value of negotiated duplex. Values can be full or half.
Speed	Value of negotiated speed.

#### **EXAMPLE #8**

To create a fail-safe network device as a virtual device, type:

\$ server\_sysconfig server\_2 -virtual -name fsn0 -create
fsn -option "primary=cge2 device=cge2,cge3"

server\_2 : done

#### **EXAMPLE #9**

To view the fail-safe network device, type:

\$ server\_sysconfig server\_2 -virtual -info fsn0

server\_2 :

\*\*\* FSN fsn0: Link is Up \*\*\*

active=cge2 primary=cge2 standby=cge3

#### Where:

Value	Definition
active	Name of the active device in the FSN.
primary	Primary device in the FSN.
standby	Standby device in the FSN.

#### **EXAMPLE #10**

To create an aggregated link using the LACP protocol with load balancing method set to **mac**, type:

\$ server\_sysconfig server\_2 -virtual -name trk0\_la
-create trk -option "device=cge2,cge3 protocol=lacp
lb=mac"

server\_2 : done

#### EXAMPLE #11

To delete an Ethernet channel, **trk0\_ec**, type:

\$ server\_sysconfig server\_2 -virtual -delete -Force
trk0\_ec

server\_2 : done

### server\_sysstat

Displays the operating system statistics for the specified Data Movers.

#### **SYNOPSIS**

```
server_sysstat {<movername> | ALL} [-blockmap]
```

#### **DESCRIPTION**

**server\_sysstat** displays the operating system statistics and the status of blockmap memory for the specified Data Movers.

The **ALL** option executes the command for all Data Movers.

#### **OPTIONS**

#### No arguments

Displays all operating system statistics.

#### [-blockmap]

Displays the status of blockmap memory in the Data Mover. Blockmaps are paged memory used by SnapSure to locate each block of data comprising the checkpoint. Twenty-five percent of physical RAM on the Data Mover is reserved for blockmap memory.

#### **EXAMPLE #1**

To display all operating system statistics, type:

```
$ server_sysstat server_2
```

```
server_2 :
    threads runnable = 6
    threads blocked = 1399
    threads I/J/Z = 2
    memory free(kB) = 2689924
    cpu idle_% = 99
```

#### Where:

Value	Definition
threads runnable	Threads running as well as the ones waiting for a cpu to become available.
threads blocked	Threads waiting on either a condition variable or a Sthread Mutex.
threads I/J/Z	Threads in the following state: IDLE, JOINED, and ZOMBIE.
memory free (kB)	The amount of free memory from the RAM that is available to Data Mover.
cpu idle_%	Idle time of the system averaged across all the processors.

### **EXAMPLE #2** To display current blockmap memory status, type:

#### \$ server\_sysstat server\_2 -blockmap

page out rate = 0block map memory quota = 1048576 (KB)block map memory consumed = 624 (KB)

### Where:

Value	Definition
total paged in	Total number of blockmap pages paged in since the system booted.
total paged out	Total number of blockmap pages paged out since the system booted.
page in rate	Number of blockmap pages paged in per second (over last 180 seconds).
page out rate	Number of blockmap pages paged out per second (over last 180 seconds).
block map memory quota	Current value of the blockmap memory quota.
block map memory consumed	Amount of memory consumed for blockmaps.

## server\_tftp

Manages the Trivial File Transfer Protocol (TFTP) for the specified Data Movers.

#### **SYNOPSIS**

```
server_tftp {<movername>|ALL}
   -service {-status|-stop|-start|-stats}
| -info
| -set [-path <pathname>][-readaccess {all|none}][-writeaccess {all|none}]
| -clear
```

#### **DESCRIPTION**

**server\_tftp** provides a service to read and write files to and from a remote server, displays the status of the service, and the number of attempted transfers.

The **ALL** option executes the command for all Data Movers.

#### **OPTIONS**

```
-service {-status|-stop|-start|-stats}
```

Displays the status of the TFTP service, stops, and starts the service on the specified Data Mover, or displays TFTP statistics.

#### -info

Displays the number of attempted TFTP transfers and the status information for ongoing transfers.

```
-set [-path <pathname>]
```

Sets the directory <pathname> for the TFTP service to use for file transfers. If the **-set** option has been executed, re-executing the -set option with a different <pathname> overwrites the original <pathname>.

```
[-readaccess {all | none}]
Sets read access for all users or no users (default).
[-writeaccess {all | none}]
Sets write access for all users or no users (default).
```

#### -clear

Clears the path/readaccess/writeaccess settings the TFTP service for the specified Data Mover.

### **SEE ALSO** *Using TFTP on Celerra Network Server.*

**EXAMPLE #1** To start TFTP service for **server\_2**, type:

\$ server\_tftp server\_2 -service -start

server\_2 : done

**EXAMPLE #2** To display the status of the TFTP service for **server\_2**, type:

\$ server\_tftp server\_2 -service -status

server\_2 :
Tftp Running

**EXAMPLE #3** To modify a path on **server\_2** for TFTP service with read access for all,

and write access for nobody, type:

\$ server\_tftp server\_2 -set -path /ufs1 -readaccess all

-writeaccess none

server\_2 : done

**EXAMPLE #4** To display TFTP information for **server\_2**, type:

\$ server\_tftp server\_2 -info

server\_2 :

path="/ufs1/" readaccess=all writeaccess=none

**EXAMPLE #5** To display statistics for **server\_2**, type:

\$ server\_tftp server\_2 -service -stats

server\_2 :

Attempted Transfers:28
Successful Transfers:27

createdthrds:28

deletedthrds:28

timedoutthrds:0

timedoutthras:U

TotalBinds:28

TotalUnbinds:28

BindFailures:0

InvalidAttempts:0

AttemptedReadTransfers:19

SuccessfulReadTransfers:19

AttemptedWriteTransfers:9

SuccessfulWriteTransfers:8

#### Where:

Value	Definition
Attempted Transfers	Total TFTP transfers attempted by that time.
Successful Transfers	Total number of successful TFTP transfers.
createdthrds	Total number of TFTP threads created (equal to total transfers).
deletedthrds	Total number of threads deleted (equal to total created threads).
timedoutthrds	Number of timed-out threads. For TFTP transfers, in case of any failures, each thread will time out and free itself.

Value	Definition
TotalBinds	Total number of binds.
TotalUnbinds	Total number of unbinds.
BindFailures	Number of bind failures. If the port we try to bind to is in use, the bind fails, and retries with a different port.
InvalidAttempts	Invalid TFTP transfer requests from clients such as trying to transfer a non-existent file.
AttemptedReadTransfers	Total TFTP read transfers initiated.
SuccessfulReadTransfers	Total TFTP read transfers successfully completed.
AttemptedWriteTransfers	Total TFTP write transfers initiated.
SuccessfulWriteTransfer	Total TFTP write transfers successfully completed.
S	

**EXAMPLE #6** To stop TFTP service on **server\_2**, type:

\$ server\_tftp server\_2 -service -stop

server\_2 : done

**EXAMPLE #7** To delete the settings for the TFTP service on **server\_2**, type:

\$ server\_tftp server\_2 -clear

server\_2 : done

## server\_umount

Unmounts filesystems.

## **SYNOPSIS**

```
server_umount {<movername> | ALL}
   [-perm | -temp] [-Force] {-all | <fs_name> | <mount_point>}
```

## **DESCRIPTION**

**server\_umount** permanently or temporarily (default) unmounts a mounted filesystem by its <mount\_point> or <fs\_name>, or unmounts all filesystems.

The **ALL** option executes the command for all Data Movers.

## **OPTIONS**

```
-all | <fs_name > | <mount_point >
```

Unmounts all filesystems, a specific filesystem, or a filesystem specified by its mount point.

**Note:** A mount point always begins with a slash (/).

#### [-perm]

Permanently deletes the mount entry from mount table.

**Note:** The export table is not affected by deletes made with **server\_umount**. Export entries remain in the export table regardless of the mount status.

## [-temp]

Indicates that the unmount does not persist across reboots. The mount entry reappears after a reboot. Default is **-temp.** 

#### [-Force]

Forces the unmount of a production filesystem.



#### **CAUTION**

Unmounting a PFS for which automatic checkpoint scheduling (configured using the Celerra Web Manager or Celerra Native Manager) has been implemented is not recommended. Doing so will corrupt the schedule and cause checkpoint data to be lost.

## **SEE ALSO**

Managing Celerra Volumes and File Systems with Automatic Volume Management, Management Celerra Volumes and File Systems Manually, Managing Celerra for the Windows Environment, Managing Celerra for a Multiprotocol Environment, Using SnapSure on Celerra, server\_mount, server\_mountpoint, and fs\_ckpt.

**EXAMPLE #1** 

To permanently unmount a filesystem, type:

\$ server\_umount server\_2 -perm /bin

server\_2: done

**EXAMPLE #2** 

To temporarily unmount a filesystem by specifying its mount point as **/bin**, type:

\$ server\_umount server\_2 -temp /bin

server\_2: done

**EXAMPLE #3** 

To temporarily unmount a filesystem by specifying its filesystem name as **ufs1**, type:

\$ server\_umount server\_2 -temp ufs1

server\_2: done

## server\_uptime

Displays the length of time that a specified Data Mover has been running since the last reboot.

## **SYNOPSIS**

server\_uptime {<movername> | ALL}

**DESCRIPTION** server\_uptime displays the length of time that the specified Data

Mover has been running since its last reboot.

The **ALL** option executes the command for all Data Movers.

**OPTIONS** No arguments

Displays the length of time that a Data Mover has been up.

**EXAMPLE #1** To display how long a Data Mover has been up, type:

\$ server\_uptime server\_2

server\_2 : up 28 days 0 hours 15 min 41 secs

**EXAMPLE #2** To display how long all the Data Movers have been up, type:

\$ server\_uptime ALL

server\_2 : up 28 days 0 hours 15 min 41 secs
server\_3 : up 2 days 5 hours 11 min 31 secs

## server\_user

Manages user accounts for the specified Data Movers.

## **SYNOPSIS**

```
server_user {<movername> | ALL}
   -list
| -add [-md5][-passwd] <name>
| -modify <name>
| -delete <name>
| -passwd [-disable|-lock|-unlock [-force]] <name>
```

### **DESCRIPTION**

**server\_user** creates, modifies, or deletes a user account for a specified Data Mover. The **server\_user** adds or disables, locks or unlocks an optional password for a user account. Data Mover user accounts provide administrative access to certain Celerra features and third-party applications.

The ALL option executes the command for all Data Movers.

