Sun Cluster 3.1 Data Service for Network File System (NFS)

Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054
U.S.A.

Part No: 817–1534
May 2003, Revision A

Copyright 2003 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. All rights reserved.

This product or document is protected by copyright and distributed under licenses restricting its use, copying, distribution, and decompilation. No part of this product or document may be reproduced in any form by any means without prior written authorization of Sun and its licensors, if any. Third-party software, including font technology, is copyrighted and licensed from Sun suppliers.

Parts of the product may be derived from Berkeley BSD systems, licensed from the University of California. UNIX is a registered trademark in the U.S. and other countries, exclusively licensed through X/Open Company, Ltd.

Sun, Sun Microsystems, the Sun logo, docs.sun.com, AnswerBook, AnswerBook2, and Solaris are trademarks, registered trademarks, or service marks of Sun Microsystems, Inc. in the U.S. and other countries. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

The OPEN LOOK and Sun™ Graphical User Interface was developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

Federal Acquisitions: Commercial Software–Government Users Subject to Standard License Terms and Conditions.

DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

Copyright 2003 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. Tous droits réservés.

Ce produit ou document est protégé par un copyright et distribué avec des licences qui en restreignent l'utilisation, la copie, la distribution, et la décompilation. Aucune partie de ce produit ou document ne peut être reproduite sous aucune forme, par quelque moyen que ce soit, sans l'autorisation préalable et écrite de Sun et de ses bailleurs de licence, s'il y en a. Le logiciel détenu par des tiers, et qui comprend la technologie relative aux polices de caractères, est protégé par un copyright et licencié par des fournisseurs de Sun.

Des parties de ce produit pourront être dérivées du système Berkeley BSD licenciés par l'Université de Californie. UNIX est une marque déposée aux Etats-Unis et dans d'autres pays et licenciée exclusivement par X/Open Company, Ltd.

Sun, Sun Microsystems, le logo Sun, docs.sun.com, AnswerBook, AnswerBook2, et Solaris sont des marques de fabrique ou des marques déposées, ou marques de service, de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays. Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux Etats-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc.

L'interface d'utilisation graphique OPEN LOOK et Sun™ a été développée par Sun Microsystems, Inc. pour ses utilisateurs et licenciés. Sun reconnaît les efforts de pionniers de Xerox pour la recherche et le développement du concept des interfaces d'utilisation visuelle ou graphique pour l'industrie de l'informatique. Sun détient une licence non exclusive de Xerox sur l'interface d'utilisation graphique Xerox, cette licence couvrant également les licenciés de Sun qui mettent en place l'interface d'utilisation graphique OPEN LOOK et qui en outre se conforment aux licences écrites de Sun.

CETTE PUBLICATION EST FOURNIE "EN L'ETAT" ET AUCUNE GARANTIE, EXPRESSE OU IMPLICITE, N'EST ACCORDEE, Y COMPRIS DES GARANTIES CONCERNANT LA VALEUR MARCHANDE, L'APTITUDE DE LA PUBLICATION A REPONDRE A UNE UTILISATION PARTICULIERE, OU LE FAIT QU'ELLE NE SOIT PAS CONTREFAISANTE DE PRODUIT DE TIERS. CE DENI DE GARANTIE NE S'APPLIQUERAIT PAS, DANS LA MESURE OU IL SERAIT TENU JURIDIQUEMENT NUL ET NON AVENU.

030314@5533

Contents

Preface

The Sun Cluster 3.1 Data Service for Network File System (NFS) contains procedures to install and configure Sun Cluster HA for NFS on your Sun Cluster nodes.

This document is intended for system administrators with extensive knowledge of Sun software and hardware. Do not use this document as a planning or presales guide. Before reading this document, you should have already determined your system requirements and purchased the appropriate equipment and software.

The instructions in this document assume knowledge of the Solaris™ operating environment and expertise with the volume manager software used with Sun Cluster.

