Sun Cluster Data Service for NFS Guide for Solaris OS

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

Part No: 820–2565–10
February 2008, Revision A

080328@19860

Contents

Preface

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

Note – This Sun Cluster release supports systems that use the SPARC™ and x86 families of processor architectures: UltraSPARC, SPARC64, and AMD64. In this document, the label x86 refers to systems that use the AMD64 family of processor architectures.

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 book assume knowledge of the Solaris™ Operating System (Solaris OS) and expertise with the volume-manager software that is used with Sun Cluster software.

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.

Using 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 conventions that are used in this book.

TABLE P–1 Typographic Conventions

Typeface	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	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*. A *cache* is a copy that is stored locally. Do *not* save the file. **Note:** Some emphasized items appear bold online.

Shell Prompts in Command Examples

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

TABLE P–2 Shell Prompts

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

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 3.1 - 3.2 Hardware Administration Manual for Solaris OS*
	Individual hardware administration guides
Data service development	*Sun Cluster Data Services Developer's Guide for Solaris OS*
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.

Documentation, Support, and Training

The Sun web site provides information about the following additional resources:

- Documentation (`http://www.sun.com/documentation/`)
- Support (`http://www.sun.com/support/`)
- Training (`http://www.sun.com/training/`)

Getting 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 number and serial number of your systems
- The release number of the Solaris Operating System (for example, Solaris 10)
- The release number of Sun Cluster (for example, Sun Cluster 3.2)

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
`/usr/cluster/bin/clnode show-rev`	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 NFS

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

Note – If you are using the Solaris 10 OS, install and configure this data service to run only in the global zone. At publication of this document, this data service is not supported in non-global zones. For updated information about supported configurations of this data service, contact your Sun service representative.

This chapter contains the following sections.

You must configure Sun Cluster HA for NFS as a failover data service. See Chapter 1, "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 the Sun Cluster Manager to install and configure this data service. See the Sun Cluster 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.

Overview of the Installation and Configuration Process for 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	"How to Install the 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 14
Secure Sun Cluster HA for NFS with Kerberos V5	"Securing Sun Cluster HA for NFS With Kerberos V5" on page 32
Tune the Sun Cluster HA for NFS fault monitor	"Tuning the Sun Cluster HA for NFS Fault Monitor" on page 36
Upgrade the `SUNW.nfs` resource type	"Upgrading the `SUNW.nfs` Resource Type" on page 39

Planning the Sun Cluster HA for NFS Installation and Configuration

This section contains the information that you need to plan the installation and configuration of your Sun Cluster HA for NFS.

Service Management Facility Restrictions

Starting with Solaris 10, the following Service Management Facility (SMF) services are related to NFS.

- `/network/nfs/cbd`
- `/network/nfs/mapid`
- `/network/nfs/server`
- `/network/nfs/rquota`
- `/network/nfs/client`
- `/network/nfs/status`
- `/network/nfs/nlockmgr`

The Sun Cluster HA for NFS data service sets the property `application/auto_enable` to `FALSE` and the property `startd/duration` to `transient` for three of these services.

- `/network/nfs/server`
- `/network/nfs/status`
- `/network/nfs/nlockmgr`

These property settings have the following consequences for these services.

- When services that depend on these services are enabled, these services are not automatically enabled.
- In the event of any failure, SMF does not restart the daemons that are associated with these services.
- In the event of any failure, SMF does not restart these services.

NFSV3 Restrictions

If you are mounting file systems on the cluster nodes from external NFS servers, such as NAS filers, and you are using the NFSv3 protocol, you cannot run NFS client mounts and the Sun Cluster HA for NFS data service on the same cluster node. If you do, certain Sun Cluster HA for NFS data-service activities might cause the NFS daemons to stop and restart, interrupting NFS services. However, you can safely run the Sun Cluster HA for NFS data service if you use the NFSv4 protocol to mount external NFS file systems on the cluster nodes.

Loopback File System Restrictions

Do *not* use the loopback file system (LOFS) if both conditions in the following list are met:

- Sun Cluster HA for NFS is configured on a highly available local file system.
- The `automountd` daemon is running.

If both of these conditions are met, LOFS must be disabled to avoid switchover problems or other failures. If only one of these conditions is met, it is safe to enable LOFS.

If you require both LOFS and the `automountd` daemon to be enabled, exclude from the automounter map all files that are part of the highly available local file system that is exported by Sun Cluster HA for NFS.

Zettabyte File System (ZFS) Restrictions

If you are using the zettabyte file system (ZFS) as the exported file system, you must set the `sharenfs` property to `off`.

To set the `sharenfs` property to `off`, run the following command.

```
$ zfs set sharenfs=off file_system/volume
```

To verify if the `sharenfs` property is set to `off`, run the following command.

```
$ zfs get sharenfs file_system/volume
```

Installing the Sun Cluster HA for NFS Packages

If you did not install the Sun Cluster HA for NFS packages during your initial Sun Cluster installation, perform this procedure to install the packages. To install the packages, use the Sun Java Enterprise System Common Installer.

▼ How to Install the Sun Cluster HA for NFS Packages

Perform this procedure on each cluster node where you are installing the Sun Cluster HA for NFS packages.

You can run the Sun Java Enterprise System Common Installer 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.

Before You Begin Ensure that you have the Sun Java™ Availability Suite DVD-ROM.

If you intend to run the Sun Java Enterprise System Common Installer with a GUI, ensure that your `DISPLAY` environment variable is set.