This command must be executed from the /nas/sbin directory. **su** to **root** to execute this command.

## **OPTIONS**

-list

Lists the user accounts.

```
-add [-md5][-passwd] <name>
Adds a new user account with the login <name>.
```

The **-md5** option allows you to specify MD5 password encryption for the new account.

The **-passwd** option allows you to specify a password for the new user account. Passwords have a six character minimum and cannot be empty.

```
-modify <name>
```

Modifies a user account.

```
-delete <name>
```

Deletes a user account.

```
-passwd [-disable|-lock|-unlock [-force]] <name>
Creates, changes, removes, locks, and unlocks the password for a
user. If a password lock is applied, the user account is available to
root only. The password does not have to be supplied during the
```

root only. The password lock is applied, the user account is available to root only. The password does not have to be supplied during the execution of the server\_user command. To unlock a password, use the -force option.

# **SEE ALSO** *Using Celerra FileMover, Using FTP on Celerra Network Server,* and *Configuring NDMP Backups on Celerra.*

## **EXAMPLE #1** To create a user account for **user1**, type:

# /nas/sbin/server\_user server\_2 -add user1

```
Creating new user user1
User ID: 100
Group ID: 101
Comment:
Home directory:
Shell:
```

**Note: Comment, Home directory** and **Shell** are optional, all others are required.

### EXAMPLE #2

To create a user account for NDMP connections, with MD5 password encryption and to configure the password, type:

# /nas/sbin/server\_user server\_2 -add -md5 -passwd
user\_name

```
Creating new user user_name
User ID: 101
Group ID: 100
Home directory:
Changing password for user user_name
New passwd:
Retype new passwd:
```

## **EXAMPLE #3** To list the user accounts, type:

# /nas/sbin/server\_user server\_2 -list

```
server_2:
APM000438070430000_APM000420008180000:LNEa7Fjh/43jQ:9000:9000:ftsQgHsc2oMrdysaOn
WeLhN8vB::ndmp_md5
user1:!!:100:101:::
user_name:WX72mBTFp/qV::101:100:W9z7HIndimdaHs2anCL20EBfNd::ndmp_md5
```

## **EXAMPLE #4** To modify account information for **user1**, type:

# /nas/sbin/server\_user server\_2 -modify user1

```
Modifying user account user1

1 User ID (100)

2 Group ID (101)

3 Home directory ()

4 Comment ()

5 Shell ()

Please select a field to modify, "done" to apply your changes or "quit" to cancel: 2

Group ID: 102
```

Please select a field to modify, "done" to apply your changes or "quit" to cancel: quit

## **EXAMPLE #5** To lock an account password for **ndmp**, type:

# /nas/sbin/server\_user server\_2 -passwd -lock user\_name
Changing password for user user\_name
Locking password for user user\_name

## **EXAMPLE #6** To disable the password for **user1**, type:

# /nas/sbin/server\_user server\_2 -passwd -disable user1
Changing password for user user1
Removing password for user user1

## **EXAMPLE #7** To unlock an account password for **user1**, type:

# /nas/sbin/server\_user server\_2 -passwd -unlock -force
user1

Changing password for user user1 Unlocking password for user user1

## **EXAMPLE #8** To delete a user account for **user1**, type:

# /nas/sbin/server\_user server\_2 -delete user1

## server\_usermapper

Provides an interface to manage the Internal Usermapper service.

## **SYNOPSIS**

```
server_usermapper {<movername> | ALL}
| -enable [primary=<ip>] [config=<path>]
| -disable
| -remove -all
| -Import {-user | -group} [-force] <pathname>
| -Export {-user | -group} <pathname>
```

## **DESCRIPTION**

**server\_usermapper** enables, disables, deletes, and displays the status of an Internal Usermapper service running on a Data Mover. Only one primary Usermapper service is recommended for a Celerra Network Server environment. In a single Celerra Network Server environment, there should only be one primary instance of the Usermapper service. All the other Data Movers in that environment are clients of the primary or secondary service.

**server\_usermapper** displays whether the Usermapper service is configured as a primary or secondary service. Additionally, it displays information such as the operational status of the service and, if the service is a secondary service, the IP address of the primary Usermapper service used by the secondary.

This command is relevant only for Internal Usermapper, which runs on the Data Mover. It cannot be used to manage External Usermapper, which runs on the Control Station or an external host.

The ALL option executes the command for all Data Movers.

## **OPTIONS**

## No arguments

Displays the state of the Internal Usermapper service.

#### -enable

Enables the Usermapper service. You do not need to issue this option if you are using the default Internal Usermapper configuration. In this case, primary Usermapper is automatically enabled when the NAS software is installed. You only need to issue this option if you are modifying a default Internal Usermapper configuration, or if you are upgrading from External to Internal Usermapper.

Use the **-enable** command with caution. It changes the relationship between the Data Mover and the Usermapper without confirming the change.

[primary=<ip\_address>] [config=<pathname>]

The **primary** option designates the local Usermapper service as secondary by indicating which primary service it depends on. The primary Usermapper is identified by its network IP address.

The **config** option indicates that the Usermapper service should use an existing Usermapper configuration file to define UID/GID ranges. This option is only relevant if you are upgrading from External to Internal Usermapper.

**Note:** If there is no specific reason to use particular UID and GID ranges for your environments' domains, EMC encourages you to use the automatic mapping method and let Internal Usermapper automatically assign new UIDs/GIDs based on the next available values. If you need to use an existing Usermapper configuration file, you must specify the **config** option during the upgrade procedure, that is, before Internal Usermapper has begun issuing default UIDs and GIDs.

#### -disable

Disables the Usermapper service.

#### -remove -all

Removes all entries from the Usermapper databases and destroys the database structure. The Usermapper service must be disabled before you can issue this option.



#### **CAUTION**

It is recommended that you consult with Customer Support before issuing the -remove -all option. This option deletes all Usermapper database entries and may result in users losing access to filesystems. If you decide to use the -remove -all option, you should first back up your existing Usermapper database file and usrmap.cfg file (if one is in use).

-Import {-user|-group} [-force] <pathname>
Imports Usermapper database information from the file specified by
<pathname>.

By default, only new entries are added to the Usermapper database. If an entry in the imported file does not match a similar entry in the existing database, the entry in the imported file is ignored unless the **-force** option is selected. If the **-force** option is selected, the existing database is deleted and replaced with new entries.



#### **CAUTION**

It is recommended that you consult with Customer Support before issuing the -force option. This option overwrites the existing Usermapper database file and may result in users losing access to filesystems. If you decide to use the -force option, you should first back up your existing Usermapper database file and usrmap.cfg file (if one is in use).

-Export {-user | -group} <pathname>

Exports all the SID, user, and group information from the Usermapper database to the file specified by <pathname>. You can specify any filename but the name should include the suffix .passwd or .group depending on the file type.

The **-Export** option is relevant only for a primary Usermapper service.

SEE ALSO

Configuring Celerra User Mapping.

**EXAMPLE #1** 

To verify the status of Internal Usermapper, type:

\$ server usermapper server 2

server\_2 : Usrmapper service: Enabled Service Class: Primary

#### Where:

Value	Definition
Usrmapper service:	The operational status of the service.
Service Class:	If the service is a primary or secondary service.
Primary	The IP address of the primary Usermapper service used by a secondary service. The (c) against the IP address indicates that the primary Usermapper is available and has been connected.

**EXAMPLE #2** To enable a secondary Usermapper service, type:

> \$ server\_usermapper server\_4 -enable primary=172.24.102.238

server\_4 : done

**EXAMPLE #3** 

To verify the status of Internal Usermapper for the primary Usermapper, type:

\$ server usermapper server 2

server\_2 : Usrmapper service: Enabled Service Class: Primary

```
EXAMPLE #4 To verify the status of Internal Usermapper for the secondary Usermapper, type:
```

```
$ server_usermapper server_4
```

server\_4 : Usrmapper service: Enabled
Service Class: Secondary
Primary = 172.24.102.238 (c)

**EXAMPLE #5** To export user information from the Usermapper database, type:

\$ server\_usermapper server\_2 -Export -user
/home/nasadmin/users\_server\_2.passwd

server\_2 : done

**EXAMPLE #6** To export group information from the Usermapper database, type:

\$ server\_usermapper server\_2 -Export -group
/home/nasadmin/group server\_2.group

server\_2 : done

**EXAMPLE #7** To import the user file users\_server\_2.passwd for server\_2, type:

\$ server\_usermapper server\_2 -Import -user
/home/nasadmin/users\_server\_2.passwd

server\_2 : done

**EXAMPLE #8** To import the group file **group\_server\_2.group** for **server\_2**, type:

\$ server\_usermapper server\_2 -Import -group
/home/nasadmin/group\_server\_2.group

server\_2 : done

**EXAMPLE #9** To disable an Internal Usermapper service, type:

\$ server\_usermapper server\_2 -disable

server\_2 : done

**EXAMPLE #10** To remove all entries from the Usermapper database, type:

\$ server\_usermapper server\_2 -remove -all

server\_2 : Warning: This operation will erase all user/group mappings. CIFS users may lose access. Continue(Y/N):

done

## server\_version

Displays the software version running on the specified Data Movers.

## **SYNOPSIS**

```
server_version {<movername> | ALL}
```

**DESCRIPTION** server\_version displays the version of the software for the specified

Data Mover.

The **ALL** option executes the command for all Data Movers.

**OPTIONS** No arguments

Displays the software version.

**SEE ALSO** nas\_version.

**EXAMPLE #1** To display the software version for a Data Mover, type:

\$ server\_version server\_2

server\_2 : Product: EMC Celerra File Server Version: T5.5.80.0

**EXAMPLE #2** To display the system software version for all Data Movers, type:

### \$ server\_version ALL

```
server_2 : Product: EMC Celerra File Server Version: T5.5.80.0
server_3 : Product: EMC Celerra File Server Version: T5.5.80.0
server_4 : Product: EMC Celerra File Server Version: T5.5.80.0
server_5 : Product: EMC Celerra File Server Version: T5.5.80.0
```

## server\_viruschk

Manages the virus checker configuration for the specified Data Movers.

### **SYNOPSIS**

```
server_viruschk {<movername> | ALL}
   [-audit | -update]
| [-set accesstime={now|none|[[[[<yy>]<mm>]<dd>]<hh>]<mm>[.<ss>]}]
| [-fsscan [<fs_name> {-list|-create [offline]|-delete}]]
```

#### DESCRIPTION

**server\_viruschk** displays, audits, and updates the virus checker configuration file for the specified Data Mover.

The virus checker configuration file is named viruschecker.conf and is located on the Control Station in the /nas/sys directory. This is a template file and should not be edited directly. Copy the viruschecker.conf file to another directory for edits using a text editor, then once the changes have been saved, the file must be copied to a Data Mover using the server\_file command.