UNIX Commands

This document contains information on commands specific to installing and configuring Sun Cluster data services. It might not contain information on basic UNIX® commands and procedures, such as shutting down the system, booting the system, and configuring devices. For that information, see one or more of the following:

- Online documentation for the Solaris software environment
- Solaris operating environment man pages
- Other software documentation that you received with your system

Typographic Conventions

Typeface or Symbol	Meaning	Examples
`AaBbCc123`	The names of commands, files, and directories; on-screen computer output	Edit your file. Use `ls -a` to list all files. `% You have mail.`
`AaBbCc123`	What you type, when contrasted with on-screen computer output	`% su` `Password:`
AaBbCc123	Book titles, new words or terms, words to be emphasized	Read Chapter 6 in the *User's Guide*. These are called *class* options. You *must* be superuser to do this.
	Command-line variable; replace with a real name or value	To delete a file, type `rm` *filename*.

Shell Prompts

Shell	Prompt
C shell	*machine_name*`%`
C shell superuser	*machine_name*#
Bourne shell and Korn shell	$
Bourne shell and Korn shell superuser	#

Related Documentation

Application	Title	Part Number
Installation	*Sun Cluster 3.1 Software Installation Guide*	816-3388
Data Services Installation and Configuration	*Sun Cluster 3.1 Data Service Planning and Administration Guide*	817–1526
	Sun Cluster 3.1 Data Service for Apache	817–1532
	Sun Cluster 3.1 Data Service for BroadVision One-To-One Enterprise	817–1542
	Sun Cluster 3.1 Data Service for Domain Name Service (DNS)	817–1533
	Sun Cluster 3.1 Data Service for Netbackup	817–1538
	Sun Cluster 3.1 Data Service for Network File System (NFS)	817–1534
	Sun Cluster 3.1 Data Service for Oracle	817–1527
	Sun Cluster 3.1 Data Service for Oracle Parallel Server/Real Application Clusters	817–1535
	Sun Cluster 3.1 Data Service for SAP	817–1536
	Sun Cluster 3.1 Data Service for SAP liveCache	817–1539
	Sun Cluster 3.1 Data Service for Siebel	817–1540
	Sun Cluster 3.1 Data Service for Sun ONE Application Server	817–1530
	Sun Cluster 3.1 Data Service for Sun ONE Directory Server	817–1529
	Sun Cluster 3.1 Data Service for Sun ONE Message Queue	817–1531
	Sun Cluster 3.1 Data Service for Sun ONE Web Server	817–1528
	Sun Cluster 3.1 Data Service for Sybase ASE	817–1537
	Sun Cluster 3.1 Data Service for WebLogic Server	817–1537
	Sun Cluster 3.1 Data Service for DHCP	817–1716
	Sun Cluster 3.1 Data Service for Samba	817–1715
	Sun Cluster 3.1 Data Service for WebSphere MQ	817–1714

Application	Title	Part Number
	Sun Cluster 3.1 Data Service for WebSphere MQ Integrator	817–1713
Hardware	*Sun Cluster 3.x Hardware Administration Manual*	817–0168
	Sun Cluster 3.x Data Service Collection at `http://docs.sun.com/db/coll/1024.1`	
API development	*Sun Cluster 3.1 Data Services Developer's Guide*	816-3385
Administration	*Sun Cluster 3.1 5/03 System Administration Guide*	816-3384
Cluster concepts	*Sun Cluster 3.1 Concepts Guide*	816-3383
Error Messages	*Sun Cluster 3.1 Error Messages Guide*	816-3382
Man Pages	*Sun Cluster 3.1 Reference Manual*	816–5251
Release notes	*Sun Cluster 3.1 Data Service 5/03 Release Notes*	817–1790
	Sun Cluster 3.1 Release Notes	816-5317
	Sun Cluster 3.1 Release Notes Supplement	816-3381

Sun Documentation Online

The `docs.sun.com`[SM] Web site enables you to access Sun technical documentation on the Web. You can browse the `docs.sun.com` archive or search for a specific book title or subject at `http://docs.sun.com`.

Help

If you have problems installing or using Sun Cluster, contact your service provider and provide the following information:

- Your name and E-mail address (if available)
- Your company name, address, and phone number
- The model and serial numbers of your systems
- The release number of the operating environment (for example, Solaris 8)
- The release number of Sun Cluster (for example, Sun Cluster 3.0)

Use the following commands to gather information about each node on your system for your service provider.

Command	Function
`prtconf -v`	Displays the size of the system memory and reports information about peripheral devices.
`psrinfo -v`	Displays information about processors.
`showrev –p`	Reports which patches are installed.
`prtdiag -v`	Displays system diagnostic information.
`scinstall -pv`	Displays Sun Cluster release and package version information.