1. **On the cluster node where you are installing the data service packages, become superuser.**

2. **Load the Sun Java Availability Suite DVD-ROM into the DVD-ROM drive.**

   If the Volume Management daemon `vold`(1M) is running and configured to manage DVD-ROM devices, the daemon automatically mounts the DVD-ROM on the `/cdrom` directory.

3. **Change to the Sun Java Enterprise System Common Installer directory of the DVD-ROM.**

   - **If you are installing the data service packages on the SPARC® platform, type the following command:**

     ```
     # cd /cdrom/cdrom0/Solaris_sparc
     ```

   - **If you are installing the data service packages on the x86 platform, type the following command:**

     ```
     # cd /cdrom/cdrom0/Solaris_x86
     ```

4. **Start the Sun Java Enterprise System Common Installer.**

   ```
   # ./installer
   ```

5. **When you are prompted, accept the license agreement.**

   If any Sun Java Enterprise System components are installed, you are prompted to select whether to upgrade the components or install new software.

6. **From the list of Sun Cluster agents under Availability Services, select the data service for Network File System (NFS).**

7. **If you require support for languages other than English, select the option to install multilingual packages.**

   English language support is always installed.

8. **When prompted whether to configure the data service now or later, choose Configure Later.**

   Choose Configure Later to perform the configuration after the installation.

9. **Follow the instructions on the screen to install the data service packages on the node.**

   The Sun Java Enterprise System Common Installer displays the status of the installation. When the installation is complete, the wizard displays an installation summary and the installation logs.

10 **(GUI only) If you do not want to register the product and receive product updates, deselect the Product Registration option.**

The Product Registration option is not available with the CLI. If you are running the Sun Java Enterprise System Common Installer with the CLI, omit this step

11 **Exit the Sun Java Enterprise System Common Installer.**

12 **Unload the Sun Java Availability Suite DVD-ROM from the DVD-ROM drive.**

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

b. **Eject the DVD-ROM.**

```
# eject cdrom
```

Next Steps See "Registering and Configuring Sun Cluster HA for NFS" on page 14 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 section describes how 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 the Sun Cluster HA for NFS Packages" on page 12 for instructions on how to install the package.

Setting Sun Cluster HA for NFS Extension Properties

The sections that follow contain instructions for registering and configuring resources. For information about the Sun Cluster HA for NFS extension properties, see Appendix A, "Sun Cluster HA for NFS Extension Properties". The Tunable entry indicates when you can update a property.

To set an extension property of a resource, include the following option in the `clresource(1CL)` command that creates or modifies the resource:

`-p` *property*=*value*

`-p` *property*
Identifies the extension property that you are setting.

value
Specifies the value to which you are setting the extension property.

You can also use the procedures in Chapter 2, "Administering Data Service Resources," in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS* to configure resources after the resources are created.

Tools for Registering and Configuring Sun Cluster HA for NFS

Sun Cluster provides the following tools for registering and configuring Sun Cluster HA for NFS:

- **The** `clsetup`(1CL) **utility.** For more information, see "How to Register and Configure the Sun Cluster HA for NFS by Using `clsetup`" on page 15.
- **Sun Cluster Manager.** For more information, see the Sun Cluster Manager online help.
- **Sun Cluster maintenance commands.** For more information, see "How to Register and Configure Sun Cluster HA for NFS by using Sun Cluster Command Line Interface (CLI)" on page 19.

The `clsetup` utility and Sun Cluster Manager each provide a wizard for configuring Sun Cluster HA for NFS. The wizards reduce the possibility for configuration errors that might result from command syntax errors or omissions. These wizards also ensure that all required resources are created and that all required dependencies between resources are set.

▼ How to Register and Configure the Sun Cluster HA for NFS by Using `clsetup`

Perform this procedure during your initial set up of Sun Cluster HA for NFS. Perform this procedure from one node only.

Note – The following instructions explain how to perform this operation by using the `clsetup` utility.

Before You Begin Before you start the Sun Cluster HA for NFS wizard, ensure that the following prerequisites are met:

- Prerequisites for configuring a logical hostname resource are met.
- Prerequisites for configuring a highly available storage resource are met.
- The name service mapping in the `/etc/nsswitch.conf` file on the cluster nodes is configured to check local files before trying to access NIS or NIS+ for remote procedure call (RPC) lookups.
- The Sun Cluster HA for NFS packages are installed.

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

2. **Start the `clsetup` utility.**

   ```
   # clsetup
   ```

   The `clsetup` main menu is displayed.

3. **Type the number that corresponds to the option for data services and press Return.**

   The Data Services menu is displayed.

4. **Type the number that corresponds to the option for configuring Sun Cluster HA for NFS and press Return.**

   The `clsetup` utility displays the list of prerequisites for performing this task.

5. **Verify that the prerequisites are met, and press Return.**

   The `clsetup` utility displays a list of all cluster nodes that are online.

6. **Select the nodes where you require Sun Cluster HA for NFS to run.**

   - **To accept the default selection of all listed nodes in an arbitrary order, type `a` and press Return.**

   - **To select a subset of the listed nodes, type a comma-separated or space-separated list of the numbers that correspond to the nodes and press Return.**

     Ensure that the nodes are listed in the order in which the nodes are to appear in the resource group's node list. The first node in the list is the primary node of this resource group.

- **To select all nodes in a particular order, type a comma-separated or space-separated ordered list of the numbers that correspond to the nodes and press Return.**

  Ensure that the nodes are listed in the order in which the nodes are to appear in the resource group's node list. The first node in the list is the primary node of this resource group.