To start and stop the virus checker for a Data Mover, server\_setup provides information.

The **ALL** option executes the command for all Data Movers.

## **OPTIONS**

## No arguments

Displays the virus checker configuration.

#### -audit

Displays the status of virus checker, including the number of files checked, and their progress.

#### -update

Integrates the changes made to the virus configuration file without stopping the virus checker.

```
-set accesstime={now|none|[[[[<yy>]<mm>]<dd>]<hh>]<mm>[.<ss>]}
```

Sets the reference time on the virus checker configuration file to **now**, disables the time scan, or sets the access time in the specified format.

#### -fsscan

Displays the filesystem scan status for all filesystems.

[<fs\_name> {-list|-create [offline]|-delete}]]
The -list option displays the scan status for the specified filesystem. The -create option initializes a full scan on the filesystem <fs\_name> and the offline options allow the filesystem scan on all offline files. By default, offline filesystems are not included. The -delete option stops the scan.

**SEE ALSO** *Using Celerra AntiVirus Agent*, server\_checkup, and server\_setup.

## **EXAMPLE #1** To display the virus checker configuration, type:

```
$ server_viruschk server_2
server_2 :
10 threads started
1 Checker IP Address(es):
172.24.102.18
               ONLINE at Mon Jan 31 18:35:43 2005 (GMT-00:00)
              RPC program version: 3
             CAVA release: 3.3.5, AV Engine: Network Associates
              Last time signature updated: Thu Jan 27 19:38:35 2005 (GMT-00:00)
31 File Mask(s):
*.exe *.com *.doc *.dot *.xl? *.md? *.vxd *.386 *.sys *.bin *.rtf *.obd *.dll
*.scr *.obt *.pp? *.pot *.ole *.shs *.mpp *.mpt *.xtp *.xlb *.cmd *.ovl *.dev
*.zip *.tar *.arj *.arc *.z
No File excluded
Share \\DM112-CGE0\CHECK$
RPC request timeout=25000 milliseconds
RPC retry timeout=5000 milliseconds
High water mark=200
Low water mark=50
Scan all virus checkers every 60 seconds
When all virus checkers are offline:
Continue to work with Virus Checking and CIFS
Scan on read if access Time less than Thu Jan 27 19:38:35 2005 (GMT-00:00)
Panic handler registered for 65 chunks
```

#### Where:

Value	Definition
threads started	The number of threads that have been started.
Checker IP Address(es)	The number of VC servers defined in /.etc/viruschecker.conf.
version=2	CAVA uses RPC program version 2.
Share	The UNC name used by CAVA to access the Data Mover.
RPC request timeout=	Time out for the full CAVA request.
RPC retry timeout=	Time out for one unitary CAVA request.
High water mark=	A log event is generated when the number of files in the request queue becomes greater than 200.
Low water mark=	A log event is generated when the number of files in the request queue become less than 50.

Value	Definition
Panic handler registered for 65 chunks	Panic is used to memorize name of unchecked files.
ERROR_SETUP	List of errors reported by CAVA.
min=, max=, average=	Min, max, and average time for CAVA requests.

## **EXAMPLE #2** To display the status of the virus checker, type:

## \$ server\_viruschk server\_2 -audit

```
server_2 :
Total Requests
                                        : 138
Requests in progress
                                        : 25
NO ANSWER from the Virus Checker Servers: 0
ERROR_SETUP
FILE_NOT_FOUND
ACCESS_DENIED
                                        : 0
FAIL
                                        : 0
TIMEOUT
                                        : 0
Total Infected Files
                                        : 875
Deleted Infected Files
                                        : 64
                                       : 0
Renamed Infected Files
Modified Infected Files
                                       : 811
min=70915 uS, max=1164891 uS, average=439708 uS
15 File(s) in the collector queue
10 File(s) processed by the AV threads
Read file '/.etc/viruschecker.audit' to display the list of pending requests
```

## Where:

Value	Definition
Total Infected Files	The number of files found that contained viruses. This displays only if infected files are found and remains visible until the Data Mover is rebooted or the CAVA viruschecking service has been restarted.
Deleted Infected Files	The number of files that contained viruses that were deleted. This displays only if infected files are found and remains visible until the Data Mover is rebooted or the CAVA viruschecking service has been restarted.
Renamed Infected Files	The number of files that contained viruses that were renamed. This displays only if infected files are found and remains visible until the Data Mover is rebooted or the CAVA viruschecking service has been restarted.
Modified Infected Files	The number of files that contained viruses that were modified. This displays only if infected files are found and remains visible until the Data Mover is rebooted or the CAVA viruschecking service has been restarted.

**EXAMPLE #3** To update the virus checker configuration file that is resident on the Data Mover, type:

\$ server\_viruschk server\_2 -update

server\_2 : done

**EXAMPLE #4** To set the access time for the virus checker configuration file, type:

\$ server\_viruschk server\_2 -set accesstime=now

server\_2 : done

**EXAMPLE #5** To start a scan on the filesystem, type:

\$ server\_viruschk server\_2 -fsscan ufs1 -create

server\_2 : done

**EXAMPLE #6** To check the scan of a filesystem, type:

\$ server\_viruschk server\_2 -fsscan ufs1 -list

server\_2 :

FileSystem 24 mounted on /ufs1:

8 dirs scanned and 22 files submitted to the scan engine firstFNN=0x0, lastFNN=0xe0f34b70, queueCount=0, burst=10

**EXAMPLE #7** To check the scan status on all filesystems, type:

\$ server\_viruschk server\_2 -fsscan

server\_2 :

FileSystem 24 mounted on /ufs1:

8 dirs scanned and 11 files submitted to the scan engine firstFNN=0x0, lastFNN=0xe0eba410, queueCount=0, burst=10 FileSystem 25 mounted on /ufs2:

9 dirs scanned and 11 files submitted to the scan engine firstFNN=0x0, lastFNN=0xe0010b70, queueCount=0, burst=10

**EXAMPLE #8** To stop a scan on a filesystem, type:

\$ server\_viruschk server\_2 -fsscan ufs1 -delete

server\_2 : done

## server\_vtlu

#### **SYNOPSIS**

Configures a virtual tape library unit (VTLU) on the specified Data Movers.

```
server_vtlu {<movername>|ALL}
   -service <service_options>
| -tlu <tlu_options>
| -storage <storage_options>
| -tape <tape_options>
| -drive <drive_options>
| -help
```

## **DESCRIPTION**

**server\_vtlu** creates, configures, and displays information on virtual tape library units on the specified Data Movers. At this time, **server\_vtlu** supports only NDMP backups.

To get detailed options for VTLU service, TLU, storage, tape, and drive, type **-help** for the full usage, or type only **-service**, **-tlu**, **-tape**, and **-drive**, respectively, after typing the <movername>.

ALL executes the command for all Data Movers.

#### SERVICE OPTIONS

```
-service {-info|-set -chain <chain_id>}
```

The **-info** option lists the chain IDs for the VTLU service on the Data Mover. The **-set -chain** option specifies a <chain\_id> for the VTLU device connection. The default chain ID is 0.

#### **TLU OPTIONS**

```
-tlu
  {-list
 -info <tlu_id>
 -new [-robot {[-vendor < vendor>][-product < product>]
    [-revision <revision>] } ]
    [-slots <num_slots>]
    [-impexp <num_impexp_slots>] [-drives <num_drives>]
    [-drive {[-vendor < vendor>][-product < product>]
    [-revision <revision>]}]
 -modify <tlu_id>
  {[-robot {[-vendor <vendor>][-product <product>][-revision <revision>]}]
  [-slots <num_slots>][-impexp <num_impexp_slots>][-drives <num_drives>]
  [-drive {[-vendor < vendor>][-product < product>]
   [-revision <revision>]}]}
 -delete <tlu_id>}
                    -tlu -list
```

Lists all VTLUs configured on the specified Data Movers.

```
-tlu -info <tlu_id>
```

Displays information for the specified VTLU.

```
-tlu -new [-robot {[-vendor <vendor>][-product
<preduct>][-revision <revision>]}]
```

Creates a VTLU on the specified Data Mover with the specified robot's vendor, product, and revision information. If this information is not specified, the default values of **EMC**, **svtlRobot**, and **1.0** are used. These values have character limits of 8, 16, and 4, respectively.

```
[-slots <num_slots>]
```

Configures the number of slots in the VTLU. If no value is defined, then the default value of 32 is used.

```
[-impexp <num_impexp_slots>]
```

Configures the number of import/export slots in the VTLU. If no value is defined, the default value of 8 is used.

```
[-drives <num_drives>]
```

Configures the number of virtual tape drives in the VTLU. If no value is defined, then the default value of 4 is used.

```
[-drive {[-vendor <vendor>][-product
<preduct>][-revision <revision>]}]
```

Specifies the vendor name, product name, and revision number of the virtual drive. If no values are defined, the default values of **EMC**, **svtlDrive**, and **1.0** are used. These values have character limits of 8, 16, and 4 respectively.

```
-tlu -modify <tlu_id> [-robot {[-vendor <vendor>]
[-product <product>][-revision <revision>]}]
```

Modifies the vendor name, product name, and revision number for the virtual robot.

```
[-slots <num slots>]
```

Modifies the number of virtual slots in the VTLU.

```
[-impexp <num_impexp_slots>]
```

Modifies the number of virtual import/export slots in the VTLU.

```
[-drives <num_drives>]
```

Modifies the number of virtual tape drives in the VTLU.

```
[-drive {[-vendor <vendor>][-product
<preduct>][-revision <revision>]}]
```

Modifies the vendor name, product name, and revision number for the VTLU drive.

#### -tlu -delete <tlu id>

Deletes a configured VTLU. The storage associated with a VTLU must be deleted before the VTLU can be deleted.

