VERITAS™

VERITAS Cluster Server 4.1

Installation Guide

Solaris

N15367F

March 2005

Disclaimer

The information contained in this publication is subject to change without notice. VERITAS Software Corporation makes no warranty of any kind with regard to this manual, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. VERITAS Software Corporation shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this manual.

VERITAS Legal Notice

VERITAS Software Corporation
350 Ellis Street
Mountain View, CA 94043
USA
Phone 650–527–8000 Fax 650–527–2908
www.veritas.com

Third-Party Legal Notices

Apache Software

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work.

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source

code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

Data Encryption Standard (DES)

Support for data encryption in VCS is based on the MIT Data Encryption Standard (DES) under the following copyright:

Copyright © 1990 Dennis Ferguson. All rights reserved.

Commercial use is permitted only if products that are derived from or include this software are made available for purchase and/or use in Canada. Otherwise, redistribution and use in source and binary forms are permitted.

Copyright 1985, 1986, 1987, 1988, 1990 by the Massachusetts Institute of Technology. All rights reserved.

Export of this software from the United States of America may require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of M.I.T. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. M.I.T. makes no representations about the suitability of this software for any purpose. It is provided as is without express or implied warranty.

Sun Microsystems Trademarks

Sun, Solaris, SunOS, Java, Sun Java System Cluster, Sun StorEdge, Solstice DiskSuite, Sun Fire, Sun Enterprise, Online: Backup, and Netra are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries.

All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

SNMP Software

SNMP support in VCS is based on CMU SNMP v2 under the following copyright:

Copyright 1989, 1991, 1992 by Carnegie Mellon University

All Rights Reserved

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of CMU not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

CMU DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL CMU BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Contents

Preface

This guide provides information on how to install VERITAS Cluster Server (VCS) version 4.1 on the Solaris operating system, versions 2.8, 2.9, and 2.10. It is intended for system and network administrators responsible for installing and configuring VCS.

For information on the hardware and software supported by VCS 4.1, and a brief overview of the features of VCS 4.1, see *VERITAS Cluster Server Release Notes*.

How This Guide Is Organized

Chapter 1. "Introduction" on page 1 describes VCS briefly. For a more comprehensive description of VCS, see the *VERITAS Cluster Server User's Guide.*

Chapter 2. "Preparing to Install VCS 4.1" on page 7 describes what you need to do before installing VCS 4.1. It describes supported hardware and software. It describes installing and configuring your hardware, including setting up the private network and configuring shared storage. It outlines the information you need to have on hand when you start installation.

Chapter 3. "Using the VCS Installation Programs" on page 23 describes using an interactive program to install VCS 4.1 on all cluster systems, and describes verifying your installation. It describes starting VCS.

Chapter 4. "Manually Installing and Configuring VCS" on page 75 describes an alternate method of installing VCS in the cluster one system at a time.

Chapter 5. "Verifying the Installation of VCS 4.1" on page 97 describes how to verify the cluster and its communication components LLT and GAB.

Chapter 6. "Setting Up I/O Fencing" on page 111 describes how to set up I/O fencing of shared storage.

Chapter 7. "Manually Upgrading VCS to Release 4.1" on page 143 describes how to upgrade your cluster from earlier versions of VCS.

Chapter 8. "Adding and Removing Cluster Systems" on page 155 describes the necessary commands to use and the configuration files to edit for adding or removing cluster systems.

Chapter 9. "Installing VCS on a Single System" on page 165 describes setting up a single system with VCS 4.1. It also describes adding a system to form a multiple system cluster.

Appendix A. "Advanced Topics Related to Installing VCS" on page 175 presents some advanced topics related to installing VCS.

Appendix B. "Upgrading From VCS QuickStart" on page 191 describes procedure to upgrade to VCS 4.1 from VCS QuickStart.

Conventions

Convention	Usage	Example
`monospace`	Used for path names, commands, output, directory and file names, functions, and parameters.	Read tunables from the `/etc/vx/tunefstab` file. See the `ls`(1) manual page for more information.
`monospace` (**`bold`**)	Indicates user input.	# **`ls pubs`** C:\> **`dir pubs`**
italic	Identifies book titles, new terms, emphasized text, and variables replaced with a name or value.	See the *User's Guide* for details. The variable *system_name* indicates the system on which to enter the command.
bold	Depicts GUI objects, such as fields, list boxes, menu selections, etc. Also depicts GUI commands.	Enter your password in the **Password** field. Press **Return.**
blue text	Indicates hypertext links.	See "Getting Help" on page xviii.
%	C shell prompt	% **`setenv MANPATH /usr/share/man:/opt/VRTS/man`**
$	Bourne/Korn shell prompt	$ **`MANPATH=/usr/share/man: /opt/VRTS/man; export MANPATH`**
#	Unix superuser prompt (all shells).	# **`cp /pubs/4.0/user_book /release_mgnt/4.0/archive`**

Getting Help

For technical assistance, visit http://support.veritas.com and select phone or email support. This site also provides access to resources such as TechNotes, product alerts, software downloads, hardware compatibility lists, and our customer email notification service. Use the Knowledge Base Search feature to access additional product information, including current and past releases of VERITAS documentation.

Diagnostic tools are also available to assist in troubleshooting problems associated with the product. These tools are available on disc or can be downloaded from the VERITAS FTP site. See the `README.VRTSspt` file in the `/support` directory for details.

For license information, software updates and sales contacts, visit http://my.veritas.com/productcenter/ContactVeritas.jsp. For information on purchasing product documentation, visit http://webstore.veritas.com.

Documentation Feedback

Your feedback on product documentation is important to us. Send suggestions for improvements and reports on errors or omissions to clusteringdocs@veritas.com. Include the title and part number of the document (located in the lower left corner of the title page), and chapter and section titles of the text on which you are reporting. Our goal is to ensure customer satisfaction by providing effective, quality documentation. For assistance with topics other than documentation, visit http://support.veritas.com.

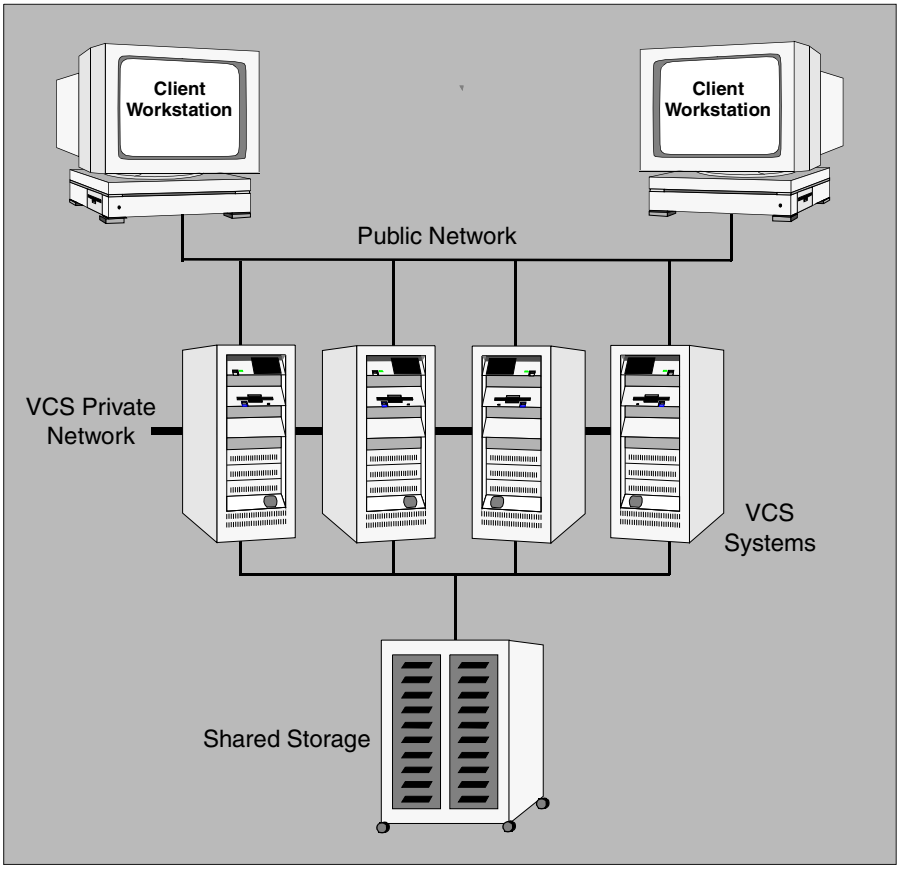
Introduction 1

VERITAS Cluster Server (VCS) is a high-availability solution for cluster configurations. VCS enables you to monitor systems and application services, and to restart services on a different system when hardware or software fails.

VCS Basics

A single VCS cluster consists of multiple systems connected in various combinations to shared storage devices. VCS monitors and controls applications running in the cluster, and restarts applications in response to a variety of hardware or software faults. Client applications continue operation with little or no downtime. In some cases, such as NFS, this continuation is transparent to high-level applications and to users. In other cases, the operation must be retried; for example, a Web page must be reloaded.

The illustration on page 2 shows a typical VCS configuration of four systems (nodes) connected to shared storage. Client workstations receive service over the public network from applications running on the VCS systems. VCS monitors the systems and their services. VCS systems in the cluster communicate over a private network.



Example of a Four-System VCS Cluster

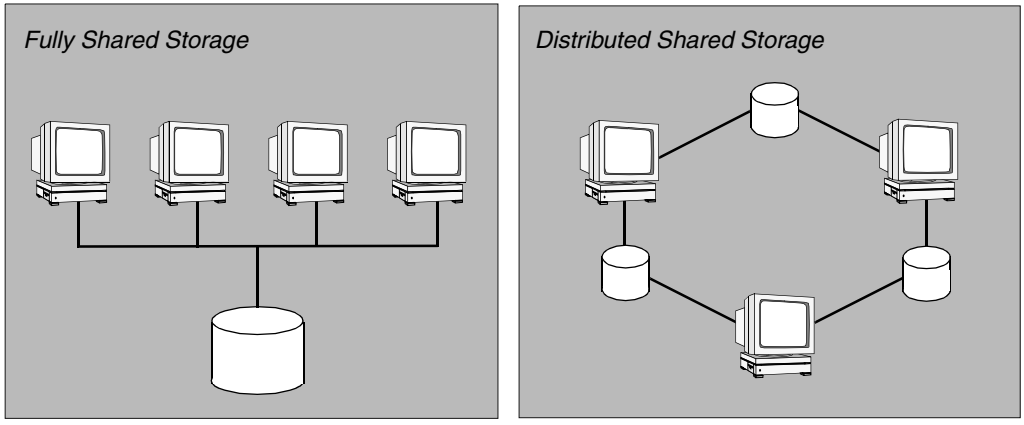
Multiple Systems

VCS runs in a replicated state on each system in the cluster. A private network enables the systems to share identical state information about all resources and to recognize which systems are active, which are joining or leaving the cluster, and which have failed. The private network requires two communication channels to guard against network partitions.

Shared Storage

A VCS hardware configuration typically consists of multiple systems connected to shared storage via I/O channels. Shared storage provides multiple systems an access path to the same data, and enables VCS to restart applications on alternate systems when a system fails, thus ensuring high availability.

The figures below illustrate the flexibility of VCS shared storage configurations. VCS systems can only access storage that is physically attached.



Two Examples of Shared Storage Configurations

LLT and GAB

VCS uses two components, LLT and GAB, to share data over private networks among systems. These components provide the performance and reliability required by VCS.

- LLT (Low Latency Transport) provides fast, kernel-to-kernel communications, and monitors network connections. The system administrator configures LLT by creating the configuration files `/etc/llthosts`, which lists all the systems in the cluster, and `/etc/lltab`, which describes the local system's private network links to the other systems in the cluster.
- GAB (Group Membership and Atomic Broadcast) provides the global message order required to maintain a synchronized state among the systems, and monitors disk communications such as that required by the VCS heartbeat utility. The system administrator configures GAB driver by creating a configuration file (`/etc/gabtab`).

See "Verifying LLT and GAB Configuration Files" on page 97.

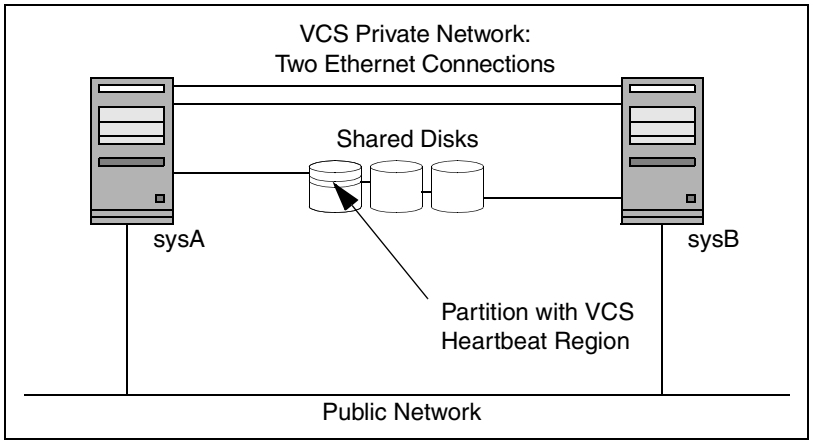
Two Types of Channels: Network and Shared Disks

For the VCS private network, two types of channels are available for heartbeating: network connections and heartbeat regions on shared disks. The shared disk region heartbeat channel is used for heartbeating only, not for transmitting information as are network channels. For information on configuring heartbeat regions on shared disks, see "Configuring Membership Heartbeat Regions on Disk (optional)" on page 86.

Each cluster configuration requires at least two channels between systems, one of which *must* be a network connection. The remaining channels may be a combination of network connections and heartbeat regions on shared disks.

This requirement for two channels protects your cluster against network partitioning. (For more about network partitioning, refer to the *VERITAS Cluster Server User's Guide*.) We recommend configuring at least one heartbeat disk region on each I/O chain shared between systems in addition to private network connections.

The following illustration shows a two-system VCS cluster in which sysA and sysB have two private network connections and another connection via the heartbeat disk region on one of the shared disks. If one of the network connections fails, two channels remain. If both network connections fail, the condition is in jeopardy, but connectivity remains via the heartbeat disk.



Two Systems Connected by Two Ethernet Connections and a Heartbeat Disk Region

Preexisting Network Partitions

A *preexisting network partition* refers to a failure in communication channels that occurs while the systems are down and VCS cannot respond. When the systems are booted, VCS is vulnerable to network partitioning, regardless of the cause of the failure.

VCS Seeding

To protect your cluster from a preexisting network partition, VCS employs the concept of a *seed*. By default, when a system comes up, it is not *seeded*. Systems can be seeded automatically or manually. Note that only systems that have been seeded can run VCS.

Systems are seeded automatically in one of two ways:

- When an unseeded system communicates with a seeded system.
- When all systems in the cluster are unseeded and able to communicate with each other.

VCS requires that you declare the number of systems to participate in the cluster. When the last system starts and joins the cluster, the cluster seeds and starts VCS on all systems. Systems can then be brought down and restarted in any combination. Seeding remains in effect as long as at least one instance of VCS is running somewhere in the cluster. Manual seeding is required only to run VCS from a cold start (all systems down) when one or more systems of the cluster are unavailable. VCS does not start service groups on a system until it is seeded.

Preparing to Install VCS 4.1 2

This chapter describes the basic preparation tasks for setting up a VCS cluster and installing the VCS 4.1 software.

If you are upgrading, and your cluster is already set up, you can run the `installvcs` program. The `installvcs` program detects the presence of your current VCS installation and upgrades VCS to release 4.1. If you cannot use the provided programs to install or upgrade VCS, refer to "Manually Installing and Configuring VCS" on page 75, or "Manually Upgrading VCS to Release 4.1" on page 143.

Preparation Tasks

Perform the following tasks before installing VCS:

- ✔ Review the hardware requirements (see "Hardware Requirements for a VCS Cluster" on page 8)
- ✔ Review the list of supported software (see "Supported Software" on page 9)
- ✔ Set the PATH variable (see "Setting the PATH and MANPATH Variables" on page 9)
- ✔ Set up the private network (see "Setting Up the Private Network" on page 10)
- ✔ Set up the shared storage (see "Setting Up Shared Storage" on page 11)
- ✔ Prepare NFS Services (see "Preparing NFS Services" on page 14)
- ✔ Disable the abort sequence (see "The Abort Sequence on SPARC Systems" on page 15)
- ✔ Enable ssh/rsh communication between systems (see "Enabling Communication Between Systems" on page 16)
- ✔ Obtain VCS license keys (see "Obtaining License Keys for VCS" on page 17)
- ✔ Prepare cluster information (see "Preparing to Use installvcs" on page 18)
- ✔ Install the Root Broker (see "VERITAS Security Services (Optional)" on page 20)

Hardware Requirements for a VCS Cluster

A VCS cluster requires the following hardware:

Item	Description
VCS systems	SPARC systems running Solaris 8 or later.
CD-ROM drive	One CD-ROM drive on each system, or a drive accessible to each.
Disks	Typical VCS configurations require that shared disks support applications that migrate between systems in the cluster. The optional VCS I/O fencing feature requires that all disks used as data disks or as coordinator disks must support SCSI-3 Persistent Reservations (PR). See http://support.veritas.com for information about supported disks. See Chapter 6. "Setting Up I/O Fencing" on page 111 for a description of I/O fencing and how to verify disks support SCSI-3 PR and I/O fencing.
Disk space	Each VCS system requires 550 MB in the `/opt` directory (additionally the language pack requires another 20 MB), 20 MB in `/usr`, 20 MB in `/var`, and 10 MB in `/` for each system.
Ethernet controllers	In addition to the built-in public Ethernet controller, VCS requires at least one more Ethernet interface per system. Two additional interfaces are recommended.
Fibre Channel or SCSI host bus adapters	VCS requires at least one built-in SCSI adapter per system to access the operating system disks, and at least one additional SCSI or Fibre Channel Host Bus Adapter per system for shared data disks.
RAM	Each VCS system requires at least 256 megabytes.

Supported Software

- Solaris 8, 9, and 10 (32-bit and 64-bit) operating systems
- For each platform, we recommend applying the latest cumulative operating system patches available from Sun. See http://sunsolve.sun.com.

Note Within the cluster, all systems must use the same operating system version and patch level.

- VERITAS Volume Manager (VxVM) 3.5 and 4.1
- VERITAS File System (VxFS) 3.5 and 4.1

Note If you plan to use the VCS I/O fencing option, you must use VxVM 4.1 and VxFS 4.1.

Setting the PATH and MANPATH Variables

Locate the installation and other commands in the `/sbin`, `/usr/sbin`, `/opt/VRTS/bin`, and `/opt/VRTSvcs/bin` directories. Add these directories to your PATH environment variable.

▼ To set the PATH variable

If you are using the Bourne Shell (sh or ksh), use:

```
$ PATH=/usr/sbin:/sbin:/opt/VRTS/bin:/opt/VRTSvcs/bin:$PATH;
  export PATH
```

If you are using the C Shell (csh or tcsh), use:

```
% setenv PATH /usr/sbin:/sbin:/opt/VRTS/bin:/opt/VRTSvcs/bin:$PATH
```

▼ To set the MANPATH variable

If you use the Bourne Shell (sh or ksh):

```
$ MANPATH=/usr/share/man:/opt/VRTS/man; export MANPATH
```

If you use the C Shell (csh or tcsh):

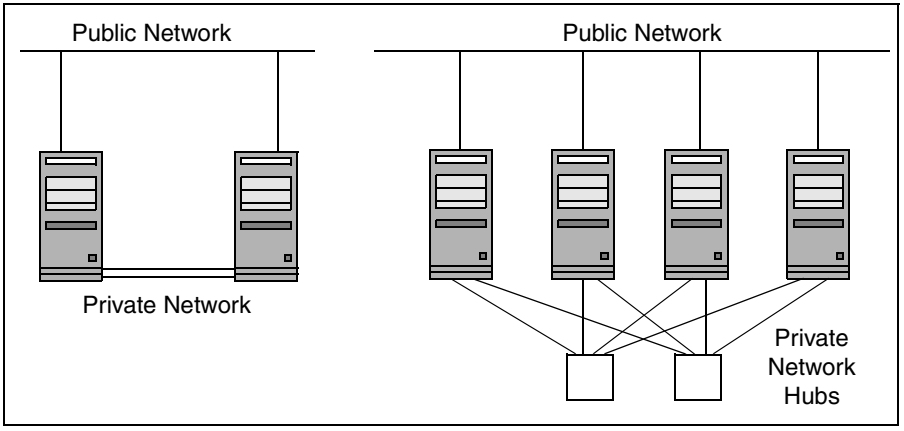
```
% setenv MANPATH /usr/share/man:/opt/VRTS/man
```


Setting Up the Private Network

1. Install the required Ethernet network interface cards.

2. Connect the VCS private Ethernet controllers on each system. Use cross-over Ethernet cables (supported only on two systems), or independent hubs, for each VCS communication network. Ensure hubs are powered from separate sources. On each system, use two independent network cards to provide redundancy.

   During the process of setting up heartbeat connections, note that a chance for data corruption exists if a failure removes all communications between the systems and still leaves the systems running and capable of accessing shared storage.



Private Network Setups: Two-node and Four-node Clusters

3. Configure the Ethernet devices used for the private network such that the auto-negotiation protocol is not used. This helps ensure a more stable configuration with cross-over cables.

   You can do this in one of two ways: by editing the `/etc/system` file to disable auto-negotiation on all Ethernet devices system-wide, or by creating a `qfe.conf` file in the `/kernel/drv` directory to disable auto-negotiation for the individual devices used for private network. Refer to the Sun Ethernet driver product documentation for information on these methods to configure device driver parameters.

4. Test network connections by temporarily assigning network addresses and use `telnet` or `ping` to verify communications.

   LLT uses its own protocol, and does not use TCP/IP. Therefore, to ensure the private network connections are used only for LLT communication and not for TCP/IP traffic, unplumb and unconfigure the temporary addresses after testing.

   The `installvcs` program, described in "Using the VCS Installation Programs" on page 23, configures the private network in the cluster during installation. If you are installing VCS manually, refer to "Manually Installing and Configuring VCS" on page 75 for information about configuring LLT for the private network links.

Using Network Switches

You can use network switches instead of hubs. However, by default, Sun systems assign the same MAC address to all interfaces. Thus, connecting two or more interfaces to a network switch can cause problems. For example, if IP is configured on one interface and LLT on another, and both interfaces are connected to a switch (assuming separate VLANs), the duplicate MAC address on the two switch ports can cause the switch to incorrectly redirect IP traffic to the LLT interface and vice-versa. To avoid this, configure the system to assign unique MAC addresses by setting the `eeprom`(1M) parameter `local-mac-address` to true.

Note Because of their performance characteristics, network switches are recommended for clusters supporting the VERITAS Storage Foundation Cluster File System and VERITAS Storage Foundation for Oracle RAC, which make extensive use of the private cluster interconnects for distributed locking. See the *VERITAS Cluster Server User's Guide* and review the chapter on VCS performance considerations.

Setting Up Shared Storage

The following sections describe setting up SCSI and Fibre Channel devices that are shared among the cluster systems.

If you intend to use VCS I/O fencing, the disks you use for data must support SCSI-3 persistent reservations. In addition, you must configure a coordinator disk group. Coordinator disks must also support SCSI-3 PR. See "Setting Up I/O Fencing" on page 111 for information on verifying SCSI-3 persistent reservation support. See also the *VERITAS Cluster Server User's Guide* for a description of I/O fencing.