7. **To confirm your selection of nodes, type `d` and press Return.**

   The `clsetup` utility displays a list of existing logical hostname resources on the cluster.

8. **You can choose any of the following options.**

   - **To use the existing logical hostname resource, type the number that corresponds to the required logical hostname and press Return. Go to Step 11.**
   - **To create a new logical hostname resource, type `c` and press Return. Go to Step 9**

9. **Type the logical hostname and press Return.**

   The `clsetup` utility displays a list of existing logical hostname resources.

10. **Type the number that corresponds to the logical hostname resource to be created and press Return.**

    The `clsetup` creates the logical hostname resource.

11. **To confirm your selection of logical hostname resource, type `d` and press Return.**

    The `clsetup` utility displays information about file system mount points.

12. **Press Return to continue.**

    The `clsetup` utility displays the existing file system mount points.

13. **Select the file system mount points for Sun Cluster HA for NFS data files.**

    - **To select a subset of the listed file system mount points, type a comma-separated or space-separated list of the numbers that correspond to the file system node point and press Return.**
    - **To select all file system mount points in a particular order, type a comma-separated or space-separated ordered list of the numbers that correspond to the file system mount points and press Return.**

14. **To confirm your selection of file system mount points, type `d` and press Return.**

    The `clsetup` utility displays a screen where you can specify the path prefix for Sun Cluster HA for NFS resource group.

15 Select the path prefix for Sun Cluster HA for NFS resource group and press Return.

The `clsetup` utility displays a screen where you can change the share option for the file system mount point that the NFS server is sharing.

16 Select the share option and press Return.

The `clsetup` utility displays the share options for the selected mount points.

17 If you require a different name for any Sun Cluster objects, change each name as follows.

a. Type the number that corresponds to the name that you are changing and press Return.

The `clsetup` utility displays a screen where you can specify the new name.

b. At the New Value prompt, type the new name and press Return.

The `clsetup` utility returns you to the list of the names of the Sun Cluster objects that the utility will create.

18 To confirm your selection of Sun Cluster object names, type `d` and press Return.

The `clsetup` utility displays information about the Sun Cluster configuration that the utility will create.

19 To create the configuration, type `c` and Press Return.

The `clsetup` utility displays a progress message to indicate that the utility is running commands to create the configuration. When configuration is complete, the `clsetup` utility displays the commands that the utility ran to create the configuration.

20 Press Return to continue.

The `clsetup` utility returns you to the Data Services Menu.

21 Type `q` and press Return.

The `clsetup` utility returns you to the Main Menu.

22 (Optional) Type `q` and press Return to quit the `clsetup` utility.

If you prefer, you can leave the `clsetup` utility running while you perform other required tasks before using the utility again. If you choose to quit `clsetup`, the utility recognizes your Sun Cluster HA for NFS resource group when you restart the utility.

23 Determine if the Sun Cluster HA for NFS resource group and its resources are online.

Use the `clresourcegroup`(1CL) utility for this purpose. By default, the `clsetup` utility assigns the name `nfs-mountpoint-admin-rg` to the Sun Cluster HA for NFS resource group.

```
# clresourcegroup status nfs-mountpoint-admin-rg
```

24 If the Sun Cluster HA for NFS resource group and its resources are *not* online, bring them online.

```
# clresourcegroup online nfs-rg
```

▼ How to Register and Configure Sun Cluster HA for NFS by using Sun Cluster Command Line Interface (CLI)

Before You Begin

- Verify that all the cluster nodes are online.

  ```
  # clnode status
  ```

- 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.

  This configuration prevents timing-related errors for `rpc` lookups during periods of public network or name service unavailability.

1 On a cluster member, become superuser or assume a role that provides `solaris.cluster.admin` RBAC authorization.

2 Create the Pathprefix directory.

Create a Pathprefix directory on the HA file system (global file system or failover file system). Sun Cluster HA for NFS resources will use this directory to maintain administrative information.

You can specify any directory for this purpose. However, you must manually create a Pathprefix directory for each resource group that you create.

```
# mkdir -p Pathprefix-directory
```

3 Create a failover resource group to contain the NFS resources.

```
# clresourcegroup create [-n nodelist] -p Pathprefix=Pathprefix-directory resource-group
```

[`-n` *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.

`-p Pathprefix=`*Pathprefix-directory*
: Specifies a directory that resources in this resource group will use to maintain administrative information. This is the directory that you created in Step 2.

resource-group
: Specifies the failover resource group.

4 **Verify that you have added all 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.

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

This modification enables HA-NFS to fail over correctly in the presence of public network failures.

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

6 **(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.**

In Solaris 10, to customize `nfsd` options, open the `/etc/default/nfs` file and edit the `NFSD_SERVERS` variable.

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)`.

---

7 **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.

```
# clreslogicalhostname create -g resource-group -h logical-hostname, ... [-N netiflist] lhresource
```

`-g` *resource-group*
Specifies the resource group that is to hold the logical hostname resources.

`-h` *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 used 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 – If you require a fully qualified hostname, you must specify the fully qualified name with the `-h` option and you cannot use the fully qualified form in the resource name.

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

---

8 From any cluster node, create the `SUNW.nfs` subdirectory.

Create a subdirectory called `SUNW.nfs` below the directory that the `Pathprefix` property identifies in Step 3.

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

9 Create a `dfstab.`*resource* file in the `SUNW.nfs` directory that you created in Step 8, 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 11). 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 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.

Note – If you want to share multiple paths, the above `share` command need to be repeated for each path that you are sharing.