#### STORAGE OPTIONS

```
-storage
{-list [<tlu_id>]
| -new <fs_name> -tlu <tlu_id> [-tapesize <size>G][-tapes <num_tapes>]
| [-barcodeprefix <barcode_prefix>]
| [-destination {slot|impexp|vault}]
| -extend <fs_name> -tlu <tlu_id> [-tapesize <size>G][-tapes <num_tapes>]
| [-destination {slot|impexp|vault}]
| -import <fs_name> -tlu <tlu_id>
| [-destination {slot|impexp|vault}]
| -export <fs_name> -tlu <tlu_id>
| -destination {slot|impexp|vault}]
| -export <fs_name> -tlu <tlu_id>
| -delete {<fsname>|-barcodeprefix <barcode_prefix>} -tlu <tlu_id>}
| -tlu <tlu_id>]
| Lists storage information on the specified VTLU.
| -storage -new <fs_name> -tlu <tlu_id>
| Creates storage on an existing VTLU.
```

## [-tapesize <size>G]

Specifies the size of the virtual tapes in the filesystem. If no value is defined, and the **-tape** <num\_tapes> option is not defined, then the default value of 50 GB is used. If <num\_tapes> is defined but the tape size is not, then the tape size is determined by the number of tapes and the filesystem size.

```
[-tapes <num_tapes>]
```

Specifies the number of virtual tapes in the filesystem. The default value is determined by the filesystem size and the value of **-tapesize** <size>.

```
[-barcodeprefix <barcode_prefix>]
```

Specifies the barcode prefix for the virtual tapes. The barcode prefix is the same across a filesystem. If no value is defined, then a four-character string is assigned automatically.

```
[-destination {slot | impexp | vault}]
```

Specifies the destination of the virtual tape as a slot, an import/export slot, or the vault. Slot is the default location.

```
-storage -extend <fs_name> -tlu <tlu_id>
```

Adds more tapes to a filesystem that has been extended.

```
[-tapesize <size>G]
```

Specifies the size of the virtual tape. If no value is defined, and the **-tapes** <num\_tapes> is not defined, then the default value of 50 GB is used. If the <num\_tapes> is defined but <size> is not, then the tape size is determined by the number of tapes and the filesystem size.

```
[-tapes <num_tapes>]
```

Specifies the number of tapes to be added to a VTLU. The default value is determined by the size of the filesystem and the tape size.

```
[-destination {slot|impexp|vault}]
```

Specifies the destination of the virtual tape as a slot, an import/export slot, or the vault. Slot is the default location.

```
-storage -import <fs_name> -tlu <tlu_id>
```

Imports a filesystem that was exported from another VTLU. The configuration of a filesystem persists when it is imported.

```
[-destination {slot|impexp|vault}]
```

Specifies the destination of the imported storage as a slot, an import/export slot, or the vault. Slot is the default location.

```
-storage -export <fs_name> -tlu <tlu_id>
```

Exports storage from a VTLU, which breaks the association between the filesystem and the VTLU. Tapes must be ejected from the VTLU and placed in the vault before you can export storage.

```
-storage -delete {<fs_name>|-barcodeprefix
<barcode_prefix>} -tlu <tlu_id>
```

Deletes the association between storage and the VTLU and cleans up related metadata.

## **TAPE OPTIONS**

## -tape

```
{-list <tlu_id> [-storage <fs_name>]
| -info {<tape_barcode> | -all} -tlu <tlu_id>
| -insert <tape_barcode> -tlu <tlu_id>
| -eject <tape_barcode> -tlu <tlu_id>}
| -drive {-list <tlu_id>
| -info <drive_id> -tlu <tlu_id>
| -unmount <drive_id> -tlu <tlu_id>}
```

```
-tape {-list <tlu_id> [-storage <fs_name>]}
```

Lists the tapes configured on a VTLU, along with their associated filesystems.

-tape -info <tape\_barcode|-all> -tlu <tlu\_id>
Lists information on a particular tape, or on all tapes in a VTLU.

-tape -insert <tape\_barcode> -tlu <tlu\_id>
Moves a virtual tape from the vault to an import/export slot.

-tape -eject <tape\_barcode> -tlu <tlu\_id>
Ejects a tape from the import/export slot and places it in the vault.

-drive {-list <tlu\_id>}

Lists the virtual tape drives associated with the specified VTLU.

-info <drive\_id> -tlu <tlu\_id> -id

Displays information on the specified tape drive.

-unmount <drive\_id> -tlu <tlu\_id>

Unmounts the tape from the drive and places it in its original source slot. If the source slot is unavailable, the tape is placed in another available slot.

## SEE ALSO

Configuring NDMP Backups to Disk on Celerra, Celerra Network Server Parameters Guide, and nas\_fs.

#### **EXAMPLE #1**

To set the chain\_ID for **server\_2**, type:

\$ server\_vtlu server\_2 -service -set -chain 1

server\_2 : done

## **EXAMPLE #2**

To list the chain\_ID for the VTLU service on **server\_2**, type:

\$ server\_vtlu server\_2 -service -info

server\_2 :
starting chain id = 1

## Where:

Value	Definition
	Starting chain of the VTLU device name. The device name format is c <x>t<y>l<z>, where <x> is the starting chain. <y> and <z> are the target and LUN, respectively. The target and LUN values cannot be modified.</z></y></x></z></y></x>

#### **EXAMPLE #3**

To create and configure a new VTLU on **server\_2**, type:

\$ server\_vtlu server\_2 -tlu -new -robot -vendor EMCCorp -product vtluRobot -revision 1.1a -slots 256 -impexp 64 -drives 2 -drive -vendor EMCCorp -product vtluDrive -revision 2.2a

server 2 :

id = 3
slots = 256
import/export slots = 64
robot vendor = EMCCorp

robot product = vtluRobot robot revision = 1.1a

robot serial number = P8gIgqs2k5
robot device name = c1t010

drives = 2

## Where:

Value	Definition
id	Unique VTLU identifier that is assigned automatically.
slots	Number of virtual slots in the VTLU.
import/export slots	Number of virtual import/export slots in the VTLU.
robot vendor	Vendor name of the virtual robot; maximum length is eight characters.
robot product	Product name of the virtual robot; maximum length is 16 characters.
robot revision	Revision number of the virtual robot; maximum length is four characters.
robot serial number	Serial number of the virtual robot that is assigned automatically.
robot device name	Device name of the virtual robot; only the first number, the starting chain, can be modified.
drives	Number of virtual drives in the VTLU.
drive vendor	Vendor name of the virtual drive; maximum length is eight characters.
drive product	Product name of the virtual drive; maximum length is 16 characters.
drive revision	Revision number of the virtual drive; maximum length is four characters.

## **EXAMPLE #4** To list all the VTLUs on **server\_2**, type:

\$ server\_vtlu server\_2 -tlu -list

server\_2 :
id vendor product revision serial\_number device\_name
3 EMCCorp vtluRobot 1.1a P8gIgqs2k5 c1t010

#### Where:

***************************************		
Value	Definition	
id	Unique VTLU identifier that is assigned automatically.	
vendor	Vendor name of the virtual robot; maximum length is eight characters.	
product	Product name of the virtual robot; maximum length is 16 characters.	
revision	VTLU robot's revision number; maximum length is four characters.	
serial_number	VTLU serial number that is assigned automatically.	
device_name	The device name of the VTLU robot; only the first number, the starting chain, can be modified.	

# **EXAMPLE #5** To display the information for the VTLU on the Data Mover identified by its ID, type:

\$ server\_vtlu server\_2 -tlu -info 3

server\_2 : id

= 3

slots = 256 import/export slots = 64 = EMCCorp robot vendor robot product = vtluRobot robot revision = 1.1arobot serial number = P8qIqqs2k5 robot device name = c1t010drives drive vendor = EMCCorp drive product = vtluDrive drive revision = 2.2a**EXAMPLE #6** To modify vendor, product, and revision information for the robot and drive of VTLU 3 for server\_2, type: \$ server vtlu server 2 -tlu -modify 3 -robot -vendor EMC -product vRobot -revision 1.1b -drives 3 -drive -vendor EMC -product vDrive -revision 2.2b server\_2 : done **EXAMPLE #7** To modify the number of virtual import/export slots and number of virtual slots of VTLU 1 for **server\_2**, type: \$ server\_vtlu server\_2 -tlu -modify 3 -slots 8 -impexp 4 server\_2 : done **EXAMPLE #8** To add new storage for VTLU 3 on server\_2, with 5 virtual tapes of 1 GB located in slots each with barcode prefix dstpre, using **ufs1** filesystem, type: \$ server\_vtlu server\_2 -storage -new ufs1 -tlu 3 -tape size 1G -tapes 5 -barcodeprefix dstpre -destination slot server\_2 : done **EXAMPLE #9** To extend VTLU 3 on server\_2 by adding 2 virtual tapes of 1 GB and placing them in the import/export virtual slots, type: \$ server\_vtlu server\_2 -storage -extend ufs1 -tlu 3 -tapesize 1G -tapes 2 -destination impexp server\_2 : done **EXAMPLE #10** To export storage from VTLU 3 stored on ufs1 located on server\_2, type: \$ server\_vtlu server\_2 -storage -export ufs1 -tlu 3 server\_2 : done **EXAMPLE #11** To import the ufs2 filesystem to VTLU 3 and place the virtual tapes in the vault, type: \$ server\_vtlu server\_2 -storage -import ufs1 -tlu 3

-destination vault

server\_2 : done

## **EXAMPLE #12** To list the storage on VTLU 3, type:

\$ server\_vtlu server\_2 -storage -list 3

server\_2 :

tlu\_id filesystem barcode\_prefix

3 ufs1 dstpre

## Where:

Value	Definition	
tlu_id	Unique VTLU identifier that is assigned automatically.	
filesystem	Name of the filesystem associated with the VTLU.	
barcode_prefix	Modifiable prefix assigned to virtual tapes that is constant across a filesystem.	

## **EXAMPLE #13** To list VTLU information on VTLU 3, type:

## \$ server\_vtlu server\_2 -tape -list 3

server_2 :				
barcode	filesystem	capacity(GB)	location	source_slot
dstpre0001	ufs1	1	vault	
dstpre0002	ufs1	1	vault	
dstpre0003	ufs1	1	vault	
dstpre0004	ufs1	1	vault	
dstpre0005	ufs1	1	vault	
dstpre0006	ufs1	1	vault	
dstpre0000	ufs1	1	impexp:0	

## Where:

Value	Definition
barcode	Virtual tape barcode, consisting of the modifiable barcode prefix and a four-digit number that is assigned automatically.
filesystem	Name of the filesystem.
capacity (GB)	Wirtual tape capacity in GB.
location	Element type and element ID of the virtual tape; possible element types are slot, drive, import/export, robot, and vault.
source_slot	Slot ID of the tape's previous location.

