Implementation Guide:

Guide to Installation—Part II: Sun™ Cluster Management Services

Chris Dotson, Enterprise Engineering
Steve Lopez, Enterprise Engineering

Sun BluePrints™ OnLine—May 2002

http://www.sun.com/blueprints

Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054 U.S.A.
650-960-1300
Part No.: 816-5133-10
Revision A
Edition: May 2002

Please
Recycle

Module 2

Part 2: Sun™ Cluster Management Services

The purpose of this module is to guide you through the tasks you must perform before you install the Sun™ Cluster 3.0 software. These tasks include setting up the administrative workstation and configuring the Sun Cluster 3.0 hardware components.

The exercises in this module explain how to install and configure a workstation to perform Sun Cluster 3.0 software administrative functions in a cluster environment. Additionally, we provide instructions for configuring the cluster, implementing key practices, and performing design verifications, as well as administering a two-node Sun Cluster 3.0 hardware cluster.

For information about managing a cluster, refer to the documents referenced in Appendix B, specifically:

- *System Administration Guide* (“Preparing to Administer the Cluster,” “Beginning to Administer the Cluster,” “Administering the Cluster,” and “Administering Sun Cluster With the Graphical User Interface”)
- S*C3.0 U1 Cluster Concepts* (“Cluster Administration and Application Development”)

Many of the steps in this guide refer to “manual (local) procedures” that you should perform only if you have local (physical) access to the SunPlex™ platform. For example, resetting the terminal concentrator in order to activate specific settings.

Objectives

After completing this module, you will have successfully verified the installation and software configuration for each cluster component. These tasks must be performed before you can install the Sun Cluster 3.0 software on each cluster node. Additionally, you will have implemented the associated key practices during each task, including:

- Configuring the Solaris™ Operating Environment (Solaris OE). This task is site-specific.
- Verifying that the administrative workstation setup on the management server.
- Verifying the terminal concentrator (TC) configuration.
- Installing and configuring the Cluster Console utility in the management server environment.
- Verifying each cluster node installs with the Solaris OE and patches.
- Configuring additional cluster management services, such as JumpStart™ software for each cluster node, plus additional platform management services. This task is performed only during manual installations.

Many of the steps for configuring the cluster require you to have physical access to the equipment. We have made special note of specific configuration procedures that require local (physical) access to the SunPlex platform. For example, forcing the TC into Monitor mode requires local, manual procedures. For this reason, these hands-on exercises specify steps that are only performed as part of local (or manual) installation procedures. These manual steps are not performed if you are accessing the Sun Cluster 3.0 hardware (SunPlex platform) remotely. Instead, these steps are included as documentation of the procedures required to complete the configuration of each cluster component (as referenced in Figure 1-1 of “Guide to Installation—Hardware Setup”). Additionally, it is assumed that it is not necessary to provide the detailed procedures for installing the Solaris OE (plus Solaris OE patches) on each SunPlex node. Instead, we define the parameters required for installation.

Note – In this module, you must perform all of the associated instructions to verify that the correct configuration has been achieved as specified. It is very important that you do this prior to configuring the additional cluster components to ensure consistent, reliable cluster services and SunPlex platform operations.

Prerequisites

As stated in the introduction, this lab guide is intended for use by qualified network administrators.

For installation queries, shell usage questions, patches, and packages, refer to the Sun Educational Services manuals for the “Solaris System Administration 1” and “Solaris System Administration 2” courses.

Required Reading

This module builds on concepts presented in the Cluster Column published in January, 2002, “Guide to Installation—Hardware Setup.” Before you read this module, familiarize yourself with the content in that publication, paying special attention to the hardware configuration and components described in Figure 1-1 and Tables 1-1 to 1-5.

Introduction

The administrative workstation (`clustadm`) is required for setting up and configuring the cluster and can be used for ongoing cluster management and operations.

The administrative workstation is a remote workstation that can be used to administer each cluster node by connecting through the terminal concentrator. You can use the administration workstation to administer and monitor the cluster through the cluster control panel, the command-line interface, or through the Sun™ Management Center console (monitoring only).

Note – Beyond these administrative workstation functions, we recommend that you implement additional cluster management services. When configuring this type of management server, consider combining JumpStart software services, applications software (and `/PATCH`) repository, along with Sun Cluster 3.0 software administrative functions appropriate to your implementation.

In this module, we describe the procedures used to install the administrative workstation (`clustadm`). We explain how to confirm that all requirements are met, verify that all cluster components (including patches) are installed correctly, and ensure that the student-selected shell environment is configured.

Management Server Functions

Management servers can be used to perform the following functions:

- Distributing (that is, saving) valuable cluster node (site-specific) configuration files
- Configuring the Solaris OE `syslogd`(1M) facility to copy messages from each cluster node to a remote administrative workstation (or another designated host) for the purpose of notifying administrators of potential cluster disruptions

  These logs should be saved and reviewed, as necessary.
- Running the Cluster Control panel (CCP) to access the Cluster Console window for each cluster node in the SunPlex platform

  As instructed throughout this guide, during the installation (and when monitoring cluster operation), commands will be entered simultaneously to each cluster node using a Cluster Console window (one window for each node), accessed from the CCP Hosts menu.

Note – These hands-on labs demonstrate remote cluster operations (for example, with CCP operations), including configuring and monitoring the cluster and status, managing changes within the cluster (for example, shutting down individual cluster nodes on the SunPlex platform prior to performing upgrades), rebooting a cluster node, and ensuring that auto-cluster formation occurs.

- Setting up time synchronization throughout the SunPlex platform
- Setting up Solaris installation services using JumpStart software
- Acting as an applications server for VERITAS Volume Manager software (VxVM), Solstice DiskSuite™ software, Sun Cluster software, and associated utilities
- Acting as an online repository for applications (binaries), patches, and site-specific data and configuration files

For more information about cluster administration functions (for example, administering cluster interconnects and public networks, patching Sun Cluster software and firmware, and backing up and restoring a cluster), refer to the documentation referenced in Appendix B, including the *Sun Cluster 3.0 U1 System Administration Guide.*

Optionally, the management server can be configured to act as a Sun Cluster 3.0 software-installation server to help simplify software installations for each cluster node on the SunPlex platform. For example, you can install the Sun Cluster 3.0 software packages on the management server and populate a custom JumpStart software directory.

Key Practice: Use the JumpStart technology to maintain consistency and fully automate the installation of the Solaris OE and additional software packages. The JumpStart software can minimize operator errors that occur during a manual installation process. Combining JumpStart software and Flash archives (for example, with the `flar`(1M) and `flarcreate`(1M) commands) can help enable quick and consistent disaster recovery operations.

Note – For information about implementing Flash archives, reference recent publications available from Sun BluePrints™ at `http://www.sun.com/blueprints/`, as well as the documentation referenced in Appendix B.

Not Required for Application Failover

Clusters do not require the use of the administrative workstation for normal application processing to occur or to achieve automatic failover to a standby node.

Section 2.7: Solaris OE Installation Preparation on Each Cluster Node

Prior to installing the Solaris OE on each cluster node, we will reconfigure the OpenBoot™ PROM settings for each cluster node.

Note – During many of the following steps, you will be required to enter commands simultaneously into each cluster node. Use the Cluster Control Panel window for this purpose. Double-click the Cluster Console (console mode) icon, as described in the previous section, “Configuring the Cluster Control Panel.”

Key Practice: Until the EEPROM configuration has been completed, you should disable the auto-boot EEPROM feature on each cluster node. Disabling the auto-boot feature will alleviate any problems that arise if both systems attempt to boot their Solaris OE, while at the same time, both systems are set with the same (that is, conflicting) SCSI-initiator ID settings, the boot would fail.

We disable auto-booting on each cluster node by configuring the `auto-boot?` variable to `FALSE` during this phase of the installation. We do this because, the system has not yet been configured. If there is an accidental reboot of a node and if the system `auto-boot?` variable has been set to `FALSE`, the system will reset to the OpenBoot PROM prompt, instead of attempting to boot from the boot disk. At this phase, an attempt to boot from the disk may require an administrator to manually put the system back to the OpenBoot PROM for further configuration changes.