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 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 the physical hostnames and logical hostnames that are associated with the cluster. Also include the hostnames for all the clients on all 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 those cluster hostnames to the appropriate net group.

The `share -o rw` command grants write access to all 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)

10 Register the NFS resource type.

```
# clresourcetype register resource-type
```

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

11 Create the NFS resource in the failover resource group.

```
# clresource create -g resource-group -t resource-type resource
```

`-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.

resource
Specifies the name of the resource to add, which you defined in Step 9. This name can be your choice but must be unique within the cluster.

The resource is created in the enabled state.

12 Run the `clresourcegroup(1CL)` command to manage the resource group.

```
# clresourcegroup online -M resource-group
```

Example 1 Setting Up and Configuring Sun Cluster HA for NFS

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

1. To create a logical host resource group and specify the path to the administrative files used by NFS (`Pathprefix`), the following command is run.

   ```
   # clresourcegroup create -p Pathprefix=/global/nfs resource-group-1
   ```

2. To add logical hostname resources into the logical host resource group, the following command is run.

   ```
   # clreslogicalhostname create -g resource-group-1 -h schost-1 lhresource
   ```

3. To make the directory structure contain the Sun Cluster HA for NFS configuration files, the following command is run.

   ```
   # mkdir -p /global/nfs/SUNW.nfs
   ```

4. To create the `dfstab.`*resource* file under the `nfs/SUNW.nfs` directory and set share options, the following command is run.

   ```
   # share -F nfs -o rw=engineering -d “home dirs” /global/nfs/SUNW.nfs
   ```

   Note – You also need to add this entry to the `dfstab.`*resource* file.

5. To register the NFS resource type, the following command is run.

   ```
   # clresourcetype register SUNW.nfs
   ```

6. To create the NFS resource in the resource group, the following command is run.

   ```
   # clresource create -g resource-group-1 -t SUNW.nfs r-nfs
   ```

   The resource is created in the enabled state.

7. To enable the resources and their monitors, manage the resource group, and switch the resource group into online state, the following command is run.

   # clresourcegroup online -M *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 the physical hosts or `netgroups` that are associated with all the Sun Cluster servers.

If you use `netgroups` in the `share`(1M) command, add all the Sun Cluster hostnames to the appropriate `netgroup`. Ideally, grant both read access and write access to all 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 the shared path 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 25.

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 On a cluster member, become superuser or assume a role that provides `solaris.cluster.admin` RBAC authorization.

2 Turn off fault monitoring on the NFS resource.

 # clresource unmonitor *resource*

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 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.

```
# unshare -F nfs  [-o specific_options] pathname
```

`-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.

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

```
# vi dfstab.resource
```

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.

```
# clresource monitor resource
```

▼ How to Dynamically Update Shared Paths on an NFS File System

You can dynamically modify the `share` command 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 `cluster status` 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 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 report an error.**

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 receive a stale file error handle and would need special care (forced unmount, or even reboot) to recover.

   b. **Remove the `share` command 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`
- `Start_timeout`
- `Stop_timeout`
- `Validate_timeout`
- `Update_timeout`
- `Monitor_Start_timeout`
- `Monitor_Stop_timeout`
- `Monitor_Check_timeout`

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

```
% clresource set -p Prenet_start_timeout=500 resource
```

Configuring SUNW.HAStoragePlus Resource Type

Sun Cluster HA for NFS is a disk-intensive data service. Therefore, you should configure the `SUNW.HAStoragePlus` resource type for use with this data service. For an overview of the `SUNW.HAStoragePlus` resource type, see "Understanding HAStoragePlus" in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

The procedure for configuring the `SUNW.HAStoragePlus` resource type depends on the type of the file system that NFS is sharing. For more information, see the following sections:

- "How to Set Up the `HAStoragePlus` Resource Type for an NFS-Exported Unix File System Using the Command Line Interface" on page 28
- "How to Set Up the `HAStoragePlus` Resource Type for an NFS-Exported Zettabyte File System" on page 30

▼ How to Set Up the HAStoragePlus Resource Type for an NFS-Exported Unix File System Using the Command Line Interface

The HAStoragePlus resource type performs the same functions as HAStorage, and synchronizes the startups between resource groups and disk device groups. The HAStoragePlus resource type has an additional feature to make a local file system highly available. For background information about making a local file system highly available, see "Enabling Highly Available Local File Systems" in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*. To use both of these features, set up the HAStoragePlus resource type.

Note – These instructions explain how to use the HAStoragePlus resource type with the UNIX file system (UFS). For information about using the HAStoragePlus resource type with the Sun StorEdge™ QFS file system, see your Sun StorEdge QFS documentation.

The following example uses a simple NFS service that exports home directory data from a locally mounted directory /global/local-fs/nfs/export/ home. The example assumes the following:

- The mount point /global/local-fs/nfs is used to mount a UFS local file system on a Sun Cluster global device partition.
- The /etc/vfstab entry for the /global/local-fs/nfs file system should omit the global option and specify that the mount at boot flag is no.
- The path-prefix directory is on the root directory of the same file system that is to be mounted, for example, /global/local-fs/nfs. The path-prefix directory is the directory that HA-NFS uses to maintain administrative information and status information.