## **EXAMPLE #14**

To insert the specified tape in a virtual import/export slot on VTLU 3, type:

\$ server\_vtlu server\_2 -tape -insert dstpre0001 -tlu 3

server\_2 : done

## **EXAMPLE #15**

To eject the specified tape from VTLU 3, type:

\$ server\_vtlu server\_2 -tape -eject dstpre0001 -tlu 3

server\_2 : done

## **EXAMPLE #16** To list the storage drive on VTLU **3**, type:

## \$ server\_vtlu server\_2 -drive -list 3

server\_2 :

drive\_id device\_name serial\_number status tape\_barcode 0 c1t0l1 NXB2w4W000 empty

1 c1t012 3u0bx4W000 empty 2 c1t013 g0pgy4W000 empty

## Where:

Value	Definition	
drive_id	Unique VTLU drive identifier that is assigned automatically.	
device_name	The device name of the VTLU drive.	
serial_number	The VTLU serial number that is automatically assigned.	
status	Status of the virtual tape drive; possible values are empty, loaded, and in use.	
tape_barcode	Barcode of the virtual tape if status is not empty.	

## **EXAMPLE #17** To display information for drive 0 on VTLU 3, type:

```
$ server_vtlu server_2 -drive -info 0 -tlu 3
```

status = empty tape\_barcode =

## **EXAMPLE #18** To delete storage from VTLU 3, type:

```
$ server_vtlu server_2 -storage -delete ufs1 -tlu 3
```

server\_2 : done

## **EXAMPLE #19** To delete VTLU **3** from **server\_2**, type:

\$ server\_vtlu server\_2 -tlu -delete 3

server\_2 : done

The server Commands	



## Using Celerra Command Scripts

This appendix contains information about command scripts that can be used to automate many Celerra tasks.

## Celerra scripting guidelines

Table 2 on page 712 lists guidelines to keep in mind when creating Celerra command scripts.

Table 2 Scripting guidelines

Issue	Action
Scheduling	NAS Database Backups:  The Celerra Network Server backs up the NAS database that stores specific configuration information required for each Data Mover every hour, at one minute after the hour. During part of the backup, the database is locked, and some commands that rely on the database might not have access. It is recommended that command scripts avoid starting at one minute after the hour. Note that scripts with complex commands that run for an extended period may overlap the backup period.  The duration of the backup may vary. Use the following Linux command to check the state of the backup process prior to executing scripts: ps -ef   grep nasdb_backup. If a lock condition occurs, wait a few minutes and retry.
Command sequencing	Some commands must lock the database in order to execute. If multiple user-entered commands or scripts are active at the same time, some of these commands may lock the database and prevent other commands from executing. To avoid this, you should arrange commands whenever possible.
Sleep statements	Some processes within a script can take time to complete. Use proper timing and adequate sleep statements to prevent timing-related issues.
Pipe and grep	Piping script outputs through <b>grep</b> is a helpful tool to check the status of the script. Use periodic checks to grep for file or database locked messages, timeouts, resource unavailable warnings, and other failure or success messages, and use this information to check status, pause the script, or halt it. Detailed information about error messages can be found in the error messages guide.
Return code check	All commands return a UNIX-style return code (for example: 0 for success or 1 for failure) or a text-based status code (for example, done) which can be used to help determine if the command completed or if there was an error or a conflict with the NAS database backup, or other commands being run. If a lock condition occurs, wait a few minutes and retry. If you create and run scripts, be sure to incorporate return code checks and verify for proper return codes from individual operations.

## Scripting examples

The RECOVERABLE variable contains the following errors to retry on:

- Unable to acquire lock
- Resource temporarily unavailable
- Unable to connect to host
- Socket: All ports in use
- Database resource vanished
- ◆ Connection timed out
- NAS\_DB locked object is stale

An example of what the RECOVERABLE variable looks like is as follows:

```
RECOVERABLE="unable to acquire lock|Resource temporarily unavailable|unable to connect to host|socket: All ports in use|database resource vanished|Connection timed out|NAS_DB locked object is stale".
```

The res variable contains the command output:

```
#!/bin/sh
# File: nas_cmdrcvr.sh
# Created by: NAS Engineering
# Date: Thursday, May 25, 2006
# Version: 1.0
# Notes:
# 1) script will retry commands for specified period of
time
#2) script will log messages to file only if there's
available disk space
NAS_DB=/nas
export NAS_DB
PATH=$PATH:$NAS_DB/bin:$NAS_DB/sbin:/usr/sbin:.
export PATH
RETRIES=60
SLEEPTIME=30
LOGDIR="/home/nasadmin"
LOGFILE="$0.log"
LOGLAST="$0.last"
DISKFULL=98
RECOVERABLE="Resource temporarily unavailable | \
unable to acquire lock \
```

```
unable to connect to host \
socket: All ports in use \
database resource vanished \
Connection timed out \
NAS_DB locked object is stale"
# function to log messages to a file
nas_log()
DISKCHK=`df -k $LOGDIR|awk 'NR>1{print $5}'|sed 's/\%//'`
# if there's enough free disk space, append to log
if [ $DISKCHK -lt $DISKFULL ]; then
TDSTAMP=`date +%Y-%m-%d' '%T`
echo $TDSTAMP: $LOGMSG >> $LOGDIR/$LOGFILE
fi
# regardless of available space, always write last error
echo $TDSTAMP: $LOGMSG > $LOGDIR/$LOGLAST
# function to execute (and potentially retry) commands
nas_cmd()
# initialize variable(s)
retry_count=0
# loop until either successful of retry count exceeded
while [ $retry_count -le $RETRIES ]; do
# execute command and gather response
RES=`$CMD 2>&1`
# check if response means command is recoverable
if [ `echo "$RES" | egrep -c "$RECOVERABLE" ` -ne 0 ]; then
# check retry count
if [ $retry_count -ne $RETRIES ]; then
# retry count has not been exceeded
LOGMSG="Command ($CMD) failed with ($RES)...retrying in
$SLEEPTIME s"
nas_log
sleep $SLEEPTIME
else
# retry count has been exceeded
```

```
LOGMSG="Command ($CMD) failed with ($RES)...exiting
(retry count of $RETRIES exceeded) "
nas_log
exit 1
fi
else
# command was either successful or failed for an unknown
reason
LOGMSG="Command ($CMD) successful with ($RES)"
nas log
retry_count=$RETRIES
exit 0
fi
#increment counter for retries
retry_count=`expr $retry_count + 1`
done
}
# main
CMD="nas_volume -d mtv1"
nas_cmd
```

# Using the NAS database and query facility

EMC has partially changed the layout or format of Celerra internal databases. This change can impact the use of **awk** or **grep** utilities when used in scripts that assume specific positions of fields in databases.

To enable searching of the NAS database, Celerra has developed a new query subsystem that appears as a hidden option on some of the nas\_commands. This query subsystem enables you to specify the information you are interested in, allows you to format the output, and is independent of the database format.



#### CAUTION

Do not use grep and awk to scan the database files. Database positions may change and substrings may return false matches for database objects.

Following is an example of a query to view unused disks:

```
nas disk -query:inuse==n -format:'%s\n' -Fields:Id
```

To filter root disks, refer to the List all non-root disks that are not in use.

## **Examples**

Use the following commands to view the tags (fields) that you can query:

```
nas_disk -query:tags
nas_fs -query:tags
nas_volume -query:tags
nas_slice -query:tags
```

Table 3 on page 716 contains a list of examples to help you get started. Note that these commands can be run on the Control Station CLI, so the hardcoded values can be replaced with shell script variables.

Table 3 Task and query examples (page 1 of 2)

Task	Example
Query the ID of a named file system	nas_fs -query:Name==RLL_fs10 -format:'%s\n' -Fields:Id
Query the ID of a named file system without the new line	nas_fs -query:Name==RLL_fs10 -format:'%s' -Fields:Id
Query the name of a file system ID that corresponds to a particular ID	nas_fs -query:id==20 -format:'%s\n' -Fields:Name
List of all server IDs	nas_server -query:* -format:'%s\n' -Fields:Id
List of all server names	nas_server -query:* -format:'%s\n' -Fields:Name
List of all the checkpoint file systems	<pre>nas_fs -query:type==ckpt -fields:name -format:"%s\n"</pre>
List type of file system with ID 20	nas_fs -query:id==20 -format:'%s\n' -Fields:Type
List the file systems that are in use	<pre>nas_fs -query:inuse==y -format:'%s\n' -Fields:Name or nas_fs -query:inuse==y -format:'%s\n' -Fields:Id</pre>
Identify file system of which file system ID 28 is a backup	nas_fs -query:id==28 -format:'%s' -Fields:BackupOf
List the name of the server with ID 2	<pre>nas_server -query:id==2 -format:'%s\n' -fields:name`"</pre>
View which volume file system is built on	<pre>nas_fs -query:Name==my_fs -format:%d -fields:VolumeID</pre>

Table 3 Task and query examples (page 2 of 2)

Task	Example
View the block count of meta volume	nas_volume -query:Name==my_meta3 -format:%d -fields:Blocks
View the block size of meta volume	nas_volume -query:Name==JAH_meta3 -format:%d -fields:BlockSize
Find which server IDs use fs123	<pre>nas_fs -query:name==fs123 -format:'%s\n' -fields:ServersNumeric</pre>
List all non-root disks that are not in use	<pre>nas_disk -query:inuse==n:IsRoot==False -format:"%s\n" -fields:name</pre>
List unused volumes that contain "dc" in the volume name	nas_volume -query:inuse==n:IsRoot==False:name=dc -format:"%s\n" -fields:name
List all available disks on a particular storage device (symm_id is a script/env variable)	<pre>nas_disk -query:inuse==n:Sym- mID==\$symm_id:IsRoot==False -format:"%s\n" -fields:name</pre>

## **Query operators** Use the operators in the table below when building your queries:

Operator	Definition
=	Having the pattern (sub-string match)
==	Exact string match
=-	Integer Minimum (not less than)
=+	Integer Maximum (not more than)
=*	Any
=^	NOT having the pattern
=^=	NOT an exact match
=^-	NOT Integer Minimum (is less than)
=^+	NOT Integer Maximum (is more than)
=^*	NOT Any (None)

# GNU General Public License

This section contains the GNU General Public License (GPL). The GPL is the license for the Linux operating system. All EMC software, including the Celerra Network Server software, is licensed by the EMC Software License included in the software kit.

## **GNU General Public License**

Version 2, June 1991

Copyright © 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330 Boston, MA 02117-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

## **Preamble**

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

## Terms and conditions for copying, distribution, and modification

This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification.") Each licensee is addressed as "you."