Also have available the contents of the `/var/adm/messages` file.

Installing and Configuring Sun Cluster HA for Network File System (NFS)

This chapter describes the steps to install and configure Sun Cluster HA for Network File System (NFS) on your Sun Cluster nodes.

This chapter contains the following procedures.

You must configure Sun Cluster HA for NFS as a failover data service. See "Planning for Sun Cluster Data Services" in *Sun Cluster 3.1 Data Service Planning and Administration Guide* and the *Sun Cluster 3.1 Concepts Guide* document for general information about data services, resource groups, resources, and other related topics.

Note – You can use SunPlex Manager to install and configure this data service. See the SunPlex Manager online help for details.

Use the worksheets in *Sun Cluster 3.1 Release Notes* to plan your resources and resource groups before you install and configure Sun Cluster HA for NFS.

The NFS mount points that are placed under the control of the data service must be the same on all of the nodes that can master the disk device group that contains those file systems.

Caution – If you use VERITAS Volume Manager, you can avoid "stale file handle" errors on the client during NFS failover. Ensure that the `vxio` driver has identical pseudo-device major numbers on all of the cluster nodes. You can find this number in the `/etc/name_to_major` file after you complete the installation.

Installing and Configuring Sun Cluster HA for NFS

The following table lists the sections that describe the installation and configuration tasks.

TABLE 1–1 Task Map: Installing and Configuring Sun Cluster HA for NFS

Task	For Instructions
Install Sun Cluster HA for NFS packages	"Installing Sun Cluster HA for NFS Packages" on page 12
Set up and configure Sun Cluster HA for NFS	"Registering and Configuring Sun Cluster HA for NFS" on page 13
Configure resource extension properties	"Configuring Sun Cluster HA for NFS Extension Properties" on page 21
View fault monitor information	"Sun Cluster HA for NFS Fault Monitor" on page 23

Installing Sun Cluster HA for NFS Packages

Use the `scinstall`(1M) utility to install the data service package, `SUNWscnfs`, on the cluster.

If you installed the `SUNWscnfs` data service package during your initial Sun Cluster installation, proceed to "Registering and Configuring Sun Cluster HA for NFS" on page 13. Otherwise, use the following procedure to install the `SUNWscnfs` package.

▼ How to Install Sun Cluster HA for NFS Packages

You need the Sun Cluster Agents CD-ROM to complete this procedure. Perform this procedure on all of the cluster nodes that can run Sun Cluster HA for NFS.

1. **Load the Sun Cluster Agents CD-ROM into the CD-ROM drive.**
2. **Run the `scinstall` utility with no options.**

This step starts the `scinstall` utility in interactive mode.

3. **Choose the menu option, Add Support for New Data Service to This Cluster Node.**
   The `scinstall` utility prompts you for additional information.
4. **Provide the path to the Sun Cluster Agents CD-ROM.**
   The utility refers to the CD as the "data services cd."
5. **Specify the data service to install.**
   The `scinstall` utility lists the data service that you selected and asks you to confirm your choice.
6. **Exit the `scinstall` utility.**
7. **Unload the CD from the drive.**

Where to Go From Here

See "Registering and Configuring Sun Cluster HA for NFS" on page 13 to register Sun Cluster HA for NFS and to configure the cluster for the data service.

Registering and Configuring Sun Cluster HA for NFS

This procedure describes how to use the `scrgadm`(1M) command to register and configure Sun Cluster HA for NFS.

Note – Other options also enable you to register and configure the data service. See "Tools for Data Service Resource Administration" in *Sun Cluster 3.1 Data Service Planning and Administration Guide* for details about these options.

Before you register and configure Sun Cluster HA for NFS, run the following command to verify that the Sun Cluster HA for NFS package, `SUNWscnfs`, is installed on the cluster.

```
# pkginfo -l SUNWscnfs
```

If the package has not been installed, see "Installing Sun Cluster HA for NFS Packages" on page 12 for instructions on how to install the package.

▼ How to Register and Configure Sun Cluster HA for NFS

1. **Become superuser on a cluster member.**

2. **Verify that all of the cluster nodes are online.**

   ```
   # scstat -n
   ```

3. **Create the Pathprefix directory.**

   The Pathprefix directory exists on the cluster file system that Sun Cluster HA for NFS uses to maintain administrative and status information.