Setting Up Shared Storage: SCSI Disks

When SCSI devices are used for storage shared between nodes, the SCSI address, or SCSI initiator ID, of each node must be unique. Since each node typically has the default SCSI address of "7," the addresses of one or more nodes must be changed to avoid a conflict. In the following example, two nodes share SCSI devices. The SCSI address of one node is changed to "5" by using `nvedit` commands to edit the `nvramrc` script.

1. Install the required SCSI host adapters on each node that connects to the storage, and make cable connections to the storage. Refer to the documentation shipped with the host adapters, the storage, and the systems.

2. With both nodes powered off, power on the storage devices.

3. Power on one system, but do not allow it to boot. Halt the system, if necessary, so that you can use the `ok` prompt. (Note that, to avoid address conflicts, it is important that only one system be running at a time.)

4. Find the paths to the host adapters:

   ```
   {0} ok show-disks
   ...b) /sbus@6,0/QLGC,isp@2,10000/sd
   ```

   The example output shows the path to one host adapter. You must include the path information, excluding the "`/sd`" directory, in the `nvramrc` script (see step 5). The path information varies from system to system.

5. Edit the `nvramrc` script on to change the `scsi-initiator-id` to 5. (The *Solaris OpenBoot 3.x Command Reference Manual* contains a full list of `nvedit` commands and keystrokes.) For example:

   ```
   {0} ok nvedit
   ```

   As you edit the script, note the following points:

   - Each line is numbered, 0:, 1:, 2:, and so on, as you enter the `nvedit` commands.
   - On the line where the `scsi-initiator-id` is set, insert exactly one space after the first quotation mark and before `scsi-initiator-id`.

   In this example, edit the nvramrc script as follows:

   ```
   0: probe-all
   1: cd /sbus@6,0/QLGC,isp@2,10000
   2: 5 " scsi-initiator-id" integer-property
   3: device-end
   4: install-console
   5: banner
   6: <CTRL-C>
   ```

6. Store the changes you make to the `nvramrc` script. The changes you make are temporary until you store them.

   ```
   {0} ok nvstore
   ```

   If you are not sure of the changes you made, you can re-edit the script without risk before you store it. You can display the contents of the `nvramrc` script by entering:

   ```
   {0} ok printenv nvramrc
   ```

   You can re-edit the file to make corrections:

   ```
   {0} ok nvedit
   ```

   Or, if necessary, discard the changes by entering:

   ```
   {0} ok nvquit
   ```

7. Instruct the OpenBoot PROM Monitor to use the `nvramrc` script on the node.

   ```
   {0} ok setenv use-nvramrc? true
   ```

8. Reboot the node. Halt the system, if necessary, so that you can use the `ok` prompt.

9. Verify that the `scsi-initiator-id` has changed. Go to the `ok` prompt. Use the output of the show-disks command to find the paths for the host adapters. Then, display the properties for the paths. For example:

   ```
   {0} ok show-disks
   ...b) /sbus@6,0/QLGC,isp@2,10000/sd
   {0} ok cd /sbus@6,0/QLGC,isp@2,10000
   {0} ok .properties
   scsi-initiator-id     00000005
   ```

   Permit the system to continue booting.

10. Boot the second node, halting the system, if necessary, to use the `ok` prompt. Verify that the `scsi-initiator-id` is 7. Use the output of the `show-disks` command to find the paths for the host adapters. Then, display the properties for that paths. For example:

    ```
    {0} ok show-disks
    ...b) /sbus@6,0/QLGC,isp@2,10000/sd
    {0} ok cd /sbus@6,0/QLGC,isp@2,10000
    {0} ok .properties
    scsi-initiator-id     00000007
    ```

    Permit the system to continue booting.

If you have more than two systems sharing the SCSI bus, use the same procedure, making sure that the storage devices have power before any of the systems, and that only one node is running at one time until each node's address is set to a unique value.

Setting Up Shared Storage: Fibre Channel

1. Install the required FC-AL controllers.

2. Connect the FC-AL controllers and the shared storage devices to the same hub or switch. If a fibre switch is being used, be sure that no zoning is implemented which would prevent all systems from seeing all shared devices required to run the critical application.

3. Boot each system with the reconfigure devices option:

   ```
   ok boot -r
   ```

4. Once all systems have booted, use the `format`(1m) command to verify that each system can see all shared devices.

   - If Volume Manager is being used, the same number of external disk devices must appear, but device nodes (c#t#d#s#) may differ.
   - If Volume Manger is not being used, the same number of external disk devices must appear and device nodes must be identical for all devices on all systems.

Preparing NFS Services

Your configuration may include disks on the shared bus that support NFS. File systems exported by NFS can be configured on disk partitions or on VERITAS Volume Manager volumes. An example disk partition name is `/dev/dsk/c1t1d0s3`. An example volume name is `/dev/vx/dsk/shareddg/vol3`. Each name represents the block device on which the file system is to be mounted.

Major and Minor Numbers

Block devices providing NFS service must have the same major and minor numbers on each system. Solaris uses major and minor numbers to identify the logical partition or disk slice. NFS also uses them to identify the exported file system. You must check major and minor numbers to ensure that the NFS identity for the file system is the same when exported from each system.

Checking Major and Minor Numbers

1. Use the following command on all systems exporting an NFS file system. This command displays the major and minor numbers for the block device. For VxVM volumes, you must first import the associated shared disk group on each system.

   ```
   # ls -lL block_device
   ```

   The variable ***block_device*** refers to a partition on which a file system is mounted for export via NFS. Use this command on each NFS file system. For example, type:

   ```
   # ls -lL /dev/dsk/c1t1d0s3
   ```

   Output on System A resembles:

   ```
   crw-r-----  1 root  sys  32,134 Dec 3 11:50 /dev/dsk/c1t1d0s3
   ```

   Output on System B resembles:

   ```
   crw-r-----  1 root  sys  32,134 Dec 3 11:55 /dev/dsk/c1t1d0s3
   ```

   Note that the major numbers, 32, and the minor numbers, 134, match.

2. If either the major or the minor numbers do not match, proceed to install VCS 4.1 and, when installation succeeds, reconcile the major numbers using the `haremajor` command. Refer to "Reconciling Major/Minor Numbers for NFS Shared Disks" on page 175 to reconcile minor numbers that do not match.

The Abort Sequence on SPARC Systems

Most UNIX operating systems provide a method to perform a "break" or "console abort." The inherent problem when you abort a hung system is that it ceases to heartbeat in the cluster. Other cluster members may begin corrective action when they believe that the aborted node is really a failed node.

In order to preserve data integrity and to prevent the cluster from taking additional corrective actions, it is critical that the only action that you perform following an abort is to reset the system in question. Do not resume the processor as cluster membership may have changed and failover actions may already be in progress.

To remove this potential problem on Sun SPARC systems, you should alias the "`go`" function in the OpenBoot eeprom to display a message.

▼ **To alias the go function to display a message**

1. At the `ok` prompt, enter:

   ```
   nvedit
   ```

2. Press Ctrl+L to display the current contents of the nvramrc buffer.

3. Press Ctrl+N until the editor displays the last line of the buffer.

4. Add the following lines exactly as shown. Press Return after adding each line.

   ```
   ." Aliasing the OpenBoot 'go' command! "
   : go ." It is inadvisable to use the 'go' command in a clustered environment. " cr
   ." Please use the 'power-off' or 'reset-all' commands instead. " cr
   ." Thank you, from your friendly neighborhood sysadmin. " ;
   ```

5. Next, press Ctrl+C to exit the nvramrc editor.

6. To verify that no errors exist, type the `nvrun` command. You should see only the following text:

   ```
   Aliasing the OpenBoot 'go' command!
   ```

7. Type the `nvstore` command to commit your changes to the non-volatile RAM (NVRAM) for use in subsequent reboots.

8. After performing these commands, at reboot you see this output:

   ```
   Aliasing the OpenBoot 'go' command! go isn't unique.
   ```

Enabling Communication Between Systems

When you install VCS using the `installvcs` program, to install and configure the entire cluster at one time, make sure that communication between systems exists. By default the installer uses `rsh`. You must grant permissions for the system where you run `installvcs` to issue `ssh` or `rsh` commands as root on all systems in the cluster. If `ssh` is used to communicate between systems, it must be configured in a way such that it operates without requests for passwords or passphrases.

If system communication is not possible between systems using `ssh` or `rsh`, refer to "Using installvcs in a Secure Environment" on page 48 or "Manually Installing VCS" on page 75.

Obtaining License Keys for VCS

VCS is a licensed software product. The `installvcs` program prompts you for a license key for each system. You cannot use your VERITAS software product until you have completed the licensing process. Use either method described in the following two sections to obtain a valid license key.

Using the VERITAS vLicense Web Site to Obtain License Key

You can obtain your license key most efficiently using the VERITAS vLicense web site. The License Key Request Form has all the information needed to establish a User Account on vLicense and generate your license key. The License Key Request Form is a one-page insert included with the disc in your product package. You must have this form to obtain a software license key for your VERITAS product.

Note Do not discard the License Key Request Form. If you have lost or do not have the form for any reason, email license@veritas.com.

The License Key Request Form contains information unique to your VERITAS software purchase. To obtain your software license key, you need the following information shown on the form:

- Your VERITAS customer number
- Your order number
- Your serial number

Follow the appropriate instructions on the vLicense web site to obtain your license key depending on whether you are a new or previous user of vLicense:

1. Access the web site at http://vlicense.veritas.com.
2. Log in or create a new login, as necessary.
3. Follow the instructions on the pages as they are displayed.

When you receive the generated license key, proceed with installation.

Faxing the License Key Request Form to Obtain License Key

If you do not have Internet access, you can fax the License Key Request Form to VERITAS. Be advised that faxing the form generally requires several business days to process in order to provide a license key. Before faxing, sign and date the form in the appropriate spaces. Fax it to the number shown on the form.

VERITAS Licensing Commands

Find the VERITAS licensing commands in the `VRTSvlic` package. You must install `VRTSvlic` for the licensing process to work. The three commands are:

`vxlicinst` Licenses a VERITAS product already installed on a system.

`vxlicrep` Enables you to view currently installed licenses.

`vxlictest` Retrieves features encoded in a license key and describes them.

You can review descriptions and options for these commands in the manual pages installed with the `VRTSvlic` package.

If you encounter problems while licensing your product, visit the VERITAS licensing support website at http://www.veritas.com/buy/vLicense/vLicenseHome.jhtml.

Patches Required for Java Run Time Environment from Sun

The GUI modules for VCS use the Java Run Time Environment from Sun Microsystems. You need to obtain and install the latest Solaris patches to enable the modules to function properly. You can obtain the patches from http://java.sun.com/j2se/1.4.2/download.html.

Preparing to Use installvcs

As you run the `installvcs` program, be prepared to answer prompts so that the installation can proceed smoothly and successfully. Use the following sections to guide you in preparing for the installation of VCS 4.1.

If you wish to install VCS packages on systems, but are not yet ready to configure the VCS cluster, refer to "Using installvcs to Install Without Configuration" on page 55. Later, when you have cluster information available, use the procedures located in "Using installvcs to Configure Without Installation" on page 55.

Required Cluster Information

Be prepared to provide the following information about the cluster and its systems:

- ✔ A name for the cluster; the name must begin with a letter of the alphabet (a-z, A-Z) and contain only the characters a through z, A through Z, and 1 through 0, hyphen (-), and underscore (_).
- ✔ A unique ID number for the cluster. Within the site containing the cluster, each cluster must have a unique ID.
- ✔ The host names of the systems in the cluster.
- ✔ Valid license keys for each system in the cluster, or a valid site or demo license key.
- ✔ Device names of the NICs used by the private networks among systems.

License Key

Be prepared to enter your VCS license key when prompted. See "Obtaining License Keys for VCS" on page 17.

Choosing Optional Packages

The optional packages included with VCS include:

- `VRTSvcsmn`: manual pages for VCS commands
- `VRTSvcsdc`: VCS documentation
- `VRTSvxfen`: I/O fencing
- `VRTSvcssim`: the VCS Simulator
- `VRTScscm`: the VCS Cluster Manager

I/O Fencing (Optional)

I/O fencing protects the data on shared disks. When nodes in a cluster detect a change in cluster membership that could indicate a split brain condition, the fencing operation proceeds to determine which nodes are to retain access to the shared storage and which nodes are to be ejected from the cluster, thus preventing possible data corruption. The *VERITAS Cluster Server User's Guide* describes I/O fencing concepts in detail.

If you select the I/O fencing option, the `installvcs` program installs the VCS I/O fencing driver, `VRTSvxfen`. After completing VCS installation, to use I/O fencing, you must:

- Install a version of VERITAS Volume Manager (VxVM) that supports SCSI-3 persistent reservations.
- Verify the disks you intend to use for shared data storage and for coordinator disks support SCSI-3 PR (Persistent Reservations). How to set up and test the storage and to enable I/O fencing are described in "Setting Up I/O Fencing" on page 111.

VERITAS Security Services (Optional)

VERITAS Security Services (VxSS) secures communication between cluster nodes and clients, including the Java and the Web consoles by using digital certificates for authentication and SSL to encrypt communication over the public network. For more information about VxSS, see the *VERITAS Cluster Server User's Guide*.

If you decide to enable VxSS, you need to:

1. Install the Root Broker.

   The Root Broker is the main registration and certification authority and can serve multiple clusters. VERITAS recommends that you install a single Root Broker on a utility computer such as an email server or domain controller, which can be highly available. To install the Root Broker see, "Installing the Root Broker" on page 27.

2. Configure VxSS during or after installation. To configure it during installation, see "Configuring the Cluster in Secure Mode" on page 37. To configure it after installation, consult the *VERITAS User's Guide*.

Virtual IP Address for Cluster Manager (Web Console)

You have the option to configure the Web-based Cluster Manager (Web Console). The Web Console is a graphical user interface that enables cluster monitoring and administration. If you choose this option, you must provide:

- ✔ The device name for the NIC providing public network access.
- ✔ A virtual IP address associated with the NIC. This virtual IP address becomes a resource for use by the ClusterService group that includes the VCS Cluster Manager (Web Console). The "Cluster Virtual IP address" can fail over to another cluster system, making the Web Console highly available.
- ✔ The subnet used with the virtual address.

Information for Configuring SMTP Notification

You have the option to configure SMTP email notification of VCS events by the VCS Notifier component. If you choose SMTP notification, be ready to answer prompts for the following information:

- ✔ The domain-based address of the SMTP server that is to send notification email about the events within the cluster. For example, `smtp.xyzstar.com`.
- ✔ The email address of each SMTP recipient to be notified. For example, `john@xyzstar.com`.
- ✔ The minimum severity of events for SMTP email notification. Events have four levels of severity: Information, Warning, Error, and SevereError.

The *VERITAS Cluster Server User's Guide* describes SMTP notification in detail; see the chapter on notification.

Information for Configuring SNMP Notification

You have the option to configure SNMP trap notification of VCS events by the VCS Notifier component. If you choose SNMP notification, be ready to answer prompts for the following information:

- ✔ The port number, 162 by default, for the SNMP trap daemon.
- ✔ The machine name for each SNMP console.
- ✔ The minimum severity of events for SNMP trap notification. Events have four levels of severity: Information, Warning, Error, and SevereError.

The *VERITAS Cluster Server User's Guide* describes SNMP notification in detail; see the chapter on notification.

Information for the Global Cluster Option

You have the option to configure the Global Cluster feature. The Global Cluster feature provides the ability to fail over applications between geographically distributed clusters when disaster occurs. The Global Cluster feature requires a license that you can add during the installation.

If you choose the Global Cluster option, the installer allows you to choose whether or not to use the same NIC, virtual IP address, and netmask as are configured for the `ClusterService` group, which are the defaults. If you choose not to use the same networking information, you must specify appropriate values for the NIC, virtual IP address, and netmask when you are prompted.

Using the VCS Installation Programs 3

While you can install VERITAS Cluster Server on clusters of up to 32 systems, the following sections present you with an example installation on two systems: `north` and `south`.

You can install the product two ways:

- The VERITAS product installer (see "Running the VERITAS Installer" on page 29)
- The `installvcs` program (see "Running the installvcs Program" on page 30)

The VERITAS product installer and the installvcs program use `rsh` to install by default. If you prefer to install using `ssh`, see "Using Other installvcs Options" on page 71 and the *Getting Started Guide* for more information.

VCS Installation Program

The `installvcs` program, which you can access from the command line or through the VERITAS product installer, manages the following tasks:

- Licensing VCS
- Installing VCS packages on multiple cluster systems
- Configuring VCS, creating several detailed configuration files on each system
- Starting VCS processes

The `uninstallvcs` program, a companion to `installvcs`, uninstalls VCS packages.

Optional Features of the installvcs Program

The `installvcs` program can also perform the following actions:

- Check the systems to verify they meet the requirements to install VCS.
- Upgrade VCS to version 4.1 if VCS currently runs on a cluster.
- Upgrade cluster systems running GCM 3.5 to VCS 4.1, provided the GCM configuration is standard—created with the aid of GCM configuration wizards. Customized GCM configurations require the assistance of a VERITAS consultant.
- Install VCS packages without configuring VCS, or, configure VCS without installing packages.
- Perform secure or automated installations using values stored in a configuration file.

Using the installvcs Program

The VCS installation program, `installvcs`, is interactive. Using information you supply to its prompts, it installs VCS packages on each cluster system and configures VCS and its communication services. During the installation, you can select the optional I/O fencing feature, the optional security services feature, and optional VCS documentation packages, and choose to configure the optional Web-based Cluster Manager (Web Console), the optional SNMP and SMTP notification features in the cluster, and the optional wide area Global Cluster feature. See "Preparing to Use installvcs" on page 18 for highlights of the information for which `installvcs` prompts you.

Interacting with the installvcs program

As you run the program, you are prompted to answer "yes or no" questions that are typically followed by a set of responses resembling `[y, n, q, ?] (y)`. The response within parentheses is the default, which you can select by pressing Return. By entering the **`?`** character, you can get help to answer the prompt. By entering **`q`**, you can quit the installation.

Note Installation of VCS packages takes place only after you have confirmed the information. However, partially installed VCS files must be removed before running the `installvcs` program again. See "Using uninstallvcs" on page 72.

At some points during the installation, the installer prompts you to type information and expects your responses to be within a certain range or in a specific format. The installer provides examples. If you are prompted to enter an item from a list, enter your selection exactly as it is shown in the list.

When the installer prompts you to answer a series of questions related to a configuration activity, you can enter the **b** character to return to the first prompt in the series. When the installer displays a set of information items you have entered, you are prompted to confirm it. If you answer **n**, the program lets you re-enter all of the information for the set.

The `installvcs` program does *not* configure GAB Disk heartbeat regions. This procedure must be done manually. Refer to "Configuring Membership Heartbeat Regions on Disk (optional)" on page 86.

Upgrading VCS Using the installvcs Program

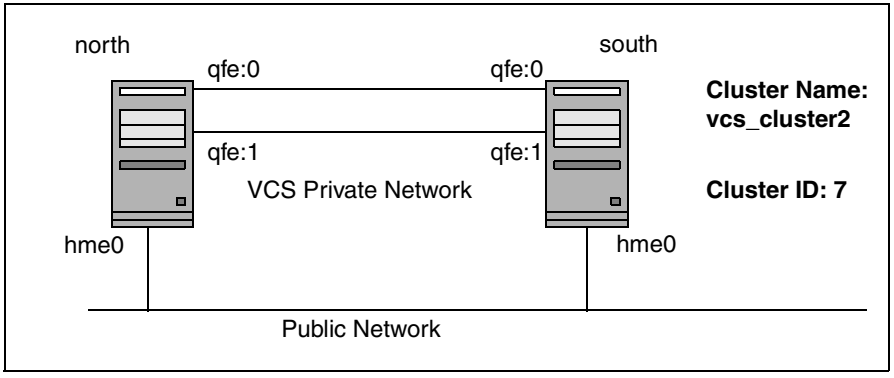
If you are currently running a VCS cluster, including a cluster running GCM 3.5, you can run the installer to upgrade to VCS 4.1. The program detects the current configuration and prompts you to indicate whether you want to upgrade. While the installer is not able to verify the validity of the existing configuration, it is able to run extensive upgrade requirement checks before proceeding.

- If you are running GCM 3.5, review this section on using the `installvcs` program and then see "Upgrading from GCM 3.5 or GCO 4.0, to VCS 4.1 with the Global Cluster Option" on page 62.
- If you are running VCS 3.5 or 4.0, see "Using installvcs to Upgrade to VCS 4.1" on page 55.

Example VCS Installation

In the example installation that follows, all optional features are chosen, including VERITAS Security Services (VxSS), Cluster Manager, SMTP notification, SNMP notification, and Global Cluster option.

The following illustration shows two systems, `north` and `south`, on which VCS is to run. For this example, the cluster's name is `vcs_cluster2` and the cluster's ID is `7`.



An Example of a VCS Installation on a Two-system Cluster

Mounting the Product Disc

1. Log in as root user on a system connected by the network to the systems where you are installing VCS. The system that you are using to install VCS does not need not be part of the cluster.

   Note If you run `installvcs` to upgrade an existing cluster running GCM, mount the disc and run the installer from the GCM master node. See "Upgrading from GCM 3.5 or GCO 4.0, to VCS 4.1 with the Global Cluster Option" on page 62.

2. Insert the software disc with the VCS software into a drive connected to the system. The Solaris volume-management software automatically mounts the disc as `/cdrom/cdrom0`. Type the command:

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

Installing the Root Broker

Install the Root Broker only if you plan on using VERITAS Security Services (VxSS). You must install and configure the Root Broker before you configure VxSS. You can configure VxSS during or after VCS installation.

VERITAS recommends that you install the Root Broker on a stable system that is outside the cluster. See "VERITAS Security Services (Optional)" on page 20 or *VERITAS Cluster Server User's Guide* for more information.

▼ To install the root broker

1. Change to the directory where you can start the `installvcs` program:

   ```
   # cd cluster_server
   ```

2. Start the Root Broker installation program by entering:

   ```
   # ./installvcs -security
   ```

3. The installer presents you with three choices, select installation:
   `[3] Install VERITAS Security Services Root Broker.`

4. When the installation program begins, it starts the VxSS installation program by presenting an informational message:

   ```
   VERITAS AUTHENTICATION SERVICE 4.1 INSTALLATION PROGRAM

   Authentication Service can be installed in three modes, Root Broker
   mode, Authentication Broker mode (AB), or both (R+AB). Typically,
   only one system per domain operates as a Root Broker, which
   validates system credentials for all other Authentication Broker
   systems within the domain.

   installvcs is used to install a system in R+AB mode to serve as the
   Root Broker for Cluster Server systems running in Secure Mode. Use
   the VERITAS Security Services CD to install Authentication Service
   in other modes, on other platforms, or to find VERITAS Security
   Services documentation.
   ```

5. After the installer prompts you to install the Authentication Service in R+AB mode, enter: **y**.

6. Enter the name of the system where you want to install the root broker:

   ```
   Enter the system name on which to install VERITAS Authentication
   Service: east
   ```


7. The installer checks to make sure that the VCS supports the operating system, verifies that you are installing from the global zone, and checks if the system already runs the security package:

```
Checking OS version on east ..... SunOS 5.10
Verifying global zone on east ..... global
Checking VRTSat package ..... not installed

Initial system check completed successfully.
```

8. The installer now checks the system for package and patch information, that sufficient space is available to install the packages, and that none of the processes and drivers related to VCS are currently are currently running.

```
Checking system installation requirements:

Checking VERITAS Authentication Service installation requirements
on east:

Checking VRTSat package ..... not installed
Checking VERITAS patch 117499 ..... not installed
Checking for any Solaris patch issues..... None
Checking file system space ..... required space is available
Checking vxatd process ..... not running

Installation requirement checks completed successfully.

Installing Authentication Service 4.1 on east:

Installing VRTSat 4.1.2.5 on east ..... Done 1 of 2 steps
Adding patch 117499-02 on east ..... Done 2 of 2 steps

Authentication Service installation completed successfully.
```

9. Start the Authentication Server processes:

```
Do you want to start Authentication Service processes now? [y,n,q] y

Authentication Service was started successfully.
Installation of Authentication Service 4.1 has completed
successfully.

The installation summary is saved at:

        /opt/VRTS/install/logs/installvcsdate_time.summary

The installvcs log is saved at:

        /opt/VRTS/install/logs/installvcsdate_time.log
```

Once you have installed the Root Broker, proceed with the next section: "Starting installvcs." In the "Starting installvcs" procedure, answer **y** to step 15 on page 37 and step 16 on page 38.