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does. This license states that:

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an

appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

- 2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:
  - a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
  - b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
  - c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise

the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

- 3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:
  - a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
  - b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
  - c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

- 4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.
- 5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.
- 6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.
- 7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

- 8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.
- 9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version," you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

## NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

## Index

/.etc/homedir 412 /etc/passwd 109 /nas/sbin 25 /nas/sys directory 696 /nasmcd/sbin 25	Authentication CIFS 411 Data Mover 118 HTTP 509 kerberos 415 Automatic checkpoint scheduling 685 Automatic Data Mover reboot 450 Automatic File System Extension 192 Auto-migration processes 385
Access control levels creating 109 deleting 109 file systems 194, 310, 332, 354 storage systems 167, 332 volumes 353 Access list, iSCSI 531 Access procedures local 13	Automount map 112 automount daemon 112 conflicting list 112 infile 112 outfile 112 Automounting 112 Autonegotiation 672 Autonegotiation, for Gigabit Ethernet 672
remote 13 ACL database 221, 422 action, nas_event 185 Activating license keys 230 time services 451 Active Directory database 419 addressing link-local 518	Backing up file systems 363 bcpio 370 BCV 173, 176, 252 Block limit, quotas 265 Blockmap memory 680 Bus speed 675, 676
Aliases, using server_cifs 415 Antivirus 394 ARP table 377 Attached storage system, displaying 331 Auditing CIFS configuration 412 AUTH_SYS 477	C Case-sensitive, names and options 16 CDMS 385 Celerra Data Migration Service 385 Celerra FileMover 58, 59, 60 CIFS connections 64 creating an HTTP connection 509

HTTP configuration file 508	CLARiiON pools 196, 199, 202
HTTP connections 67,70	CLARiiON storage pools 202, 250
NFS connections 59, 61	CLATA 173, 175, 252
NFS server 63	Clone volumes 353
parameters 59	Clone, volume 354
Celerra Replicator 287	CLSTD 173, 175, 252
Chain number, SCSI 463, 464	CMATA 173, 175, 252
CHAP security 623	CMSTD 173, 175, 252
Checking configuration parameters 394	CNS series 676
Checkpoint 28	Command set
Checkpoint file system 30	conventions 15
creating 30	synopsis 15
refreshing 30	Command-line interface
Checkpoint scheduling 135	local access 13
CIFS 411	remote access 13
ACL database 422	Comments 9
activating 626	COMPAT, MPD 219
adding NFS service 418	Component 394
alias 415	Configuration parameters 394
auditing configuration 412	Configuration settings 150
authentication 411	Connection ID 60
computer name 414	Control Station 13
configuration 411	accessing the command-line interface 13
connections 64	dual Control Stations 25
creating a share 475	external Usermapper 691
dialect 420	failover 25
DNS 415	local access 13
FQDN 418	NAS database 193
homedir 412	nas_commands 16
local users 413	remote access 13
NetBIOS 412	starting NFSv4 service 592
network backups 59	volume table 353
Network neighborhood 413	Conventions 15
passwords 418	Converting hostnames
security 419	using DNS 473
security IDs 422	Converting IP addresses
security,CIFS	using DNS 473
Kerberos 411	cpio 370
statistics 422	CPU, monitoring 449
usermapper 411	Creating
CIFS client 59	TimeFinder/FS copy 100
CIFS shares 475	Current log 564
cifsnanoroundup 574	cvfsname 572
cifssyncwrite 573	
CLARiiON 331	<b>D</b>
CLARiiON array properties 332	D
CLARiiON backend support 202, 250	Daemons

automount 112	DIR3, MPD 219
Data migration 385	Disk space 471
Data Mover	Disk table 171
activating the standby 645	deleting disks 171
authentication 118	Disk types
current log 564	BCV 173, 175, 252
date and time 451	CLATA 173, 175, 252
device configuration 463	CLSTD 173, 175, 252
disk space 471	CMATA 173, 175, 252
halting 449	CMSTD 173, 175, 252
hardware configuration 463	R1BCV 173, 175, 252
inodes 471	R2BCV 173, 175, 252
internal Usermapper 691	R2STD 173, 175, 252
listing the server table 309	STD 173, 175, 252
mounting file systems 570	Displaying
operating system statistics 680	disk space 471
pinging 615, 617	Displaying system parameters 603
rebooting 449	Distinguished name 557
security policy settings 623	DNLC 658
standby	DNS 415, 473
designating a policy 646	activating 473
policy types 646	lookup server 473
system software version 695	server 473
time zone 451	Documentation, related 7
types	Domain, Windows NT 413
nas 309	Dual Control Stations, cs_standby 25
rdf 309	,, ,, ,, ,
standby 309	E
vdm 309	<del>-</del>
unmounting file systems 685, 696	ECHO_REPLY
user accounts 688	messages receive 615, 617
usermapper 691	ECHO_REQUEST 615, 617
Data Mover, GPO, using server_security 623	messages send 615, 617
Data Mover, motherboard type 671	EMC Active Directory 419
Data Mover, protocol 626	emctar 371
Data Movers	Enabling
standby 645	licenses 230
Dependency 394	software 230
Destination Celerra Network Servers 167	Enabling software 230
Destination file, using server_file 491	Error messages
Destination volume 355	server_archive 376
Device configuration 463	Ethernet
probe, save, display 463	changing default values 672
Devices 463	speed, duplex 672
Diagnostic messages	Event configuration file 190
server_archive 364	Event configuration files 185
Dialect, using server cifs 420	Event store 185