   You can specify any directory for this purpose. However, you must manually create an *admin-dir* directory for each resource group that you create. For example, create the directory `/global/nfs`.

   ```
   # mkdir -p /global/admin-dir
   ```

4. **Create a failover resource group to contain the NFS resources.**

   ```
   # scrgadm -a -g resource-group -y Pathprefix=/global/admin-dir [-h nodelist]
   ```

`-a`	Specifies that you are adding a new configuration.
`-g` *resource-group*	Specifies the failover resource group.
`-y Pathprefix=`*path*	Specifies a directory on a cluster file system that Sun Cluster HA for NFS uses to maintain administrative and status information.
[`-h` *nodelist*]	Specifies an optional, comma-separated list of physical node names or IDs that identify potential masters. The order here determines the order in which the Resource Group Manager (RGM) considers primary nodes during failover.

5. **Verify that you have added all of your logical hostname resources to the name service database.**

   To avoid any failures because of name service lookup, verify that all of the logical hostnames are present in the server's and client's `/etc/inet/hosts` file.

6. **Configure name service mapping in the `/etc/nsswitch.conf` file on the cluster nodes to first check the local files before trying to access NIS or NIS+ for rpc lookups.**

   Doing so prevents timing-related errors for rpc lookups during periods of public network or nameservice unavailability.

7. **Modify the hosts entry in `/etc/nsswitch.conf` so that upon resolving a name locally the host does not first contact NIS/DNS, but instead immediately returns a successful status.**

Doing this enables HA-NFS to failover correctly in the presence of public network failures.

```
# hosts: cluster files [SUCCESS=return] nis
# rpc: files nis
```

8. **(Optional) Customize the `nfsd` or `lockd` startup options.**

   a. **To customize `nfsd` options, on each cluster node open the `/etc/init.d/nfs.server` file, find the command line starting with `/usr/lib/nfs/nfsd`, and add any additional arguments desired.**

   b. **To customize `lockd` startup options, on each cluster node open the `/etc/init.d/nfs.client` file, find the command line starting with `/usr/lib/nfs/lockd`, and add any command line arguments desired.**

Note – The command lines must remain limited to a single line. Breaking the command line into multiple lines is not supported. The additional arguments must be valid options documented in man pages `nfsd(1M)` and `lockd(1M)`.

---

9. **Add the desired logical hostname resources into the failover resource group.**

   You must set up a logical hostname resource with this step. The logical hostname that you use with Sun Cluster HA for NFS **cannot** be a `SharedAddress` resource type.

   # **`scrgadm -a -L -g`** *resource-group* **`-l`** *logical-hostname, ...* [**`-n`** *netiflist*]

`-a`	Specifies that you are adding a new configuration.
`-L -g` *resource-group*	Specifies the resource group that is to hold the logical hostname resources.
`-l` *logical-hostname, …*	Specifies the logical hostname resource to be added.
`-n` *netiflist*	Specifies an optional, comma-separated list that identifies the IP Networking Multipathing groups that are on each node. Each element in *netiflist* must be in the form of `netif@node`. `netif` can be given as an IP Networking Multipathing group name, such as `sc_ipmp0`. The node can be identified by the node name or node ID, such as `sc_ipmp0@1` or `sc_ipmp@phys-schost-1`.

Note – Sun Cluster does not currently support using the adapter name for `netif`.

---

10. **From any cluster node, create a directory structure for the NFS configuration files.**

Create the administrative subdirectory below the directory that the `Pathprefix` property identifies in Step 4, for example, `/global/nfs/SUNW.nfs`.

```
# mkdir Pathprefix/SUNW.nfs
```

11. Create a `dfstab.`*resource* file in the `SUNW.nfs` directory that you created in Step 10, and set up share options.

a. Create the *Pathprefix*`/SUNW.nfs/dfstab.`*resource* file.

This file contains a set of `share` commands with the shared path names. The shared paths should be subdirectories on a cluster file system.

Note – Choose a *resource* name suffix to identify the NFS resource that you plan to create (in Step 13). A good resource name refers to the task that this resource is expected to perform. For example, a name such as `user-nfs-home` is a good candidate for an NFS resource that shares user home directories.

---

b. Set up the `share` options for each path that you have created to be shared.