1 **On a cluster node, become superuser or assume a role that provides solaris.cluster.modify RBAC authorization.**

2 **Determine whether the HAStoragePlus resource type and the SUNW.nfs resource type are registered.**

The following command prints a list of registered resource types.

```
# clresourcetype show | egrep Type
```

3 **If necessary, register the HAStoragePlus resource type and the SUNW.nfs resource type.**

```
# clresourcetype register SUNW.HAStoragePlus
# clresourcetype register SUNW.nfs
```

4 **Create the failover resource group nfs-rg.**

```
# clresourcegroup create -p PathPrefix=/global/local-fs/nfs nfs-rg
```

5 **Create a logical host resource of type** `SUNW.LogicalHostname`**.**

```
# clreslogicalhostname create -g nfs-rg -L -h log-nfs nfs-lh-rs
```

Note – If you require a fully qualified hostname, you must specify the fully qualified name with the `-h` option and you cannot use the fully qualified form in the resource name.

---

6 **Create the resource** `nfs-hastp-rs` **of type** `HAStoragePlus`**.**

```
# clresource create -g nfs-rg -t SUNW.HAStoragePlus \
-p FilesystemMountPoints=/global/local-fs/nfs \
-p AffinityOn=True nfs-hastp-rs
```

The resource is created in the enabled state.

Note – You can use the `FilesystemMountPoints` extension property to specify a list of one or more mount points for file systems. This list can consist of mount points for both local file systems and global file systems. The mount at boot flag is ignored by `HAStoragePlus` for global file systems.

---

7 **Bring online the resource group** `nfs-rg` **on a cluster node.**

The node or zone where the resource group is brought online becomes the primary node for the `/global/local-fs/nfs` file system's underlying global device partition. The file system `/global/local-fs/nfs` is then mounted on this node or zone.

```
# clresourcegroup online -M nfs-rg
```

8 **Create the resource** `nfs-rs` **of type** `SUNW.nfs` **and specify its resource dependency on the resource** `nfs-hastp-rs`**.**

The file `dfstab.nfs-rs` must be present in `/global/local-fs/nfs/SUNW.nfs`.

```
# clresource create -g nfs-rg -t SUNW.nfs \
-p Resource_dependencies=nfs-hastp-rs nfs-rs
```

The resource is created in the enabled state.

Note – Before you can set the dependency in the `nfs-rs` resource, the `nfs-hastp-rs` resource must be online.

---

9 **Take offline the resource group** `nfs-rg`**.**

```
# clresourcegroup offline nfs-rg
```

10 **Bring online the** `nfs-rg` **group on a cluster node or zone.**

```
# clresourcegroup online -M nfs-rg
```

Caution – Ensure that you switch only the resource group. Do *not* attempt to switch the device group. If you attempt to switch the device group, the states of the resource group and the device group become inconsistent, causing the resource group to fail over.

Whenever the service is migrated to a new node, the primary I/O path for `/global/local-fs/nfs` will always be online and colocated with the NFS servers. The file system `/global/local-fs/nfs` is locally mounted before the NFS server is started.

How to Set Up the HAStoragePlus Resource Type for an NFS-Exported Zettabyte File System

The following procedure uses a simple NFS service.

See "Creating a ZFS Storage Pool in *Solaris ZFS Administration Guide*" for information about how to create a ZFS pool. See "Creating a ZFS File System Hierarchy" in *Solaris ZFS Administration Guide* for information about how to create a ZFS in that ZFS pool.

1. **On a cluster node, become superuser or assume a role that provides `solaris.cluster.modify` RBAC authorization.**

2. **Determine whether the `HAStoragePlus` resource type and the `SUNW.nfs` resource type are registered.**

   The following command prints a list of registered resource types.

   ```
   # clresourcetype list
   ```

3. **If necessary, register the `HAStoragePlus` resource type and the `SUNW.nfs` resource type.**

   ```
   # clresourcetype register SUNW.HAStoragePlus SUNW.nfs
   ```

4. **Create the failover resource group.**

   ```
   # clresourcegroup create -p PathPrefix=path resource-group
   ```

5. **Create a logical host resource of type `SUNW.LogicalHostname`.**

   ```
   # clreslogicalhostname create -g resource-group \
   -h logical-hostname logicalhost-resource
   ```

   Note – If you require a fully qualified hostname, you must specify the fully qualified name with the `-h` option and you cannot use the fully qualified form in the resource name.

6. **Create the ZFS resource of type `HAStoragePlus`.**

   ```
   # clresource create -g resource-group -t SUNW.HAStoragePlus \
   -p Zpools=zpool HASP-resource
   ```

The resource is created in the enabled state.

Note – You can specify a list of one or more ZFS pools for the `Zpools` extension property.

---

7 Bring online the resource group on a cluster node in a managed state.

The node on which the resource group is brought online becomes the primary node for the ZFS. The ZFS pool *zpool* is imported on this node. The ZFS is consequently mounted locally on this node.

```
# clresourcegroup online -M resource-group
```

8 Create the resource of type `SUNW.nfs` and specify its resource dependency on the resource of type `SUNW.HAStoragePlus`.

The file `dfstab.nfs-rs` must be present in *zpool*`/nfs/SUNW.nfs`.

```
# clresource create -g resource-group -t SUNW.nfs \
-p Resource_dependencies=HASP-resource NFS-resource
```

The resource is created in the enabled state.

Note – Before you can set the dependency in the *NFS-resource* resource, the *HASP-resource* resource must be online.