You will be instructed to re-enable this important feature at the end of this procedure.

Step 2.7.1—Disabling the Automatic Boot Capability

Disable auto-booting by entering the following command into both nodes (we want to disable the auto boot capability temporarily until we have built a stable Solaris OE on the machine). The auto-boot variable will be re-enabled later in the exercise.

```
ok setenv auto-boot? false
auto-boot? = false
```

Step 2.7.2—Changing the SCSI Initiator ID

Enter the following command into the cconsole: host clustnode2 window.

```
ok setenv scsi-initiator-id 6
scsi-initiator-id = 6
```

Note – For SCSI initiator ID modification, refer to Figure 1-1 and Tables 1-1 through 1-5, specifically noting the disk subsystem cabling and configuration. Because two cluster nodes (both Sun Enterprise™ 220R hardware servers) are connected to the same pair of Sun StorEdge™ D1000 arrays, the OpenBoot PROM settings require modification. We will set the SCSI initiator ID on one of the cluster nodes (clustnode2 in this exercise) to a value of 6 and insert a script into the nvramrc (nonvolatile memory) on clustnode2 to maintain a SCSI initiator ID of 7 for the internal SCSI controller on clustnode2. Setting the global SCSI initiator ID on clustnode2 to 6 will prevent a conflict on the shared SCSI bus that connects both Sun Enterprise 220R servers to the Sun StorEdge D1000 arrays.

By changing the SCSI initiator ID on clustnode2 to a value of 6, we are making a global variable change that affects other devices attached to the internal SCSI controller on clustnode2. Specifically, setting the global SCSI initiator ID to 6 will create a conflict between the internal SCSI controller and the internal CD-ROM drive. To solve this conflict, we will explicitly set the SCSI initiator ID of the internal SCSI controller on clustnode2 to a value of 7 by including specific script commands into the nvramrc.

Note – Use the OpenBoot PROM nvedit command in the following procedure. The nvram editor is always set to *insert* mode. Use the following keystrokes when editing.

Using `nvedit` Keystrokes

Keystroke	Action
Ctrl+B	Moves backward one character.
Ctrl+C	Exits the `nvramrc` editor, returning to the Open Boot PROM command interpreter. The temporary buffer is preserved, but it is not written back to the `nvramrc` editor. (Use `nvstore` afterwards to write it back.)
Delete	Deletes previous character.
Ctrl+F	Moves forward one character.
Ctrl+K	From current position in a line, deletes all text after the cursor and joins the next line to the current line (that is, it deletes the new line).
Ctrl+L	Lists all lines.
Ctrl+N	Moves to the next line of the `nvramrc` editing buffer.
Ctrl+O	Inserts a new line at the cursor position and stays on the current line.
Ctrl+P	Moves to the previous line of the `nvramrc` editing buffer.
<CR>	Inserts a new line at the cursor position and advances to the next line.

Note – Using `nvedit` can be tricky because there is no *delete* command. If you want to delete a line, you must delete all of the characters for the line. Then, use Ctrl+K to join the empty line with the subsequent line.

Step 2.7.3—Setting the Internal SCSI Controller

Set the internal SCSI controller (/pci@1f,4000/scsi@3) SCSI-initiator-ID value to 7 by using the nvedit command. Enter the following commands into the cconsole: host clustnode2 window.

```
ok printenv nvramrc

nvramrc =

{{ensure that no previous commands or entries exist in nvram before
proceeding}}

ok nvedit {{invoke the nvram editor}}
   0: probe-all
   1: cd /pci@1f,4000/scsi@3
   2: 7 " scsi-initiator-id" integer-property {{note the space before AND
after the first quotation prior to the word “scsi” in this line}}
   3: device-end
   4: install-console
   5: banner
   6: {{at this point use Ctrl + C to exit nvedit}}

ok nvstore
ok printenv nvramrc {{verify that this exactly matches with your screen
output}}

nvramrc =         probe-all
                  cd /pci@1f,4000/scsi@3
                  7 " scsi-initiator-id" integer-property
                  device-end
                  install-console
                  banner
```

Step 2.7.4—Enabling the nvramrc Variable

Enter the following command into the cconsole: host clustnode2 window to enable the nvramrc variable.

```
ok setenv use-nvramrc? true
use-nvramrc? = true
```

Step 2.7.5—Verifying the nvramrc Script

Verify that the nvramrc script works by performing the following steps on clustnode2.

1. On clustnode2, reset the system by entering the reset command into the OpenBoot PROM prompt.
2. After clustnode2 resets, enter the printenv scsi-initiator-id command into the OpenBoot PROM prompt to confirm that the global SCSI-initiator-ID is set to 6.

```
ok reset
Resetting ...
ok printenv scsi-initiator-id
scsi-initiator-id = 6
```

3. On clustnode2, use the cd(1) command to navigate to the node (that is, directory) that represents the internal SCSI controller /pci@1f,4000/scsi@3.
4. At the OpenBoot PROM prompt, enter the .properties command to verify that the internal SCSI initiator ID on clustnode2 is set to 7, as indicated in the next codebox.

```
ok cd /pci@1f,4000/scsi@3
ok .properties
scsi-initiator-id 00000007
.
.
.
```

Step 2.7.6—Resetting the auto-boot Variable

After you have verified that the SCSI-initiator-ID settings are correct, reset the EEPROM auto-boot variables back to true on each cluster node.

```
ok setenv auto-boot? true
auto-boot? = true
```

Summary of Key Practices

Disable the OpenBoot PROM `auto-boot` feature on each cluster node until EEPROM configuration is complete.

Be careful when using `nvedit` because editing can be tricky (that is, it is always set to insert mode).

Section 2.8: Install the Solaris 8 Operating Environment on Each Cluster Node

For local (manual) installations, configure the management server (administrative workstation) as a Solaris OE installation (JumpStart software) server by modifying the JumpStart software configuration to include new class file. These class files should configure a machine to operate as a node within a cluster, according to the following guidelines.

For these hands-on labs, a JumpStart software server has already been configured.

Note – The Solaris OE, plus patch, installation is *not* performed during these hands-on labs. Instead, each cluster nodes has been previously installed with the Solaris OE and patches.

Caution – The same version of the Solaris OE must be installed on the Sun Cluster 3.0 software administrative workstation and each of the cluster nodes in the SunPlex platform. In this procedure, Solaris OE version 8 07/01 is used.

Key Practice: Use the Solaris JumpStart software to maintain consistency and to fully automate the installation of the Solaris OE and additional software packages. The JumpStart software can minimize operator errors that occur during a manual installation process.

Step 2.8.1—Creating a New Class File

For local (manual) installations only, on the management server (the JumpStart software install server), create a new class file (profile) in the `/JumpStart` directory called `cluster_node.class` and include the following lines.

```
install_type initial_install
system_type server
partitioning explicit
cluster   SUNWCXall {{we select Entire Distribution plus
OEM}}
usedisk   c0t0d0
filesys   c0t0d0s0   32768   /
filesys   c0t0d0s1   1024    swap
```

Note – The above code is an example only in which we configure 1 Gbyte of swap space and leave all of the remaining space to root. For Sun Cluster 3.0 software, a minimum of 750 Mbytes of swap space is recommended for each cluster node. For site-specific (actual) implementations, swap size should be configured based on the actual requirements of the application(s) to be hosted. For additional information, see the JumpStart software references in Appendix B.

Step 2.8.2—Ensuring Proper Hostname Configuration

The following sample shows the `/etc/inet/hosts` file entries required for these exercises. At this time, ensure the correct (site-specific) hostnames are configured, as shown in the following example.

```
clustadm# more /etc/inet/hosts

. . . {{output omitted}}

xxx.xxx.xx.xxx  clustadm loghost
xxx.xxx.xx.xxx  clustnode1
xxx.xxx.xx.xxx  clustnode2
xxx.xxx.xx.xxx  tc nhl-tc

. . . {{output omitted}}

xxx.xxx.xx.xxx  lh-hanfs
xxx.xxx.xx.xxx  lh-apache

. . . {{output omitted}}
```

Note – The preceding hostname and IP address information is an example. Carefully note the actual values required at your site. We do not use a naming service or domain name. Instead, we rely on local (`/etc`) files. For additional configuration information regarding these exercises, see "Section 2.10: The Solaris OE Post Installation and Configuration" section on page Module 2-29.