The format of this file is exactly the same as the format that is used in the `/etc/dfs/dfstab` file.

```
share [-F nfs] [-o] specific_options [-d "description"] pathname
```

`-F nfs`	Identifies the file system type as `nfs`.
`-o` *specific_options*	Grants read-write access to all of the clients. See the `share`(1M) man page for a list of options. Set the `rw` option for Sun Cluster.
`-d` *description*	Describes the file system to add.
pathname	Identifies the file system to share.

When you set up your share options, consider the following points.

- When constructing `share` options, do not use the `root` option, and do not mix the `ro` and `rw` options.
- Do not grant access to the hostnames on the cluster interconnect.

  Grant read and write access to all of the cluster nodes and logical hosts to enable the Sun Cluster HA for NFS monitoring to do a thorough job. However, you can restrict write access to the file system or make the file system entirely read-only. If you do so, Sun Cluster HA for NFS fault monitoring can still perform monitoring without having write access.
- If you specify a client list in the `share` command, include all of the physical hostnames and logical hostnames that are associated with the cluster, as well as the hostnames for all of the clients on all of the public networks to which the cluster is connected.

- If you use net groups in the `share` command (rather than names of individual hosts), add all of those cluster hostnames to the appropriate net group.

The `share -o rw` command grants write access to all of the clients, including the hostnames that the Sun Cluster software uses. This command enables Sun Cluster HA for NFS fault monitoring to operate most efficiently. See the following man pages for details.

- `dfstab`(4)
- `share`(1M)
- `share_nfs`(1M)

12. Register the NFS resource type.

```
# scrgadm -a -t resource-type
```

`-a -t` *resource-type*	Adds the specified resource type. For Sun Cluster HA for NFS, the resource type is `SUNW.nfs`.

13. Create the NFS resource in the failover resource group.

```
# scrgadm -a -j resource -g resource-group -t resource-type
```

`-a`	Specifies that you are adding a configuration.
`-j` *resource*	Specifies the name of the resource to add, which you defined in Step 11. This name can be your choice but must be unique within the cluster.
`-g` *resource-group*	Specifies the name of a previously created resource group to which this resource is to be added.
`-t` *resource-type*	Specifies the name of the resource type to which this resource belongs. This name must be the name of a registered resource type.

14. Run the `scswitch`(1M) command to perform the following tasks.

- Enable the resource and the resource monitor.
- Manage the resource group.
- Switch the resource group into the `ONLINE` state.

```
# scswitch -Z -g resource-group
```

Example – Setting Up and Configuring Sun Cluster HA for NFS

The following example shows how to set up and configure Sun Cluster HA for NFS.

(Create a logical host resource group and specify the path to the administrative files used by NFS (`Pathprefix`).)

```
# scrgadm -a -g resource-group-1 -y Pathprefix=/global/nfs
```

(Add logical hostname resources into the logical host resource group.)
```
# scrgadm -a -L -g resource-group-1 -l schost-1
```

(Make the directory structure contain the Sun Cluster HA for NFS configuration files.)
```
# mkdir -p /global/nfs/SUNW.nfs
```

(Create the dfstab.resource file under the nfs/SUNW.nfs directory and set share options.)
```
# share -F nfs -o rw=engineering -d "home dirs" nfs/SUNW.nfs
```

(Register the NFS resource type.)
```
# scrgadm -a -t SUNW.nfs
```

(Create the NFS resource in the resource group.)
```
# sc
rgadm -a -j r-nfs -g resource-group-1 -t SUNW.nfs
```

(Enable the resources and their monitors, manage the resource group, and switch the resource group into online state.)
```
# scswitch -Z -g resource-group-1
```

Where to Go From Here

See "How to Change Share Options on an NFS File System" on page 18 to set share options for your NFS file systems. See "Configuring Sun Cluster HA for NFS Extension Properties" on page 21 to review or set extension properties.

▼ How to Change Share Options on an NFS File System

If you use the `rw`, `rw=`, `ro`, or `ro=` options to the `share -o` command, NFS fault monitoring works best if you grant access to all of the physical hosts or `netgroups` that are associated with all of the Sun Cluster servers.

If you use `netgroups` in the `share`(1M) command, add all of the Sun Cluster hostnames to the appropriate `netgroup`. Ideally, grant both read access and write access to all of the Sun Cluster hostnames to enable the NFS fault probes to do a complete job.

