Sun Cluster Data Service for Network File System (NFS) Guide for Solaris OS

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

Part No: 817–4646–11
September 2004, Revision A

040812@9495

Contents

Preface

Sun Cluster Data Service for Network File System (NFS) Guide for Solaris OS explains how to install and configure Sun™ Cluster HA for Network File System (NFS) on both SPARC® and x86 based systems.

Note – In this document, the term "x86" refers to the Intel 32-bit family of microprocessor chips and compatible microprocessor chips made by AMD.

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 System and expertise with the volume manager software that is used with Sun Cluster.

Note – Sun Cluster software runs on two platforms, SPARC and x86. The information in this document pertains to both platforms unless otherwise specified in a special chapter, section, note, bulleted item, figure, table, or example.

UNIX Commands

This document contains information about commands that are specific to installing and configuring Sun Cluster data services. The document does *not* contain comprehensive information about basic UNIX® commands and procedures, such as shutting down the system, booting the system, and configuring devices. Information about basic UNIX commands and procedures is available from the following sources:

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

Typographic Conventions

The following table describes the typographic changes that are used in this book.

TABLE P–1 Typographic Conventions

Typeface or Symbol	Meaning	Example
`AaBbCc123`	The names of commands, files, and directories, and onscreen computer output	Edit your `.login` file. Use `ls -a` to list all files. `machine_name% you have mail.`
`AaBbCc123`	What you type, contrasted with onscreen computer output	`machine_name%` **`su`** `Password:`
AaBbCc123	Command-line placeholder: replace with a real name or value	The command to remove a file is `rm` *filename*.
AaBbCc123	Book titles, new terms, and terms to be emphasized	Read Chapter 6 in the *User's Guide*. These are called *class* options. Do *not* save the file. (Emphasis sometimes appears in bold online.)

Shell Prompts in Command Examples

The following table shows the default system prompt and superuser prompt for the C shell, Bourne shell, and Korn shell.

TABLE P–2 Shell Prompts

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

Related Documentation

Information about related Sun Cluster topics is available in the documentation that is listed in the following table. All Sun Cluster documentation is available at `http://docs.sun.com`.

Topic	Documentation
Data service administration	*Sun Cluster Data Services Planning and Administration Guide for Solaris OS*
	Individual data service guides
Concepts	*Sun Cluster Concepts Guide for Solaris OS*
Overview	*Sun Cluster Overview for Solaris OS*
Software installation	*Sun Cluster Software Installation Guide for Solaris OS*
System administration	*Sun Cluster System Administration Guide for Solaris OS*
Hardware administration	*Sun Cluster 3.x Hardware Administration Manual for Solaris OS*
	Individual hardware administration guides
Data service development	*Sun Cluster Data Services Developer's Guide for Solaris OS*

Topic	Documentation
Error messages	*Sun Cluster Error Messages Guide for Solaris OS*
Command and function reference	*Sun Cluster Reference Manual for Solaris OS*

For a complete list of Sun Cluster documentation, see the release notes for your release of Sun Cluster at `http://docs.sun.com`.

Related Third-Party Web Site References

Third-party URLs that are referenced in this document provide additional related information.

Note – Sun is not responsible for the availability of third-party web sites mentioned in this document. Sun does not endorse and is not responsible or liable for any content, advertising, products, or other materials that are available on or through such sites or resources. Sun will not be responsible or liable for any actual or alleged damage or loss caused or alleged to be caused by or in connection with use of or reliance on any such content, goods, or services that are available on or through such sites or resources.

Accessing Sun Documentation Online

The docs.sun.com^{SM} Web site enables you to access Sun technical documentation online. You can browse the docs.sun.com archive or search for a specific book title or subject. The URL is `http://docs.sun.com`.

Ordering Sun Documentation

Sun Microsystems offers select product documentation in print. For a list of documents and how to order them, see “Buy printed documentation” 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 email address (if available)
- Your company name, address, and phone number
- The model and serial numbers of your systems
- The release number of the Solaris Operating System (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
SPARC: `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 Data Services Planning and Administration Guide for Solaris OS* and the *Sun Cluster Concepts Guide for Solaris OS* 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 "Configuration Worksheets" in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS* 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.

Sun Cluster HA for NFS requires that all NFS client mounts be "hard" mounts.

No Sun Cluster node may be an NFS client of a file system that is exported by Sun Cluster HA for NFS and is being mastered on a node in the same cluster. Such cross-mounting of Sun Cluster HA for NFS is prohibited. Use the cluster file system to share files among cluster nodes.

Starting with Solaris 9, if Solaris Resource Manager is used to manage system resources allocated to NFS on a cluster, all Sun Cluster HA for NFS resources which can fail over to a common cluster node must have the same Solaris Resource Manager project ID. This project ID is set with the `Resource_project_name` resource property.