Step 2.8.3—Editing the `/JumpStart/rules` File

For local (manual) installations, on the JumpStart software install server (management server), edit the `/JumpStart/rules` file, and add the following lines.

```
hostname clustnode1 - cluster_node.class set_root_pw
hostname clustnode2 - cluster_node.class set_root_pw
```

Note – The `/JumpStart/rules` file should have an entry for *each* node in the cluster.

Substep 1—Creating a Root Password for Each Cluster Node

For local (manual) installations, on the management server (administrative workstation), edit the `/JumpStart/set_root_pw` file to create a root password for each cluster node.

Examine the `/etc/shadow` file, and record the encrypted root password (the UNIX™ software encrypted value).

Note – Previously, we set the password on the management server (administrative workstation) to *abc*. Because we want to use the same password across all systems (nodes) that have to be started with the JumpStart software, we need to edit the `set_root_pw` script. The encrypted value of the password (for this lab exercise) is `tby83XuShUxKM`; however, the value of the password will be different for your installation.

Caution – If the password file is not edited to reflect the correct password, the machine will have a password that is unknown. It is possible, though *not* recommended, to set the `root` password to null by deleting all of the characters that follow the `PASSWD=` line in the `set_root_pw` script.

Substep 2—Editing the /JumpStart/set_root_pw File

For local (manual) installations, edit the /JumpStart/set_root_pw file. Modify the PASSWD= line by entering the encrypted password value recorded in Substep 1, from the /etc/shadow/ file. The edited file should look as follows:

```
clustadm# more /JumpStart/set_root_pw
# !/bin/sh
#
#       @(#)set_root_pw 1.6 97/02/26 SMI
#
# This is an example bourne shell script to be run
# after installation. It sets the system's root
# password to the entry defined in PASSWD. The
# encrypted password is obtained from an existing root
# password entry in /etc/shadow from an installed
# machine.
echo "setting password for root"
# set the root password
PASSWD=tby83XuShUxKM {{edit this line with encrypted
password value}}
# create a temporary input file

. . .{{the rest of the file has been omitted}}. . .
```

Step 2.8.4—Verifying the rules and class Files

For local (manual) installations, this final step verifies the rules and class files. A script named check (copied to this directory during a previous step) will be run. This script generates a file named rules.ok, which is read during the JumpStart software installation process.

Execute the /JumpStart/check script and verify the output of the check script, as follows:

```
clustadm# cd /JumpStart
clustadm# ./check
validating rules...
validating profile standard_load_.class...
The custom JumpStart configuration is ok.
```

Step 2.8.5—Verifying the rules.ok File

For local (manual) installations, verify that the rules.ok file contains the correct data. Enter the following command, and confirm that the output looks as follows:

```
clustadm# more rules.ok
hostname clustnode1 - cluster_node.class set_root_pw
hostname clustnode2 - cluster_node.class set_root_pw
# version=2 checksum=5105
```

Note – The checksum and version values may be different for your installation.

Step 2.8.6—Performing the JumpStart Software Installation

For local (manual) installations, perform a JumpStart software installation for each cluster node (for example, clustnode1 and clustnode2). Enter the following command into the Cluster Console Window for each cluster node being installed:

```
ok boot net - install
{{note the spaces before and after the hyphen}}
```

Note – The boot net - install command takes approximately one hour to complete. If error messages appear during this phase, indicating ARP/RARP errors that occur during a boot over the network, the Ethernet address might be incorrect. Before proceeding to the next step, you must troubleshoot and resolve the problem.

Note – You will see Link Down - cable problems? error messages for interfaces that are not connected.

Step 2.8.7—Verifying the Installation

At this point in the installation, verify that each cluster node has been installed successfully. Each node should have been rebooted into multiuser mode, and a login prompt should be displayed.

On each cluster node, log in as superuser (root) from the `cconsole:host` *`hostname`* window for each cluster node.

```
clustnode1 console login: root
Password: abc
```

```
clustnode2 console login: root
Password: abc
```

Note – Prior to using a text editor (such as, `vi`) to view or modify files, you must verify that your terminal environment variable is set to `TERM=vt220` for proper video display.

Key Practice: Review the `/var/sadm/README` file to determine the location of the most recent installation logs (for example, `/var/sadm/system/logs`). Examine the most recently updated log files for potential errors (that is, `begin.log`, `sysidtool.log`, or `install_log`). Confirm the cause of any patch installation error messages that may have occurred. Refer to the *Sun Cluster 3.0 U1 Error Messages Manual* for error message translations (see Appendix B for information on obtaining this manual).

Step 2.8.8—Verifying the Installation Logs

After the Solaris OE installation is completed, examine the current (that is, the most recent) installation log files, and verify that any installation errors do not go undetected or unresolved on each cluster node, as follows:

```
# cd /var/sadm/system/logs
# pwd
/var/sadm/system/logs
# ls -t
sysidtool.log
begin.log
install_log
. . .
. . . {{this output lists the most recently modified files
first}})

# more install_log

. . .
{{It is important to look for installation error messages
in the logs, as in the following example:

pkgadd: ERROR: postinstall script did not complete
successfully}}
```

Note the `/var/sadm/README` file (when navigating the directory structure). Verify correct log files are referenced (that is, the most recent installation logs). For example, the `begin.log` and `finish.log` filenames should have the following form:

`begin.log_`*`YYYY_MM_DD`* and `final.log_`*`YYYY_MM_DD`* (where *`YYYY_MM_DD`* are the install date in year_month_day format)

Step 2.8.9—Verifying the Partitioning

Verify that the primary boot disk was partitioned correctly.

Note – It is easier to use the `format`(1M) command to calculate the exact number of (even) cylinders to be configured, as when determining the size of the root (/) filesystem. Using these guidelines, the size of the root filesystem is dependent on the actual size of the disk.

On each cluster node, verify that the primary boot disk (`c0t0`) is partitioned correctly.

The following example is with 18 Gbyte disk:

```
Slice 0 = cylinders 510 - 7461 assigned to "/" (all
unallocated space; approximately 10GB)

Slice 1 = cylinders 7 - 500 assigned to "swap" (750MB min.)

Slice 2 = cylinders 0 - 7505 assigned as "backup"(full
extent of the disk)

Slice 6 = cylinders 7462 - 7505 assigned to the
"/globaldevices" filesystem (100MB)

Slice 7 = cylinders 1 - 6 assigned to "alternates" for SDS
metadata* (reserve cylinders 1- 6 for use by a volume
manager)

*SDS requires slice 7 for storing metadata; VxVM requires
slices 3 and 4.
```

Caution – The previous example assumes that the boot disk is an 18 Gbyte disk drive with 7506 cylinders. For each configuration, you must ensure that the slicing information matches the actual disk geometry.

The following example is with 36 Gbyte disk:

```
Slice 0 = cylinders 510 - 7461 assigned to "/"
(approximately 10GB)

Slice 1 = cylinders 7 - 500 assigned to "swap" (750MB min.)

Slice 2 = cylinders 0 - 24619 assigned as "backup" (full
extent of the disk)

Slice 6 = cylinders 7462 - 7505 assigned to the
"/globaldevices" filesystem (100MB)

Slice 7 = cylinders 1 - 6 assigned to "alternates" for SDS
metadata* (reserve cylinders 1- 6 for use by a volume
manager)

*SDS requires slice 7 for storing metadata; VxVM requires
slices 3 and 4.
```

Caution – The previous example assumes that the boot disk is a 36 Gbyte disk drive with 24620 cylinders. For each configuration, you must ensure that the slicing information matches the actual disk geometry.

Note – The partition information shown is required to support the Sun Cluster 3.0 software, per the guidelines established in Section 2.1.