Running the VERITAS Installer

You have two ways to start VCS installation:

- Using the `installvcs` program directly; skip to "Running the installvcs Program" on page 30, or
- Using the VERITAS product installer program on the software disc. Refer to the next procedure:

▼ To use the product installer

1. Confirm that you are logged in as the root user with the disc mounted at: **`/cdrom/cdrom0`**.

2. Start the installer:

```
# ./installer
```

3. From the opening Selection Menu, choose: "**I**" for "Install/Upgrade a Product."

4. From the displayed list of products to install, choose: **`VERITAS Cluster Server`**.

5. When the installation program begins, it starts the product installation program by presenting a copyright message and prompting you for the names of the systems where you want to install VCS. Skip to step 3.

Running the installvcs Program

With the software disc mounted, you can start the `installvcs` program right away.

If you currently have VCS 3.5 or 4.0 installed on your cluster systems, this program can upgrade the systems to 4.1. See "Using installvcs to Upgrade to VCS 4.1" on page 55 for more information.

Using the installvcs -precheck Option

Before beginning the installation of VCS software, you can check the readiness of the systems where you plan to install VCS. The command to start the pre-installation check is:

```
installvcs -precheck system1 system2 ...
```

For example:

```
# cd /cdrom/cdrom0/cluster_server
# ./installvcs -precheck north south
```

The program proceeds in a non-interactive mode, examining the systems for licenses, packages, disk space, and system-to-system communications. The program displays the results of the check and saves the results of the check in a log file.

Starting installvcs

1. Confirm that you are logged in as root user with the disc mounted at: **`/cdrom/cdrom0/cluster_server`**.

2. Start the `installvcs` program:

   ```
   # ./installvcs
   ```

3. The installer begins with the following introduction:

   ```
           VERITAS CLUSTER SERVER 4.1 INSTALLATION PROGRAM

   Copyright (c) 2003 VERITAS Software Corporation. All rights
   reserved.

   VERITAS, the VERITAS Logo and all other VERITAS product names
   and slogans are trademarks or registered trademarks of VERITAS
   Software Corporation. VERITAS and the VERITAS Logo Reg. U.S.
   Pat. & Tm. Off. Other product names and/or slogans mentioned
   herein may be trademarks or registered trademarks of their
   respective companies.
   ```

4. Enter the names of the systems in the cluster:

```
Enter the system names separated by spaces on which to install
VCS: north south
```

Performing Initial System Checks

5. The installer verifies that the systems you specify use the proper operating system, that they are installing from the global zone, and that they are configured with `ssh` or `rsh` for system-to-system communication. If the installer finds `ssh` binaries, it confirms that `ssh` can operate without requests for passwords or passphrases.

```
Checking OS version on north ..... SunOS 5.10
Verifying global zone on north ..... global
Checking VRTSvcs package ..... not installed
Verifying communication with south ..... ping successful
Attempting rsh with south ..... rsh successful
Attempting rcp with south ..... rcp successful
Checking OS version on south .....SunOS 5.10
Checking VRTSvcs package ..... not installed
Creating log directory on south ..... Done

Logs for installvcs are being created in /var/tmp/
installvcsdate_time.

Using /usr/bin/rsh and /usr/bin/rcp to communicate with remote
systems.

Initial system check completed successfully.
```

Installing the VERITAS Infrastructure Packages

6. After verifying that the infrastructure packages are not installed, the installer verifies they are not already installed and that disk space is available:

```
Installing VERITAS Infrastructure packages on north:
   Checking VRTSvlic package ..... not installed
   Checking VRTScpi package ..... not installed
   Checking file system space ..... required space available
   Installing VRTScpi 4.1.0.54 on north ..... Done
    Installing VRTSvlic 3.02.005h on north .....
Done

Installing VERITAS Infrastructure packages on south:
   Checking VRTSvlic package ..... not installed
   Checking VRTScpi package ..... not installed
```


```
    Checking file system space ..... required space available
    Copying VRTScpi package to south..... Done
    Installing VRTScpi 4.1.0.54 on south ..... Done
    Copying VRTSvlic.tar.gz to south ..... Done
     Installing VRTSvlic 3.02.005h on south .....
Done

 VERITAS Infrastructure packages installed successfully.
```

Verifying VCS Licenses

7. The installer checks for VCS license keys currently in place on each system. You can enter a VCS license and add licenses for additional product features.

```
Each system requires a VCS product license before installation.
License keys for additional product features should also be added
at this time.

Some license keys are node locked and are unique per system. Other
license keys, such as demo keys and site license keys, are
registered on all systems and must be entered on the first system.

VCS Licensing Verification:

Checking VCS license key on north ................. not licensed
Enter a VCS license key for north: [?] XXXX-XXXX-XXXX-XXXX-XXX
Registering XXXX-XXXX-XXXX-XXXX-XXX on north .............. Done
```

Note You can add other licenses at this time.

```
Do you want to enter another license key for north? [y,n,q,?] (n)

Registering XXXX-XXXX-XXXX-XXXX-XXX on south
Checking VCS license key on south .................Cluster Server

Do you want to enter another license key for south? [y,n,q,?] (n)

VCS licensing completed successfully.
```

Choosing Optional Packages Before Adding VCS Packages

8. The installer prompts you to install optional VCS packages. You can select from the optional packages, and see their descriptions.

To use the Java Console with VCS Simulator, you must install the VRTScssim and VRTScscm packages.

```
installvcs can install the following optional VCS packages:


 VRTSvxfen    VERITAS I/O Fencing
 VRTSvcsmn    VERITAS Cluster Server Man Pages
 VRTSvcsdc    VERITAS Cluster Server Documentation
 VRTScssim    VERITAS Cluster Server Simulator
 VRTScscm     VERITAS Cluster Server Cluster Manager

  1)  Install all of the optional packages
  2)  Install none of the optional packages
  3)  View package description and select optional packages

Select the optional packages to be installed on all systems?
[1-3,q,?]  (1)
```

9. After you choose whether to install optional packages, the installer lists all of the packages it installs:

```
installvcs will install the following VCS packages:

VRTSperl     VERITAS Perl 5.8.0 Redistribution
VRTSat       VERITAS Authentication Service
VRTSllt      VERITAS Low Latency Transport
VRTSgab      VERITAS Group Membership and Atomic Broadcast
VRTSvxfen    VERITAS I/O Fencing
VRTSvcs      VERITAS Cluster Server
VRTSvcsmg    VERITAS Cluster Server Message Catalogs
VRTSvcsag    VERITAS Cluster Server Bundled Agents
VRTSvcsmn    VERITAS Cluster Server Man Pages
VRTSvcsdc    VERITAS Cluster Server Documentation
VRTSjre      VERITAS Java Runtime Environment Redistribution
VRTScutil    VERITAS Cluster Utilities
VRTScssim    VERITAS Cluster Server Simulator
VRTScscw     VERITAS Cluster Server Configuration Wizards
VRTSweb      VERITAS Java Web Server
VRTSvcsw     VERITAS Cluster Manager (Web Console)
VRTScscm     VERITAS Cluster Server Cluster Manager
```


10. The installer checks both systems to make sure none of the packages are already installed, that sufficient space is available to install the packages, and that none of the processes and drivers related to VCS are currently are currently running.

```
Checking VCS installation requirements on north:

   Checking VRTSperl package ..................... not installed
   Checking VRTSat package ....................... not installed
   Checking VRTSllt package ...................... not installed
   Checking VRTSgab package ...................... not installed
   Checking VRTSvxfen package .................... not installed
   Checking VRTSvcs package ...................... not installed
   Checking VRTSvcsmg package .................... not installed
   Checking VRTSvcsag package .................... not installed
   Checking VRTSvcsmn package .................... not installed
   Checking VRTSvcsdc package .................... not installed
   Checking VRTSjre package ...................... not installed
   Checking VRTScutil package .................... not installed
   Checking VRTScssim package .................... not installed
   Checking VRTScscw package ..................... not installed
   Checking VRTSweb package ...................... not installed
   Checking VRTSvcsw package ..................... not installed
   Checking VRTScscm package ..................... not installed
   Checking VERITAS patch 117499.................. not installed
   Checking for any Solaris patch issues ................... None
   Checking file system space ...... required space is available
   Checking VRTSweb process ........................ not running
   Checking had process ............................ not running
   Checking hashadow process ....................... not running
   Checking CmdServer process ...................... not running
   Checking notifier process ....................... not running
   Checking vxatd process .......................... not running
   Checking vxfen driver ........................... not running
   Checking gab driver ............................. not running
   Checking llt driver ............................. not running
```

The same checks are made on `south` and the following message displays:

```
Installation requirement checks completed successfully.
```

In some cases, you might find packages already installed on a system. If the current version of a package is on a system, the installer removes it from the package installation list for the system. If a previous version of a package is installed, it is removed and the current version is installed.

Configuring the Cluster

11. The installer describes the VCS options that you have selected. While you must configure VCS before it can be used, you can choose to install and configure VCS now, or to install packages on the systems and leave the cluster configuration steps for later. See "Using installvcs to Configure Without Installation" on page 55.

```
It is optional to configure VCS now. If you choose to
configure VCS later, you can either do so manually or run the
installvcs -configure command. Are you ready to configure VCS?
[y,n,q] (y) y
```

12. The installer lists the information it requires to configure a VCS cluster:

```
To configure VCS the following is required:

  A unique Cluster name
  A unique Cluster ID number between 0-255
  Two or more NIC cards per system used for heartbeat links

  One or more heartbeat links are configured as private links
  One heartbeat link may be configured as a low priority link

All systems are being configured to create one cluster

Enter the unique cluster name: [?] vcs_cluster2
Enter the unique Cluster ID number between 0-255: [b,?] 7
```


13. The installer discovers the NICs available on the first system and reports them:

```
Discovering NICs on north ...discovered hme0 qfe0 qfe1 qfe2 qfe3
```

The installer presents questions about configuring the discovered heartbeat NICS:

```
Enter the NIC for the first private heartbeat NIC on north:
[b,?] qfe0
Would you like to configure a second private heartbeat link?
[y,n,q,b,?] (y)
Enter the NIC for the second private heartbeat NIC on north:
[b,?] qfe1
Would you like to configure a third private heartbeat link?
[y,n,q,b,?](n)
Do you want to configure an additional low priority heartbeat
link? [y,n,q,b,?] (n)
Are you using the same NICs for private heartbeat links on all
systems? [y,n,q,b,?] (y)
```

Note When answering **y** be sure that the same NICs are available on each system; the installer does not verify this.

Notice that in this example, `hme0` is not selected for use as a private heartbeat NIC because it already in use as the public network interface. The default responses are chosen.

14. The installer summarizes the information and prompts you to confirm that it is correct:

```
Cluster information verification:

   Cluster Name: vcs_cluster2
   Cluster ID Number: 7
   Private Heartbeat NICs for north: link1=qfe0 link2=qfe1
   Private Heartbeat NICs for south: link1=qfe0 link2=qfe1

Is this information correct? [y,n,q] (y)
```

- If the information is *not* correct, answer **n**. The installer prompts you to enter the information again.
- If the information is correct, press Return.

Configuring the Cluster in Secure Mode

15. The installer begins with the following introduction, and asks if you want to proceed to install VERITAS Security Services (VxSS):

```
Cluster Server can be configured to utilize VERITAS Security
Services.

Running VCS in Secure Mode guarantees that all inter-system
communication is encypted and that users are verified with security
credentials.

When running VCS in Secure Mode, NIS and system usernames and
passwords are used to verify identity. VCS usernames and  passwords
are no longer utilized when a cluster is running in Secure Mode.

Before configuring a cluster to operate using VERITAS Security
Services, another system must already have VERITAS Security
Services installed and be operating as a Root Broker. Refer to the
Cluster Server Installation and Configuration Guide for more
information on configuring a VxSS Root Broker.

Would you like to configure VCS to use VERITAS Security Serivces?
[y,n,q] (n)
```

- If you want to configure VxSS, make sure that you have installed the root broker (see "Installing the Root Broker" on page 27), and answer **y**.
- If you do not want to configure VxSS, press Return.

16. The installer now checks for installed credentials and packages on the cluster.

    - If you see a message similar to the following, proceed to step 17.

      ```
      Checking VERITAS Security Services on system north:

         Checking VRTSat package ..................... not installed

      Checking VERITAS Security Services on system south:

         Checking VRTSat package ..................... not installed
      ```

    - If you see a message similar to the following, VERITAS Security Services are already installed, skip to step 19.

      ```
      Checking VERITAS Security Services credentials

      Checking VERITAS Security Services on system north:

         Checking VRTSat package .......... version 4.1.2.5 installed
         Checking root credential ............................ None

      Checking VERITAS Security Services on system south:

         Checking VRTSat package .......... version 4.1.2.5 installed
         Checking root credential ............. root@east.xyzstar.com

      Systems have credentials from root@east.xyzstar.com
      Using root@east.xyzstar.com as root broker for other
      cluster systems
      ```

17. It then informs you that you must establish the Root Broker, and asks for the Root Broker's name:

    ```
    In order to Enable VERITAS Security Services on a VCS Cluster,
    VERITAS Authorization Services (VRTSat package) must be installed
    on a system and operating as a Root Broker. Refer to the VCS
    Installation and Configuration Guide for more information on
    installing and configuring VERITAS Authorization Services.

    Enter the name of the VxSS Root Broker system: east

       Checking vxatd process ............................. running
       Checking vxatd version ............................. 4.1.2.5

    Systems will use root@east.xyzstar.com as its VxSS Domain.
    ```

If VERITAS Security Services are already installed, you should see output similar to:

```
Configuring Cluster Server:

  Creating north security principal on east ............... Done
  Starting VERITAS Security Services on north .............. Done
  Creating south security principal on east ............... Done
  Starting VERITAS Security Services on south .............. Done
  Creating Cluster Server configuration files .............. Done
  Copying configuration files to north ..................... Done
  Copying configuration files to south ..................... Done

Cluster Server configured successfully.
```

Adding VCS Users

18. If you have enabled VxSS, you do not need to add VCS users; skip to step 19. Otherwise, on systems operating under an English locale, you can add VCS users at this time. For each user you want to add, the installer prompts you for the user's name, password, and level of privileges. You also have the opportunity to reset the password for the Admin user.

```
The following information is required to add VCS users:

    A user name
    A password for the user
    User privileges (Administrator, Operator, or Guest)

Do you want to set the password for the Admin user
(default password='password')? [y,n,q] (n) y
Enter New Password:******

Enter Again:******
Do you want to add another user to the cluster? [y,n,q] (y)

Enter the user name: [?] smith
Enter New Password:*******

Enter Again:*******
Enter the privilege for user smith (A=Administrator, O=Operator,
 G=Guest): [?] a

Would you like to add another user? [y,n,q] (n)

       User: admin     Privilege: Administrators
       User: smith     Privilege: Administrators

       Passwords are not displayed

Is this information correct? [y,n,q] (y)
```


Configuring Cluster Manager

19. The installer describes information required to configure Cluster Manager:

```
The following information is required to configure Cluster
Manager:

  A public NIC used by each system in the cluster
  A Virtual IP address and netmask for Cluster Manager

 Do you want to configure Cluster Manager (Web Console)
 [y,n,q](Y)
```

Press Return to configure Cluster Manager (Web Console) on the systems. Enter **n** to skip configuring Cluster Manager and advance to configure SMTP notification.

20. Confirm whether you want to use the discovered public NIC on the first system.

```
Active NIC devices discovered on north: hme0
Enter the NIC for Cluster Manager (Web Console) to use on north:
[b,?](hme0)
```

Press Return if the discovered NIC is the one to use. Otherwise, type the name of a NIC to use and press Return.

```
Is hme0 to be the public NIC used by all systems [y,n,q,b,?] (y)
```

Press Return if all systems use the same public NIC. You are prompted to enter a NIC for each system if unique NICs are used.

21. Enter the virtual IP address that the Cluster Manager uses:

```
Enter the Virtual IP address for Cluster Manager: [b,?]
11.136.88.199
```

22. Confirm the default netmask or enter another one:

```
Enter the netmask for IP 11.136.88.199: [b,?] (255.255.240.0)
```

23. The installer prompts you to verify Cluster Manager information:

```
Cluster Manager (Web Console) verification:

      NIC: hme0
      IP: 11.136.88.199
      Netmask: 255.255.240.0

Is this information correct? [y,n,q] (y)
```

- If the information is *not* correct, answer **n**. The installer prompts you to enter the information again.
- If the information is correct, press Return.

Configuring SMTP Email Notification

24. The installation program describes the information required to configure the SMTP notification feature of VCS:

```
The following information is required to configure SMTP
notification:

   The domain-based hostname of the SMTP server
   The email address of each SMTP recipient
   A minimum severity level of messages to send to each recipient

Do you want to configure SMTP notification? [y,n,q] (y) y
```

You can enter **n** and skip configuring SMTP notification. The program advances you to the screen enabling you to configure SNMP notification (see step 27).

25. Respond to the prompts and provide information to configure SMTP notification.

```
Enter the domain-based hostname of the SMTP server
  (example: smtp.yourcompany.com): [b,?] smtp.xyzstar.com
Enter the full email address of the SMTP recipient
  (example: user@yourcompany.com): [b,?] ozzie@xyzstar.com
Enter the minimum severity of events for which mail should be
  sent to ozzie@xyzstar.com  [I=Information, W=Warning,
  E=Error, S=SevereError]: [b,?] w
Would you like to add another SMTP recipient? [y,n,q,b] (n) y
Enter the full email address of the SMTP recipient
  (example: user@yourcompany.com): [b,?] harriet@xyzstar.com
Enter the minimum severity of events for which mail should be
  sent to harriet@xyzstar.com  [I=Information, W=Warning,
  E=Error, S=SevereError]: [b,?] E
Would you like to add another SMTP recipient? [y,n,q,b] (n)
```

26. The installer prompts you to verify the SMTP notification information:

```
SMTP Address: smtp.xyzstar.com
Recipient: ozzie@xyzstar.com receives email for Warning or
   higher events
Recipient: harriet@xyzstar.com receives email for Error or
   higher events

Is this information correct? [y,n,q] (y)
```

- If the information is *not* correct, answer **n**. The installer prompts you to enter the information again.
- If the information is correct, press Return.

Configuring SNMP Trap Notification

27. The installation program describes the information required to configure the SNMP notification feature of VCS:

```
System names of SNMP consoles to receive VCS trap messages
SNMP trap daemon port numbers for each console
A minimum severity level of messages to send to each console

Do you want to configure SNMP notification? [y,n,q] (y)
```

You can enter **n** and skip configuring SNMP notification. The program advances you to the screen enabling you to configure the Global Cluster option.

28. Respond to the prompts and provide information to configure SNMP trap notification:

```
Enter the SNMP trap daemon port: [b,?] (162)
Enter the SNMP console system name: [b,?] saturn
Enter the minimum severity of events for which SNMP traps should
  be sent to saturn [I=Information, W=Warning, E=Error,
  S=SevereError]: [b,?] E
Would you like to add another SNMP console? [y,n,q,b] (n) y
Enter the SNMP console system name: [b,?] jupiter
Enter the minimum severity of events for which SNMP traps should
  be sent to jupiter [I=Information, W=Warning, E=Error,
  S=SevereError]: [b,?] S
Would you like to add another SNMP console? [y,n,q,b] (n)
```

29. The installer prompts you to verify the SNMP trap notification information:

```
SNMP Port: 162
Console: saturn receives SNMP traps for Error or higher events
Console: jupiter receives SNMP traps for SevereError or higher
   events

Is this information correct? [y,n,q] (y)
```

- If the information is *not* correct, answer **n**. The installer prompts you to enter the information again.
- If the information is correct, press Return.

Configuring the Global Cluster Option

30. The installation program describes the information required to configure the Global Cluster option of VCS:

```
The following is required to configure the Global Cluster Option:

  A public NIC used by each system in the cluster
  A Virtual IP address and netmask

The Virtual IP address and NIC may be the same as those
configured for Cluster Manager (Web Console)

Do you want to configure the Global Cluster Option? [y,n,q]
  (y)
```

You can enter **n** and skip configuring the Global Cluster Option. The installation program starts installation of the packages; see step 33.

31. Respond to the prompts and provide information to configure the Global Cluster option. As the prompts suggest, you can use the same virtual IP address and netmask used by Cluster Manager:

```
Enter the Virtual IP address for Global Cluster Manager: [b,?]
  (11.136.88.199)
```

Press return to accept the default, which is the virtual IP address, NIC, and netmask used by Cluster Manager (see step 23). If you enter another IP address, the installer prompts you for a NIC and value for the netmask.

32. The installer prompts you to verify the configuration of the Global Cluster option:

```
Global Cluster Option configuration verification:

        NIC: hme0
        IP: 11.136.88.199
        Netmask: 255.255.240.0

Matching Cluster Manager (Web Console) Virtual IP configuration

Is this information correct? [y,n,q] (y)
```

- If the information is *not* correct, answer **n**. The installer prompts you to enter the information again.
- If the information is correct, press Return.