Note – Before you change share options, read the `share_nfs`(1M) man page to understand which combinations of options are legal.

---

1. **Become superuser on a cluster node.**

2. **Turn off fault monitoring on the NFS resource.**

   ```
   # scswitch -n -M -j resource
   ```

`-M`	Disables the resource monitor

3. **Test the new `share` options.**

   a. **Before you edit the `dfstab.`*resource* file with new share options, execute the new `share` command to verify the validity of your combination of options.**

      ```
      # share -F nfs [-o] specific_options [-d "description"] pathname
      ```

`-F nfs`	Identifies the file system type as NFS.
`-o` *specific_options*	Specifies an option. You might use `rw`, which grants read-write access to all of the clients.
`-d` *description*	Describes the file system to add.
pathname	Identifies the file system to share.

   b. **If the new `share` command fails, immediately execute another `share` command with the old options. When the new command executes successfully, proceed to Step 4.**

4. **Edit the `dfstab.`*resource* file with the new share options.**

   a. **To remove a path from the `dfstab.`*resource* file, perform the following steps in order.**

      i. **Execute the `unshare`(1M) command.**

      ii. **From the `dfstab.`*resource* file, delete the `share` command for the path that you want to remove.**

      ```
      # unshare [-F nfs] [-o specific_options] pathname
      # vi dfstab.resource
      ```

`-F nfs`	Identifies the file system type as NFS.
`-o` specific_options	Specifies the options that are specific to NFS file systems.
pathname	Identifies the file system that is made unavailable.

   b. **To add a path or change an existing path in the `dfstab.`*resource* file, verify that the mount point is valid, then edit the `dfstab.`*resource* file.**

Note – The format of this file is exactly the same as the format that is used in the `/etc/dfs/dfstab` file. Each line consists of a `share` command.

5. **Enable fault monitoring on the NFS resource.**

```
# scswitch -e -M -j resource
```

▼ How to Tune Sun Cluster HA for NFS Method Timeouts

The time that Sun Cluster HA for NFS methods require to finish depends on the number of paths that the resources share through the `dfstab.`*resource* file. The default timeout for these methods is 300 seconds.

As a general guideline, allocate 10 seconds toward the method timeouts for each path that is shared. Default timeouts are designed to handle 30 shared paths.

- If the number of shared paths is less than 30, do not reduce the timeout.
- If the number of shared paths exceeds 30, multiply the number of paths by 10 to compute the recommended timeout. For example, if the `dfstab.`*resource* file contains 50 shared paths, the recommended timeout is 500 seconds.

Update the following method timeouts if the number of shared paths is greater than 30.

`Prenet_start_timeout`	`Postnet_stop_timeout`	`Monitor_Start_timeout`
`Start_timeout`	`Validate_timeout`	`Monitor_Stop_timeout`
`Stop_timeout`	`Update_timeout`	`Monitor_Check_timeout`

To change method timeouts, use the `scrgadm -c` option, as in the following example.

```
% scrgadm -c -j resource -y Prenet_start_timeout=500
```

▼ How to Configure `SUNW.HAStoragePlus` Resource Type

The `SUNW.HAStoragePlus` resource type was introduced in Sun Cluster 3.0 5/02. This new resource type performs the same functions as `SUNW.HAStorage`, and synchronizes actions between HA storage and Sun Cluster HA for NFS.

`SUNW.HAStoragePlus` also has an additional feature to make a local file system highly available. Sun Cluster HA for NFS is both failover and disk-intensive, and therefore, you should set up the `SUNW.HAStoragePlus` resource type.

See the `SUNW.HAStoragePlus(5)` man page and "Relationship Between Resource Groups and Disk Device Groups" in *Sun Cluster 3.1 Data Service Planning and Administration Guide* for background information. See "Synchronizing the Startups Between Resource Groups and Disk Device Groups" in *Sun Cluster 3.1 Data Service Planning and Administration Guide* for the procedure. (If you are using a Sun Cluster 3.0 version prior to 5/02, you must set up `SUNW.HAStorage` instead of `SUNW.HAStoragePlus`. See "Synchronizing the Startups Between Resource Groups and Disk Device Groups" in *Sun Cluster 3.1 Data Service Planning and Administration Guide* for the procedure.)