---

9 Bring online the *resource-group* group on a cluster node in a managed state.

```
# clresourcegroup online -M resource-group
```

Example 2 Setting Up the `HAStoragePlus` Resource Type for an NFS-Exported ZFS

The following example uses a simple NFS service. The example assumes the following:

- The `nfs/export` directory exists in the ZFS pool `/nfszpool`.
- The `dfstab.`*resource* file exists in the `/nfszpool/nfs/SUNW.nfs` directory.
- The path-prefix directory is on the root directory of the same file system that is to be mounted, for example, `/nfszpool/nfs`. The path-prefix directory is the directory that HA-NFS uses to maintain administrative information and status information.

```
phys-schost-1% su
Password:
# clresourcetype list
SUNW.LogicalHostname:2
SUNW.SharedAddress:2
# clresourcetype register SUNW.HAStoragePlus SUNW.nfs
# clresourcegroup create -p PathPrefix=/nfszpool/nfs nfs-rg
# clreslogicalhostname create -g nfs-rg -h log-nfs nfs-lh-rs
# clresource create -g nfs-rg -t SUNW.HAStoragePlus \
```

```
                    -p Zpools=nfszpool nfs-hastp-rs
# clresourcegroup online -M nfs-rg
# clresource create -g nfs-rg -t SUNW.nfs \
                    -p Resource_dependencies=nfs-hastp-rs nfs-rs
# clresourcegroup online -M nfs-rg
```

Securing Sun Cluster HA for NFS With Kerberos V5

You can secure Sun Cluster HA for NFS with Kerberos V5 by configuring the Kerberos client. This configuration includes adding a Kerberos principal for NFS over the logical hostnames on all cluster nodes.

To configure the Kerberos client, perform the following procedures.

- Prepare the nodes. See "How to Prepare the Nodes" on page 32.
- Create Kerberos principals. See "How to Create Kerberos Principals" on page 33.
- Enable the secured NFS. See "Enabling Secure NFS" on page 36.

▼ How to Prepare the Nodes

1 Configure the Key Distribution Center (KDC) server that will be used by the cluster nodes.

Refer to Solaris Kerberos/SEAM (Sun Enterprise Authentication Mechanism) documentation for details.

2 Set up the time synchronization.

The KDC server must be time synchronized with the cluster nodes as well as any clients that will be using the Sun Cluster HA for NFS services from the cluster. The Network Time Protocol (NTP) method performs time corrections with greater granularity than other methods, and therefore the time synchronization is more reliable. To benefit from this greater reliability, use NTP for the time synchronization.

3 Verify the DNS client configuration.

The DNS client configuration must be complete and working on all cluster nodes as well as on any NFS clients that will be using secure NFS services from the cluster. Use `resolv.conf(4)` to verify the DNS client configuration.

The DNS domain name must be made known to the Kerberos configuration by keeping a mapping in the `domain_realm` section of `krb5.conf(4)` file.

The following example shows a mapping of DNS domain name `mydept.company.com` to Kerberos realm `ACME.COM`.

```
[domain_realm]
.mydept.company.com = ACME.COM
```

4 **Ensure that the Master KDC server is up when the Kerberos client software is configured on the cluster nodes.**

5 **Ensure that the same configuration file and the same service key table file are available to all cluster nodes.**

The `/etc/krb5/krb5.conf` file must be configured the same on all the cluster nodes. In addition, the default Kerberos keytab file (service key table), `/etc/krb5/krb5.keytab`, must be configured the same on all the cluster nodes. Consistent configuration can be achieved by copying the files to all cluster nodes. Alternately, you can keep a single copy of each file on a global file system and install symbolic links to `/etc/krb5/krb5.conf` and `/etc/krb5/krb5.keytab` on all cluster nodes.

You can also use a failover file system to make files available to all cluster nodes. However, a failover file system is visible on only one node at a time. Therefore, if Sun Cluster HA for NFS is being used in different resource groups, potentially mastered on different nodes, the files are not visible to all cluster nodes. In addition, this configuration complicates Kerberos client administrative tasks.

6 **Ensure that all Kerberos-related entries in the file `/etc/nfssec.conf` are uncommented.**

On all cluster nodes, as well as on any NFS clients that are configured to use secure NFS services from the cluster, all Kerberos-related entries in the file `/etc/nfssec.conf` must be uncommented. See `nfssec.conf(4)`.

▼ How to Create Kerberos Principals

The following steps create the required Kerberos principals and keytab entries in the KDC database. For each cluster node, the keytab entries for which service principals are created depend on the version of Solaris that is running on the cluster node.

- In Solaris 8, both the "root" and the "host" entries must be created.
- In Solaris 9, only the "host" entry must be created.

The principal for the "nfs" service over the logical hostname is created on one node only and then added manually to the default Kerberos keytab file on each cluster node. The Kerberos configuration file `krb5.conf` and the keytab file `krb5.keytab` must be stored as individual copies on each cluster node and must not be shared on a global file system.

1 **On each cluster node, log in to the KDC server as the administrator and create the host principal for each cluster node.**

Note that, with Solaris 8, you must create both host and root principals for each cluster node.

Principals must be created using the fully qualified domain names.

Add these entries to the default keytab file on each node. These steps can be greatly simplified with the use of cluster console utilities (see `cconsole(1M)`).

The following example creates the root and host entries. Perform this step on all cluster nodes, substituting the physical hostname of each cluster node for the hostname in the example.

```
# kadmin -p username/admin
Enter Password:
kadmin: addprinc -randkey host/phys-red-1.mydept.company.com
Principal "host/phys-red-1.mydept.company.com@ACME.COM" created.

kadmin: addprinc -randkey root/phys-red-1.mydept.company.com
Principal "root/phys-red-1.mydept.company.com@ACME.COM" created.

kadmin: ktadd host/phys-red-1.mydept.company.com
Entry for principal host/phys-red-1.mydept.company.com with kvno 2,
encryption type DES-CBC-CRC added to keytab WRFILE:/etc/krb5/krb5.keytab.

kadmin: ktadd root/phys-red-1.mydept.company.com
Entry for principal root/phys-red-1.mydept.company.com with kvno 2,
encryption type DES-CBC-CRC added to keytab WRFILE:/etc/krb5/krb5.keytab.

kadmin: quit
#
```

2 On one cluster node, create the principal for the Sun Cluster HA for NFS service for the logical hostnames which provide Sun Cluster HA for NFS service.