Installing the VCS Packages

33. After you have verified that the information for the Global Cluster option you have entered is correct, the installation program begins installing by prompting you to indicate whether you want to install the packages consecutively or simultaneously.

```
VCS packages can be installed on systems consecutively or
simultaneously. Installing packages on systems consecutively
takes more time but allows for better error handling.
```

By default, installation occurs on systems simultaneously.

```
Would you like to install Cluster Server packages on all
systems simultaneously? [y,n,q,?] (y) y
 Installing Cluster Server 4.1 on all systems simultaneously:

Copying VRTSperl.tar.gz to south ..... Done 1 of 54 steps
Installing VRTSperl 4.0.12 on north ..... Done 2 of 54 steps
Installing VRTSperl 4.0.12 on south ..... Done 3 of 54 steps
Copying VRTSat.tar.gz to south ..... Done 4 of 54 steps
Installing VRTSat 4.1.2.5 on north ..... Done 5 of 54 steps
Installing VRTSllt 4.1 on north ..... Done 6 of 54 steps
Installing VRTSgab 4.1 on north ..... Done 7 of 54 steps
Installing VRTSat 4.1.2.5 on south ..... Done 8 of 54 steps
Copying VRTSllt.tar.gz to south ..... Done 9 of 54 steps
Installing VRTSvxfen 4.1 on north ..... Done 10 of 54 steps
Installing VRTSllt 4.1 on south ..... Done 11 of 54 steps
Copying VRTSgab.tar.gz to south ..... Done 12 of 54 steps
Installing VRTSvcs 4.1 on north ..... Done 13 of 54 steps
Installing VRTSgab 4.1 on south ..... Done 14 of 54 steps
Installing VRTSvcsmg 4.1 on north ..... Done 15 of 54 steps
Copying VRTSvxfen.tar.gz to south ..... Done 16 of 54 steps
Installing VRTSvcsag 4.1 on north ..... Done 17 of 54 steps
Installing VRTSvcsmn 4.1 on north ..... Done 18 of 54 steps
Installing VRTSvcsdc 4.1 on north ..... Done 19 of 54 steps
Installing VRTSvxfen 4.1 on south ..... Done 20 of 54 steps
Copying VRTSvcs.tar.gz to south ..... Done 21 of 54 steps
Installing VRTSvcs 4.1 on south ..... Done 22 of 54 steps
Copying VRTSvcsmg.tar.gz to south ..... Done 23 of 54 steps
Installing VRTSjre 1.4 on north ..... Done 24 of 54 steps
Installing VRTScutil 4.1 on north ..... Done 25 of 54 steps
Installing VRTSvcsmg 4.1 on south ..... Done 26 of 54 steps
Copying VRTSvcsag.tar.gz to south ..... Done 27 of 54 steps
Installing VRTScssim 4.1 on north ..... Done 28 of 54 steps
Installing VRTScscw 4.1 on north ..... Done 29 of 54 steps
Installing VRTSweb 4.2 on north ..... Done 30 of 54 steps
Installing VRTSvcsag 4.1 on south ..... Done 31 of 54 steps
Installing VRTSvcsw 4.3 on north ..... Done 32 of 54 steps
```

```
Copying VRTSvcsmn.tar.gz to south .......... Done 33 of 54 steps
Installing VRTScscm 4.3 on north ........... Done 34 of 54 steps
Installing VRTSvcsmn 4.1 on south .......... Done 35 of 54 steps
Copying VRTSvcsdc.tar.gz to south .......... Done 36 of 54 steps
Installing VRTSvcsdc 4.1 on south .......... Done 37 of 54 steps
Copying VRTSjre.tar.gz to south ............ Done 38 of 54 steps
Installing VRTSjre 1.4 on south ............ Done 39 of 54 steps
Copying VRTScutil.tar.gz to south .......... Done 40 of 54 steps
Installing VRTScutil 4.1 on south .......... Done 41 of 54 steps
Copying VRTScssim.tar.gz to south .......... Done 42 of 54 steps
Installing VRTScssim 4.1 on south .......... Done 43 of 54 steps
Copying VRTScscw.tar.gz to south ........... Done 44 of 54 steps
Installing VRTScscw 4.1 on south ........... Done 45 of 54 steps
Copying VRTSweb.tar.gz to south ............ Done 46 of 54 steps
Adding patch 117499-02 on north ............ Done 47 of 54 steps
Installing VRTSweb 4.2 on south ............ Done 48 of 54 steps
Copying VRTSvcsw.tar.gz to south ........... Done 49 of 54 steps
Installing VRTSvcsw 4.3 on south ........... Done 50 of 54 steps
Copying VRTScscm.tar.gz to south ........... Done 51 of 54 steps
Installing VRTScscm 4.3 on south ........... Done 52 of 54 steps
Copying 117499-02.tar.gz to south .......... Done 53 of 54 steps
Adding patch 117499-02 on south ............ Done 54 of 54 steps

  Cluster Server installation completed successfully.

  Press [Return] to continue:
```

Creating VCS Configuration Files

34. The installation program continues by creating configuration files and copying them to each system:

```
  Creating Cluster Server configuration files ............ Done
  Copying configuration files to north................... Done
  Copying configuration files to south................... Done

Cluster Server configured successfully.
```


Starting VCS

35. You can now start VCS and its components on each system:

```
Do you want to start Cluster Server processes now? [y,n,q] (y)

Starting Cluster Server:

  Starting LLT on north ............................. Started
  Starting LLT on south ............................. Started
  Starting GAB on north ............................. Started
  Starting GAB on south ............................. Started
  Starting Cluster Server on north ................... Started
  Starting Cluster Server on south ................... Started
  Confirming Cluster Server startup ........ 2 systems RUNNING

Cluster Server was started successfully.

Press [Return] to continue:
```

36. When Cluster Server 4.1 installation completes successfully, the installation program displays the following messages:

```
Installation of Cluster Server 4.1 has completed successfully.

  The installation summary is saved at:

    /opt/VRTS/install/logs/installvcsdate_time.summary

  The installvcs log is saved at:

    /opt/VRTS/install/logs/installvcsdate_time.log

  The installation response file is saved at:

    /opt/VRTS/install/logs/installvcsdate_time.response
```

These files provide useful information that can assist you with this and future installations:

- The “summary” file lists packages installed on each system, describes the cluster and its configured resources, and provides information for managing the cluster.
- The “log” file details the entire installation.
- The “response” file contains configuration information that can be used to perform secure or unattended installations on other systems (see “Example Response File” on page 51).

Verifying the Cluster After Installation

When you have used `installvcs` and chosen to configure and start VCS, it is expected that VCS and all components are properly configured and can start correctly. To verify that your cluster is operating properly following installation, review "Verifying the Installation of VCS 4.1" on page 97.

Copying the Installation Guide to Each System

After you install VCS, VERITAS recommends that you copy the PDF version of this guide from the installation disc (`cluster_server/docs/vcs_install.pdf`) to the directory `/opt/VRTS/docs` on each node for reference.

Installing Language Packages

If you are installing a language version, install the language packages required by the VERITAS packages you have installed.

For following procedure, the `install_lp` command must use the `ssh` or `rsh` commands as root on all systems in the cluster. Make sure that permissions are granted for the system on which `install_lp` is run.

1. Insert the "Language CD" into the CD-ROM drive. The Solaris volume-management software automatically mounts the disc as `/cdrom/cdrom0`. The disc is automatically mounted.

2. Change to the `/cdrom/cdrom0` directory and install the language package:

```
# cd /cdrom/cdrom0
# ./install_lp
```


Using installvcs in a Secure Environment

In secure enterprise environments, `ssh` or `rsh` communication is not allowed between systems. In such cases, `installvcs` can install and configure VCS only on systems with which it can communicate—most often the local system only. When installation is complete, a "response" file is created. The "Example Response File" on page 51 resembles the file created by `installvcs`. Note that a response file generated by `installvcs` contains descriptions and explanations of the variables and their values. By copying this file to the other systems in the cluster and editing it to reflect the current local system, you can use the installation program with the `-responsefile` option to install and configure VCS identically on each system without being prompted.

▼ To use installvcs in a secure environment

1. On one system in the cluster, perform the steps listed in "Starting installvcs" on page 30. In step 5, the inability to communicate between systems is detected.

```
Verifying communication with south ..... ping successful
Attempting rsh with south ............... Cannot rsh to south

CPI WARNING V-9-10-1020
north cannot communicate with or does not have rsh permissions
  with the following systems: south

Would you like to install Cluster Server on systems north only
  and create a responsefile for systems south? [y,n,q] (y)
```

2. Enter all cluster information in the steps that follow step 5 on page 31. VCS is installed and configured on systems where communication is possible. Once installation is complete, the installation program reports that the response file is stored within the file `/opt/VRTS/install/logs/installvcs`*`date_time`*`.response`. Note that the date and time the installation began is part of the file's name.

Note Until VCS is installed and started on all systems in the cluster, the following appears when VCS is started: `VCS:11306:Did not receive cluster membership, manual intervention may be needed for seeding`

3. Using a method of your choice (for example, by using NFS, `ftp`, or a floppy disk), place a copy of the response file in a directory such as `/tmp` on the next system to install VCS.

4. On the next system, edit the response file. For the variables described in the following example, change the name of the system to reflect the current local system:

```
.
$CFG{INSTALL}{SYSTEMS} = ["south"];
.
.
$CFG{KEYS}{south} = ["XXXX-XXXX-XXXX-XXXX-XXXX-XXX"];
.
```

   For demo or site licenses, the license key need not be changed. When license keys are "node-locked" to specific cluster nodes, you must edit the license key.

5. On the next system, follow the steps listed in "Mounting the Product Disc" on page 26, but modify the command in step 2 on page 30 by starting VCS installation using the `-responsefile` option:

```
# ./installvcs -responsefile /tmp/installvcsdate_time.response
```

6. Repeat step 3 through step 5 until VCS has been installed on all systems in the cluster.

Using installvcs to Perform Unattended Installations

Using `installvcs` with the `-responsefile` option is useful not only for installing and configuring VCS within a secure environment, but for conducting unattended installations to other clusters as well. Typically, you can use the response file generated during the installation of VCS on one cluster to install VCS on other clusters. You can copy the file to a system in another cluster and manually edit the file to contain appropriate values.

Assuming the systems are set up and meet the requirements for installation, you can enter the following command from one of the cluster systems where you have copied the response file. For example, if `/tmp/response_file` is the response file's full path name:

```
# cd /cdrom/cdrom0
# ./installvcs -responsefile /tmp/response_file
```

Syntax Used in Response File

The syntax of Perl statements included in the response file varies, depending on whether "Scalar" or "List" values are required by the variables.

For example,

```
$CFG{Scalar_variable}="value";
```

or, in the case of an integer value:

```
$CFG{Scalar_variable}=123;
```

or, in the case of a list:

```
$CFG(List_variable}=["value", "value", "value"];
```

Example Response File

The following example response is a modified version of the response file generated on `vcs_cluster2` that you can use to install VCS on `vcs_cluster3`. The table on the following pages define the variables required for installation.

```
$CFG{Scalar} = ## (in the case of integers)
# installvcs configuration values:
#
$CFG{CLUSTERID}=8;
$CFG{CLUSTERNAME}="vcs_cluster3";
$CFG{CSGNETMASK}="255.255.240.0";
$CFG{CSGNIC}{ALL}="hme0";
$CFG{CSGVIP}="11.136.88.189";
$CFG{DONOTINSTALL}=[];
$CFG{DONOTREMOVE}=[];
$CFG{GCONETMASK}="255.255.240.0";
$CFG{GCONIC}{ALL}="hme0";
$CFG{GCOVIP}="11.136.88.189";
$CFG{INSTALL}{AUTOSTART}=1;
$CFG{INSTALL}{SIMULTANEOUS}=0;
$CFG{INSTALL}{SYSTEMS}=["east", "west"];
$CFG{INSTALL}{USESSH}=0;
$CFG{KEYS}{north}=["XXXX-XXXX-XXXX-XXXX-XXXX-XXX"];
$CFG{KEYS}{south}=["XXXX-XXXX-XXXX-XXXX-XXXX-XXX"];
$CFG{LLTLINK1}{east}="qfe0";
$CFG{LLTLINK1}{west}="qfe0";
$CFG{LLTLINK2}{east}="qfe1";
$CFG{LLTLINK2}{west}="qfe1";
$CFG{SMTPRECP}=["earnie@xyzstar.com"];
$CFG{SMTPRSEV}=["Warning"];
$CFG{SMTPSERVER}="smtp.xyzstar.com";
$CFG{SNMPCONS}=["neptune"];
$CFG{SNMPCSEV}=["Information"];
$CFG{SNMPPORT}=162;
$CFG{USERENPW}=["ghiHhgGnhDhqGohF", "fopLoqNxpJlpKp"];
$CFG{USERNAME}=["admin", "major"];
$CFG{USERPRIV}=["Administrators", "Administrators"];
```


Response File Variable Definitions

The variables used in the response file are defined in the following table. Note that while some variables are labeled as required and others as optional, some of the optional variables, if used, make it necessary that other optional variables be defined. For example, all variables related to the cluster service group (CSGNIC, CSGVIP, and CSGNETMASK) must be defined if any are defined. The same is true for the SMTP notification (SMTPSEVER, SMTPRECP, and SMTPRSEV), SNMP trap notification (SNMPPORT, SNMPCONS, and SNMPCSEV), and the Global Cluster Option (CGONIC, GCOVIP, and GCONETMASK).

Variable	List/ Scalar	Opt'l/ Req'd	Description
$CFG{INSTALL}{SYSTEMS}	List	Req'd	List of systems to be installed.
$CFG{INSTALL}{SYSTEMSCONFIG}	List	Opt'l	List of systems to be recognized in configuration if secure environment prevents all systems from being installed at once.
$CFG{INSTALL}{AUTOSTART}	Scalar	Opt'l	Defines whether the product is to be started following installation (1=yes/0=no).
$CFG{INSTALL}{SIMULTANEOUS}	Scalar	Opt'l	Defines if the product is to be installed on systems consecutively or simultaneously (1=simultaneous/0=consecutive).
$CFG{INSTALL}{USESSH}	Scalar	Opt'l	Defines whether ssh and scp are configured to be used to execute the installation or remote systems (1=ssh/0=rsh).
$CFG{DONOTINSTALL}{<PACKAGE>}	List	Opt'l	Instructs the installation to not install the optional packages designated in the list.
$CFG{CLUSTERNAME}	Scalar	Req'd	Defines the name of the cluster.
$CFG{CLUSTERID}	Scalar	Req'd	An integer between 0 and 255 that uniquely identifies the cluster.

Variable	List/ Scalar	Opt'l/ Req'd	Description
$CFG{KEYS}{<SYSTEM>}	Scalar	Opt'l	List of keys to be registered on the system.
$CFG{LLTLINK#}{<SYSTEM>}	Scalar	Req'd	Defines the NIC to be used for a private heartbeat link on each system. Two LLT links are required per system (LLTLINK1 and LLTLINK2). Up to four LLT links can be configured.
$CFG{LLTLINKLOWPRI}{<SYSTEM>}	Scalar	Opt'l	Defines a low priority heartbeat link. Typically, LLTLINKLOWPRI is used on a public network link to provide an additional layer of communication.
$CFG{CSGNIC}{<SYSTEM>}	Scalar	Opt'l	Defines the NIC for Cluster Manager (Web Console) to use on a system. 'ALL' can be entered as a system value if the same NIC is used on all systems.
$CFG{CSGVIP}	Scalar	Opt'l	Defines the virtual IP address to be used by the Cluster Manager (Web Console).
$CFG{CSGNETMASK}	Scalar	Opt'l	Defines the Netmask of the virtual IP address to be used by the Cluster Manager (Web Console).
$CFG{SMTPSERVER}	Scalar	Opt'l	Defines the domain-based hostname (example: smtp.yourcompany.com) of the SMTP server to be used for web notification.
$CFG{SMTPRECP}	List	Opt'l	List of full email addresses (example: user@yourcompany.com) of SMTP recipients
$CFG{SMTPRSEV}	Scalar	Opt'l	Defines minimum severity level of messages (Information, Warning, Error, SevereError) that listed SMTP recipients are to receive. Note that the ordering of severity levels must match that of the addresses of SMTP recipients.

Variable	List/ Scalar	Opt'l/ Req'd	Description
$CFG{SNMPPORT}	Scalar	Opt'l	Defines the SNMP trap daemon port (default=162).
$CFG{SNMPCONS}	List	Opt'l	List of SNMP console system names
$CFG{SNMPCSEV}	List	Opt'l	Defines minimum severity level of messages (Information, Warning, Error, SevereError) that listed SNMP consoles are to receive. Note that the ordering of severity levels must match that of the SNMP console system names.
$CFG{GCONIC}{<SYSTEM>}	Scalar	Opt'l	Defines the NIC for the Virtual IP used for the Global Cluster Option. 'ALL' can be entered as a system value if the same NIC is used on all systems.
$CFG{GCOVIP}	Scalar	Opt'l	Defines the virtual IP address to be used by the Global Cluster Option.
$CFG{GCONETMASK}	Scalar	Opt'l	Defines the Netmask of the virtual IP address to be used by the Global Cluster Option).
$CFG{USERENPW}	List	Opt'l	List of encoded passwords for users
$CFG{USERNAME}	List	Opt'l	List of names of users
$CFG{USERPRIV}	List	Opt'l	List of privileges for users

Using installvcs to Install Without Configuration

In certain situations, users may choose to install the VCS packages on a system before they are ready for cluster configuration. During such situations, the `installvcs -installonly` option can be used. The installation program licenses and installs VCS packages on the systems entered without creating any VCS configuration files.

Using installvcs to Configure Without Installation

When VCS has been installed without configuration, use the `installvcs -configure` option to configure VCS when you are ready for cluster configuration. The `installvcs` program prompts for cluster information as described in the section, "Example VCS Installation" on page 26, and creates VCS configuration files without performing installation.

The `-configure` option can be used to reconfigure a VCS cluster. VCS must not be running on systems when this reconfiguration is performed.

Using installvcs to Upgrade to VCS 4.1

The `installvcs` program can perform the following automatic upgrades:

- From VCS 3.5 or 4.0 to VCS 4.1 (see the following section)
- From GCM 3.5, or GCO 4.0, to VCS 4.1 with Global Cluster option (see "Upgrading from GCM 3.5 or GCO 4.0, to VCS 4.1 with the Global Cluster Option" on page 62)

Note If you want to upgrade from an earlier version of VCS to VCS 4.1 and use the Global Cluster option, you must first upgrade to standard VCS 4.1. After adding a license for the Global Cluster option, you can run the `gcoconfig` wizard. See the *VERITAS Cluster Server User's Guide* for instructions.

Upgrading from VCS 3.5 or 4.0

When you run `installvcs` on cluster systems that currently run VCS 3.5 or 4.0 the program guides you through an upgrade procedure. The following example shows `installvcs` when run on an existing VCS 4.0 cluster.

Starting the Upgrade

1. Start the program (see "Starting installvcs" on page 30, if necessary) on any system in the cluster. In this example, the system is `north`:

   ```
   # ./installvcs
   ```

   After displaying a copyright notice, the program examines the configuration files and discovers the existing cluster configuration (including the `ClusterService` group, if it is defined):

   ```
   VCS configuration files exist on this system with the following
   information:

      Cluster Name: vcs_cluster2
      Cluster ID Number: 7
      Systems: north south
      Service Groups: ClusterService XYZ_group

   No checks have been made to ensure the validity of the
   configuration.

   VCS version 4.0 is installed on this system

   Do you want to upgrade to version 4.1 on these systems using
   the current configuration? [y,n,q,?] (y) y
   ```

2. When you answer **y**, the program verifies that the upgrade can proceed on the systems by checking the operating system level and system-to-system communication. It also creates logs for the VCS update.

   ```
   Checking system communication:

     Verifying communication with north ........ ping successful
     Attempting rsh with north ................... rsh successful
     Attempting rcp with north ................... rcp successful
     Checking OS version on north ..................... SunOS 5.9
     Checking VRTSvcs package ............ version 4.0 installed
     Creating log directory on north ...................... Done
     Checking OS version on south ..................... SunOS 5.8
     Checking VRTSvcs package ............ version 4.0 installed

   Logs for installvcs are being created in var/tmp/
   installvcsdate_time.

   Using /usr/bin/rsh and /usr/bin/rcp to communicate with remote
   systems.

   Initial system check completed successfully.
   ```

3. When systems are verified, the installer adds the infrastructure packages on each system, upgrading them where necessary:

```
VERITAS Infrastructure package installation:

Installing VERITAS Infrastructure packages on north

 Checking VRTScpi package ..... version 4.0.5 installed
 Checking VRTSvlic package ........version 3.02.005d installed
 Checking file system space ...... required space is available
 Uninstalling VRTSvlic 3.02.005d on north ............... Done
 Uninstalling VRTScpi 4.0.5 on north ..................... Done
 Copying VRTScpi package to north ........................ Done
 Installing VRTScpi 4.1.0.54 on north .................... Done
 Copying VRTSvlic.tar.gz to north ........................ Done
 Installing VRTSvlic 3.02.005h on north .................. Done


Installing VERITAS Infrastructure packages on south
 .
 .
 .
 Installing VRTSvlic 3.02.005h on south .................. Done

VERITAS Infrastructure packages installed successfully.
```

4. The installer checks the licenses currently in place on the cluster systems:

```
Each system requires a VCS product license before installation.
License keys for additional product features should also be
added at this time.

Some license keys are node locked and are unique per system.
Other license keys, such as demo keys and site license keys,
are registered on all systems and must be entered on the first
system.

VCS Licensing Verification:

 Checking VCS license key on north .. Cluster Server Permanent
 Checking VCS license key on south .. Cluster Server Permanent
VCS licensing verified successfully.
```


Checking Upgrade Requirements and Changing Passwords

5. The installer checks for existing packages and available file system space on the first system, and backs up the current configuration files.

```
Checking VCS upgrade requirements on south

  Checking VRTSperl package ............ version 4.02 installed
  .
  .
  Checking file system space ...... required space is available
  Backing up VCS configuration files ..................... Done

installvcs must now make configuration updates and stop the
cluster before upgrading VCS packages

Are you ready to begin the Cluster Server upgrade at this time?
[y,n,q](y) y
```

6. This section only applies to version 3.5 to 4.1 upgrades.

When you indicate that upgrade can begin, the installer starts by prompting for a new administrator password:

```
VCS 4.1 uses a new advanced password encryption algorithm.
Therefore, passwords for all VCS users must be reset at this
time.  Be sure to write down the new VCS user passwords at this
time for future reference as they are not written to the install
log for your protection.

Resetting VCS password for user admin:
Enter New Password: ******

Enter Again: ******

Resetting VCS password for user smith:
Enter New Password:********

Enter Again:******
```

7. The program updates the passwords, freezes service groups, updates resource types, stops processes, and unloads the GAB and LLT modules on the first system.

```
Updating password for user admin ..... Done
Updating password for user smith ..... Done
Freezing group ClusterService ..... Done
Freezing group XYZ_group ..... Done
Updating VCS 4.1 Resource types ..... Done
Checking VRTSweb process ..... not running
Checking had process ..... running
Stopping had .....
.
.
.
Unloading gab module on north ..... Done
Checking llt driver ..... llt module loaded
Stopping llt driver ..... Done
Unloading llt module on north ..... Done
```

8. On each system, the installer checks for existing packages, patches, patch issues, and available file system space; and backs up VCS configuration files:

```
   Checking VRTSperl package ..... version 4.0.2 installed
.
.
.
 Checking required SunOS patch 113277-08 ... 113277-09 installed
 Checking VERITAS patch 117499 ..... not installed
 Checking for any Solaris patch issues ..... None
 Checking file system space ..... required space is available
 Backing up VCS configuration files ..... Done

installvcs must now make configuration updates and stop the
cluster before upgrading VCS packages
```

It then freezes groups, updates resources, checks and stops processes and drivers, and unloads drivers on each system.

```
Are you ready to begin the Cluster Server upgrade at this time?
[y,n,q] (y) y

   Freezing group ClusterService ..... Done
   .
   .
   .
   Checking llt driver ..... llt module loaded
   Stopping llt driver ..... Done
   Unloading llt module on vcstc10 ..... Done
```


Removing VCS 4.0 Packages and Installing VCS 4.1 Packages

9. When the program is ready to remove the VCS 3.5 packages and install the VCS 4.1 packages one each system, you can choose to have them installed consecutively on each of the systems (the default) or simultaneously. Errors are more clearly handled on a consecutive installation.

```
Would you like to upgrade Cluster Server packages on all
systems simultaneously? [y,n,q,?] (y) n
```

10. The program uninstalls packages from each system:

```
Uninstalling Cluster Server packages on north:
  Uninstalling VRTSvcsw 3.5 on north .......Done 1 of 20 steps
  Uninstalling VRTSweb 3.5 on north.........Done 2 of 20 steps
  .
  .
Cluster Server package uninstall completed successfully.
```

11. The VCS 4.1 packages are installed after the packages from the previous version are uninstalled.

```
Installing Cluster Server 4.1 on north:

 Installing VRTSperl 4.1.2 on north /....... Done 1 of 68 steps
 .
 .
 .
 Copying patch 115210-05 to south.......... Done 67 of 68 steps
 Adding patch 115210-05 on south .......... Done 68 of 68 steps

Cluster Server installation completed successfully.
```

Starting VCS

12. When the installation portion of the upgrade is complete, the program prompts you whether to start the cluster server processes. If you answer **y**, LLT, GAB, and VCS are started on each system:

```
Do you want to start Cluster Server processes now? [y,n,q] (y)

 Starting Cluster Server:

  Starting LLT on north ............................. Started
  Starting LLT on south ............................. Started
  Starting GAB on north ............................. Started
  Starting GAB on south ............................. Started
  Starting VCS on north ............................. Started
  Starting VCS on south ............................. Started
  Confirming Cluster Server startup ......... 2 systems RUNNING
  Unfreezing ClusterService ............................. Done
  Unfreezing XYZ_group .................................. Done
  Onlining ClusterService Group on north ................ Done

Cluster Server was started successfully.
```

Summarizing the Upgrade

13. After starting the cluster server processes, the upgrade is complete. The program lists the locations of the summary and log files.

```
Upgrade of Cluster Server 4.1 has completed successfully.

The upgrade summary is saved at:

      /opt/VRTS/install/logs/installvcsdate_time.summary

The installvcs log is saved at:

      /opt/VRTS/install/logs/installvcsdate_time.log
```

14. You can verify the cluster is operating properly. The procedures described in "Verifying the Installation of VCS 4.1" on page 97 can also be used to verify the upgrade.