Configuring Sun Cluster HA for NFS Extension Properties

Typically, you use the command line `scrgadm -x` *parameter=value* to configure extension properties when you create the NFS resource. You can also use the procedures in "Administering Data Service Resources" in *Sun Cluster 3.1 Data Service Planning and Administration Guide* to configure these properties later. You do not need to set any extension properties for Sun Cluster HA for NFS. See "Standard Properties" in *Sun Cluster 3.1 Data Service Planning and Administration Guide* for details on all of the Sun Cluster properties.

Table 1–2 describes extension properties that you can configure for Sun Cluster HA for NFS. You can update some properties dynamically. You can update others, however, only when you create the resource. The Tunable entries indicate when you can update the property.

TABLE 1–2 Sun Cluster HA for NFS Extension Properties

Name/Data Type	Default
`Lockd_nullrpc_timeout` (integer)	The time-out value (in seconds) to use when probing `lockd`. **Default:** `120` **Range:** Minimum = `60` **Tunable:** Any time

TABLE 1–2 Sun Cluster HA for NFS Extension Properties *(Continued)*

Name/Data Type	Default
`Monitor_retry_count` (integer)	The number of times that the process monitor facility (PMF) restarts the fault monitor during the time window that the `Monitor_retry_interval` property specifies. Note that this property refers to restarts of the fault monitor itself, rather than the resource. The system-defined properties `Retry_interval` and `Retry_count` control restarts of the resource. See the `scrgadm`(1M) man page for a description of these properties. **Default:** `4` **Range:** `0` – `2,147,483,641` `-1` indicates an infinite number of restart attempts. **Tunable:** Any time
`Monitor_retry_interval` (integer)	The time (in minutes) over which failures of the fault monitor are counted. If the number of times that the fault monitor fails is more than the value that is specified in the extension property `Monitor_retry_count` within this period, the PMF restarts the fault monitor. **Default:** `2` **Range:** `0` – `2,147,483,641` `-1` indicates an infinite amount of time. **Tunable:** Any time
`Mountd_nullrpc_restart` (Boolean)	A Boolean to indicate whether to restart `mountd` when a null `rpc` call fails. **Default:** `True` **Range:** None **Tunable:** Any time
`Mountd_nullrpc_timeout` (integer)	The time-out value (in seconds) to use when probing `mountd`. **Default:** `120` **Range:** Minimum = `60` **Tunable:** Any time

TABLE 1–2 Sun Cluster HA for NFS Extension Properties *(Continued)*

Name/Data Type	Default
`Nfsd_nullrpc_restart` (Boolean)	A Boolean to indicate whether to restart `nfsd` when a null `rpc` call fails. **Default:** `False` **Range:** None **Tunable:** Any time
`Nfsd_nullrpc_timeout` (integer)	The time-out value (in seconds) to use when probing `nfsd`. **Default:** `120` **Range:** Minimum = 60 **Tunable:** Any time
`Rpcbind_nullrpc_reboot` (Boolean)	A Boolean to indicate whether to reboot the system when a null `rpc` call on `rpcbind` fails. **Default:** `False` **Range:** None **Tunable:** Any time
`Rpcbind_nullrpc_timeout` (integer)	The time-out value (in seconds) to use when probing `rpcbind`. **Default:** `120` **Range:** Minimum = 60 **Tunable:** Any time
`Statd_nullrpc_timeout` (integer)	The time-out value (in seconds) to use when probing `statd`. **Default:** `120` **Range:** Minimum = `60` **Tunable:** Any time

Sun Cluster HA for NFS Fault Monitor

The Sun Cluster HA for NFS fault monitor uses the following two processes.

- NFS system fault monitoring, which involves monitoring the NFS daemons (`nfsd`, `mountd`, `statd`, and `mountd`) and resolving any problems that occur.

- Status check, which is specific to each NFS resource. The fault monitor of each resource checks the status of each shared path to monitor the file systems that the resource exports.

Fault Monitor Startup

An NFS resource `MONITOR_START` method starts the NFS system fault monitor. This start method first checks if the NFS system fault monitor (`nfs_daemons_probe`) already runs under the process monitor `pmfadm`. If the NFS system fault monitor is not running, the start method starts the `nfs_daemons_probe` process under the control of the process monitor. The start method then starts the resource fault monitor (`nfs_probe`), also under the control of the process monitor.