Step 2.8.10—Verifying the Hostname Entries

At this time, on each cluster node, verify that the corresponding hostname entries have been created in the `/etc/inet/hosts` file, as required for the SunPlex platform. Also, verify that any additional or site-specific entries have been created, such as the two logical host entries (as required for our two data services).

```
# more /etc/inet/hosts

. . . {{output omitted}}

xxx.xxx.xx.xxx  clustadm loghost
xxx.xxx.xx.xxx  clustnode1
xxx.xxx.xx.xxx  clustnode2
xxx.xxx.xx.xxx  tc nhl-tc

. . . {{output omitted}}

xxx.xxx.xx.xxx  lh-hanfs
xxx.xxx.xx.xxx  lh-apache

. . . {{output omitted}}
```

Note – Example shows standard hostname entries for the cluster. You must verify that your configuration matches the actual (site-specific) installation requirements.

On each node in the SunPlex platform, examine the `/etc/inet/hosts` file, noting that the site-specific (actual) IP addresses and host names are included, as indicated in the codebox. Specifically, on each cluster node, examine this file, and note the entries for `clustnode1`, `clustnode2`, `clustadm` (the administrative workstation), `tc` (the terminal concentrator), and `lh-hanfs` (the first logical host), and `lh-apache` (the second logical host).

Summary of Key Practices

Use the Solaris JumpStart software to maintain consistency and to fully automate the installation of the Solaris OE and additional software packages.

Verify the Solaris OE installation was successful. Prior to installing the Sun Cluster software, or additional software, ensure each node has been installed correctly, as per site-specific requirements.

Section 2.9: Install Recommended Solaris OE Patches on Each Cluster Node

For local (manual) installations, you must ensure that the latest Solaris OE patches are installed on each cluster node, per the following recommendations.

Note – The Solaris OE (and patch) installation is performed only during local (manual) installations. Always verify installation is correct for each cluster node, as described below.

Key Practice: On each cluster node, install the latest Solaris OE recommended patches from the SunSolve℠ program website, as described in this section. Maintaining the latest recommended Solaris OE patches assures that your system has its highest reliability.

Note – The method of obtaining patches through the `http://sunsolve.sun.com` website may deviate from the following procedure. Check with your local Sun service provider on the best method for getting the required patch clusters for your current operating environment.

If you have access to the Internet, go to `http://sunsolve.sun.com`, and click the Patches option on the left-hand column.

Key Practice: The SunSolve program is a contract service from Sun Enterprise Services. It is a good idea to subscribe to this service, especially if you are running a production server.

Key Practice: Create a `/PATCHES` directory on a dedicated server (`clustadm` repository) to store all of the required patches. This enables centralized patch management.

Step 2.9.1—Verifying the Existence of the Patch Directory

For local (manual) installations, change the directory to where the appropriate Solaris OE patch-cluster files are stored. Following these key practices, this next example assumes that the patch repository has been created on `clustadm`:

```
clustadm# cd /JumpStart/PATCHES/SOLARIS_8/8_Recommended
clustadm# ls {{verify existence of patches}}
```

Step 2.9.2—Sharing the Patch Directory

For local (manual) installations, verify that the `PATCHES` directory is shared to each cluster node by entering the following command on the Sun Cluster 3.0 software administration workstation (`clustadm`). If the `/JumpStart` directory is not already shared, repeat steps from Section 2.3 (refer to the `share`(1M) man page).

```
clustadm# share
- /JumpStart ro,anon=0 “”
```

Key Practice: Always refer to the individual patch `README` files to review any install prerequisites before installing patches. Using this practice could possibly prevent conflicts with other patches, software, boot-prom variables, or other unknowns.

Step 2.9.3—Installing the Patches

For local (manual) installations, change the directory to the Solaris OE patches directory, and install all of the required patches on each cluster node. For example, enter the following commands on each cluster node:

```
# mount -F nfs clustadm:/JumpStart/PATCHES/SOLARIS_8/8_Recommended /mnt

# cd /mnt
# ./install_cluster
Are you ready to continue with install? [y/n]: y
Installing <PATCH NUMBER>...
Installing <PATCH NUMBER>...
...
Installing <PATCH NUMBER>...
```

Note – Some patches included in the patch cluster may fail installation with a `Return code 2` error message because they were already included in the Solaris OE base software installation.

Key Practice: Review the installation log file for each patch applied (for example, `/var/sadm/patch/..`) to verify that it was successfully installed. Determine and resolve the cause of any patch installation error messages that occurred. Refer to the *Sun Cluster 3.0 U1 Error Messages Manual* for error message translations (see Appendix B for information on obtaining this manual).

Step 2.9.4—Verifying the Patch Installation

For local (manual) installations, verify that the required patches are installed. For example, to list all of the patches that were applied, enter the following command into both nodes.

```
# patchadd -p | more
```

For specific installation messages, refer to the `/var/sadm/patch/..` directory. Note the `/var/sadm/README` file when navigating the directory structure.

Note – Both the `/usr/sbin/patchadd -p` and `/usr/bin/showrev -p` commands displays a list of patches that have been added to the system.

Use the `/usr/bin/patchadd -p | grep` *`patch_number`* to check individual patches.

```
# patchadd -p | grep patch_number
```

Step 2.9.5—Rebooting the Cluster Nodes

For local (manual) installations, reboot both cluster nodes by performing the next two steps in succession.

First, enter the following command into each cluster node:

```
# shutdown -g0 -y
```

Second, reboot each cluster node by entering the following command at the OpenBoot PROM prompt:

```
ok reset
```

Summary of Key Practices

Always install the latest (required) Solaris OE patches.

Create a `/PATCHES` directory on a dedicated server to store all of the patches. This enables centralized patch management.

Refer to the individual patch `README` files to review any install pre-requisites before installing the patches.

Review the installation (and patch) log files to determine, and resolve, the cause of any software installation error messages.

Section 2.10: The Solaris OE Post Installation and Configuration

After successfully installing the Solaris OE on each cluster node and before you build the Sun Cluster software environment, you must perform the post-installation and site-specific procedures in this section to verify that the operating environment was configured correctly.

Step 2.10.1—Verifying the Date and Time on the Nodes

Before you install additional software, ensure that the date and time are correct on each SunPlex platform node, using the management server or a designated time host.

Note – This is *not* intended to synchronize the date and time on a *running, production* cluster. For additional references, and when configuring NTP in a running cluster, see Appendix B, specifically "How to Update Network Time Protocol (NTP)" in the *Sun Cluster 3.0 (U1) Installation Guide* and "Cluster Time" in *Sun Cluster 3.0 U1 Concepts.*

Key Practice: In preparation for installing additional software, verify (set) the system date and time on each SunPlex platform node. In the future, this can also help ensure that relevant time-stamps (that is, log file entries and time-stamps, error messages, and cluster events) are synchronized, as when correlating a sequence of system events involving multiple nodes.

Note – It is often useful to track events closely and accurately over specific time periods. It is also important when auditing system statistics and logs (for example, for performance monitoring, capacity planning, configuration management, troubleshooting, and ensuring scheduled daily or sustaining operations are completed successfully).

For a running cluster, the date and time on all of the cluster nodes must be synchronized. For this, the SunPlex platform employs the NTP to synchronize the clocks between nodes. In normal cluster operation, you should not need to reset the date and time, unless they were set incorrectly.

At this time, ensure the system date and time is correct on each node. Synchronize the date and time to the management server (`clustadm`) on each cluster node:

```
# ping clustadm
clustadm is alive

# rdate clustadm
```

Note – Verify that the `rdate`(1M) command completes successfully and that the date and time are set correctly on each SunPlex platform node.

Step 2.10.2—Setting the Environment Variables

Use a text editor to set the required environment variables listed in the Appendix A samples. Establish your selected shell environment by configuring each variable indicated.

Note – For example, if not done previously, it is important to set your terminal environment variable correctly (for example, `TERM=vt220`) for proper video display before you use a text editor to view or modify files.