Upgrading from VCS 3.5 or 4.0 in a Secure Environment

In a secure environment, run the `installvcs` program on each system to upgrade a cluster to VCS 4.1. On the first system, the program updates the configuration and stops the cluster before upgrading the system. On the other systems, it uninstalls the previous version and installs VCS 4.1. Once the final system is upgraded and started, the upgrade is complete.

Note To upgrade VCS manually, see "Manually Upgrading VCS to Release 4.1" on page 143.

Upgrading from GCM 3.5 or GCO 4.0, to VCS 4.1 with the Global Cluster Option

When you run `installvcs` on cluster systems that currently run GCM 3.5, the program detects that GCM is running and prompts you to indicate whether or not you want to upgrade to VCS 4.1. If you agree, the installer guides you through an automated upgrade procedure that includes the configuration of the Global Cluster Option. For a description of the VCS 4.1 Global Cluster Option, see the *VERITAS Cluster Server User's Guide*.

Note The installer is capable of upgrading a standard GCM configuration, ideally set up using the GCM configuration wizards. If you want to upgrade a highly customized GCM configuration to VCS 4.1, contact a VERITAS consultant to assist you.

In the following example, step 1 through step 9 are identical to the standard VCS upgrade and are included here for completeness. GCM sites, `mtv` and `svl` are currently running GCM version 3.5. The `installvcs` program is run on the CGMmaster node `north` in the site `mtv`, which also includes the node `south`. Use the same procedure to upgrade other sites.

Starting the Upgrade

1. Start the program on the GCM master node (see "Starting installvcs" on page 30, if necessary).

   ```
   # ./installvcs
   ```

2. The program examines the configuration files and discovers the existing cluster configuration (including the `ClusterService` group, if it is defined):

   ```
   VCS configuration files exist on this system with the following
   information:

      Cluster Name: vcs_cluster2
      Cluster ID Number: 7
      Systems: north south
      Service Groups: ClusterService XYZ_group appgrp repgrp

   No checks have been made to ensure the validity of the
   configuration.

   VCS version 3.5 is installed on this system
   ```

3. The installer prompts you to upgrade VCS on the cluster systems:

   ```
   Do you want to upgrade to version 4.1 on these systems using
   the current configuration? [y,n,q,?] (y) y
   ```

4. When you answer **y**, the program verifies that the VCS upgrade can proceed on the systems by checking the operating system levels and system-to-system communication.

   ```
   Checking system communication:

     Checking OS version on north .....................SunOS 5.7
     Checking VRTSvcs package ..... version 3.5 installed
     Verifying communication with south..... ping successful
     Attempting rsh with south ..... rsh successful
     Attempting rcp with south ..... rcp successful
     Checking OS version on south .....................SunOS 5.8
     Checking VRTSvcs package ..... version 3.5 installed
     Creating log directory on south ..... Done
   ```


Adding the Infrastructure Packages and Checking Licenses

5. When systems are verified, the installer adds the infrastructure packages on each system, upgrading them where necessary:

```
Checking VRTScpi package ..................... not installed
Checking VRTSvlic package ..................... not installed
Checking file system space ..... required space is available
Uninstalling VRTSvlic 3.00.007d on north /.............. Done
Installing VRTScpi 4.1.6.8 on north..................... Done
.
.
Installing VRTSvlic 3.02.005b on south ................. Done
VERITAS Infrastructure packages installed successfully.
```

After you install the VERITAS licensing package (`VRTSvlic`), the installer checks the status of the current licenses on each system.

6. The installer checks the licenses currently in place on the cluster systems:

```
Each system requires a VCS product license before installation.
License keys for additional product features should also be
added at this time.

Some license keys are node locked and are unique per system.
Other license keys, such as demo keys and site license keys,
are registered on all systems and must be entered on the first
system.

VCS Licensing Verification:

 Checking VCS license key on north ... Cluster Server Permanent
 Checking VCS license key on south ... Cluster Server Permanent
VCS licensing verified successfully.
```

Checking Upgrade Requirements and Changing Passwords

7. The installer then checks for existing packages and available file system space on the first system, and backs up the current configuration files.

```
Checking VCS upgrade requirements on north

  Checking VRTSperl package ............. version 3.5 installed
  .
  .
 Checking file system space ....... required space is available
 Backing up VCS configuration files ..................... Done
```

8. The installer can begin the upgrade of VCS and prompts you to indicate you are ready:

```
installvcs must now make configuration updates and stop the
cluster before upgrading VCS packages

Are you ready to begin the Cluster Server upgrade at this time?
[y,n,q](y)
```

9. When you indicate that upgrade can begin, the installer starts by prompting for a new administrator passwords, including the password for GCM users:

```
Resetting VCS password for user admin:
Enter New Password: ******

Enter Again: ******

Resetting VCS password for user GCMmaster:
Enter New Password:********

Enter Again:********

Updating password for user admin .........................Done
Updating password for user GCMmaster......................Done
```

Capturing the GCM Configuration

10. The installer prompts you to indicate you are ready for the CGM upgrade:

```
Upgrade procedure will replace GCM 3.5 with VCS 4.1

Do you want to upgrade GCM at this time? [y,n,q] (y) y
```

11. The installer must modify the designation of the configured CGM components, renaming the GCM sites to VCS cluster names, and revising the name of GCM service groups. For example:

```
Capturing CGM configuration:

 site mtv will become cluster vcspri
 site sv1 will become remote cluster vcsdr
 Group appgrp will become Global Group with Manual policy

Do you want to continue? [y,n,q] (y)
```

These changes take place after VCS 4.1 has been installed and started, but you have an opportunity to stop the upgrade at this time.

Completing Check of Upgrade Requirements

12. When you continue, the installer begins the upgrade by freezing service groups, updating resource types, and stopping processes, and unloading the GAB and LLT modules on the first system:

```
Offlining ClusterService ..............................Done
Freezing group ClusterService ......................... Done
Freezing group XYZ_group .............................. Done
Updating VCS 4.1 Resource types ....................... Done
Checking VRTSweb process ..................... not running
Checking had process ................................ running
Stopping had .....................
.
.
```

It continues checking the requirements and packages on the other node, where processes are stopped, and drivers are unconfigured and unloaded.

```
.
.
Stopping llt driver ................................... Done
Unloading llt module on south ......................... Done

Upgrade requirement checks completed successfully.
```

Removing VCS 3.5 Packages, Installing VCS 4.1 Packages

13. When the program is ready to remove the VCS 3.5 packages and install the VCS 4.1 packages one each system, you can choose to have them installed consecutively on each of the systems (the default) or simultaneously. Errors are more clearly handled on a consecutive installation.

```
Would you like to upgrade Cluster Server packages on all
systems simultaneously? [y,n,q,?] (y)

Uninstalling Cluster Server 4.1 on all systems simultaneously:

Uninstalling Cluster Server packages on north:
  Uninstalling VRTSvcsw 3.5 on north .......Done 1 of 23 steps
  Uninstalling VRTSweb 3.5 on north.........Done 2 of 23 steps
  .
  .
Cluster Server package uninstall completed successfully.
```

14. The VCS 4.1 packages are installed after the packages from the previous version are uninstalled.

```
Installing Cluster Server 4.1 on all systems simultaneously:

 Copying VRTSperl.tar.gz to south ..........Done 1 of 66 steps
 Installing VRTSperl 4.1.2 on north ........ Done 2 of 66 steps
 Installing VRTSperl 4.1.2 on south ........ Done 3 of 66 steps
 .
 .
 Copying patch 115210-05 to south.......... Done 65 of 66 steps
 Adding patch 115210-05 on south .......... Done 66 of 66 steps

Cluster Server installation completed successfully.
```

Starting VCS

15. When the installation portion of the upgrade is complete, the program asks you if you want to start the cluster server processes. If you answer **y**, LLT, GAB, and VCS are started on each system:

```
Do you want to start Cluster Server processes now? [y,n,q] (y)

 Starting Cluster Server:

  Starting LLT on north ............................. Started
  Starting LLT on south ............................. Started
  Starting GAB on north ............................. Started
  Starting GAB on south ............................. Started
  Starting VCS on north ............................. Started
  Starting VCS on south ............................. Started
  Confirming Cluster Server startup ......... 2 systems RUNNING
  Unfreezing ClusterService ............................. Done
  Unfreezing XYZ_group .................................. Done
  Onlining ClusterService Group on north ................ Done

Cluster Server was started successfully.
```


Completing the Upgrade from CGM to VCS 4.1 Global Cluster

16. After VCS 4.1 is started on each node, the automatic upgrade GCM to VCS 4.1 continues by configuring the VCS 4.1 global cluster option features:

```
Do you want to automatically upgrade GCM to VCS 4.1: [y,n,q] (y)
   Adding remote cluster vcsdr ..........................Done
   Adding Icmp heartbeat to vcsdr .......................Done
   Adding RVGPrimary type ...............................Done
   Processing group appgrp .............................. Done
   deleting GCM resources ...............................Done
   Onlining ClusterService Group on north ...............Done

Upgrade requirment checks completed successfully.
Press [Return] to continue:
```

Summarizing the Upgrade

17. After starting the cluster server processes, the upgrade is complete. The program lists the locations of the summary and log files.

```
Upgrade of Cluster Server 4.1 has completed successfully.

The upgrade summary is saved at:

      /opt/VRTS/install/logs/installvcsdate_time.summary

The installvcs log is saved at:

      /opt/VRTS/install/logs/installvcsdate_time.log
```

You can verify the cluster is operating properly. The procedures described in "Verifying the Installation of VCS 4.1" on page 97 can also be used to verify the upgrade.

Completing the Upgrade of GCM to VCS 4.1 with GCO

Perform the following steps on each site.

1. Use the `vxlicinst` utility to add the VCS 4.1 license with the Global Cluster Option on each cluster node. The GCM 3.5 license discovered during the upgrade must be upgraded. Refer to "Using vxlicinst to Update Product Licenses" on page 70.

2. Run the `gcoconfig` utility to configure the WAC (wide-area connector) resource.

   ```
   # gcoconfig
   ```

   Refer to the *VERITAS Cluster Server User's Guide* for information on running the GCO Configuration Wizard to complete the configuration of VCS 4.1 with the Global Cluster Option.

3. Stop VCS on all cluster nodes:

   ```
   # hastop -all -force
   ```

4. When VCS is stopped on all nodes, start it again on each node.

   ```
   # hastart
   ```


Checking Licensing Information on the System

You can use the `vxlicrep` utility to display information about the licenses on a system. For example, enter:

```
# cd /opt/VRTS/bin
# ./vxlicrep

VERITAS License Manager vxlicrep utility version 3.00.004
Copyright (C) VERITAS Software Corp 2002. All Rights reserved.

Creating a report on all VERITAS products installed on this system

 ------------------**************************-----------------

   License Key                          = xxx-xxx-xxx-xxx-xxx
   Product Name                         = VERITAS Cluster Server
   Serial Number                        = 1249
   License Type                         = PERMANENT
   OEM ID                               = 478

 Features :=
   Platform                             = Solaris
   Version                              = 4.1
   Tier                                 = 0
   Reserved                             = 0

   Mode                                 = VCS
```

From the output, you can determine the license key, the type of license, the product for which it applies, and its expiration date, if any. Demo keys have expiration dates, permanent keys and site keys do not.

Using vxlicinst to Update Product Licenses

Use the `vxlicinst` command to add the VCS license key on each system:

```
# cd /opt/VRTS/bin
# ./vxlicinst -k XXXX-XXXX-XXXX-XXXX-XXXX-XXX
```

If you have VCS already installed and configured and you are using a demo license, you can replace the demo license using the procedure "Replacing a VCS Demo License with a Permanent License" on page 94.

Using Other installvcs Options

In addition to the `-precheck`, `-responsefile`, `-installonly`, and `-configure` options, the `installvcs` program has other useful options.

Option and Syntax	Description
`-license`	Update product licenses. Useful for replacing demo license.
`-nolic`	Install product packages on systems without licensing or configuration. License-based features or variants are not installed when using this option.
`-usessh` *`system1 system2`*	Specifies that `ssh` and `scp` are to be used for communication between systems instead of `rsh` and `rcp`. This option requires that systems be pre-configured such that `ssh` commands between systems execute without prompting for passwords or confirmations.
`-pkgpath` *`pkg_path`*	Specifies that *pkg_path* contains all packages to be installed by `installvcs` on all systems; *pkg_path is* the complete path of a directory, usually NFS mounted.
`-tmppath` *`tmp_path`*	Specifies that *tmp_path*, not `/var/tmp`, is the working directory for `installvcs`. This destination is where initial logging is performed and where packages are copied on remote systems before installation.

Using uninstallvcs

Before removing VCS from any system in the cluster, shut down applications such as Java Console or any VCS enterprise agents that depend on VCS.

The `uninstallvcs` program does not remove shared packages such as `VRTSllt`, `VRTSgab`, and `VRTSweb` if they are also used as a part of other VERITAS software programs. The `uninstallvcs` program does not automatically uninstall VCS enterprise agents, but offers uninstallation if proper package dependencies on `VRTSvcs` are found. See the documentation for the specific enterprise agent for instructions on removing it if it is not uninstalled by `uninstallvcs`.

▼ To uninstall VCS

1. If you can execute commands as root on the remote systems in the cluster using `ssh` or `rsh`, run `uninstallvcs` on one system to uninstall VCS on all systems in the cluster. If you cannot execute commands as root on remote systems in the cluster using `ssh` or `rsh`, you must run `uninstallvcs` each system in the cluster.

2. Enter the command to start `uninstallvcs`:

   ```
   # cd /opt/VRTS/install
   # ./uninstallvcs
   ```

   The program begins with a copyright notice followed by a description of the cluster and a prompt to proceed uninstalling software:

   ```
   VCS configuration files exist on this system with the following
   information:

      Cluster Name: VCS_cluster2
      Cluster ID Number: 7
      Systems: north south
      Service Groups: ClusterService groupA groupB

   Do you want to uninstall VCS from these systems? [y,n,q] (y)
   ```

   Enter **`y`** to uninstall VCS on these systems. If you enter **`n`** or if no VCS configuration files are found on the local system, the program prompts you for a list of systems to uninstall.

Note Before removing VCS from fewer than all systems in a cluster, make sure that no service groups are running on the systems from which VCS is uninstalled. You must also reconfigure VCS on the remaining systems. Refer to "Adding and Removing Cluster Systems" on page 155 for instructions on how to remove systems from a cluster.

3. The `uninstallvcs` program continues by verifying communication between systems and checking the installations on each system to determine the packages to be uninstalled. If packages, such as enterprise agents, are found to be dependent on a VCS package, you are prompted on whether you want them removed:

```
.
.
.
Checking VRTSvcs package ..............  version 4.1 installed
Checking VRTSvcs dependencies .....VRTScscw VRTSvcsmn VRTSvcsor
.
.
.
Do you want to uninstall VRTSvcsor which is dependent on
package VRTSvcs? (Y)
```

   Enter `y` if you wish to remove the designated package.

4. After the program verifies that uninstallation can proceed, it displays the following message:

```
uninstallvcs is now ready to uninstall VCS packages.
All VCS processes that are currently running will be stopped.

Are you sure you want to uninstall VCS packages? [y,n,q] (y)
```

   If you press Enter, uninstallation stops processes and unloads kernel modules:

```
Stopping VCS processes on north:

  Checking VRTSweb process ..... not running
  Checking had process ..... running
  Stopping had ..... Done
  Checking hashadow process ..... not running
  Checking CmdServer process ..... running
  Killing CmdServer ..... Done
  Checking notifier process ..... running
  Killing notifier ..... Done
  Checking vxfen driver ..... vxfen module loaded
  Stopping vxfen driver ..... Done
  Unloading vxfen module on south ..... Done
  Checking gab driver ..... gab module loaded
  Stopping gab driver ..... Done
  Unloading gab module on north  ..... Done
  Checking llt driver ..... llt module loaded
  Stopping llt driver ..... Done
  Unloading llt module on north  ..... Done
```

   The program performs the same actions on the other systems in the cluster.

5. After stopping processes on each system, the program removes the packages:

```
Uninstalling Cluster Server 4.1 on all systems simultaneously:

  Uninstalling VRTScssim 4.1 on south ...... Done 1 of 30 steps
  Uninstalling VRTScssim 4.1 on north ...... Done 2 of 30 steps
  Uninstalling VRTScscm 4.3 on north ....... Done 3 of 30 steps
  .
  .
  .
  Uninstalling VRTSvxfen 4.1 on south ..... Done 28 of 30 steps
  Uninstalling VRTSgab 4.1 on south ....... Done 29 of 30 steps
  Uninstalling VRTSllt 4.1 on south ....... Done 30 of 30 steps

Cluster Server package uninstall completed successfully.
```

6. After all packages are successfully removed, the program indicates the location of summary and log files:

```
Uninstallation of Cluster Server has completed successfully.

  The uninstallation summary is saved at:

       /opt/VRTS/install/logs/uninstallvcsdate_time.summary

  The uninstallvcs log is saved at:

       /opt/VRTS/install/logs/uninstallvcsdate_time.log
```

Uninstalling VERITAS Infrastructure Packages

VERITAS products use shared packages called infrastructure packages. When uninstalling a single VERITAS product, the shared infrastructure packages remain. If you want to remove all VERITAS products and packages from a system, run the `uninstallinfr` program.

```
# cd /opt/VRTS/install
# ./uninstallinfr
```

This program removes the `VRTSvlic` licensing package and the `VRTScpi` and `VRTSperl` packages required for product installation.

Running uninstallvcs from the VCS 4.1 Disc

If you need to uninstall VCS after an incomplete installation, or if the `uninstallvcs` program is not available in `/opt/VRTS/install`, you may need to use the `uninstallvcs` program on the VCS 4.1 disc.

Manually Installing and Configuring VCS 4

This chapter describes the manually installing VCS:

- Copying VCS patches and compressed VCS packages from the software disc to a local temporary directory
- Unzipping compressed package files
- Installing VCS packages using `pkgadd`
- Installing VCS patches using `patchadd`
- Installing language packages
- Licensing VCS
- Configuring LLT, GAB, and VCS
- Configuring membership heartbeat regions on disk (optional)
- Starting LLT, GAB, and VCS
- Removing VCS packages using `pkgrm` and VCS patches using `patchrm`

Manually Installing VCS

You can manually install and configure VCS instead of using the `installvcs` program. Manually installing VCS is appropriate when:

- You are installing a single VCS package.
- You are installing VCS to one system in a cluster already running VCS 4.1.
- You are unable to install on a system over the network. This can occur if the user does not have remote root user access.

Because of the number of steps and care required to install VCS, VERITAS recommends that you avoid installing VCS manually. Use the `installvcs` program described in "Using the VCS Installation Programs" on page 23 whenever possible.

Requirements for Installing VCS

Review "Preparing to Install VCS 4.1" on page 7 and verify that you are ready to install VCS software.

Disk Space Required for Manual Installation

Note that full VCS installation requires 550 MB in the `/opt` directory (additionally the language pack requires another 20 MB), 20 MB in `/usr`, 20 MB in `/var`, and 10 MB in `/` for each system.

JumpStart

VCS is JumpStart compliant. When configuring the JumpStart server, make sure to install the following sections, and the packages and patches therein, in this order:

- "Installing the Infrastructure Packages" on page 77
- "Installing VCS Packages" on page 78
- "Installing VCS Patches" on page 79
- "Installing Language Packages" on page 80 for the Japanese language packages

For more information on using JumpStart, refer to the appropriate Sun Solaris documentation.

Installing VCS Software Manually

On each system in the cluster, do the following steps to install VCS.

1. Log in as root user on the system where you are installing VCS.
2. Create a directory for installation:

   ```
   # mkdir /tmp/install
   ```

3. Insert the software disc with the VCS software into a drive connected to the system. The Solaris volume-management software automatically mounts the disc as `/cdrom/cdrom0`. Type the command:

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

4. Copy the compressed package files from the software disc to the temporary directory:

   ```
   # cp -r cluster_server/pkgs/* /tmp/install
   ```

5. Go to the temporary directory and unzip the compressed package files:

   Note If your system does not have the `gunzip` utility, copy it from the disc:
   `# cp /cdrom_path/gnu/gunzip /tmp/install`

```
# cd /tmp/install
# gunzip VRTS*.gz
```

6. List the files in the temporary directory:

```
# ls /tmp/install
VRTSat.tar       VRTSjre.tar       VRTSvcsmn.tar
VRTScpi/         VRTSllt.tar       VRTSvcsw.tar
VRTScscm.tar     VRTSperl.tar      VRTSvlic.tar
VRTScscw.tar     VRTSvcs.tar       VRTSvxfen.tar
VRTScssim.tar    VRTSvcsag.tar     VRTSweb.tar
VRTScutil.tar    VRTSvcsdc.tar     info/
VRTSgab.tar      VRTSvcsmg.tar
```

Installing the Infrastructure Packages

The packages collectively known as infrastructure packages are non-VCS packages that VCS requires for installation.

1. Extract the compressed files from the tar files:

```
# tar -xvf VRTSvlic.tar
# tar -xvf VRTSperl.tar
```

2. Install the infrastructure packages using `pkgadd` (note that VRTScpi was not compressed):

```
# pkgadd -d . VRTScpi
# pkgadd -d . VRTSvlic
# pkgadd -d . VRTSperl
```


Installing VCS Packages

The VCS packages include required packages and optional packages. Install the required packages first. All packages are installed in the `/opt` directory.