Fault Monitor Stops

The NFS resource `MONITOR_STOP` method stops the resource fault monitor. This method also stops the NFS system fault monitor if no other NFS resource fault monitor runs on the local node.

NFS Fault Monitor Process

To check for the presence of the process and its response to a null `rpc` call, the system fault monitor probes `rpcbind`, `statd`, `lockd`, `nfsd`, and `mountd`. This monitor uses the following NFS extension properties.

`Rpcbind_nullrpc_timeout`	`Lockd_nullrpc_timeout`
`Nfsd_nullrpc_timeout`	`Rpcbind_nullrpc_reboot`
`Mountd_nullrpc_timeout`	`Nfsd_nullrpc_restart`
`Statd_nullrpc_timeout`	`Mountd_nullrpc_restart`

See “Configuring Sun Cluster HA for NFS Extension Properties” on page 21 to review or set extension properties.

If a daemon needs to be stopped, the calling method sends a kill signal to the process id (PID) and waits to verify that the PID disappears. The amount of time that the calling method waits is a fraction of the method’s timeouts. If the process does not stop within that period of time, the fault monitor assumes that the process failed.

Note – If the process needs more time to stop, increase the timeout of the method that was running when the process was sent the kill signal.

After the daemon is stopped, the fault monitor restarts the daemon and waits until the daemon is registered under RPC. If a new RPC handle can be created, the status of the daemon is reported in the fault monitor internally as a success. If the RPC handle cannot be created, the status of the daemon is returned to the fault monitor as unknown, and no error messages are printed.

Each system fault-monitor probe cycle performs the following steps in a loop.

1. **Sleeps for `Cheap_probe_interval`.**
2. **Probes `rpcbind`.**

   If the process fails and `Failover_mode=HARD`, then the fault monitor reboots the node.

   If a null `rpc` call to the daemon fails, `Rpcbind_nullrpc_reboot=False`, and `Failover_mode=HARD`, then the fault monitor reboots the node.
3. **Probes `statd` and `lockd`.**

   If `statd` or `lockd` fail, the fault monitor attempts to restart both daemons. If the fault monitor cannot restart the daemons, all of the NFS resources fail over to another node.

   If a null `rpc` call to these daemons fails, the fault monitor logs a message to `syslog` but does not restart `statd` or `lockd`.
4. **Probe `mountd`.**

   If `mountd` fails, the fault monitor attempts to restart the daemon.

   If the `kstat` counter, `nfs_server:calls`, is not increasing, the following actions occur.

   a. **A null `rpc` call is sent to `mountd`.**

   b. **If the null `rpc` call fails and `Mountd_nullrpc_restart=True`, the fault monitor attempts to restart `mountd` if the cluster file system is available.**

   c. **If the fault monitor cannot restart mountd and the number of failures reaches `Retry_count`, then all of the NFS resources fail over to another node.**
5. **Probe `nfsd`.**

   If `nfsd` fails, the fault monitor attempts to restart the daemon.

   If the `kstat` counter, `nfs_server:calls`, is not increasing, the following actions occur.

   a. **A null `rpc` call is sent to `nfsd`.**

b. If the null `rpc` call fails and `Nfsd_nullrpc_restart=TRUE`, then the fault monitor attempts to restart `nfsd`.

c. If the fault monitor cannot restart `nfsd` and the number of failures reaches `Retry_count`, then all of the NFS resources fail over to another node.

If any of the NFS daemons fail to restart, the status of all of the online NFS resources is set to `FAULTED`. When all of the NFS daemons are restarted and healthy, the resource status is set to `ONLINE` again.

NFS Resource Monitor Process

Before starting the resource monitor probes, all of the shared paths are read from the `dfstab` file and stored in memory. In each probe cycle, all of the shared paths are probed in each iteration by performing `stat()` on the path.

Each resource monitor fault probe performs the following steps.

1. Sleeps for `Thorough_probe_interval`.
2. Refreshes the memory if `dfstab` has been changed since the last read.
3. Probes all of the shared paths in each iteration by performing `stat()` of the path.

If any path is not functional, the resource status is set to `FAULTED`. If all of the paths are functional, the resource status is set to `ONLINE` again.

Index