The syntax for the following command is shell specific (in this case, from the korn shell).

```
# TERM=vt220
# export TERM
```

At this time, ensure that the correct settings are configured on each cluster node. Specifically, note the following:

- The `/etc/passwd` file is set to the root shell, as appropriate.
- All of the required variables listed in Appendix A have been configured and exported.

For example, use the `/.profile` or `/.login` file to make the settings listed in Appendix A (sample `root` user startup files) and to establish these settings for the `root` user, as appropriate for your shell environment.

Step 2.10.3—Verifying the Superuser's Shell Environment

On each cluster node, verify the superuser (`root`) shell environment startup files have been established with the following variables:

Table 2-1 Superuser Shell Environment

Variable	Label
`TERM`	`vt220`
`stty`	`istrip`
Prompt	*`hostname`*`#` {{for example, `clustnodeX#`}}
Add the following to the `PATH` variable	`PATH=/usr/bin:/usr/ucb:/etc:/sbin:` `/usr/sbin:/usr/cluster/bin:` `/opt/SUNWcluster/bin`
Add the following to the `MANPATH` variable	`MANPATH=/usr/dt/man:/usr/man:` `/usr/openwin/share/man:` `/usr/cluster/man`

Step 2.10.4—Verifying the Superuser's Group Membership

On each cluster node, verify that the superuser (`root`) is a member of the `sysadmin` group (14) or that the `/.rhosts` file contains a hostname entry for each of the other nodes, as indicated in the next step (sample `/.rhosts` file).

Note – In Module 4, the Solstice DiskSuite software requires this to coordinate activities between nodes.

Step 2.10.5—Verifying the /.rhosts File

On each cluster node, verify that the nine entries in the following code example have been added correctly to the /.rhosts file. Note that the first six entries reference the cluster interconnect and that the final three entries are for the Solstice DiskSuite software.

On the system administration workstation, verify that the final three entries are included in the /.rhosts file on clustadm:

```
# more /.rhosts

204.152.65.33

204.152.65.1

204.152.65.17

204.152.65.34

204.152.65.2

204.152.65.18

{{ the entries below must be included on each node in the
SunPlex platform}}

clustadm

clustnode1

clustnode2
```

Step 2.10.6—Excluding the Cluster Nodes as IP Routers

Ensure that each cluster node will not come up as an IP router. They are not supported on Sun Cluster 3.0 software cluster nodes. If not already done, create the /etc/notrouter file by entering the following command on each cluster node.

```
# touch /etc/notrouter
```

Key Practice: The cluster environment requires that local `/etc` files supporting network services are searched ahead of any naming services. This increases the availability by not relying on an outside agent.

Note – We are not using NIS in the hands-on lab environment. However, it is a good idea to verify the `/etc/nsswitch.conf` file. See Appendix A for an example `nsswitch.conf` file using NIS.

Step 2.10.7—Verifying the NIS Settings

For these hands-on labs, on each cluster node, view the `/etc/nsswitch.conf` file, and verify the following settings:

```
passwd: files
group: files
hosts: files
rpc: files
netmasks: files
services: files
```

Step 2.10.8—Changing the CONSOLE Setting for the Superuser

When set, the CONSOLE setting requires the superuser (root) to log in *only* on the device; however, for these exercises, the SunPlex platform nodes must allow the superuser to log in from other devices.

Note – During Module 6, we reverse this procedure, conforming to best practices, to minimize potential security vulnerabilities.

For these hands-on labs, edit the `/etc/default/login` file, and comment out the following entry (that is, insert the comment character (#) in front of the `CONSOLE=/dev/console` entry).

```
#If console is set, root can only. . .
# Comment. . .
#
# CONSOLE=/dev/console {{comment out this line}}
```

Step 2.10.9—Configuring the Alternate Boot Disk

After ensuring the primary book disk (c0t0) has been configured (as in Step 2.8.9), verify and / or configure the alternate boot disk partitions (`c0t1`) on each cluster node.

For the Sun Cluster 3.0 software, implement the following guidelines when partitioning the primary and alternate boot disks on each cluster node. These guidelines are intended to provide a flexible and consistent format of the system disk that easily enables the use of either the Solstice DiskSuite software or the VxVM software.

- Always configure a mirrored boot environment for each cluster node.
- Reserve cylinders 1 through 6 for use by a volume manager. This requires approximately 10 Mbytes of disk space.
- Configure the `swap` space. This requires a minimum of 750 Mbytes of disk space.
- Configure the `/globaldevices` partitions. This requires a minimum of 100 Mbytes of disk space.
- Allocate all of the *unused* disk space to the root (`/`) directory, slice 0.

For in-depth information on building an HA-boot environment, see Appendix B.

Caution – Refer to Figure 1-1 and Table 1-1 to 1-5 that describe the hardware configuration. Note that both `disk0` and `disk1` are connected to the embedded (`c0`) SCSI controller. This provides only a single datapath to the devices. A single boot path does not support highly available boot configurations.

These partitioning guidelines must be implemented on each cluster node. They must also take into consideration the special requirements for adding a volume manager and the additional space reserved for the Sun Cluster 3.0 software global devices filesystem (`/globaldevices`).

In prepartation for mirroring, ensure that the local disks (that is, `c0t0` and `c0t1`) are partitioned per these guidelines.

Note – Use the `format`(1M) command to calculate the exact number of (even) cylinders to be configured, as when determining the size of the `root` file system. Using these guidelines, the size of the `root` filesystem is dependent on the actual size of the disk.

Using `format`, verify that the primary boot disk is partitioned correctly on each cluster node.

The following is an example of an 18 Gbyte boot disk:

```
Slice 0 = cylinders 510 - 7461 assigned to "/" (all
unallocated space; approximately 10GB)

Slice 1 = cylinders 7 - 500 assigned to "swap" (750MB min.)

Slice 2 = cylinders 0 - 7505 assigned as "backup" (full
extent of the disk)

Slice 6 = cylinders 7462 - 7505 assigned to the
"/globaldevices" filesystem (100MB)

Slice 7 = cylinders 1 - 6 assigned to "alternates" for SDS
metadata* (reserve cylinders 1- 6 for use by a volume
manager)

*The Solstice DiskSuite software requires slice 7 for
storing metadata; VxVM requires slices 3 and 4.
```

Caution – The above example shows an 18 Gbyte disk with 7506 cylinders. For each configuration, you must ensure the slicing information matches the actual disk geometry. After rewriting the new virtual table of contents (VTOC), verify that it was rewritten correctly before proceeding to configure the volume manager.

The following is an example of a 36 Gbyte boot disk:

```
Slice 0 = cylinders 510 - 7461 assigned to “/”
(approximately 10GB)

Slice 1 = cylinders 7 - 500 assigned to “swap” (750MB min.)

Slice 2 = cylinders 0 - 24619 assigned as “backup“ (full
extent of the disk)

Slice 6 = cylinders 7462 - 7505 assigned to the
“/globaldevices” filesystem (100MB)

Slice 7 = cylinders 1 - 6 assigned to “alternates” for SDS
metadata* (reserve cylinders 1-6 for use by a volume
manager)

*SDS requires slice 7 for storing metadata; VxVM requires
slices 3 and 4.
```

Caution – The above example shows an 36 Gbyte disk with 24620 cylinders. For each configuration, you must ensure the slicing information matches the actual disk geometry. After rewriting the new virtual table of contents (VTOC), verify that it was rewritten correctly before proceeding to configure the volume manager.

Note – For the Sun Cluster 3.0 software and site-specific implementations, the configured swap space should be sized based on the actual requirements of the application(s) to be hosted.

Ensure that the alternate boot disk (`c0t1`) is configured (and partitioned) correctly. Use the `format`(1M) command to examine the current settings for *each slice*.

On each cluster node, execute the `format`(1M) command, and verify that the primary boot disk (`c0t0`) is properly partitioned, as previously described in Section 2.8, Step 2.8.9.

Configure the alternate boot disk (`c0t1`) to match the primary boot disk (`c0t0`). Execute the `format`(1M) command, and select the alternate boot disk (`c0t1`), then create the partitions to match the primary boot disk.