Principals must be created using the fully qualified domain names. Perform this step on only one cluster node.

```
# kadmin -p username/admin
Enter Password:
kadmin: addprinc -randkey nfs/relo-red-1.mydept.company.com
Principal "nfs/relo-red-1.mydept.company.com@ACME.COM" created.

kadmin: ktadd -k /var/tmp/keytab.hanfs nfs/relo-red-1.mydept.company.com
Entry for principal nfs/relo-red-1.mydept.company.com with kvno 3,
encryption type DES-CBC-CRC added to keytab WRFILE:/var/tmp/keytab.hanfs.

kadmin: quit
#
```

In the above example, `relo-red-1` is the logical hostname used with Sun Cluster HA for NFS.

3 Securely copy the keytab database `/var/tmp/keytab.hanfs` specified in Step 2 to the rest of the cluster nodes.

Do not use insecure copying methods such as regular `ftp` or `rcp`, and so forth. For additional security, you can use the cluster private interconnect to copy the database.

The following example copies the database.

```
# scp /var/tmp/keytab.hanfs clusternode2-priv:/var/tmp/keytab.hanfs
# scp /var/tmp/keytab.hanfs clusternode3-priv:/var/tmp/keytab.hanfs
```

4 On all cluster nodes, add the keytab entry for the "nfs" service over logical hostname to the local keytab database.

The following example uses the `ktutil(1M)` command to add the entry. Remove the temporary keytab file `/var/tmp/keytab.hanfs` on all cluster nodes after it has been added to the default keytab database `/etc/krb5/krb5.keytab`.

```
# ktutil
ktutil: rkt /etc/krb5/krb5.keytab
ktutil: rkt /var/tmp/keytab.hanfs
ktutil: wkt /etc/krb5/krb5.keytab
ktutil: quit#
# rm /var/tmp/keytab.hanfs
```

5 Verify the Kerberos client configuration.

List the default keytab entries on each cluster node and make sure that the key version number (KVNO) for the "nfs" service principal is the same on all cluster nodes.

```
# klist -k
Keytab name: FILE:/etc/krb5/krb5.keytab
KVNO Principal
---- --------------------------------
2    host/phys-red-1.mydept.company.com@ACME.COM
2    root/phys-red-1.mydept.company.com@ACME.COM
3    nfs/relo-red-1.mydept.company.com@ACME.COM
```

On all cluster nodes, the principal for the "nfs" service over the logical host must have the same KVNO number. In the above example, the principal for the "nfs" service over the logical host is `nfs/relo-red-1.mydept.company.com@ACME.COM`, and the KVNO is 3.

6 (Solaris 9 only) The user credentials database `gsscred` must be up-to-date for all users who access secure NFS services from the cluster.

Build the user credential database by running the following command on all cluster nodes.

```
# gsscred -m kerberos_v5 -a
```

See `gsscred(1M)` man pages for details.

Note that the above approach builds the user credentials database only once. Some other mechanism must be employed, for example, `cron(1M)`, to keep the local copy of this database up-to-date with changes in the user population.

This step is not necessary for Solaris release 10.

Enabling Secure NFS

Use the `-o sec=`*option* option of the `share(1M)` command in the `dfstab.`*resource-name* entry to share your file systems securely. See `nfssec(5)` man pages for details of specific option settings. If the Sun Cluster HA for NFS resource is already configured and running, see "How to Change Share Options on an NFS File System" on page 24 for information about updating the entries in the `dfstab.`*resource-name* file. Note that the `sec=dh` option is not supported on Sun Cluster configurations.

Tuning the Sun Cluster HA for NFS Fault Monitor

The Sun Cluster HA for NFS fault monitor is contained in a resource whose resource type is `SUNW.nfs`.

For general information about the operation of fault monitors, see "Tuning Fault Monitors for Sun Cluster Data Services" in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*.

Fault Monitor Startup

The 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

The NFS resource `MONITOR_STOP` method stops the resource fault monitor. If no other NFS resource fault monitor is running on the local node, the stop method stops the NFS system fault monitor.

Operations of Sun Cluster HA for NFS Fault Monitor During a Probe

This section describes the operations of the following fault monitoring processes:

- NFS system fault monitoring
- NFS resource fault monitoring

- Monitoring of file sharing

NFS System Fault Monitoring Process

The NFS system fault monitor probe monitors the NFS daemons (`nfsd`, `mountd`, `statd`, and `lockd`) and the RPC portmapper service daemon (`rpcbind`) on the local node. The probe checks 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`
- `Rpcbind_nullrpc_reboot`
- `Statd_nullrpc_timeout`
- `Lockd_nullrpc_timeout`
- `Mountd_nullrpc_timeout`
- `Mountd_nullrpc_restart`
- `Nfsd_nullrpc_timeout`
- `Nfsd_nullrpc_restart`

See "Setting Sun Cluster HA for NFS Extension Properties" on page 14.

Each NFS system fault monitor probe cycle performs the following steps in a loop. The system property `Cheap_probe_interval` specifies the interval between probes.