Caution – If you use VERITAS Volume Manager (available for use in SPARC based clusters only), 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 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 13
Set up and configure Sun Cluster HA for NFS	"Registering and Configuring Sun Cluster HA for NFS" on page 15
Configure resource extension properties	"Configuring Sun Cluster HA for NFS Extension Properties" on page 24
View fault monitor information	"Sun Cluster HA for NFS Fault Monitor" on page 27

Installing Sun Cluster HA for NFS Packages

If you did not install the Sun Cluster HA for Network File System (NFS) packages during your initial Sun Cluster installation, perform this procedure to install the packages. Perform this procedure on each cluster node where you are installing the Sun Cluster HA for Network File System (NFS) packages. To complete this procedure, you need the Sun Java Enterprise System Accessory CD Volume 3.

If you are installing more than one data service simultaneously, perform the procedure in "Installing the Software" in *Sun Cluster Software Installation Guide for Solaris OS*.

Install the Sun Cluster HA for Network File System (NFS) packages by using one of the following installation tools:

- The Web Start program
- The `scinstall` utility

Note – The Web Start program is *not* available in releases earlier than Sun Cluster 3.1 Data Services 10/03.

▼ How to Install Sun Cluster HA for NFS Packages Using the Web Start Program

You can run the Web Start program with a command-line interface (CLI) or with a graphical user interface (GUI). The content and sequence of instructions in the CLI and the GUI are similar. For more information about the Web Start program, see the `installer`(1M) man page.

1. **On the cluster node where you are installing the Sun Cluster HA for Network File System (NFS) packages, become superuser.**
2. **(Optional) If you intend to run the Web Start program with a GUI, ensure that your `DISPLAY` environment variable is set.**
3. **Load the Sun Java Enterprise System Accessory CD Volume 3 into the CD-ROM drive.**

   If the Volume Management daemon `vold`(1M) is running and configured to manage CD-ROM devices, it automatically mounts the CD-ROM on the `/cdrom/cdrom0` directory.
4. **Change to the Sun Cluster HA for Network File System (NFS) component directory of the CD-ROM.**

The Web Start program for the Sun Cluster HA for Network File System (NFS) data service resides in this directory.

```
# cd /cdrom/cdrom0/\
components/SunCluster_HA_NFS_3.1
```

5. **Start the Web Start program.**

```
# ./installer
```

6. **When you are prompted, select the type of installation.**

- To install only the C locale, select Typical.
- To install other locales, select Custom.

7. **Follow instructions on the screen to install the Sun Cluster HA for Network File System (NFS) packages on the node.**

After the installation is finished, the Web Start program provides an installation summary. This summary enables you to view logs that the Web Start program created during the installation. These logs are located in the `/var/sadm/install/logs` directory.

8. **Exit the Web Start program.**

9. **Unload the Sun Java Enterprise System Accessory CD Volume 3 from the CD-ROM drive.**

   a. **To ensure that the CD-ROM is not being used, change to a directory that does *not* reside on the CD-ROM.**

   b. **Eject the CD-ROM.**

```
# eject cdrom
```

Where to Go From Here

Go to "Registering and Configuring Sun Cluster HA for NFS" on page 15.

▼ How to Install Sun Cluster HA for NFS Packages Using the `scinstall` Utility

This section describes how to install Sun Cluster HA for NFS packages by using the `scinstall` utility.

1. **Load the Sun Java Enterprise System Accessory CD Volume 3 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 Java Enterprise System Accessory CD Volume 3.**
   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

Go to "Registering and Configuring Sun Cluster HA for NFS" on page 15.

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 Data Services Planning and Administration Guide for Solaris OS* 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 13 for instructions on how to install the package.

▼ How to Register and Configure Sun Cluster HA for NFS

This section describes 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 lookups, verify that all IP addresses to hostname mappings that are used by Sun Cluster HA for NFS are present in the server's and client's `/etc/inet/hosts` file. This requirement also applies to any IPMP test IP addresses of logical hostnames that are used by Sun Cluster HA for NFS.

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 fail over 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.**

      Starting with Solaris 9, you can set the `lockd` grace period with the `LOCKD_GRACE_PERIOD` parameter in the `/etc/default/nfs` file. However, if the grace period is set in a command line argument in the `/etc/init.d/nfs.client` file, this will override the value set in `LOCKD_GRACE_PERIOD`.