When selecting optional packages, note:

- The packages for VCS manual pages (`VRTSvcsmn`) and VCS documentation (`VRTSvcsdc`) are recommended; it is not necessary to install the documentation package on all nodes.
- The I/O fencing package (`VCSvxfen`) can be used only with shared disks that support SCSI-3 Persistent Reservations (PR). See "Setting Up I/O Fencing" on page 111 for the procedures to test shared storage for SCSI-3 Persistent Reservations and for implementing I/O fencing. See the *VERITAS Cluster Server User's Guide* for a conceptual description of I/O fencing.
- The VCS configuration wizard (`VRTScscw`) package includes wizards for the installation and/or configuration of VERITAS products for which VCS configuration is required.
- To use the Java Console with VCS Simulator, you must install the VRTScssim and VRTScscm packages.

1. Extract the required VCS files from the compressed files:

```
# tar -xvf VRTSat.tar
# tar -xvf VRTSllt.tar
# tar -xvf VRTSgab.tar
# tar -xvf VRTSvcs.tar
# tar -xvf VRTSvcsmg.tar
# tar -xvf VRTSvcsag.tar
# tar -xvf VRTSjre.tar
# tar -xvf VRTScutil.tar
# tar -xvf VRTScscw.tar
# tar -xvf VRTSweb.tar
# tar -xvf VRTSvcsw.tar
```

2. Install the required VCS packages. Do not install any packages already installed on the system. As you enter the command, be sure to list the packages in the order shown in the following example:

```
# pkgadd -d . VRTSat VRTSllt VRTSgab VRTSvcs VRTSvcsmg VRTSvcsag
  VRTSjre VRTScutil VRTScscw VRTSweb VRTSvcsw
```

3. Extract the optional VCS packages from the compressed files:

```
# tar -xvf VRTSvxfen.tar
# tar -xvf VRTSvcsmn.tar
# tar -xvf VRTSvcsdc.tar
# tar -xvf VRTScssim.tar
# tar -xvf VRTScscm.tar
```

4. Install the optional VCS packages. As you enter the command, use the following example; you may omit packages you do not want to install, but be sure to list those you are installing in the order shown:

```
# pkgadd -d . VRTSvxfen VRTSvcsmn VRTSvcsdc
  VRTScssim VRTScscm
```

5. Perform step 1 through step 4 on each of the other cluster systems.

Installing VCS Patches

1. Change to the directory containing the patches:

```
# cd /cdrom/cdrom0/cluster_server/patches
```

2. Copy the compressed files from the software disc to the temporary directory:

```
# cp -r * /tmp/install
```

3. Go to the temporary directory and unzip the compressed patch files:

```
# cd /tmp/install
# gunzip 117499*.gz
```

4. Extract the compressed files from the tar files:

```
# tar -xvf 117499-02.tar
```

5. Install the required VCS patch using the patchadd command:

```
# patchadd 117499-02
```

6. Perform step 1 and step 5 on each cluster system.

Installing Language Packages

If you are installing the Japanese language version of VCS, you can install the language packages required by VCS after you have installed the base VCS packages. The Japanese language packages are:

Package Name	Description
VRTSjacs	Japanese VERITAS Cluster Server Message Catalogs
VRTSjacsd	Japanese VERITAS Cluster Server Documentation
VRTSjacsj	Japanese VERITAS Cluster Server Cluster Manager
VRTSjacsw	Japanese VERITAS Cluster Manager (Web Console)
VRTSjaweb	Japanese VERITAS Java Web Server Language Pack
VRTSjacsu	Japanese VERITAS Cluster Utility Language Pack
VRTSjacsm	Japanese VERITAS Cluster Server Simulator (optional)
VRTSmulic	Multi-language VERITAS License Utilities

▼ To install the Japanese language packages

1. Insert the Language CD into the CD-ROM drive.
2. Go to the directory containing the Japanese language packages required for VCS:

   ```
   # cd /cdrom/cdrom0/ja/cluster_server/pkgs
   ```

3. Copy the compressed package files from the software disc to the temporary directory:

   ```
   # cp -r * /tmp/install
   ```

4. Go to the temporary directory and unzip the compressed package files:

   Note If your system does not have the `gunzip` utility, copy it from the disc:
   `# cp /cdrom_path/gnu/gunzip /tmp/install`

   ```
   # cd /tmp/install
   # gunzip VRTS*.gz
   ```

5. List the files in the temporary directory:

```
# ls /tmp/install
  VRTSjacsj.tar  VRTSjacsw.tar    VRTSjacs.tar
  VRTSjacsm.tar  VRTSjaweb.tar    VRTSjacsd.tar
  VRTSjacsu.tar  VRTSmulic.tar    /info
```

6. Extract the compressed files from the tar files:

```
# tar -xvf VRTSmulic.tar
# tar -xvf VRTSjaweb.tar
# tar -xvf VRTSjacs.tar
# tar -xvf VRTSjacsd.tar
# tar -xvf VRTSjacsj.tar
# tar -xvf VRTSjacsm.tar
# tar -xvf VRTSjacsu.tar
# tar -xvf VRTSjacsw.tar
```

7. Install the infrastructure packages using `pkgadd` (note that VRTScpi was not compressed):

```
# pkgadd -d . VRTSmulic
# pkgadd -d . VRTSjaweb
# pkgadd -d . VRTSjacs
# pkgadd -d . VRTSjacsd
# pkgadd -d . VRTSjacsj
# pkgadd -d . VRTSjacsu
# pkgadd -d . VRTSjacsw
```

8. Install the optional packages using the following command:

```
# pkgadd -d . VRTSjacsm
```

9. Repeat step 1 through step 8 on each system in the cluster.

Adding a License Key

After all packages have been installed on each cluster node, use the `vxlicinst` command to add the VCS license key on each system:

```
# cd /opt/VRTS/bin
# ./vxlicinst -k XXXX-XXXX-XXXX-XXXX-XXXX-XXX
```


Checking Licensing Information on the System

Use the `vxlicrep` utility to display information about all VERITAS licenses on a system. For example, enter:

```
# cd /opt/VRTS/bin
# ./vxlicrep
```

From the output, you can determine the license key, the type of license, the product for which it applies, and its expiration date, if any. Demo keys have expiration dates, while permanent keys and site keys do not.

Upgrading

If you have manually added 4.1 packages to upgrade your cluster to VCS 4.1, you need to restore the configuration files from your previous VCS installation. Refer to "Manually Upgrading VCS to Release 4.1" on page 143 for instructions on restoring the configuration files.

Installing Cluster Manager

If you did not elect to install Cluster Manager (the VCS Java-based graphical user interface package), `VRTScscm`, you can do it later. See "Installing the VCS Java Console" on page 108.

Copying the Installation Guide to Each System

After you install VCS, we recommend that you copy the PDF version of this guide from the installation software disc (`cluster_server/docs/vcs_ig.pdf`) to the directory `/opt/VRTSvcs/docs` on each cluster system to make it available for reference.

Configuring LLT and GAB

VCS uses LLT and GAB to replace the functions of TCP/IP for VCS private network communications. LLT and GAB provide the performance and reliability required by VCS for these and other functions.

LLT and GAB must be configured as described in the following sections.

Configuring Low Latency Transport (LLT)

To configure LLT, set up two files: `/etc/llthosts` and `/etc/lltab` on each system in the cluster.

Setting Up /etc/llthosts

The file `llthosts`(4) is a database, containing one entry per system that links the LLT system ID (in the first column) with the LLT host name. You must create an identical file on each system in the cluster.

Using `vi`, or another editor, create the file `/etc/llthosts` that contains entries resembling:

```
0 north
1 south
```

Setting Up /etc/lltab

The `/etc/lltab` file must specify the system's ID number (or, its node name), and the network links that correspond to the system. In addition, the file can contain other directives. See "LLT Directives" on page 84. Refer also to the sample `lltab` file in `/opt/VRTSllt`.

Using `vi`, or another editor, create the file `/etc/lltab` that contains entries that resemble:

```
set-node north
set-cluster 2
link qfe0 /dev/qfe:0 - ether - -
link qfe1 /dev/qfe:1 - ether - -
```

The first line must identify the system on which the file exists. In the example above, the value for `set-node` could be `north, 0`, or the file name `/etc/nodename`, provided the file contains the name of the system (`north` in this example). The next two lines, beginning with the link command, identify the two private network cards that the LLT protocol uses. The order of directives must be the same as in the sample file `/opt/VRTSllt/sample-lltab`.

LLT Directives

`set-node`	Assigns the system ID or symbolic name. The system ID number must be unique for each system in the cluster, and must be in the range 0-31. The symbolic name corresponds to the system ID listed in `/etc/llthosts` file. *Note that LLT fails to operate if any systems share the same ID.*
`link`	Attaches LLT to a network interface. At least one link is required, and up to eight are supported. The first argument to link is a user-defined tag shown in the `lltstat`(1M) output to identify the link. It may also be used in `lltab` to set optional static MAC addresses. The second argument to link is the device name of the network interface. Its format is *device_name:device_instance_number*. The remaining four arguments to link are defaults; these arguments should be modified only in advanced configurations. There should be one `link` directive for each network interface. LLT uses an unregistered Ethernet SAP of 0xCAFE. If the SAP is unacceptable, refer to the `lltab`(4) manual page for information on how to customize SAP. Note that IP addresses do not need to be assigned to the network device; LLT does not use IP addresses.
`set-cluster`	Assigns a unique cluster number. Use this directive when more than one cluster is configured on the same physical network connection. LLT uses a default cluster number of zero.
`link-lowpri`	Use this directive in place of `link` for public network interfaces. This directive prevents VCS communication on the public network until the network is the last link, and reduces the rate of heartbeat broadcasts. Note that LLT distributes network traffic evenly across all available network connections and, in addition to enabling VCS communication, broadcasts heartbeats to monitor each network connection.

For more information about LLT directives, refer to the `lltab`(4) manual page.

Additional Considerations for LLT

- Each network interface configured for LLT (Low Latency Transport) must be attached to a separate and distinct physical network.
- By default, Sun systems assign the same MAC address to all interfaces. Thus, connecting two or more interfaces to a network switch can cause problems. For example, if IP is configured on one public interface and LLT on another, and both interfaces are connected to a switch, the duplicate MAC address on the two switch ports can cause the switch to incorrectly redirect IP traffic to the LLT interface and vice-versa. To avoid this, configure the system to assign unique MAC addresses by setting the `eeprom`(1M) parameter `local-mac-address?` to true.

Optimizing LLT Media Speed Settings on Private NICs

For optimal LLT communication among the cluster nodes, the interface cards on each node must use the same media speed settings. Also, the settings for switches or hubs used for the interconnects must match that of the interface cards. Incorrect settings can cause poor network performance or even network failure.

Guidelines of Setting Media Speed of LLT Interconnects

- If you have hubs or switches for LLT interconnects, VERITAS recommends using the Auto_Negotiation media speed setting on each Ethernet card on each node.
- If you have hubs or switches for LLT interconnects and you do not use the Auto_Negotiation media speed setting, set the hub or switch port to the same setting as that used on the cards on each node.
- If you use directly connected Ethernet links (using crossover cables), set the media speed to the highest value common to both cards, typically 100_Full_Duplex.
- VERITAS does not recommend using dissimilar network cards for private links.

Details for setting the media speeds for specific devices are outside of the scope of this manual. Consult the device's documentation for more information.

Configuring Group Membership and Atomic Broadcast (GAB)

To configure GAB, use `vi` or another editor to set up an `/etc/gabtab` configuration file on each system in the cluster. The following example shows a simple `/etc/gabtab` file:

```
/sbin/gabconfig -c -nN
```

where the `-c` option configures the driver for use and `-nN` specifies that the cluster is not formed until at least *N* systems are ready to form the cluster. By default, *N* is the number of systems in the cluster.

Note VERITAS does not recommend the use of the `-c -x` option for `/sbin/gabconfig`. Using `-c -x` dramatically increases configuration time for the Gigabit Ethernet controller and can lead to a split-brain condition.

Configuring Membership Heartbeat Regions on Disk (optional)

You can set up disk heartbeating on a shared disk to provide an additional path for VCS heartbeating (see "Two Types of Channels: Network and Shared Disks" on page 4). With disk heartbeating configured in addition to the private network connections, VCS has multiple heartbeat paths available. For example, if one of two private network connections fails, VCS has the remaining network connection and the disk heartbeat region that allow heartbeating to continue normally.

With disk heartbeating configured, each system in the cluster periodically writes to and reads from specific regions on a dedicated shared disk. This exchange consists of heartbeating only, and does not include communication about cluster status.

Because disk heartbeats do not support cluster communication, a failure of private network links that leaves only a disk heartbeat link between one system and the remaining nodes in the cluster causes the system to have a special jeopardy status. The system is excluded from regular cluster membership with the other systems because the status of its resources cannot be known by other systems. While the system in special jeopardy can continue to function, its resources are prevented from failing over or being switched over. This prevents possible data corruption in a split-brain situation.

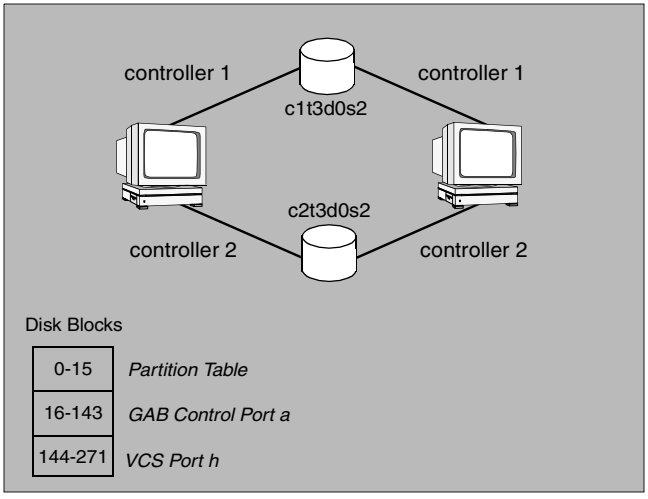
The *VERITAS Cluster Server User's Guide* contains a description of how VCS uses heartbeating to provide cluster systems a means to determine the status of their peers and to prevent possible data corruption on shared storage. The *VERITAS Cluster Server User's Guide* also describes reconnecting private networks.

Editing the /etc/gabtab File to Add Heartbeat Regions

You can set up heartbeat regions on a shared disk by using `gabdiskhb(1M)` commands. You must enter these commands in the `/etc/gabtab` file identically on each system (see "Configuring Group Membership and Atomic Broadcast (GAB)" on page 86). The heartbeat regions on the shared disk are configured when the systems start up.

When configuring disk heartbeating, you must create two membership heartbeat regions on the disk, each consisting of 128 blocks: one for the GAB control (port a) and one for the VCS (port h).

In the following illustrated example, two shared disks connect two systems. Each system uses a separate controller for each disk.



Allocation of Heartbeat Disk Regions

The illustrated configuration is specified in a `/etc/gabtab` file that resembles:

```
/sbin/gabdiskhb -a /dev/dsk/c1t3d0s2 -s 16 -p a
/sbin/gabdiskhb -a /dev/dsk/c1t3d0s2 -s 144 -p h
/sbin/gabdiskhb -a /dev/dsk/c2t3d0s2 -s 16 -p a
/sbin/gabdiskhb -a /dev/dsk/c2t3d0s2 -s 144 -p h
/sbin/gabconfig -c -n2
```

The `-s` option to the `gabdiskhb` command specifies the start location of each 128-block region.

The `-p` option specifies the port: the value "`a`" specifies the GAB control port, and the value "`h`" specifies the VCS port.

The regions should not overlap. Two adjacent regions must have starting blocks separated by 128 blocks.

Usually, the first 16 blocks of the first partition of the disk are reserved. If the partition you are using is not the first partition on the disk, the start locations may be 0 and 128.

Note the following considerations when configuring heartbeat disk regions.

- A disk partition containing a heartbeat region cannot be used for any other purpose, such as a file system or volume.
- If a disk containing heartbeat regions is also used for other purposes, the traffic could adversely affect performance of the heartbeating.

The /etc/gabtab file is used at startup to create the regions on the disk. Reboot each system to implement the configuration. After the system starts up, you can display the configured heartbeat regions by entering:

```
# /sbin/gabdiskhb -l

Port  Disk                Major   Minor    Start    Active
=========================================================
a     /dev/dsk/c1t3d0s2   37      8        16       01
h     /dev/dsk/c1t3d0s2   37      8        144      01
a     /dev/dsk/c2t3d0s2   37      7        16       01
h     /dev/dsk/c2t3d0s2   37      7        144      01
```

Adding GAB Disk Region Signatures (Optional) for Integrity

To guarantee the integrity of the GAB disk region, GAB can be directed to verify a signature in that region on a periodic basis. This optional feature ensures that valuable data on the disk, such as a filesystem, is not accidentally overwritten.

You can use the `gabdiskconf`(1M) command to initialize the region with the specified signature. This must be done before the gabdiskhb command is run manually or from the `/etc/gabtab` file during boot.

Example, Configuring and Checking for a Signature

In the following example, GAB disk regions are initialized by assigning signatures.

```
gabdiskconf -i /dev/dsk/c1t1d2s3 -s 16 -S 1123
gabdiskconf -i /dev/dsk/c1t1d2s3 -s 144 -S 1124
```

The disk regions, starting at block 16 and 144 of the block device `/dev/dsk/c1t1d2s3`, are assigned the 4-byte strings of 1123 and 1124, respectively, as signatures.

Later, the regions are configured as heartbeating regions by the `gabdiskhb` command. In the following example, the `gabdiskhb` command specifies that GAB check the signatures on a periodic basis.

```
gabdiskhb -a /dev/dsk/c1t1d2s3 -s 16 -p a -S 1123
```

```
gabdiskhb -a /dev/dsk/c1t1d2s3 -s 144 -p h -S 1124
```

If GAB determines that a signature does not match the user's specified value, it marks the disk as faulted.

Initializing File Systems and Disk Groups on Shared Storage

In addition to the shared disk partitions used for VCS communications, your configuration may include disks on the shared bus that contain VERITAS Volume Manager™ (VxVM) disk groups or file systems.

For VxVM configurations, install VxVM as instructed in the *Storage Foundation Installation Guide*. Disks on the shared bus must be configured into disk groups other than `rootdg`. Create disk groups on one system only. VCS deports and imports them onto the other system as necessary. Similarly, use `mkfs` to make shared file systems from one system only. They are mounted on other systems by VCS as necessary.

Note Do not add exported file systems to `/etc/vfstab` or `/etc/dfs/dfstab`. VCS mounts and exports these file systems automatically.

Configuring Heartbeat Disk Regions on VxVM Disks

Heartbeat disk regions and service group heartbeat disk regions can coexist on a disk controlled by VxVM. However, these disk regions cannot be configured on VxVM volumes, and must be configured instead on the block ranges of the underlying physical device. The space for these partitions must be allocated before a disk is initialized by the VxVM.

Follow the steps below to prepare a disk for VCS communication and VxVM storage:

1. Install VxVM as instructed in the *Storage Foundation Installation Guide*.
2. Identify the disk by its VxVM tag name, for example, `c1t1d0`.
3. If the disk contains data, migrate the data to another storage media.
   a. Unmount all file systems on the disk.
   b. Remove any volumes, plexes, or subdisks from the disk.
   c. Remove the disk from any active disk group or deport its disk group.

4. Allocate a VCS partition on the disk. Type:

   ```
   # /opt/VRTSvcs/bin/hahbsetup disk_tag
   ```

   Enter **y** when prompted. The `hahbsetup` command sets up disk communication for VxVM and VCS. The variable *`disk_tag`* refers to the name you identified in step 2. For example:

   ```
   # /opt/VRTSvcs/bin/hahbsetup c1t1d0
   ```

   Output resembles:

   ```
   The hadiskhb command is used to set up a disk for combined use
   by VERITAS Volume Manager and VERITAS Cluster Server for disk
   communication.
   WARNING: This utility will destroy all data on c1t1d0
   Have all disk groups and file systems on disk c1t1d0 been either
   unmounted or deported? y
   There are currently slices in use on disk /dev/dsk/c1t1d0s2
   Destroy existing data and reinitialize disk? y
   1520 blocks are available for VxCS disk communication and
   service group heartbeat regions on device /dev/dsk/c1t1d0s7
   This disk can now be configured into a Volume Manager disk
   group. Using vxdiskadm, allow it to be configured into the disk
   group as a replacement disk. Do not select reinitialization of
   the disk.
   After running vxdiskadm, consult the output of prtvtoc to
   confirm the existence of slice 7. Reinitializing the disk
   under VxVM will delete slice 7. If this happens, deport the disk
   group and rerun hahbsetup.
   ```

5. The disk should now be initialized, even though it has not been added to a disk group. To add the disk to a disk group, run the `vxdg addisk` command (refer to the `vxdg`(1M) manual page for more information). For example, after running `hahbsetup` to allocate a VCS partition on `c1t1d0`, add `c1t1d0` to the `sharedg` disk group as `disk01` by typing the following command:

   ```
   # vxdg -g sharedg adddisk disk01=c1t1d0
   ```

6. Display the partition table. Type:

   ```
   # prtvtoc /dev/dsk/disk_tags0
   ```

   For example:

   ```
   # prtvtoc /dev/dsk/c1t1d0s0
   ```

   Output resembles:

Partition	Tag	Flags	First Sector	Sector Count	Last Sector	Mount Directory
2	5	01	0	8887440	8887439	
3	15	01	0	1520	1519	
4	14	01	3040	8884400	8887439	
7	13	01	1520	1520	3039	

7. Confirm that slice 7 exists and that its tag is 13.

8. Configure partition `/dev/dsk/c1t1d0s7` into VCS.

Configuring VCS

Configuration of VCS requires two files: `types.cf` and `main.cf` on each system in the cluster. Both of the files are located in the `/etc/VRTSvcs/conf/config` directory.

The `main.cf` configuration file requires the following minimum essential elements:

- An "`include`" statement that specifies the file, `types.cf`, which defines the VCS bundled agent resources.
- The name of the cluster.
- The name of the systems that make up the cluster.

Editing the main.cf File

When you use `pkgadd` to install VCS, the file `/etc/VRTSvcs/conf/config/main.cf` contains only the line:

```
include "types.cf"
```

1. Log in as superuser, and move to the directory containing the configuration file:

   ```
   # cd /etc/VRTSvcs/conf/config
   ```

2. Using `vi`, or another text editor, edit the `main.cf` file, defining your cluster name and system names (refer to the example below).

   ```
   # vi main.cf
   ```

3. Save and close the file.

Example, main.cf

```
include "types.cf"
cluster VCSCluster2 ( )
system north
system south
```

Starting LLT

To start LLT, on each system, type:

```
# /etc/rc2.d/S70llt start
```

If LLT is configured correctly on each system, the console output resembles:

```
Apr  5 14:46:18 north llt: LLT:10009: LLT Protocol available
```

To verify LLT is operating, see "Verifying LLT" on page 102.

Starting GAB

To start GAB, on each system, type:

```
# /etc/rc2.d/S92gab start
```

If GAB is configured correctly on each system, the console output resembles:

```
Apr  5 14:46:29 north gab: GAB:20021: GAB available
Apr  5 14:51:50 north gab: GAB:20026: Port a registration
waiting for seed port membership
```

To verify GAB is operating, see "Verifying GAB" on page 104.