1. The fault monitor probes `rpcbind`.

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

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

   If a null `rpc` call to the daemon terminates unexpectedly, `Rpcbind_nullrpc_reboot=True`, and `Failover_mode=HARD`, the fault monitor reboots the node.

2. The fault monitor 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`.

3. The fault monitor 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.

4. The fault monitor 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`, the fault monitor attempts to restart `nfsd` if the cluster file system is available.

5. If any one of the above NFS daemons (except `rpcbind`) fails to restart during a probe cycle, the NFS system fault monitor will retry the restart in the next cycle. When all 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.
6. At the end of each probe cycle, if all daemons are healthy, the monitor clears the history of failures.

NFS Resource Fault Monitoring Process

NFS resource fault monitoring 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.

Before starting the NFS resource fault monitor probes, all the shared paths are read from the `dfstab` file and stored in memory. In each probe cycle, the probe performs the following steps.

1. If `dfstab` has been changed since the last read, the probe refreshes the memory.

   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.
2. The fault monitor probes all 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`.
3. The probe checks for the presence of NFS daemons (`nfsd`, `mountd`, `lockd`, `statd`) and `rpcbind`.
4. If any of these daemons are down, the resource status is set to `FAULTED`.
5. If all shared paths are valid and NFS daemons are present, the resource status is reset to `ONLINE`.

Monitoring of File Sharing

The Sun Cluster HA for NFS fault monitor probe monitors the success or failure of file sharing by monitoring the following files:

- `/etc/dfs/sharetab`
- `/etc/mnttab`
- *Pathprefix*`/SUNW.nfs/dfstab.`*resource*

The *Pathprefix* part of the file path is the value of the `Pathprefix` extension property for the resource group, and *resource* is the resource name.

If the probe detects any modification to any of these files, it shares the paths in `dfstab.`*resource* again.

Upgrading the `SUNW.nfs` Resource Type

Upgrade the `SUNW.nfs` resource type if the following conditions apply:

- You are upgrading the Sun Cluster HA for NFS data service to Sun Cluster 3.2 12/07 from an earlier version of the data service.
- You are upgrading to Solaris 10 from an earlier version of the operating system.

For general instructions that explain how to upgrade a resource type, see "Upgrading a Resource Type" in *Sun Cluster Data Services Planning and Administration Guide for Solaris OS*. The information that you require to complete the upgrade of the resource type is provided in the subsections that follow.

Information for Registering the New Resource Type Version

The release of Sun Cluster data services indicates the release in which the version of the resource type was introduced.

To determine the version of the resource type that is registered, use the `clresourcetype show` command.

The resource type registration (RTR) file for this resource type is `/opt/SUNWscnfs/etc/SUNW.nfs`.

Information for Migrating Existing Instances of the Resource Type

The information that you require to edit each instance of the resource type is as follows:

- You must perform the migration when the resource group is in an unmanaged state.
- For Sun Cluster 3.2 2/08, the required value of the `Type_version` property is `3.2`.

The following example shows a command for modifying an instance of the `SUNW.nfs` resource type.

EXAMPLE 3 Migrating Instances of the SUNW.nfs Resource Type

```
# clresource set -p Type_version=3.2 nfs-rs
```

This command modifies the Type_version property of the nfs-rs resource to 3.2.

◆ ◆ ◆ APPENDIX A

Sun Cluster HA for NFS Extension Properties

This section describes the extension properties for the resource type `SUNW.nfs`. This resource type represents the Network File System (NFS) application in a Sun Cluster configuration.

For details about system-defined properties, see the `r_properties(5)` man page and the `rg_properties(5)` man page.

The extension properties of the `SUNW.nfs` resource type are as follows:

`Lockd_nullrpc_timeout`
The time-out value (in seconds) to use when probing `lockd`

Data type Integer

Default 120

Range Minimum = 60

Tunable At any time

`Monitor_retry_count`
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.

Data type Integer

Default 4

Range 0 – 2,147,483,641

A value of –1 indicates an infinite number of restart attempts.

Tunable At any time

`Monitor_retry_interval`
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.

Data type Integer

Default 2

Range 0 – 2,147,483,641

–1 indicates an infinite amount of time.

Tunable At any time

`Mountd_nullrpc_restart`
A Boolean to indicate whether to restart `mountd` when a null `rpc` call fails.

Data type Boolean

Default `True`

Range Not applicable

Tunable At any time

`Mountd_nullrpc_timeout`
The time-out value (in seconds) to use when probing `mountd`.

Data type Integer

Default 120

Range Minimum = 60

Tunable At any time

`Nfsd_nullrpc_restart`
A Boolean to indicate whether to restart `nfsd` when a null `rpc` call fails.

Data type Boolean

Default `False`

Range Not applicable

Tunable At any time

`Nfsd_nullrpc_timeout`
The time-out value (in seconds) to use when probing `nfsd`.

Data type Integer

Default 120

Range Minimum = 60

Tunable At any time

`Rpcbind_nullrpc_reboot`
A Boolean to indicate whether to reboot the system when a null `rpc` call on `rpcbind` fails.

Data type Boolean

Default `False`

Range Not applicable

Tunable At any time

`Rpcbind_nullrpc_timeout`
The time-out value (in seconds) to use when probing `rpcbind`.

Data type Integer

Default 120

Range Minimum = 60

Tunable At any time

`Statd_nullrpc_timeout`
The time-out value (in seconds) to use when probing `statd`.

Data type Integer

Default 120

Range Minimum = 60

Tunable At any time

Index

R

S

T

U

V

Z