F acility, nas_event 185 Failover     activating a standby 645     Control Station 25 Fail-safe network device 673, 679 FarCopy environment 20 Fiber channel 463 File system     access control values 194, 310, 332, 354     aclchk 221     Auto extension 192     checking inodes 471     checkpoint 28, 30     connections 58     default type 198, 201     deleting 40, 193, 521, 522     extending 195     file-level retention 200     fisch 221     high watermark 201     information 193     mount points 578     notifications 573 File-Level Retention 91, 192, 198 FileMover connections 82 FQDN 64 Free disk space 471 FTP users 196 FTP, using server_file 491 Full sanity check 394  Gateway address 621 getreason 26 GID 109 GID, quotas 261, 262 Gigabit Ethernet 516 Gigabit Ethernet 516 Gigabit Ethernet Fiber 672 Gigabit Ethernet Copper 672 Gigabit Ethernet, autonegotiation 672 Gigabit Ethernet, autonegotiation 672 Gigabit switch 672 Gipabit switch 672 Gipabit switch 672 Gipabit Ethernet, autonegotiation 672 Gipabit Ethernet, autonegotiation 672 Gipabit Ethernet, autonegotiation 672 Gipabit switch 672 Gipabit switch 672 Gipabit switch 672 Gipabit ethernet, autonegotiation
File-Level Retention 91, 192, 198 Failover     activating a standby 645     Control Station 25 Fail-safe network device 673, 679 FarCopy environment 20 Fiber channel 463 File system     access control values 194, 310, 332, 354     aclchk 221     Auto extension 192     checking inodes 471     checkpoint 28, 30     connections 58     default type 198, 201     deleting 40, 193, 521, 522     extending 195     file-level retention 200     figle-level retention 200     figle-level retention 200     figle-level retention 193     maximum size 201     minimum volume size 354     mounting 570     nbpi 198, 201
Faillover activating a standby 645 Control Station 25 Fail-safe network device 673, 679 FarCopy environment 20 Fiber channel 463 File system access control values 194, 310, 332, 354 aclchk 221 Auto extension 192 checking inodes 471 checkpoint 28, 30 connections 58 default type 198, 201 deleting 40, 193, 521, 522 extending 195 file-level retention 200 fisck 221 high watermark 201 information 193 maximum size 201 minimum volume size 354 mixed-access checking 194 mounting 570 nbpi 198, 201  FileMover HTTP 509 FileMover connections 82 FQDN 64 Free disk space 471 Fru using server_file 491 Full sanity check 394  FileMover HTTP 509 File Author File Aut
Failover activating a standby 645 Control Station 25 Fail-safe network device 673, 679 FarCopy environment 20 Fiber channel 463 File system access control values 194, 310, 332, 354 aclchk 221 Auto extension 192 checking inodes 471 checkpoint 28, 30 connections 58 default type 198, 201 deleting 40, 193, 521, 522 extending 195 file-level retention 200 fsck 221 high watermark 201 information 193 maximum size 201 minimum volume size 354 mixed-access checking 194 mounting 570 nbpi 198, 201
activating a standby 645 Control Station 25 Fail-safe network device 673, 679 FarCopy environment 20 Fiber channel 463 File system  access control values 194, 310, 332, 354 aclchk 221 Auto extension 192 checking inodes 471 checkpoint 28, 30 connections 58 default type 198, 201 deleting 40, 193, 521, 522 extending 195 file-level retention 200 file-level retention 200 file-level retention 193 maximum size 201 minimum volume size 354 mixed-access checking 194 mounting 570 nbpi 198, 201  File Mover connections 82 FQDN 64 Free disk space 471 FTP users 196 FTP, using server_file 491 Full sanity check 394  File disk space 471 FTP users 196 FTP, using server_file 491 Full sanity check 394  File disk space 471 FTP users 196 FTP, using server_file 491 Full sanity check 394  File disk space 471 FTP users 196 FTP, using server_file 491 Full sanity check 394  File disk space 471 FTP users 196 FTIP users 196 FTP, using server_file 491 Full sanity check 394  File disk space 471 FTP users 196 FTP users 196 FTP, using server_file 491 Full sanity check 394  File disk space 471 FTP users 196 FTP users 196 FTP, using server_file 491 Full sanity check 394  File disk space 471 FTP users 196 FTP users 196 FTP, using server_file 491 Full sanity check 394  File disk space 471 FTP users 196 FTP users 196 FTP, using server_file 491 Full sanity check 394  File disk space 471 FTP users 196 FTP
Control Station 25 Fail-safe network device 673, 679 FarCopy environment 20 Fiber channel 463 File system  access control values 194, 310, 332, 354 aclchk 221 Auto extension 192 checking inodes 471 checkpoint 28, 30 connections 58 default type 198, 201 deleting 40, 193, 521, 522 extending 195 file-level retention 200 fisck 221 high watermark 201 information 193 maximum size 201 minimum volume size 354 mixed-access checking 194 mounting 570 nbpi 198, 201  FQDN 64 Free disk space 471 FTP users 196 FTP, using server_file 491 Full sanity check 394  Gateway address 621 getreason 26 GID 109 GID, quotas 261, 262 Gigabit Ethernet 516 Gigabit Ethernet Fiber 672 Gigabit Ethernet Fiber 672 Gigabit switch 672 GPO settings 624 GPO, group policy object 623 Group quotas 260 Group, file system 87
Fail-safe network device 673, 679 FarCopy environment 20 Fiber channel 463 File system  access control values 194, 310, 332, 354 aclchk 221 Auto extension 192 checking inodes 471 checkpoint 28, 30 connections 58 default type 198, 201 deleting 40, 193, 521, 522 extending 195 file-level retention 200 file-level retention 200 file watermark 201 information 193 maximum size 201 minimum volume size 354 mixed-access checking 194 mounting 570 nbpi 198, 201  Free disk space 471 FTP users 196 FTP, using server_file 491 Full sanity check 394  Gateway address 621 getreason 26 GID 109 GID, quotas 261, 262 Gigabit Ethernet 516 Gigabit Ethernet 516 Gigabit Ethernet Fiber 672 Gigabit switch 672 Gipo group policy object 623 Group policy object 623 Group, file system 87
FarCopy environment 20 Fiber channel 463 File system  access control values 194, 310, 332, 354 aclchk 221 Auto extension 192 checking inodes 471 checkpoint 28, 30 connections 58 default type 198, 201 deleting 40, 193, 521, 522 extending 195 file-level retention 200 file-level retention 200 file-level retention 193 maximum size 201 minimum volume size 354 mixed-access checking 194 mounting 570 nbpi 198, 201  FTP users 196 FTP, using server_file 491 Full sanity check 394  Gateway address 621 getreason 26 GID 109 GID, quotas 261, 262 Gigabit Ethernet 516 Gigabit Ethernet Copper 672 Gigabit Ethernet Fiber 672 Gigabit Ethernet, autonegotiation 672 Gigabit switch 672 GPO settings 624 GPO, group policy object 623 Group quotas 260 Group, file system 87
Fiber channel 463 File system  access control values 194, 310, 332, 354 aclchk 221 Auto extension 192 checking inodes 471 checkpoint 28, 30 connections 58 default type 198, 201 deleting 40, 193, 521, 522 extending 195 file-level retention 200 fisck 221 high watermark 201 information 193 maximum size 201 minimum volume size 354 mixed-access checking 194 mounting 570 nbpi 198, 201  Auto extension 194 Gateway address 621 getreason 26 GID 109 GID, quotas 261, 262 Gigabit Ethernet 516 Gigabit Ethernet Copper 672 Gigabit Ethernet Fiber 672 Gigabit Ethernet, autonegotiation 672 GPO settings 624 GPO, group policy object 623 Group quotas 260 Group, file system 87
File system  access control values 194, 310, 332, 354 aclchk 221  Auto extension 192 checking inodes 471 checkpoint 28, 30 connections 58 default type 198, 201 deleting 40, 193, 521, 522 extending 195 file-level retention 200 fisck 221 high watermark 201 information 193 maximum size 201 minimum volume size 354 mixed-access checking 194 mounting 570 nbpi 198, 201   Gateway address 621 getreason 26 GID 109 GID, quotas 261, 262 Gigabit Ethernet 516 Gigabit Ethernet Fiber 672 Gigabit Ethernet Fiber 672 Gigabit Switch 672 Gigabit switch 672 GPO settings 624 GPO, group policy object 623 Group quotas 260 Group, file system 87
access control values 194, 310, 332, 354 aclchk 221 Auto extension 192 checking inodes 471 checkpoint 28, 30 connections 58 default type 198, 201 deleting 40, 193, 521, 522 extending 195 file-level retention 200 fsck 221 high watermark 201 information 193 maximum size 201 minimum volume size 354 mixed-access checking 194 mounting 570 nbpi 198, 201  Gateway address 621 getreason 26 GID 109 GID, quotas 261, 262 Gigabit Ethernet 516 Gigabit Ethernet 516 Gigabit Ethernet Fiber 672 Gigabit Ethernet Fiber 672 Gigabit Ethernet, autonegotiation 672 GPO settings 624 GPO, group policy object 623 Group quotas 260 Group, file system 87
aclchk 221 Auto extension 192 checking inodes 471 checkpoint 28, 30 connections 58 default type 198, 201 deleting 40, 193, 521, 522 extending 195 file-level retention 200 fsck 221 high watermark 201 information 193 maximum size 201 minimum volume size 354 mixed-access checking 194 mounting 570 nbpi 198, 201  Gateway address 621 getreason 26 GID 109 GID, quotas 261, 262 Gigabit Ethernet 516 Gigabit Ethernet 516 Gigabit Ethernet Copper 672 Gigabit Ethernet Fiber 672 Gigabit Ethernet, autonegotiation 672 GPO, group policy object 623 Group policy object 623 Group quotas 260 Group, file system 87
checking inodes 471 checkpoint 28, 30 connections 58 default type 198, 201 deleting 40, 193, 521, 522 extending 195 file-level retention 200 figh watermark 201 information 193 maximum size 201 minimum volume size 354 mixed-access checking 194 mounting 570 nbpi 198, 201  Gateway address 621 getreason 26 GID 109 GID, quotas 261, 262 Gigabit Ethernet 516 Gigabit Ethernet Filer 672 Gigabit Ethernet Fiber 672 Gigabit switch 672 Gigabit switch 672 GPO settings 624 GPO, group policy object 623 Group policy object 623 Group, file system 87
checking inodes 471 checkpoint 28, 30 connections 58 default type 198, 201 deleting 40, 193, 521, 522 extending 195 file-level retention 200 figh watermark 201 information 193 maximum size 201 minimum volume size 354 mixed-access checking 194 mounting 570 nbpi 198, 201  GID 109 GID, quotas 261, 262 Gigabit Ethernet 516 Gigabit Ethernet Fiber 672 Gigabit Ethernet Fiber 672 Gigabit Ethernet, autonegotiation 672 Gigabit switch 672 GPO settings 624 GPO, group policy object 623 Group quotas 260 Group, file system 87
checkpoint 28, 30 connections 58 default type 198, 201 deleting 40, 193, 521, 522 extending 195 file-level retention 200 fisck 221 high watermark 201 information 193 maximum size 201 minimum volume size 354 mixed-access checking 194 mounting 570 nbpi 198, 201  GID, quotas 261, 262 Gigabit Ethernet 516 Gigabit Ethernet Copper 672 Gigabit Ethernet Fiber 672 Gigabit Ethernet, autonegotiation 672 Gigabit switch 672 Gigabit switch 672 Gigabit switch 672 Gigabit switch 672 Group policy object 623 Group policy object 623 Group quotas 260 Group, file system 87
connections 58 default type 198, 201 deleting 40, 193, 521, 522 extending 195 file-level retention 200 fisck 221 high watermark 201 information 193 maximum size 201 minimum volume size 354 mixed-access checking 194 mounting 570 nbpi 198, 201  GID, quotas 261, 262 Gigabit Ethernet 516 Gigabit Ethernet Copper 672 Gigabit Ethernet Fiber 672 Gigabit Switch 672 Gigabit ethernet 516 Gigabit Ethernet 516 Gigabit Ethernet 516 Gigabit Ethernet 70 Gigabit 80 Gigabit Ethernet 70 Gigabit 80 Gi
default type 198, 201 deleting 40, 193, 521, 522 extending 195 file-level retention 200 fisck 221 high watermark 201 information 193 maximum size 201 minimum volume size 354 mixed-access checking 194 mounting 570 nbpi 198, 201  GID, quotas 261, 262 Gigabit Ethernet 516 Gigabit Ethernet Copper 672 Gigabit Ethernet Fiber 672 Gigabit Ethernet, autonegotiation 672 Gigabit switch 672 Gigabit ethernet Copper 672 Gigabit Ethernet Copper 672 Gigabit Ethernet Fiber 672 Gigabit Ethernet Fiber 672 Gigabit Ethernet Copper 672 Gigabit Ethernet Fiber 672 Gigabit Et
deleting 40, 193, 521, 522  extending 195  Gigabit Ethernet 516  Gigabit Ethernet Copper 672  Gigabit Ethernet Fiber 672  Gigabit Ethernet Fiber 672  Gigabit Ethernet, autonegotiation 672  Gigabit switch 672  Gigabit switch 672  Gigabit switch 672  Gigabit Sthernet, autonegotiation 672  Gigabit switch 672  Gigabit Sthernet, autonegotiation 672  Gigabit Sthernet, auto
extending 195  file-level retention 200  fisck 221  high watermark 201  information 193  maximum size 201  minimum volume size 354  mixed-access checking 194  mounting 570  nbpi 198, 201  Gigabit Ethernet Copper 672  Gigabit Ethernet, autonegotiation 672  Gigabit switch 672  Gigabit Switch 672  Gigabit Ethernet, autonegotiation 672  Gigabit Stepper 672  Gigabit Ethernet Copper 672  Gigabit Ethernet, autonegotiation 672  Gigabit Ethernet, aut
file-level retention 200 fisck 221 Gigabit Ethernet Fiber 672 Gigabit Ethernet, autonegotiation 672 Gigabit switch 672 Gigabit switch 672 Gigabit switch 672 Gigabit switch 672 GPO settings 624 GPO, group policy object 623 minimum volume size 354 mixed-access checking 194 Group quotas 260 Group, file system 87  Gigabit Ethernet Fiber 672 Gigabit Ethernet, autonegotiation 672 GPO settings 624 GPO, group policy object 623 Group policy object 623 Group quotas 260 Group, file system 87
fsck 221 high watermark 201 information 193 maximum size 201 minimum volume size 354 mixed-access checking 194 mounting 570 nbpi 198, 201  Gigabit Ethernet, autonegotiation 672 Gigabit switch 672 GPO settings 624 GPO, group policy object 623 Group policy object 623 Group quotas 260 Group, file system 87
high watermark 201 information 193 GPO settings 624 maximum size 201 GPO, group policy object 623 minimum volume size 354 Group policy object 623 Group policy object 623 Group quotas 260 Group, file system 87 nbpi 198, 201
maximum size 201 GPO, group policy object 623 minimum volume size 354 Group policy object 623 mixed-access checking 194 Group quotas 260 mounting 570 Group, file system 87 nbpi 198, 201
maximum size 201 GPO, group policy object 623 minimum volume size 354 Group policy object 623 mixed-access checking 194 Group quotas 260 mounting 570 Group, file system 87 nbpi 198, 201
minimum volume size 354 Group policy object 623 mixed-access checking 194 Group quotas 260 mounting 570 Group, file system 87 nbpi 198, 201
mixed-access checking 194 Group quotas 260 Group, file system 87 nbpi 198, 201
nbpi 198, 201
percentage threshold 29, 30, 31
read-only 571
read-write 571  Halting a Data Mover 449  Hauting a Data Mover 449  Health checkup 120
renaming 194 Health checkup 130 High watermark 201
SIZE 174
total capacity 471 Home directory, enabling 412 Hostname requirement
types
ckpt 193 server_ping 615, 617 HTTP
group 193 authentication 509
mgfs 193, 385 migrants 103 client 68, 71
mirrorfs 193 configuration file 508
nmfs 193 connections 67
for Celerra FileMover, 508
uxfs 193 LINIX permissions 104 protocol 512
ONIX perhussions 194 requests 67 71
threads 509
Windows permissions 194 HTTP connections, Celerra FileMover 67, 70