Starting VCS

To start VCS, on each system, type:

```
# /etc/rc3.d/S99vcs start
```

If VCS is configured correctly on each system, the console output resembles:

```
Apr  5 14:51:52 north qfe: SUNW,qfe0: 100 Mbps full duplex link up
  - internal  transceiver
Apr  5 14:51:52 north qfe: SUNW,qfe1: 100 Mbps full duplex link up
  - internal  transceiver
Apr  5 14:51:52 north llt: LLT:10024: link 0 (qfe0) node 0 active
Apr  5 14:51:52 north llt: LLT:10024: link 1 (qfe1) node 0 active
VCS:10619:'HAD' starting on: north
VCS:10620:Waiting for local cluster configuration status
VCS:10625:Local cluster configuration valid
VCS:11034:registering for cluster membership
Apr  5 14:51:55 north gab: GAB:20005: Port h registration waiting
  for seed port membership
VCS:11035:Waiting for cluster membership
Apr  5 14:51:57 north gab: GAB:20036: Port a gen 4570ae01
  membership 01
Apr  5 14:52:02 north gab: GAB:20036: Port h gen 3972a201
  membership 01
VCS:10077:received new cluster membership
VCS:10075:building from remote system
VCS:10066:entering RUNNING state
```

To verify VCS is operating, see "Verifying the Cluster" on page 105.

Modifying the VCS Configuration

After the successful installation of VCS, you can modify the configuration of VCS using several methods. You can dynamically modify the configuration by using the command line, the VCS Cluster Manager (Web Console), or Cluster Manager (the VCS Java GUI). Refer to the *VERITAS Cluster Server User's Guide* for information on using the Web Console and the Java Console.

You can also edit the `main.cf` file directly. See the *VERITAS Cluster Server User's Guide* for information on the structure of the `main.cf` file.

Configuring the ClusterService Group

When you have successfully installed VCS, and verified that LLT, GAB, and VCS are working correctly, you can create a service group to include the optional features including the Web Console, the VCS notification components, and the Global Cluster option. If you used `pkgadd` to add VCS to your cluster systems, you must create the `ClusterService` group manually. For reference, you can see the "main.cf Example, for Clusters Without the GCO Option" on page 100 for an example of a system configured with a `ClusterService` group.

Replacing a VCS Demo License with a Permanent License

When a VCS demonstration key license expires, you can replace it with a permanent license using the `vxlicinst`(1) utility. Replace a demo key using the following procedure:

1. Make sure you have permissions to log in as root on each of the systems in the cluster.

2. Shut down VCS on all systems in the cluster:

   ```
   # hastop -all -force
   ```

   This does not shut down any running applications.

3. Enter the permanent license key using the following command on *each* system:

   ```
   # cd /opt/VRTS/bin
   # ./vxlicinst -k xxxx-xxxx-xxxx-xxxx-xxxx-xxx
   ```

Note Make sure demo licenses are replaced on all cluster systems before starting VCS.

4. Start VCS on each node:

   ```
   # hastart
   ```

Removing VCS Packages Using pkgrm

1. Shut down VCS on the local system using the `hastop`(1m) command.

   ```
   # hastop -local
   ```

2. Unconfigure the GAB and LLT utilities.

   ```
   # /sbin/gabconfig -U
   # /sbin/lltconfig -U
   ```

3. Unload the GAB and LLT modules from the kernel.

   a. Determine the kernel module IDs:

      ```
      # modinfo | grep gab
      # modinfo | grep llt
      ```

      The module IDs are in the left-hand column of the output.

   b. Unload the module from the kernel:

      ```
      # modunload -i gab_id
      # modunload -i llt_id
      ```

4. Use `pkgrm` to remove the VCS 4.1 packages in the following order:

   ```
   # pkgrm VRTScscm
   # pkgrm VRTSvcsw
   # pkgrm VRTSweb
   # pkgrm VRTScscw
   # pkgrm VRTScssim
   # pkgrm VRTScutil
   # pkgrm VRTSjre
   # pkgrm VRTSvcsdc
   # pkgrm VRTSvcsmn
   # pkgrm VRTSvcsag
   # pkgrm VRTSvcsmg
   # pkgrm VRTSvcs
   # pkgrm VRTSvxfen
   # pkgrm VRTSgab
   # pkgrm VRTSllt
   # pkgrm VRTSperl
   ```


5. Use pkgrm to remove the VCS 4.1 language packages in the following order:

```
# pkgrm VRTSjacsw
# pkgrm VRTSjacsu
# pkgrm VRTSjacsj
# pkgrm VRTSjacsd
# pkgrm VRTSjacs
# pkgrm VRTSjaweb
# pkgrm VRTSmulic
```

6. Remove the optional language package:

```
# pkgrm VRTSjacsm
```

7. On each system, perform step 1 through step 4 on each system to uninstall VCS and step 5 and step 6 to remove the language packages.

Verifying the Installation of VCS 4.1 5

After successfully installing VCS, you can inspect the contents of the key configuration files that have been installed and modified during the process. These files reflect the configuration based on the information you supplied.

Verifying LLT and GAB Configuration Files

The following files are required by the VCS communication services, LLT (Low Latency Transport) and GAB (Group Membership and Atomic Broadcast).

/etc/llthosts

The file `llthosts`(4) is a database, containing one entry per system, that links the LLT system ID (in the first column) with the LLT host name. This file is identical on each system in the cluster.

For example, the file `/etc/llthosts` contains entries that resemble:

```
0 north
1 south
```

/etc/lltab

The file `lltab`(1M) contains information that is derived during installation and used by the utility `lltconfig`(1M). After installation, this file lists the private network links that correspond to the specific system.

For example, the file `/etc/lltab` contains entries that resemble:

```
set-node north
set-cluster 2
link qfe:0 /dev/qfe:0 - ether - -
link qfe:1 /dev/qfe:1 - ether - -
```


The first line identifies the system. The second line identifies the cluster—the cluster ID you entered during installation. The next two lines, beginning with the `link` command, identify the two network cards used by the LLT protocol.

Refer to the `lltab`(4) manual page for details about how to modify the LLT configuration. The manual page describes the ordering of the directives in the `lltab` file.

/etc/gabtab

After installation, the file `/etc/gabtab` contains a `gabconfig`(1) command that configures the GAB driver for use.

The file `/etc/gabtab` contains a line that resembles:

```
/sbin/gabconfig -c -nN
```

where the `-c` option configures the driver for use and `-n`*N* specifies that the cluster is not formed until at least *N* systems are ready to form the cluster. By default, *N* is the number of systems in the cluster.

Note VERITAS does not recommend the use of the `-c -x` option for `/sbin/gabconfig`. Using `-c -x` dramatically increases configuration time for the Gigabit Ethernet controller and can lead to a split-brain condition.

Verifying the main.cf File

The VCS configuration file `/etc/VRTSvcs/conf/config/main.cf` is created during the installation process. Examples are shown on the next pages. The `main.cf` file contains the minimum information that defines the cluster and its systems. In addition, the file `types.cf`, which is listed in the `include` statement, defines the VCS bundled types for VCS resources. The file `types.cf` is also located in the directory `/etc/VRTSvcs/conf/config` after installation.

Notice that the cluster has an attribute `UserNames`. The `installvcs` program creates a user "admin" whose password is encrypted; the word "password" is the password.

With the information you provide, `installvcs` configures the VCS Cluster Manager (Web Console) into a service group, `ClusterService`, that includes the IP, NIC, and VRTSWebApp resources. The service group also includes the notifier resource configuration, which is based on your input to `installvcs` prompts about notification. A resource dependency tree has also been created.

If you have installed VCS with the Global Cluster Option, the `ClusterService` service group contains an Application resource, `wac` (wide-area connector), whose attributes contain definitions for controlling the cluster in a Global Cluster environment. Refer to the *VERITAS Cluster Server User's Guide* for information about managing clusters that use the Global Cluster option.

Refer to the *VERITAS Cluster Server User's Guide* and review the chapter on configuration concepts for descriptions and examples of `main.cf` and `types.cf` files for UNIX systems.

main.cf Example, for Clusters Without the GCO Option

```
include "types.cf"

cluster VCSCluster2 (
        UserNames = { admin = cDRpdxPmHpzS, smith = dKLhKJkHLh }
        ClusterAddress = "11.136.88.199"
        Administrators = { admin, smith }
        CounterInterval = 5
        )

system north (
        )

system south (
        )

group ClusterService (
        SystemList = { north = 0, south = 1 )
        UserStrGlobal = "LocalCluster@https://10.182.2.76:8443;"
        AutoStartList = { north, south )
        OnlineRetryLimit = 3
        OnlineRetryInterval = 120
        )

        IP webip (
               Device = hme0
               Address = "11.136.88.199"
               NetMask = "255.255.240.0"
               )

        NIC csgnic (
               Device = hme0
               )

        NotifierMngr ntfr (
               SnmpConsoles = { "saturn" = Error, "jupiter" =
                    SevereError }
               SmtpServer = "smtp.xyzstar.com"
               SmtpRecipients =  { "ozzie@xyzstar.com" =
                   Warning,"harriet@xyzstar.com" = Error }
               )

        VRTSWebApp VCSweb (
               Critical = 0
               AppName = vcs
               InstallDir = "/opt/VRTSweb/VERITAS"
               TimeForOnline = 5
               RestartLimit = 3
               )
```

```
        VCSweb requires webip
        ntfr requires csgnic
        webip requires csgnic

// resource dependency tree
        //
        //      group ClusterService
        //      {
        //      VRTSWebApp VCSweb
        //          {
        //          IP webip
        //              {
        //              NIC csgnic
        //              }
        //          }
        //      NotifierMngr ntfr
        //          {
        //          NIC csgnic
        //          }
        //      }
```

main.cf Example, for Clusters With the GCO Option

If you have installed VCS with the Global Cluster option, note that the `ClusterService` group also contains the Application resource, `wac`, required to control the cluster in a Global Cluster environment.

```
.
.
group ClusterService (
        SystemList = { north = 0, south = 1 }
        UserStrGlobal = "LocalCluster@https://10.182.2.78:8443;"
        AutoStartList = { north, south }
        OnlineRetryLimit = 3
        OnlineRetryInterval = 120
        )

        Application wac (
                StartProgram = "/opt/VRTSvcs/bin/wacstart"
                StopProgram = "/opt/VRTSvcs/bin/wacstop"
                MonitorProcesses = { "/opt/VRTSvcs/bin/wac" }
                RestartLimit = 3
                )
.
.
```


Verifying LLT, GAB, and Cluster Operation

Before attempting to verify the operation of LLT, GAB, or the cluster, you must:

- ✔ Log in to any system in the cluster as `root`.
- ✔ Place the VCS command directory in your `PATH` variable:

```
# export PATH=$PATH:/usr/sbin:/sbin:/opt/VRTS/bin:/opt/VRTSvcs/bin
```

Note If you are using SUN SCI adapters for your private network, move the scripts `S70llt` and `S92gab` from the directory `/etc/rc2.d` to directory `/etc/rc3.d` so that they are run after the `S19sci` and `S23scid` scripts.

Verifying LLT

Use the `lltstat` command to verify that links are active for LLT. This command returns information about the links for LLT for the system on which it is typed. Refer to the `lltstat`(1M) manual page for more information.

Using lltstat -n

In the following example, type `lltstat -n` on each system in the cluster:

System 1

```
# lltstat -n
```

Output resembles:

```
LLT node information:
     Node      State     Links
    *0 north   OPEN      2
     1 south   OPEN      2
```

System 2

```
# lltstat -n
```

Output resembles:

```
LLT node information:
     Node      State     Links
     0 north   OPEN      2
    *1 south   OPEN      2
```

Note that each system has two links and that each system is in the OPEN state. The asterisk (*) denotes the system on which the command is typed.

Using lltstat -nvv

With LLT configured correctly, the output of `lltstat -n` shows all the systems in the cluster and two links for each system. If the output shows otherwise, you can use the verbose option of `lltstat`. Type `lltstat -nvv | more` on a system to view additional information about LLT.

For example, type the command on a system in a two-system cluster:

```
# lltstat -nvv | more
```

The output resembles:

```
Node         State    Link          Status   Address
*0 north     OPEN
                      qfe:0         UP       08:00:20:93:0E:34
                      qfe:1         UP       08:00:20:93:0E:35
1 south      OPEN
                      qfe:0         UP       08:00:20:8F:D1:F2
                      qfe:1         DOWN
2            CONNWAIT
                      qfe:0         DOWN
                      qfe:1         DOWN
             CONNWAIT
                      qfe:0         DOWN
                      qfe:1         DOWN
.
.
.
1            CONNWAIT
                      qfe:0         DOWN
                      qfe:1         DOWN
```

Note that the output lists 32 nodes. It reports on the two cluster systems, `north` and `south`, plus non-existent nodes. For each correctly configured system, the information should show a state of OPEN, a status for each link of UP, and an address for each link. However, the output in the example shows that for the system `south` the private network connection is possibly broken or the information in the `/etc/lltab` file may be incorrect.

To obtain information about the ports open for LLT, type `lltstat -p` on any system. In the following example, `lltstat -p` is typed on one system in a two-system cluster:

System 1

```
# lltstat -p
```

Output resembles:

```
LLT port information:
    Port   Usage     Cookie
      0    gab       0x0
        opens:       0 2 3 4 5 6 7 8 9 10 11 … 28 29 30 31
        connects:    0 1
      7    gab       0x7
        opens:       0 2 3 4 5 6 7 8 9 10 11 … 28 29 30 31
        connects:    0 1
      31   gab       0x1F
        opens:       0 2 3 4 5 6 7 8 9 10 11 … 28 29 30 31
        connects:    0 1
```

Verifying GAB

To verify that GAB is operating, type the following command on each system:

```
# /sbin/gabconfig -a
```

If GAB is operating, the following GAB port membership information returns:

```
GAB Port Memberships
===================================
Port a gen a36e0003 membership 01
Port h gen fd570002 membership 01
```

`Port a` indicates that GAB is communicating, `gen a36e0003` is a random generation number, and `membership 01` indicates a connection between systems `0` and `1`.

`Port h` indicates that VCS is started, `gen fd570002` is a random generation number, and `membership 01` indicates that systems `0` and `1` are both running VCS.

If GAB is not operating, no GAB port membership information returns:

```
GAB Port Memberships
===================================
```

If only one private network link is connected, the following GAB port membership information returns:

```
GAB Port Memberships
===================================
Port a gen a36e0003 membership 01
Port a gen a36e0003 jeopardy    1
Port h gen fd570002 membership 01
Port h gen fd570002 jeopardy    1
```

For more information on GAB, refer to the *VERITAS Cluster Server User's Guide*.

Verifying the Cluster

To verify that the cluster is operating, type the following command:

```
# hastatus -summary
```

The output resembles:

```
-- SYSTEM STATE
-- System               State                Frozen

A  north                RUNNING              0
A  south                RUNNING              0

-- GROUP STATE
-- Group            System       Probed  AutoDisabled  State

B  ClusterService   north        Y       N             ONLINE
B  ClusterService   south        Y       N             OFFLINE
```

Note the SYSTEM STATE. If the value is RUNNING, VCS is successfully installed and running. The GROUP STATE lists the ClusterService group, which is ONLINE on north and OFFLINE on south. Refer to the hastatus(1M) manual page. In the *VERITAS Cluster Server User's Guide*, look for a description of system states and the transitions between them.

hasys -display

On one of the systems, use the `hasys`(1M) command:

```
# /opt/VRTSvcs/bin/hasys -display
```

On each system, the output should be similar. For more information on the `hasys -display` command, refer to the `hasys`(1M) manual page. Also refer to the *VCS Cluster Server User's Guide* for information about administering VCS from the command line.

The example on the this page shows the output the `hasys -display` command run on the system `north`; the list continues with similar information for `south` (not shown) and any other systems in the cluster:

```
#System    Attribute           Value
north      AgentsStopped       0
north      AvailableCapacity   100
north      CPUBinding          BindTo None CPUNumber 0
north      CPUUsage            0
north      CPUUsageMonitoring  Enabled 0 ActionThreshold 0
                               ActionTimeLimit 0 Action NONE
                               NotifyThreshold 0 NotifyTimeLimit 0
north      Capacity            100
north      ConfigBlockCount    100
north      ConfigCheckSum      29776
north      ConfigDiskState     CURRENT
north      ConfigFile          /etc/VRTSvcs/conf/config
north      ConfigInfoCnt       0
north      ConfigModDate       Mon Aug 11 23:00:00 2004
north      CurrentLimits
north      DiskHbStatus
north      DynamicLoad         0
north      EngineRestarted     0
north      Frozen              0
north      GUIIPAddr
north      LLTNodeId           0
north      LicenseType         PERMANENT SITE
north      Limits
north      LinkHbStatus        qfe:0 UP qfe:1 UP
north      LoadTimeCounter     1890
north      LoadTimeThreshold   600
north      LoadWarningLevel    80
north      MajorVersion        4
north      MinorVersion        0
north      NoAutoDisable       0
north      NodeId              0
north      OnGrpCnt            1
```

```
north       ShutdownTimeout     120
north       SourceFile          ./main.cf
north       SysInfo             Solaris:north,Generic_108528-04,5.8,su
                                n4u
north       SysName             north
north       SysState            RUNNING
north       SystemLocation
north       SystemOwner
north       TFrozen2            0
north       TRSE                0
north       UpDownState         Up
north       UserInt             0
north       UserStr
north       VCSFeatures         DR
north       VCSMode             VCS
```

Accessing the VCS Cluster Manager (Web Console)

The VCS Web-based Cluster Manager (Web Console) enables you to monitor the cluster from any workstation on the public network. Supported browsers are Netscape Navigator 4.1 or later, or Internet Explorer 4.1 or later.

When VCS starts running in the cluster and the `ClusterService` group comes up, the Web Console server starts. To access the Web Console:

1. From the browser, navigate to the Web Console by entering:

   `http://`*`web_gui_IP_address`***`:8181/vcs`**

   For example:

   `http://10.129.96.64:8181/vcs`

   The IP address is the “Cluster virtual IP address” configured into the `ClusterService` group.

2. On the Login screen, enter a valid user name and password. By default, the administrator of a new installation can log in as “admin” and use “password” as a password. For security, change your password at your earliest convenience.

3. Click Login to enter the Cluster Summary view.

Accessing the VCS Documentation

The directory `/opt/VRTS/docs` contains the documentation for VCS in Portable Document Format (PDF). The directory contains the following documents:

- `vcs_users.pdf` (*VERITAS Cluster Server User's Guide)*
- `vcs_bundled_agents.pdf` (*VERITAS Cluster Server Bundled Agents Reference Guide)*
- `vcs_agent_dev.pdf` (*VERITAS Cluster Server Agent Developer's Guide)*
- `vcs_appnote_e10k.pdf` (*VERITAS Cluster Server Application Note: Sun Enterprise 10000 Dynamic Reconfiguration)*
- `vcs_appnote_f15k.pdf` (*VERITAS Cluster Server Application Note: Sun 12K/15K Dynamic Reconfiguration)*
- `vcs_appnote_s6800.pdf` (*VERITAS Cluster Server Application Note: Sun Fire 6800 Dynamic Reconfiguration)*

Installing the VCS Java Console

You can administer VCS using the VCS Java-based graphical user interface, Java Console. After installing VCS, install the Java Console on a UNIX system with X-Windows, or on a Windows NT, Windows 2000 Professional, Window XP, or Windows 2003 system. The system from which you run the Java Console can be a system in the cluster or a remote workstation; the latter enables each system in the cluster to be administered remotely.

For information about using the Cluster Manager and the Configuration Editor components of the Java Console, see the applicable chapter in the *VERITAS Cluster Server User's Guide*.

When installing the Java Console on a Solaris system, make sure a printer is configured to that system. On a system without a configured printer, printing from the online JavaHelp could cause the Java Console to hang.

Installing the Java Console on UNIX (Solaris)

1. Create a directory for installation of the Java Console:

   ```
   # mkdir /tmp/install
   ```

2. Insert the software disc with the VCS software into a drive connected to the system. The Solaris volume-management software automatically mounts the disc as `/cdrom/cdrom0`. Type the command:

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

3. Copy the compressed package files from the software disc to the temporary directory:

```
# cp -r cluster_server/pkgs/VRTScscm* /tmp/install
```

4. Go to the temporary directory and unzip the compressed package file:

Note If your system does not have the `gunzip` utility, copy it from the disc:
`# cp /cdrom_path/gnu/gunzip /tmp/install`

```
# cd /tmp/install
# gunzip VRTScscm.tar.gz
```

The file `VRTScscm.tar` is now present in the temporary directory.

5. Extract the compressed file from the tar file:

```
# tar -xvf VRTScscm.tar
```

6. Install the software:

```
# pkgadd -d . VRTScscm
```

7. Answer **Yes** if prompted.

Installing the Java Console on a Windows System

If you are installing the VCS Java Console (Cluster Manager) on a Windows NT, Windows 2000 Professional, Windows XP, or Windows 2003 system to administer the cluster, do the following:

1. Insert the software disc with the VCS software into a drive on your Windows system.
2. Using Windows Explorer, select the CD drive.
3. Go to `\windows\WindowsInstallers\WindowClusterManager\EN`.
4. Double-click `setup.exe`.
5. The VERITAS Cluster Manager Install Wizard guides you through the installation process.

Setting Up I/O Fencing 6

This chapter describes:

- VCS I/O fencing, its components, and how it works. See "I/O Fencing" on page 112.
- Procedures for setting up I/O fencing, including:
  - Requirements for using I/O fencing (see "Setting Up Shared Storage for I/O Fencing" on page 115)
  - Adding disks (see "Adding Disks" on page 115)
  - Testing the data disks for SCSI-3 Persistent Reservations (an I/O fencing requirement; see "Testing Data Storage Disks Using vxfentsthdw" on page 116)
  - Setting up coordinator disks (see "Setting Up Coordinator Disks" on page 120)
  - Configuring the coordinator disk group (see "Setting Up the Disk Group for Coordinator Disks" on page 121)
  - Testing the I/O fencing coordinator disk group for SCSI-3 Persistent Reservations (see "Requirements for Testing the Coordinator Disk Group" on page 122)
  - Enabling I/O fencing (see "Creating /etc/vxfendg to Configure the Disk Group for Fencing" on page 124)

In addition, the chapter provides:

- Troubleshooting information (see "I/O Fencing" on page 112)
- Information for testing many data disks (with the `vxfentsthdw` utility), whether they are set up in disk groups or listed in a file (see "vxfentsthdw Options" on page 134)
- Scenarios in which I/O fencing functions to prevent data corruption (see "How I/O Fencing Works in Different Event Scenarios" on page 137)
- A description of the `vxfenadm` command, which you can use to test and troubleshoot I/O fencing configurations (see "The vxfenadm Utility" on page 140)
- Information about the vxfen tunable parameters (see "VXFEN Tunable Parameters" on page 141)

I/O Fencing

I/O fencing is a feature within a kernel module of VCS designed to guarantee data integrity, even in the case of faulty cluster communications causing a split brain condition.

Understanding Split Brain and the Need for I/O Fencing

Split brain is an issue faced by all cluster solutions. To provide high availability, the cluster must be capable of taking corrective action when a node fails. In VCS, this is carried out by the reconfiguration of CVM and CFS to change membership. Problems arise when the mechanism used to detect the failure of a node breaks down. The symptoms look identical to a failed node. For example, if a system in a two-node cluster were to fail, it would stop sending heartbeats over the private interconnects and the remaining node would take corrective action. However, the failure of the private interconnects would present identical symptoms. In this case, both nodes would determine that their peer has departed and attempt to take corrective action. This typically results in data corruption when both nodes attempt to take control of data storage in an uncoordinated manner.