Alternatively, we'll present another efficient method for replicating valid VTOC data, quickly (and consistently) for populating multiple disk spindles.

Caution – The `disk0` VTOC (partitioning) should have been verified previously in Step 2.8.9 in ''Section 2.8: Install the Solaris 8 Operating Environment on Each Cluster Node'' section on page Module 2-13.

Note – In this step, the partition information shown is for *typical* disk drives, where `disk0` (`c0t0`) is the primary boot disk and `disk1` (`c0t1`) is the alternate boot disk (mirror). Both physical disks are identical (that is, that are the same model and type) and are formatted identically.

Key Practice: For each cluster node, the Solstice DiskSuite software configuration will create three separate metastate database replicas on three separate disk spindles, further maximizing availability by ensuring that a dual disk failure would have to occur for the Solstice DiskSuite software to be unable to determine a valid state database (that is, for a valid Solstice DiskSuite software quorum, at least 51% of the quorum devices must be available).

Additionally, placing a metastate database replica on *each* of the disk arrays (that is, for a pair of Sun StorEdge D1000 arrays, placing replicas on different arrays and physically on opposite sides of each array) provides an additional measure of redundancy. The replicas can also provide a higher level of availability (for example, having separate data channels).

Example: Replicate VTOC Information

For easy and consistent replication of a *standard* disk drive partitioning (for example, when configuring multiple spindles), consider the following example, which uses the `prtvtoc`(1M) and `fmthard`(1M) commands to replicate a *valid* VTOC.

Caution – Always save a copy of the original VTOC before modifying or rewriting to disk. *Take precautions when using these commands to modify or rewrite disk information.* Also, in the following example, the original VTOC data is saved *temporarily* in the `/tmp` directory (note that the `/tmp/vtoc.orig` file would be erased during a system reboot).

This example assumes that both disk spindles are identical (that is, in type, model, and geometry) and that before reading the existing VTOC from the primary boot disk (c0t0), a new VTOC is written on the alternate boot disk (c0t1).

The following is an example of how to use the prtvtoc(1M) and fmthard(1M) commands:

```
# prtvtoc /dev/rdsk/c0t1d0s2 > /tmp/vtoc.orig
{{saves copy of original disk1 VTOC}}

# prtvtoc /dev/rdsk/c0t0d0s2 > /tmp/vtoc.new
{{creates copy of a "valid" VTOC; here, disk0 was
previously verified and will be replicated}}

# fmthard -s /tmp/vtoc.new /dev/rdsk/c0t1d0s2
{{writes the correct disk0 VTOC to disk1}}

# prtvtoc /dev/rdsk/c0t1d0s2
{{verifies that disk1 VTOC matches the disk0 VTOC}}
```

Note – You can use the OpenBoot PROM commands probe-scsi or probe-ide to determine disk information. Look at both disk drives to ensure they are the same size. If they are not, use the values for the smaller drive.

Caution – Prior to issuing either of these OpenBoot PROM commands, you should perform the reset - all command at the ok prompt to avoid hanging the system, which would require local, manual intervention.

Step 2.10.10—Configuring the Shared Storage

Configure disk partitions for all shared storage (disk spindles). Ensure that each shared disk is configured correctly, according to site-specific requirements.

1. Ensure each shared disk (spindle) is configured and verified before proceeding to the next step.
2. Examine the partition table for *each* shared disk (that is, each Sun StorEdge D1000 array spindle), including: c1t0, c1t1, c1t2, c1t8, c1t9, c1t10, c2t0, c2t1, c2t2, c2t8, c2t9, and c2t10.

3. Verify the configuration follows the following guidelines:

- Slice 0 is approximately 2 Gbytes in size for the shared data.
- Slice 2, the backup, is defined as the full extent of the disk.
- Slice 7, the alternate, Reserve cylinders 1 through 6 (the first six cylinders.

Note – For our hardware configuration, the twelve disks (total) are divided into disk groups for creating the mirrored metavolumes within shared storage. Shared volumes are required for HANFS and the Apache data services.

Key Practice: When partitioning many disks, save time using a standard partitioning scheme. For example, this implementation configures six disk spindles in each array. Array 1 is connected to each cluster node by way of controller `c1`, and array 2 is connected to each cluster node by way of controller `c2`. A *total* of 12 disks are configured. Furthermore, implement a partitioning scheme that is flexible and allows for the use of a couple of volume managers. Partitioning each disk spindle identically can save time and provide additional configuration flexibility.

Summary of Key Practices

Verify system date and time is set correctly, for all production nodes.

Configure `/etc/nsswitch.conf` to search local (`/etc`) files ahead of any naming services. This increases availability by not having to rely on an outside agent.

Always mirror the primary boot disk. Preferably, configure an HA-boot environment (that is, multiple datapaths) on each cluster node.

For the Solstice DiskSuite software configuration, creating three separate metastate database replicas on three separate disk spindles can further maximize availability in the event of a single disk failure.

Easily and consistently replicate a standard VTOC when configuring multiple disk spindles. Use a simple script to create the required partitions when configuring many shared disks.

When partitioning multiple (similar) disk spindles and planning for filesystem layout, save time and reduce the opportunity for errors by using a standard partitioning scheme for multiple disk spindles.

The size of swap space should be based on the actual application requirements, although the Sun Cluster 3.0 software requires a minimum of 750 Mbytes on each cluster node.

Section 2.11: Configure Additional Cluster Management Services

In this section, you will configure additional management services that can help the cluster administrator by making cluster operations more efficient and less error prone.

Note – In future modules, you will be instructed further on how to setup additional Sun Cluster 3.0 software features, as required to complete the Sun Cluster 3.0 software configuration.

Step 2.11.1—Setting Up the syslog(1M) Feature

As per site-specific requirements, we recommend setting up the `syslog`(1M) facility by configuring the `/etc/syslog.conf` file on each cluster node and verifing that the messages are logged properly.

Key Practice: On each cluster node, set up the `syslog`(1M) facility to forward errors and system messages to the management server (administrative workstation). The logged message includes a message header and a message body. The message header consists of a facility indicator, a severity level indicator, a timestamp, a tag string, and optionally the process ID. See the `syslogd`(1M) and `syslog`(3C) man pages for additional information.

Step 2.11.2—Implementing a Repository

Implement a repository on the management server (`clustadm`) for saving a snapshot of site-specific and system configuration files. This includes all modified system files specific to each node. Additionally, for each cluster node, ensure that a valid backup (most recent snapshot) is maintained for each file and that the owner, group, and permissions are preserved correctly.

Key Practice: Simplify cluster administration and management operations by ensuring consistent pathnames within the managed cluster environment. For example, when performing the Sun Cluster 3.0 software installation procedures, you may be required to re-access specific software (applications) or be required to re-enter certain command sequences (steps) repeatedly. This may be necessary if a data entry error occurs or when a given series of *setup* instructions have failed and the subsequent *verification* instructions do not produce the intended results. To further simplify the numerous tasks required to successfully complete the setup of a cluster, implement time-saving practices.

Simplify the numerous "management server-to-cluster node" NFS mounting of directories (and NFS exported filesystems) by creating a mount point (directory name) that can be shared from the management server and accessed remotely by each cluster node using the following syntax:

```
/net/mgmt_server/directory
```

Where *mgmt_server* is `clustadm` and *directory* is the name of the cluster node (for example, `clustnode1`).

This ensures that the path naming to the referenced resources is consistent, regardless of where the following administration functions are being performed.

Execute the following commands on the management server:

```
clustadm# mkdir /saved_files/clustnode1
clustadm# mkdir /saved_files/clustnode1
clustadm# chmod -R 777 /saved_files
```

Key Practice: Decrease administrative overhead of repetitive cluster operations and improve efficiency by implementing a centralized repository for site-specific configuration data files. Whenever any of these site-dependent files are modified, ensure a consistent, valid backup is created on the management sever (the central repository).

When setting up and maintaining the SunPlex platform, certain procedures can become repetitive when configuring multiple cluster nodes, or you may be required to reconstruct the cluster configuration to resolve problems caused by mistakes and operator errors. In theory, almost *any* system file that has to be modified during the configuration process can be saved to the management server (and easily restored, as needed). For each file, ensure backups are valid (copied without errors) and have retained the proper owner, group, and permissions.