HTTPS	realm 547
connections 70	Kerberos Domain Controller 419
	Kerberos, authentication 415
	Kerberos, user and data authentication 477
	keyword, nas_event 185
I18N 312	
ICMP	L
ECHO_REPLY 615,617	
ECHO_REQUEST 615, 617	LACP, aggregation 674
Initiator, iSCSI 531	Ldap 557
Inodes 471	License key 230
Inodes, displaying 471	License, software 230
Internal Usermapper service 691	Licensing
IP protocol, assigning 515	nas_license 230
iPlanet	Link aggregation 673, 674
client 555, 556	Link Aggregation protocol 671
domain 557	Linked Celerra Network Servers 117, 347
IP-to-MAC address 377	link-local address 518
setting 377	Linux 25
iSCSI	Load balancing 674
access list 531	Loading a debug/patch image 626
configuration 526	Local access 13
iSNS 526	Log files 232, 563
logical unit numbers (LUNs) 526	LUN masks 301
LUN 529	LUNs, for Mirror View/S 167
name service management 526	
portal group 530	M
Qualified Name 527	MAC addresses 674
standard 527	Magnetic tape drive 583
targets 526	Managing protocols 626
Unicode characters 527	Mask, iSCSI 530
virtual provisioning 529	Maximum Transmission Unit size 515
ISCSI replication	MD5 password encryption 688
nas_replicate 154, 192, 272, 287, 347, 526, 585,	MGFS 385
603	mgfs 385
iSCSI replication 287	Migrated file system 385
nas_replicate 526, 555	MirrorView/S 167, 236
iSNS 526	MirrorView/Synchronous 167, 236
	Mixed access-checking, for file systems 194
K	Mixed backend system 354
Kadmin 549	Mother board 675
KDC 547	Mount options 571
Kerberos authentication 590	read-only 571
Kerberos configuration 547	read-write 571
domain 547	Mount points
Kadmin 549	creating, listing, deleting 578
KDC 547	Mount protocol 386
	r

mountd daemon 62	MTU size 515
Mounting a file system 570	Network interfaces, changing default values 672
MPD 194, 218	Network Neighborhood 413
MPFS 580	NFS connections 61
protocol 580	NFS connections, Celerra FileMover 59, 61
threads 580, 581	NFS port 386
MPFS client 581	NFS protocol 61
MPFS clients 571	NFS server, Celerra FileMover 63
MTU 515	nfsd daemon 62
MTU size	NFSv2 385, 590
for Ethernet 516	NFSv3 385, 590
for Gigabit Ethernet 516	NFSv4 590, 592
Multi-Protocol Directory 219	NFSv4 clients 596
Multiprotocol directory 194	nfsv4delegation 574
Waldprotocol affectory 191	NIS
11	activating 601
N	deleting 601
Name service managemen, iSNSt 526	NIS client 601
NAS 626	NIS domain 557
NAS database 193	NIS server requirement
nas_dbtable 163	
nbpi 198, 201	server_ping 615, 617 NMFS 202
NDMP	nonotify 573
connections 689	
creating user accounts 690	nooplock 573
passwords 690	noprefetch 572 noscan 573
NDMP backups 700	
NearCopy environment 20	Notifications, for mounted file systems 573
Nested mount file system 193	notifyonaccess 573
Nested mountpoints 193	notifyonwrite 573
NetBIOS name 415	ntcredential 573
NetBIOS name, alias 415	NTLM 415, 416
NetBIOS names 481	_
Netgroup 479	0
Netgroups 479	Offset default 320
Network	Offset, Slice Volume 320
activate DNS client 473	OpenLDAP
display statistics 586	client 556
operating system statistics 680	OpenLDAP client 555
send ECHO_REQUEST messages 615, 617	OpenLDAP domain 557
Network adapter card 671	Operating system statistics 680
Network interface cards	Oplock, default 573
speed 671	7,
Network interfaces	n
creating an interface 515	P
enabling and disabling 515	Parameters 603, 604
IP protocol 515	PAX 604
protect 010	Password encryption 688

Password, for users 688	Quotas, block limit 265
PAX 611	_
PAX parameters 604  Paymissions for UNIX and Windows 104	R
Permissions, for UNIX and Windows 194	R1 20
PFS 28	R1 and R2 20
Ping 615, 617	R1BCV 173, 176, 252
Policy type, for Standby	R1STD 173, 176, 252
auto 646	R2 20
manual 646	R2BCV 173, 176, 252
retry 646	R2STD 173, 176, 252
Portal groups 530	RDF file systems 91
Portmapper 386	RDF relationship for remote Celerra 117, 348
POSIX compatibility 366	RDF relationships 645
Power fail/crash recovery 450	Realm, for Kerberos 549
Primary Control Station 25	Rebooting a Data Mover 449
Primary Data Mover	Remote access 13
activating the standby 645	Remote Celerra
restoring the standby 645	cel_fs 20
Production file system 28	R1 and R2 20
Production file system (PFS) 28	SRDF 20
Protocol, setting up 626	Remote Celerra Network Server 117, 348
Protocols	Remote Celerra Network Servers
CDMS 626	
CIFS 626	data migration 385 Remote disk 173
ICMP 586	
IP 586, 674	Remote file system 20
MAC 674	Remote NFS server, communicating with 385
managing 626	Remotely-linked, Celerra Network Server 20
MPFS 580, 626	Remotely-linked, file systems 20
NFS 592	Replication for iSCSI 287
RIP 619	Replication for remote Celerra 117, 348
RPC 592	Replication sessions
TCP 586, 674	displaying 289
TFTP 682	listing 288
UDP 586	Restoring the standby Data Mover 645
virus checker 696	RIP 619
VIRUSCHK 626	RIP configuration 619
	Root 478
^	root 109
Q	Routing entries 620
Quota configuration 261	Routing table 586
Quotas	RPC bind 386
edit 261, 262	RPC statistics 593
group 260, 261, 262	
proto user 261, 262	S
report 261, 262	Sanity check 394
user 260	SavVol 28

sbin 26	setting policy type 646
Scheduling 135	Standby Data Mover, activating 626
SCSI device table 464	Statistics
listing 464	CIFS 422
SCSI devices 179, 463	network 586
health check failures 464	NFS 592
listing 464	operating system 680
Secondary file system 59, 61, 64, 67, 70	routing table 586
Secure NFS service 590	RPC 592
Security IDs 422	STD 173, 176, 252
Security modes 420	Storage pools
Security policy settings 623	creating 245
Security, CIFS 411	extending 245
Security, for CIFS 419	system-defined 245
Server tables, listing 309	user-defined 245
Server types	Storage systems 167, 332
nas, standby 626	Stub files 60
server_log	Subnet 479
examples 563	sv4cpio 371
Slice entry	sv4crc 371
deleting 320	Symmetrix backend support 202, 251, 313
Slice information	Symmetrix pools 196
displaying 320	Symmetrix storage pools 202, 251, 313
Slice table	Synchronizing time 451
listing 320	Synopsis 15
Slice Volume	syslog file 29
offset 320	System events 185
Slice volumes	System health checkup 130
creating 320	System parameters 603
SMB statistics 422, 423	System version, displaying 351
SnapSure 28	systems 570
SnapSure checkpoint 135	
SnapSure checkpoints 194	T
SNMP	Tables
agent 629	ARP 377
values 629	mounted file systems 570
Software packages, nas_license 230	NIS 601
Software version, for Data Movers 695	routing 620
Source file, using server_file 491	SCSI device table 464
Source volume 355	slice 320
SRDF ink 117	Tape backup 583
SRDF relationship, initializing 274	tar 371
SRDF-linked, Celerra 117 Standalona CIES carvar, 416	TCP 386, 473, 516
Standalone CIFS server 416	TFTP 682
Standby	Threads
activating 645	for CIFS, CDMS, or virus checker 626
restoring 645	·

HTTP 509	V
MPFS 580	VDMs 309
Time services	displaying 310
using server_date 451	loading 311
Time synchronization 451	Virtual Data Movers 309
Time zone 451	virtual device 674
TimeFinder/FS 87, 98	Virtual devices 673
TimeFinder/FS Far Copy 91	Virtual provisioning 192, 196, 200, 529
Timing sources 451	Virtual robot 701
Trap, nas_event 185	Virtual tape library unit 700
Tree IDs 264	Virtual tapes 706
Tree quotas 262	Virus Checker
triggerlevel 573	noscan 573
Trivial File Transfer Protocol (TFTP) 682	Virus checker 696, 697
Trunked Ethernet channel 674	auditing 696
Trusted relationship 118	configuration file 696
	upgrading 696
U	viruschecker.conf 696
UDP 386, 473, 516	VLAN 516
UDP sockets 586	VLTU 700
UID 109	Volume table 353
UID, quotas 261, 262	Volumes
Umask 481	access control levels 353
uncached 573	cloning 353
Unicode characters 527	creating 353
UNIX permissions 194	deleting 353
UNIX/Linux Kerberos KDC 547	displaying information 353
User accounts 688	extending 354
User accounts for NDMP 690	getting info 353
User-defined storage pools 245	listing 353
Usermapper 421, 691	minimum size for file system 354
Usermapper database 694	renaming 353
Usermapper, CIFS 411	
usrmap.cfg file 692	W
ustar 371	Windows NT 4.0 387
UTC 423	Windows NT 4.0 domain 413
UTF-8 419	Windows NT Credential 573
Utilities	Windows permissions 194
aclchk 220	Windows popups 59
fsck 220	WINS server(s) 420

Index	