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 fault 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
```

▼ 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.

You can also modify shared paths and options dynamically without bringing offline the Sun Cluster HA for NFS resource. See "How to Dynamically Update Shared Paths on an NFS File System" on page 22.

To modify the share options on an NFS file system while the Sun Cluster HA for NFS resource is offline, perform the following steps.

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 fault 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 Dynamically Update Shared Paths on an NFS File System

You can dynamically modify the shared paths on an NFS file system without bringing offline the Sun Cluster HA for NFS resource. The general procedure consists of modifying the `dfstab.`*resource* file for Sun Cluster HA for NFS and then manually running the appropriate command, either the `share` command or the `unshare` command. The command is immediately effective, and Sun Cluster HA for NFS handles making these paths highly available.

Ensure that the paths that are shared are always available to Sun Cluster HA for NFS during failover so that local paths (on non-HA file systems) are not used.

If paths on a file system that is managed by HAStoragePlus are shared, the HAStoragePlus resource must be in the same resource group as the Sun Cluster HA for NFS resource, and the dependency between them must be set correctly.

1. Use the `scstat -g` command to find out the node on which the Sun Cluster HA for NFS resource is online.

2. On this node run the `/usr/sbin/share` command to see the list of paths currently shared. Determine the changes you want to make to this list.

3. To add a new shared path, perform the following steps.

a. Add the `share` command to the `dfstab.`*resource* file.

Sun Cluster HA for NFS shares the new path the next time it checks the file. The frequency of these checks is controlled by the `Thorough_Probe_Interval` property (by default 120 seconds).

b. Run the `share` command manually to make the newly added shared path effective immediately. Running the command manually is recommended because the user can be certain that the shared paths are available to potential clients. Sun Cluster HA for NFS detects that the newly added path is already shared and does not complain.

4. To unshare a path, perform the following steps.

a. Run the `dfmounts(1M)` command to ensure that no clients are currently using the path.

Although a path can be unshared even if clients are using it, these clients would get a stale file error handle and would need special care (forced umount, or even reboot) to recover.

b. Remove the shared path from the `dfstab.`*resource* file.

c. Run the `unshare` command manually.

5. To modify options for an existing shared path, perform the following steps.

a. Modify the `dfstab.`*resource* file as needed.
b. Run the appropriate command (`share` or `unshare`) manually.

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 Data Services Planning and Administration Guide for Solaris OS* for background information. See "Synchronizing the Startups Between Resource Groups and Disk Device Groups" in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS* 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 Data Services Planning and Administration Guide for Solaris OS* 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 Data Services Planning and Administration Guide for Solaris OS* 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 Data Services Planning and Administration Guide for Solaris OS* for details on all of the Sun Cluster properties.

For information about the extension properties that you can configure for Sun Cluster HA for NFS, see Table 2. 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 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
`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

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

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

- NFS system fault monitoring, which involves monitoring the NFS daemons (`nfsd`, `mountd`, `statd`, and `lockd`) and resolving any problems that occur. The NFS system fault monitoring also monitors the RPC portmapper service daemon (`rpcbind`).
- NFS resource fault monitoring, 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

First, 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`) is already running under the process monitor daemon (`rpc.pmfd`). 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 Stop

First, the NFS resource `MONITOR_STOP` method stops the resource fault monitor. Then, this method stops the NFS system fault monitor if no other NFS resource fault monitor runs on the local node.

NFS System Fault Monitor Process

The NFS system fault monitor probes `rpcbind`, `statd`, `lockd`, `nfsd`, and `mountd` on the local node by checking for the presence of the process and its response to a null `rpc` call. 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 24 to review or set extension properties.

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 terminates unexpectedly, but a warm restart of the daemon is in progress, `rpcbind` continutes to probe other daemons.

   If the process terminates unexpectedly, then the fault monitor reboots the node.

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

   If `statd` or `lockd` terminate unexpectedly, the system fault monitor attempts to restart both daemons.

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

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

   If the null `rpc` call to the daemon terminates unexpectedly and `Mountd_nullrpc_restart=True`, the fault monitor attempts to restart `mountd` if the cluster file system is available.
5. **Probes `nfsd`.**

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

   If the null `rpc` call to the daemon terminates unexpectedly and `Nfsd_nullrpc_restart=TRUE`, then the fault monitor attempts to restart `nfsd` if the cluster file system is available.

If any of the above NFS daemons (except `rpcbind`) fail to restart during a probe cycle, the NFS system fault monitor will retry the restart in the next cycle. When all of the NFS daemons are restarted and healthy, the resource status is set to `ONLINE`. The monitor tracks unexpected terminations of NFS daemons in the last `Retry_interval`. When the total number of unexpected daemon terminations has reached `Retry_count`, the system fault monitor issues a `scha_control` giveover. If the giveover call fails, the monitor attempts to restart the failed NFS daemon.

At the end of each probe cycle, if all daemons are healthy, the monitor clears the history of failures.

NFS Resource Fault Monitor Process

Before starting the resource fault 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 fault monitor fault probe performs the following steps in a loop.

1. Sleeps for `Thorough_probe_interval`.
2. Refreshes the memory if `dfstab` has been changed since the last read.

   If an error occurs while reading the `dfstab` file, the resource status is set to `FAULTED`, and the monitor skips the remainder of the checks in the current probe cycle.
3. Probes all of the shared paths in each iteration by performing `stat()` on the path.

   If any path is not functional, the resource status is set to `FAULTED`.
4. Probes for the presence of NFS daemons (`nfsd`, `mountd`, `lockd`, `statd`) and `rpcbind`.
5. If any of these daemons are down, the resource status is set to `FAULTED`.

If all shared paths are valid and NFS daemons are present, the resource status is reset to `ONLINE` at the end of the probe cycle.

Index