Create a repository for the following site-specific files:

- `/.profile`
- `/.cshrc`
- `/.login`
- `/.rhosts`
- `/etc/inet/hosts`
- `/etc/ftpusers`
- `/etc/defaultlogin`
- `/etc/group`
- `/etc/inet/ntp.conf`

Additional examples of site-specific system files modified during these exercises includes, but are not limited to, the following:

- `/etc/ethers`
- `/etc/vfstab`
- `/etc/dfs/dfstab`
- `/etc/defaultrouter`
- `/etc/serialports`
- `/etc/notrouter`
- `/etc/syslog.conf`
- `/kernel/drv/md.conf`
- `/etc/lvm/md.tab`
- VTOC information (for all disks and/or types, the output from `prtvtoc`(1M) is saved)

- Cluster post-installation information (for example, after the installation is successfully completed, the output from the `scconf`(1M), `scstat`(1M), and `scrgadm`(1M) utilities is saved to document the installation and configuration of the cluster)

Consider the following for each node being managed:

- Cluster node host ID
- BIOS information
- OpenBoot PROM settings
- Logical host attributes

Step 2.11.3—Setting Up the /saved_files Directory

On the management server (`clustadm`), create a new entry in the `/etc/dfs/dfstab` file so that the `/saved_files` directory is added and made exportable (that is, shared) with read and write (`rw`) permission for each cluster node. See to Step 2.3.4 and Step 2.3.5 for an example on creating a `dfstab` entry and sharing the directory.

Verify that the entry created in the `/etc/dfs/dfstab` file is correct and that the directory is shared correctly and accessible on each cluster node, before proceeding to the next step.

Step 2.11.4—Copying Site-Specific Configuration Files From clustnode1

Copy all of the site-specific configuration files from `clustnode1` to the `/saved_files/clustnode1` directory on the management server, as in the following example.

```
clustnode1# cd /net/clustadm/saved_files/clustnode1
clustnode1# pwd

/net/clustadm/saved_files/clustnode1
```

Step 2.11.5—Copying Site-Specific Configuration Files From clustnode2

Copy all of the site-specific configuration files from clustnode2 to the /saved_files/clustnode2 directory on the management server.

```
clustnode2# cd /net/clustadm/saved_files/clustnode2
clustnode2# pwd

/net/clustadm/saved_files/clustnode2
```

Step 2.11.6—Creating a Backup of Site-Specific System Configuration Files

Create a backup of all of the site-specific system configuration files. Execute the following commands on each cluster node:

```
clustnode1# cp /.profile .
clustnode1# cp /.cshrc .
clustnode1# cp /.login .
clustnode1# cp /.rhosts .
clustnode1# cp /etc/inet/hosts .
clustnode1# cp /etc/ftpusers .
clustnode1# cp /etc/defaultlogin .
```

Key Practice: When making modifications to the hosts file, always reference the /etc/inet/hosts (source) file because the /etc/hosts file is a link to /etc/inet/hosts.

Step 2.11.7—Verifying the Backup Files

Verify that the files were copied correctly (that is, there is a valid backup), including the correct filenames, owner, group, and permissions. For any system file that gets modified and backed up (and which may need to be restored later, simply reverse the process of copying these files). Care must be taken to know the correct file owner, group, and permission settings for each backup created. Ensure a valid backup was created with the following command:

```
# ls -lisa

{{Verify snapshot was successful, and files were copied
correctly.}}
```

Key Practice: Create an consistent, automated method for ensuring that all of the site-dependent system files are backed up correctly.

A simple `tar`(1) command can make an efficient backup method for each cluster node, saving site-specific files and retaining the correct owner, group, and permissions by executing the following commands:

```
# cd /net/mgmt_server/saved_files/node-specific-directory
# tar cvf ./systemfiles.tar /.profile /.cshrc /.login
/.rhosts /etc/inet/hosts /etc/ftpusers /etc/defaultlogin

{{Verify files are copied to the correct destination
directory, and a valid backup was created.}}
```

Ensure that a valid restoration can be achieved. After verifying the backup is created correctly, ensure that the system files can be restored correctly (that is, a valid restoration) by using the most recent (valid) backup before continuing to build the cluster.

Verify that a complete restoration occurs for all site-specific and system files and that each file retains the correct owner, group, and permissions by executing the following command:

```
clustnode2# cd /net/mgmt_server/saved_files/clustnode2
clustnode2# tar xvfp ./systemfiles.tar /.profile /.cshrc
/.login /.rhosts /etc/inet/hosts /etc/ftpusers
/etc/defaultlogin

{{verify files are copied to the correct destination
directory, along with correct owner, group, and
permissions, and a valid system file has been restored on
clustnode2.}}
```

Note – It is a good idea to re-verify the root login, as part of ensuring a valid restoration of all site-specific and system configuration files because the startup scripts may have been altered. One method is to simply verify that you can remotely log in (`rlogin`) from `clustnode1` to `clustnode2` as the superuser (`root`) (in this example, this can be verified *after* `clustnode2` has been restored and *before* logging off the `clustnode2` console). Verify the superuser login for each node that has been restored, as described.

Summary of Key Practices

On each cluster node, set up the `syslog`(1M) facility to forward errors and system messages to the management server (Sun Cluster 3.0 software administrative workstation).

Simplify cluster administration and repetitive operations by creating a mount point that can be shared from the management server and accessed remotely by each cluster node.

Decrease administrative *overhead* of repetitive cluster operations and improve efficiency by implementing a centralized repository for site-specific configuration data (files).

Always reference the `/etc/inet/hosts` file when making modifications.

Implement a consistent, automated method for saving and restoring site-specific and other system files.

Before logging off the console, verify that the superuser can log in successfully after the system files have been restored from the most recent *backup*.

End of Module Two

Module 2 is now complete. You have successfully performed the following procedures:

1. Completed the administrative workstation (management server) setup.
2. Reviewed the configuration of the JumpStart software services (for example, the Solaris OE and the Sun Cluster 3.0 software, plus all patches).
3. Configured the terminal concentrator.
4. Configured the Cluster Console (CCP) utility.
5. Prepared each cluster node before the Solaris OE installation.
6. Verified that the Solaris OE installation (plus patches) was successful and performed post installation and configuration procedures on each cluster node.

7. Configured `root` workspace on each cluster node.
8. Backed up (saved) all of the site-specific files.

Appendix A: System Configuration Files

This appendix presents the following information:

- Sample `/.profile` and `/.login` startup scripts
- Sample `/etc/inet/hosts` file

Output From `/.profile` and `/.login` Files

The following is an example of the `/.profile` configuration file:

```
#!/bin/ksh
# root ksh environment settings
TERM=vt220
stty=istrip
Prompt='/bin/hostname'
PATH=/usr/bin:/usr/ucb:/etc:/sbin:/usr/sbin:\
/usr/cluster/bin:/opt/SUNWcluster/bin:.
MANPATH=/usr/dt/man:/usr/man:/usr/openwin/share/man:\
/usr/cluster/man:/opt/SUNWcluster/man:/opt/SUNWconn/man:
export TERM stty Prompt PATH MANPATH
umask 022
```

The following is an example of the `/.login` configuration file:

```
# The root csh environment settings
setenv TERM vt220
setenv stty istrip
set prompt = 'bin/hostname'%
setenv PATH /usr/bin:/usr/ucb:/etc:/sbin:/usr/sbin:\
/usr/cluster/bin:/opt/SUNWcluster/bin:.
setenv MANPATH
/usr/dt/man:/usr/man:/usr/openwin/share/man:\
/usr/cluster/man:/opt/SUNWcluster/man:/opt/SUNWconn/man
umask 022
```

Note – The preceding scripts have been written as generic as possible. You are welcome to add to the scripts or not use them at all; however, keep in mind that you will have to troubleshoot any errors the shell scripts create.

The following is an example of the `/.login` configuration file:

```
# more /etc/inet/hosts
# Internet host table
127.0.0.1 localhost
129.153.xx.xxx clustadm loghost
129.153.xx.xxx clustnode1
129.153.xx.xxx clustnode2
129.153.xx.xxx lh-hanfs
129.153.xx.xxx lh-apache
129.153.xx.xxx tc
```

Note – The preceding is an example only. It shows standard hostname entries for the SunPlex platform. Note that the output includes two logical host entries (`lh-hanfs` and `lh-apache`), plus the terminal concentrator (`tc`). Verify that your configuration matches the actual site-specific configuration requirements.

Appendix B: References

This appendix contains a list of URLs and references to online documents and Sun BluePrints articles to review when implementing Sun Cluster software solutions.

On-Line References to Sun Cluster 3.0 Software Documents and Sun BluePrints Publications

Cluster Platform 220/100 Architecture - A Product from SunTone Platforms Portfolio, by Enrique Vargas, can be found by browsing:

`http://www.sun.com/software/solutions/blueprints/browsesubject.html`

Excellent (comprehensive) examples of Sun Cluster software installation checklists (configuration worksheets) can be found by browsing:

`http://sunweb.germany/EIS/Web/inst-support/checklists/cluster30-e.pdf`

Recommend reviewing: Sun Cluster software Site Planning (chapter 3), Site Preparation (chapter 4), and Acceptance Testing (chapter 8), which can be downloaded by browsing:

`http://sunweb.germany/EIS/Web/inst-support/EDocu/server/index.html`

Sun Cluster 3.0 System Administration Guide can be found by browsing for User Documentation at:

`http://suncluster.eng.sun.com/products/SC3.0/`

Sun Cluster 3.0 (U1) Installation Guide can be found by browsing for User Documentation at:

`http://suncluster.eng.sun.com/products/SC3.0/`

Sun Cluster 3.0 (U1) Hardware Guide can be found by browsing for User Documentation at:

`http://suncluster.eng.sun.com/products/SC3.0/`

Sun Cluster 3.0 (U1) Release Notes: Refer to Patches and Firmware requirements; Appendix A, Sun Cluster Installation and Configuration Worksheets which can be found by browsing for User Documentation at:

`http://suncluster.eng.sun.com/products/SC3.0/`

Sun Cluster 3.0 (U1) Concepts can be found by browsing for User Documentation at:

`http://suncluster.eng.sun.com/products/SC3.0/`

Sun Cluster 3.0 (U1) Error Messages Manual can be found by browsing for User Documentation at:

`http://suncluster.eng.sun.com/products/SC3.0/`

Sun Cluster 3.0 (U1) Data Services Installation and Configuration Guide can be found by browsing for User Documentation at:

`http://suncluster.eng.sun.com/products/SC3.0/`

Sun Cluster 3.0 Cluster File System (CFS): Making the most of the global file service, by Tim Read, Senior Consultant, Sun Microsystems UK and many valuable contributions can be found by browsing:

`http://llgweb.uk.sun.com/~timr/`

Sun Web Start Flash Archive (Flash Project) Functional Specification, by Matt Simmons, can be downloaded by browsing:

`http://sps.central/Flash/func_spec.wp.pdf`

Sun WebStart software Flash information can be found by browsing:

`http://www.sun.com/solaris/webstartflash/`

Sun BluePrints Online Articles and valuable writings are available by browsing:

`http://www.sun.com/blueprints`

Availability - What It Means, Why It's Important, and How to Improve It, by Richard McDougal, can be found by browsing:

http://www.sun.com/software/solutions/blueprints/browsesubject.html

Disaster Recovery Requirements, by Stan Stringfellow, can be found by browsing:

http://www.sun.com/software/solutions/blueprints/browsesubject.html

High Availability Best Practices, by Enrique Vargas, can be found by browsing:

http://www.sun.com/software/solutions/blueprints/browsesubject.html

HA: Boot/Root/Swap, by Jeannie Johnstone Kobert, can be found by browsing:

http://www.sun.com/software/solutions/blueprints/browsesubject.html

The Art of Production Environment Engineering, by Bill Walker, can be found by browsing:

http://www.sun.com/software/solutions/blueprints/browsesubject.html

Datacenter Naming Scheme, by Mark Garner, can be found by browsing:

http://www.sun.com/software/solutions/blueprints/browsesubject.html

Robust Clustering: A Comparison of SC3.0 vs. SC 2.2 can be found by browsing:

http://www.sun.com/blueprints/0901/sc30vs22.html

Architecting a Service Provider Infrastructure for Maximum Growth, by Stan Stringfellow, can be found by browsing:

http://www.sun.com/software/solutions/blueprints/browsesubject.html

Building Secure N-Tier Environments, by Alex Noordergraaf, can be found by browsing:

```
http://www.sun.com/software/solutions/blueprints/
browsesubject.html
```

Planning for Large Configurations of Netra t1 Servers, by Stan Stringfellow, can be found by browsing:

```
http://www.sun.com/software/solutions/blueprints/
browsesubject.html
```

Planning to Fail, by John S. Howard, can be found by browsing:

```
http://www.sun.com/software/solutions/blueprints/
browsesubject.html
```

High Availability Fundamentals, by Enrique Vargas, can be found by browsing:

```
http://www.sun.com/software/solutions/blueprints/
browsesubject.html
```

Tales from the Trenches: "The Case of the RAM Starved Cluster", by Richard McDougal, can be found by browsing:

```
http://www.sun.com/software/solutions/blueprints/
browsesubject.html
```

Building a JumpStart Infrastructure, by Alex Noordergraaf, can be found by browsing:

```
http://www.sun.com/software/solutions/blueprints/
browsesubject.html
```

Using NTP to control and Synchronize System Clocks (Parts 1 - 3) can be found by browsing:

```
http://www.sun.com/software/solutions/blueprints/
browsesubject.html
```

Solaris 8 Additions to "sysidcfg", by Rob Snevely, can be found by browsing:

http://www.sun.com/software/solutions/blueprints/browsesubject.html

An Overview of Methodology, by Adrian Cockcroft, can be found by browsing at:

http://www.sun.com/software/solutions/blueprints/browsesubject.html

JumpStart™ Mechanics: Using JumpStart Application for Hands-Free Installation of Unbundled Software parts 1 and 2, by John S. Howard, can be found by browsing:

http://www.sun.com/software/solutions/blueprints/browsesubject.html

Setting Up A Solaris Operating Environment Install Server and the Solaris JumpStart Feature, by Rob Snevely, can be found by browsing:

http://www.sun.com/software/solutions/blueprints/browsesubject.html

Sun GigaBit Ethernet vs. Alteon WebSystems, Inc.: Performance (Throughout) Benchmark Analysis and Datasheet can be found by browsing:

http://www.sun.ca/products-n-solutions/hw/networking/connectivity/sungigabitethernet/gem.html#analysis

Scrubbing Disks using Solaris OE "format" Program can be found by browsing:

http://www.sun.com/software/solutions/blueprints/browsesubject.html

Fast Oracle Exports, by Stan Stringfellow, can be found by browsing:

http://www.sun.com/software/solutions/blueprints/browsesubject.html

IP Network Multipathing, by Mark Garner, can be found by browsing:

http://www.sun.com/software/solutions/blueprints/browsesubject.html

Sun/Oracle Best Practices can be found by browsing:

```
http://www.sun.com/software/solutions/blueprints/
browsesubject.html
```

MR System for Rapid Recovery, by John S. Howard, can be found by browsing:

```
http://www.sun.com/software/solutions/blueprints/
browsesubject.html
```

VxVM Private Regions: Mechanics and Internals of the VxVM Configuration Database, by Gene Trantham, can be found by browsing:

```
http://www.sun.com/software/solutions/blueprints/
browsesubject.html
```

Veritas VXVM Storage Management Software, by Gene Trantham, can be found by browsing:

```
http://www.sun.com/software/solutions/blueprints/
browsesubject.html
```

SSP Best Practices, by John S. Howard, can be found by browsing:

```
http://www.sun.com/software/solutions/blueprints/
browsesubject.html
```