In addition to a broken set of private networks, other scenarios can cause this situation. If a system were so busy as to appear hung, it would be declared dead. This can also happen on systems where the hardware supports a “break” and “resume” function. Dropping the system to PROM level with a break and subsequently resuming means the system could be declared as dead, the cluster could reform, and when the system returns, it could begin writing again.

VCS uses a technology called I/O fencing to remove the risk associated with split brain. I/O fencing blocks access to storage from specific nodes. This means even if the node is alive, it cannot cause damage.

SCSI-3 Persistent Reservations

VCS uses an enhancement to the SCSI specification, known as SCSI-3 Persistent Reservations, (SCSI-3 PR). SCSI-3 PR is designed to resolve the issues of using SCSI reservations in a modern clustered SAN environment. SCSI-3 PR supports multiple nodes accessing a device while at the same time blocking access to other nodes. SCSI-3 reservations are persistent across SCSI bus resets and SCSI-3 PR also supports multiple paths from a host to a disk.

SCSI-3 PR uses a concept of registration and reservation. Systems wishing to participate register a “key” with a SCSI-3 device. Each system registers its own key. Multiple systems registering keys form a membership. Registered systems can then establish a reservation. This is typically set to “Write Exclusive Registrants Only” (WERO). This means registered systems can write, and all others cannot. For a given disk, there can only be one reservation, while there may be many registrations.

With SCSI-3 PR technology, blocking write access is as simple as removing a registration from a device. Only registered members can "eject" the registration of another member. A member wishing to eject another member issues a "*preempt and abort*" command that ejects another node from the membership. Nodes not in the membership cannot issue this command. Once a node is ejected, it cannot in turn eject another. This means ejecting is final and "atomic."

In the VCS implementation, a node registers the same key for all paths to the device. This means that a single preempt and abort command ejects a node from all paths to the storage device.

To summarize:

- Only a registered node can eject another
- Since a node registers the same key down each path, ejecting a single key blocks all I/O paths from the node
- Once a node is ejected, it has no key registered and it cannot eject others

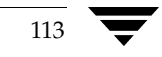
The SCSI-3 PR specification simply describes the method to control access to disks with the registration and reservation mechanism. The method to determine who can register with a disk and when a registered member should eject another node is implementation specific. The following paragraphs describe VCS I/O fencing components and implementation.

I/O Fencing Components

I/O fencing, or simply fencing, allows write access to members of the active cluster and blocks access to non-members. I/O fencing in VCS uses several components. The physical components are *coordinator* disks and *data* disks. Each has a unique purpose and uses different physical disk devices.

Data Disks

Data disks are standard disk devices used for data storage. These can be physical disks or RAID Logical Units (LUNs). These disks must support SCSI-3 PR. Data disks are incorporated in standard VxVM/CVM disk groups. In operation, CVM is responsible for fencing data disks on a disk group basis. Since VxVM enables I/O fencing, several other features are provided. Disks added to a disk group are automatically fenced, as are new paths discovered to a device.



Coordinator Disks

Coordinator disks are special purpose disks in a VCS environment. Coordinator disks are three (or an odd number greater than three) standard disks, or LUNs, set aside for use by I/O fencing during cluster reconfiguration. These disks must support SCSI-3 PR.

The coordinator disks act as a global lock device during a cluster reconfiguration. This lock mechanism is used to determine who gets to fence off data drives from other nodes. From a high level, a system must eject a peer from the coordinator disks before it can fence the peer from the data drives. This concept of racing for control of the coordinator disks to gain the capability to fence data disks is key to understanding the split brain prevention capability of fencing.

Coordinator disks cannot be used for any other purpose in the VCS configuration. The user must not store data on these disks, or include the disks in a disk group used by user data. The coordinator disks can be any three disks that support SCSI-3 PR. VERITAS typically recommends the smallest possible LUNs for coordinator use. Since coordinator disks do not store any data, cluster nodes need only register with them and do not need to reserve them.

I/O Fencing Operation

I/O fencing provided by the kernel-based fencing module (VXFEN) performs identically on node failures and communications failures. When the fencing module on a node is informed of a change in cluster membership by the GAB module, it immediately begins the fencing operation. The node immediately attempts to eject the key for departed node(s) from the coordinator disks using the preempt and abort command. When the node has successfully ejected the departed nodes from the coordinator disks, it ejects the departed nodes from the data disks. If this were a split brain scenario, both sides of the split would be "racing" for control of the coordinator disks. The side winning the majority of the coordinator disks wins the race and fences the loser. The loser then panics and reboots.

The *VERITAS Cluster Server User's Guide* describes I/O fencing concepts in detail.

Setting Up Shared Storage for I/O Fencing

Note that to use I/O fencing you must:

- ✔ Have installed the `VRTSvxfen` package when you installed VCS
- ✔ Have installed a version of VERITAS Volume Manager (VxVM) that supports SCSI-3 persistent reservations (SCSI-3 PR). Refer to the installation guide accompanying the Storage Foundation product you are using.

The shared storage you add for use with VCS software must support SCSI-3 persistent reservations, a functionality that enables the use of I/O fencing.

Adding Disks

After you physically add shared disks to cluster systems, you must initialize them as VxVM disks. Use the following examples. The *VERITAS Volume Manager Administrator's Guide* has more information about adding and configuring disks.

1. The disks you add can be listed by the command:

   ```
   # lsdev -Cc disk
   ```

2. Use the `vxdisk scandisks` command to scan all disk drives and their attributes, to update the VxVM device list, and to reconfigure DMP with the new devices. For example:

3. `# vxdisk scandisks`

4. To initialize the disks as VxVM disks, use either of two methods:

   a. Use the interactive `vxdiskadm` utility to initialize the disks as VxVM disks. As you answer prompts while running the utility, we recommend specifying that the disks support Cross-platform Data Sharing (CDS) format.

   b. You can also use the command `vxdisksetup` to initialize a disk as a VxVM disk. The example that follows specifies the CDS format:

      ```
      vxdisksetup -i device_name format=cdsdisk
      ```

      For example:

      ```
      # vxdisksetup -i /dev/rdsk/c2t0d2s2 format=cdsdisk
      ```


Verifying that Systems See the Same Disk

To perform the test that determines whether a given disk (or LUN) supports SCSI-3 persistent reservations, two systems must simultaneously have access to the same disks. Because a given shared disk is likely to have a different name on each system, a way is needed to make sure of the identity of the disk. Typically the method to check the identity of a given disk, or LUN, is to check its serial number.

You can use the `vxfenadm` command with the `-i` option to verify that the same serial number for a LUN is returned on all paths to the LUN.

For example, an EMC array is accessible by the path `/dev/rrdsk/c2t13d0s2` on node A and by the path `/dev/rdsk/c2t11d0s2r` on node B. From node A, the command is given:

```
# vxfenadm -i /dev/rdsk/c2t13d0s2
Vendor id       : EMC
Product id      : SYMMETRIX
Revision        : 5567
Serial Number   : 42031000a
```

The same serial number information should be returned when the equivalent command is given on node B using the path `/dev/rrdsk/c2t11d0s2`.

On a disk from another manufacturer, Hitachi Data Systems, for example, the output is different. It may resemble:

```
# vxfenadm -i /dev/rdsk/c2t0d2s2
Vendor id       : HITACHI
Product id      : OPEN-3      -SUN
Revision        : 0117
Serial Number   : 0401EB6F0002
```

Refer to the `vxfenadm`(1M) manual page.

Testing Data Storage Disks Using vxfentsthdw

Use the `vxfentsthdw` utility to test the shared storage arrays that are to be used for data. The utility verifies the disks support SCSI-3 persistent reservations and I/O fencing.

Note Disks used as coordinator disks must also be tested. See "Setting Up Coordinator Disks" on page 120.

General Guidelines for Using vxfentsthdw

- Connect the shared storage to be used for data to two cluster systems.

  Caution The tests overwrite and destroy data that may be present on a disk unless you use the -r option.

- The two systems must have `rsh` permission set so that each node has root user access to the other. Temporarily modify the `/.rhosts` file to enable cluster communications for the `vxfentsthdw` utility, placing a "+" character in the first line of the file. You can also limit the remote access to specific systems. Refer to the manual page for the `/.rhosts` file for more information. See "Removing rsh Permissions and Restoring Public Network Connections" on page 125 when you complete testing.
- To ensure both systems are connected to the same disk during the testing, use the `vxfenadm -i` *`diskpath`* command to verify a disk's serial number. See "Verifying that Systems See the Same Disk" on page 116.

Running vxfentsthdw

This section describes the steps required to set up and test data disks for your initial installation. It describes using the `vxfentsthdw` utility with the default options. The `vxfentsthdw` utility and it options are described in detail in the section "vxfentsthdw Options" on page 134.

The `vxfentsthdw` utility indicates a disk can be used for I/O fencing with a message resembling:

```
The disk /dev/rdsk/c4t8d0s2 is ready to be configured for I/O
Fencing on node south
```

If the utility does not show a message stating a disk is ready, verification has failed.

For the following example, assume you must check a shared device known by two systems as `/dev/rdsk/c4t8d0s2`. (Each system could use a different name for the same device.)

1. Make sure system-to-system communication is set up. See "Enabling Communication Between Systems" on page 16.

2. On one system, start the utility:

   ```
   # /opt/VRTSvcs/vxfen/bin/vxfentsthdw
   ```

   The utility begins by providing an overview of its function and behavior. It warns you that its tests overwrite any data on the disks you check:

   ```
   ******** WARNING!!!!!!!! ********
     THIS UTILITY WILL DESTROY THE DATA ON THE DISK!!

     Do you still want to continue : [y/n] (default: n)
     y
     Enter the first node of the cluster:
     north
     Enter the second node of the cluster:
     south
   ```

3. Enter the name of the disk you are checking. For each node, the disk may be known by the same name, as in our example.

   ```
     Enter the disk name to be checked for SCSI-3 PR on node north in
     the format: /dev/rdsk/cxtxdxsx
     /dev/rdsk/c4t8d0s

     Enter the disk name to be checked for SCSI-3 PR on node south in
     the format: /dev/rdsk/cxtxdxsx
     Make sure it's the same disk as seen by nodes north and south
     /dev/rdsk/c4t8d0s2
   ```

   Note the disk names, whether or not they are identical, must refer to the same physical disk.

4. The utility starts to perform the check and report its activities. For example:

```
Testing north /dev/rdsk/c4t8d0s2 south /dev/rdsk/c4t8d0s2

Registering keys on disk /dev/rdsk/c4t8d0s2 from node galaxy
.....................................................Passed.
Verifying registrations for disk /dev/rdsk/c4t8d0s2 on node
galaxy ..............................................Passed.
Reads from disk /dev/rdsk/c4t8d0s2 on node galaxy .......Passed.
Writes to disk /dev/rdsk/c4t8d0s2 from node galaxy ......Passed.
Reads from disk /dev/rdsk/c4t8d0s2 on node nebula .......Passed.
Writes to disk /dev/rdsk/c4t8d0s2 from node nebula ......Passed.
Reservations to disk /dev/rdsk/c4t8d0s2 from node galaxy ....
 .....................................................Passed.
Verifying reservation for disk /dev/rdsk/c4t8d0s2 on node
galaxy.....................................................Passed.
 .
 .
```

5. For a disk that is ready to be configured for I/O fencing on each system, the utility reports success. For example:

```
ALL tests on the disk /dev/rdsk/c4t8d0s2 have PASSED
The disk is now ready to be configured for I/O Fencing on node
north
ALL tests on the disk /dev/rdsk/c4t8d0s2 have PASSED
The disk is now ready to be configured for I/O Fencing on node
south

Cleaning up...
Removing temporary files...
Done.
```

6. Run the `vxfentsthdw` utility for each disk you intend to verify.

Note The `vxfentsthdw` utility has additional options suitable for testing many disks. The options for testing disk groups (`-g`) and disks listed in a file (`-f`) are described in detail in "vxfentsthdw Options" on page 134. You can also test disks without destroying data using the `-r` option.

Setting Up Coordinator Disks

I/O Fencing requires coordinator disks configured in a disk group accessible to each system in the cluster. The use of coordinator disks enables the `vxfen` driver to resolve potential split brain conditions and prevent data corruption. See the topic "I/O Fencing" on page 112 for a discussion of I/O fencing and the role of coordinator disks. See also "How I/O Fencing Works in Different Event Scenarios" on page 137 for additional description of how coordinator disks function to protect data in different split brain scenarios.

A coordinator disk is not used for data storage, and so may be configured as the smallest possible LUN on a disk array to avoid wasting space.

Requirements for Coordinator Disks

Coordinator disks must meet the following requirements:

- ✔ There must be at least three coordinator disks and the total number of coordinator disks must be an odd number. This ensures a majority of disks can be achieved.
- ✔ The coordinator disks must support SCSI-3 persistent reservations. See "Requirements for Testing the Coordinator Disk Group" on page 122.
- ✔ Each of the coordinator disks must use a physically separate disk or LUN.
- ✔ Each of the coordinator disks should be on a different disk array, if possible.
- ✔ Each disk must be initialized as a VxVM disk. The default (CDS) format is recommended.
- ✔ The coordinator disks must be included in a disk group. See "Setting Up the Disk Group for Coordinator Disks."

It is recommended that coordinator disks use hardware-based mirroring.

Setting Up the Disk Group for Coordinator Disks

If you have already added and initialized disks you intend to use as coordinator disks, you can begin the following procedure at step 4.

1. Physically add the three disks you intend to use for coordinator disks. Add them as physically shared by all cluster systems. It is recommended you use the smallest size disks/LUNs, so that space for data is not wasted.

2. If necessary, use the `vxdisk scandisks` command to scan the disk drives and their attributes. This command updates the VxVM device list and reconfigures DMP with the new devices. For example:

   ```
   # vxdisk scandisks
   ```

3. You can use the command `vxdisksetup` command to initialize a disk as a VxVM disk. The example command that follows specifies the CDS format:

   ```
   vxdisksetup -i device_name format=cdsdisk
   ```

   For example:

   ```
   # vxdisksetup -i /dev/rdsk/c2t0d2s2 format=cdsdisk
   ```

   Repeat this command for each disk you intend to use as a coordinator disk.

4. From one system, create a disk group for the coordinator disks (for example, `vxfencoorddg`). This group must contain an odd number of disks/LUNs and a minimum of three disks.

   For example, assume the disks have the device names `rdsk/c1t1d0s2`, `rdsk/c2t1d0s2rdsk/c1t1d0s2`, and `rdsk/c3t1d0s2rdsk/c1t1d0s2`.

   a. On any node, create the disk group by specifying the device name of one of the disks.

   b. `# vxdg init vxfencoorddg c1t1d0s2rdsk/c1t1d0s2`Add the other two disks to the disk group.

      ```
      # vxdg -g vxfencoorddg adddisk c2t1d0s2rdsk/c1t1d0s2
      # vxdg -g vxfencoorddg adddisk c3t1d0s2rdsk/c1t1d0s2
      ```

   Refer to the *VERITAS Volume Manager Administrator's Guide* for more information about creating disk groups.

Requirements for Testing the Coordinator Disk Group

- The utility requires that the coordinator disk group be accessible from two systems. For example, if you have a four-system cluster, select any two systems for the test.
- The two systems must have `rsh` permission set so that each node has root user access to the other. Temporarily modify the `/.rhosts` file to enable cluster communications for the `vxfentsthdw` utility, placing a "+" character in the first line of the file. You can also limit the remote access to specific systems. Refer to the manual page for the `/.rhosts` file for more information. See "Removing rsh Permissions and Restoring Public Network Connections" on page 125 when you complete testing.
- To ensure both systems are connected to the same disks during the testing, you can use the `vxfenadm -i` *`diskpath`* command to verify a disk's serial number. See "Verifying that Systems See the Same Disk" on page 116.

Using the vxfentsthdw -c to Test the Coordinator Disk Group

In the example that follows, the three disks are tested by the `vxfentsthdw` utility, one disk at a time from each node. From the node `north`, the disks are `rhdisk75`, `rhdisk76`, and `rhdisk77`. From the node `south`, the same disks are seen as `rhdisk65`, `rhdisk66`, `rhdisk67`.

1. Use the `vxfentsthdw` command with the `-c` option. For example:

   ```
   # /opt/VRTSvcs/vxfen/bin/vxfentsthdw -c vxfencoorddg
   ```

2. The program prompts you for the names of the systems you are using to test the coordinator disks.

   ```
   Enter the first node of the cluster:
   north
   Enter the second node of the cluster:
   south

   ********************************************
   Testing north /dev/rhdisk75 south /dev/rhdisk65

   Evaluating the disk before testing ..... pre-existing keys.
   Registering keys on disk /dev/rhdisk75 from node north.........
   Passed
   Verifying registrations for disk /dev/rhdisk75 on node north ..
   Passed.
   Registering keys on disk /dev/rhdisk65 from node south.........
   Passed.
   Verifying registrations for disk /dev/rhdisk75 on node north ..
   Passed.
   Verifying registrations for disk /dev/rhdisk65 on node south ..
   ```

```
Passed.
Preempt and aborting key KeyA using key KeyB on node south.....
Passed.
Verifying registrations for disk /dev/rhdisk75 on node north ..
Passed.
Verifying registrations for disk /dev/rhdisk65 on node south ..
Passed.
Removing key KeyB on node south.............................
Passed.
Check to verify there are no keys from node north ............
Passed.

ALL tests on the disk /dev/rhdisk75 have PASSED.
The disk is now ready to be configured for I/O Fencing on node
north as a COORDINATOR DISK.

ALL tests on the disk /dev/rhdisk65 have PASSED.
The disk is now ready to be configured for I/O Fencing on node
south as a COORDINATOR DISK.

****************************************
Testing north /dev/rhdisk75 south /dev/rhdisk65
.
.
```

The preceding shows the output of the test utility as it tests one disk. The disk group is ready for use when all disks in the disk group are successfully tested.

Removing and Adding a Failed Disk

If a disk in the coordinator disk group fails verification, remove the failed disk or LUN from the coordinator disk group, replace it with another, and retest the disk group.

- Use the `vxdiskadm` utility to remove the failed disk from the disk group. Refer to the *VERITAS Volume Manager Administrator's Guide*.
- Add a new disk to the system, initialize it, and add it to the coordinator disk group. See "Setting Up the Disk Group for Coordinator Disks" on page 121.
- Retest the disk group. See "Requirements for Testing the Coordinator Disk Group" on page 122.

Note If you need to replace a disk in an active coordinator disk group, refer to the troubleshooting procedure, "Removing or Adding Coordinator Disks" on page 132.

Creating /etc/vxfendg to Configure the Disk Group for Fencing

After you have set up and tested the coordinator disk group, configure it for use.

1. Deport the disk group. For example:

   ```
   # vxdg deport vxfencoorddg
   ```

2. Import the disk group with the `-t` option so that it is not automatically imported when the systems are restarted:

   ```
   # vxdg -t import vxfencoorddg
   ```

3. Deport the disk group again. Deporting the disk group prevents the coordinator disks from being used for other purposes.

   ```
   # vxdg deport vxfencoorddg
   ```

4. On all systems, enter the command:

   ```
   # echo "vxfencoorddg" > /etc/vxfendg
   ```

   No spaces should appear between the quotes and the text containing the name of the coordinator disk group; for example: **`"vxfencoorddg"`**

   This command creates the file `/etc/vxfendg`, which includes the name of the coordinator disk group.

   Based on the contents of the `/etc/vxfendg` file, the `rc` script creates the file `/etc/vxfentab` for use by the `vxfen` driver when the system starts. The `/etc/vxfentab` file is a generated file and should not be modified.

5. Go to "Editing VCS Configuration to Add the UseFence Attribute" on page 125 to edit the `main.cf` file and add the `UseFence = SCSI3` attribute to the VCS configuration.

Note Do *not* shut down the system at this time. Stop and restart the system after you have edited the `main.cf` file to add the `UseFence = SCSI3` attribute.

An Example /etc/vxfentab File

On each system, the coordinator disks are listed in the file `/etc/vxfentab`. The same disks may be listed using different names on each system. An example `/etc/vxfentab` file on one system resembles:

```
/dev/rdsk/c1t1d0s2
/dev/rdsk/c2t1d0s2
/dev/rdsk/c3t1d0s2
```

When the system starts, the `rc` startup script automatically creates `/etc/vxfentab` and then invokes the `vxfenconfig` command, which configures the `vxfen` driver to start and use the coordinator disks listed in `/etc/vxfentab`.

If you must remove disks from or add disks to an existing coordinator disk group, please see "Removing or Adding Coordinator Disks" on page 132.

Removing rsh Permissions and Restoring Public Network Connections

When you have completed setting I/O fencing, remove the temporary `rsh` access permissions you have set for the systems in the cluster and restore the connections of the cluster systems to the public network.

Note If your cluster systems use `ssh` for secure communications, and you temporarily removed the connections to the public network, restore them at this time.

Editing VCS Configuration to Add the UseFence Attribute

After adding coordinator disks and configuring I/O fencing, edit the VCS configuration file, `/etc/VRTSvcs/conf/config/main.cf`, and add the `UseFence` cluster attribute.

1. Save the existing configuration:

```
# haconf -dump -makero
```

2. Stop VCS on all nodes.

```
# hastop -all
```

3. Make a backup copy of the `main.cf` file:

```
# cd /etc/VRTSvcs/conf/config
# cp main.cf main.orig
```


4. On one node, use `vi` or another text editor to edit the `main.cf` file. Modify the list of cluster attributes by adding the attribute, `UseFence`, and assign it a value of `SCSI3`. For example, with the attribute added this portion of the file resembles:

```
cluster vcs_cluster2 (
        UserNames = { admin = "cDRpdxPmHpzS." }
        Administrators = { admin }
        HacliUserLevel = COMMANDROOT
        CounterInterval = 5
        UseFence = SCSI3
        )
```

5. Save and close the file.

6. Verify the syntax of the file `/etc/VRTSvcs/conf/config/main.cf`:

```
# hacf -verify .
```

7. Using `rcp`, or some other available utility, copy the VCS configuration file to the other nodes. For example, on each node:

```
# rcp north:/etc/VRTSvcs/conf/config/main.cf
        /etc/VRTSvcs/conf/config
```

8. With the configuration file in place on each system, shut down and then restart each system.

```
# shutdown -y -i6
```

Note To ensure that I/O fencing is shut down properly, use the `shutdown` command instead of the `reboot` command.

Troubleshooting I/O Fencing

The following troubleshooting topics have headings that indicate likely symptoms or that indicate procedures required for a solution.

vxfentsthdw Fails When SCSI TEST UNIT READY Command Fails

If you see a message resembling:

```
Issuing SCSI TEST UNIT READY to disk reserved by other node FAILED.
Contact the storage provider to have the hardware configuration
fixed.
```

The disk array does not support returning success for a SCSI TEST UNIT READY command when another host has the disk reserved using SCSI-3 persistent reservations. This happens with Hitachi Data Systems 99XX arrays if bit 186 of the system mode option is not enabled.

vxfentsthdw Fails When Prior Registration Key Exists on Disk

Although unlikely, you may attempt to use the `vxfentsthdw` utility to test a disk that has a registration key already set. If you suspect a key exists on the disk you plan to test, use the `vxfenadm -g` command to display it.

```
# vxfenadm -g diskname
```

- If the disk is not SCSI-3 compliant, an error is returned indicating: `Inappropriate ioctl for device`.
- If you have a SCSI-3 compliant disk and no key exists, then the output resembles:

  ```
  Reading SCSI Registration Keys...
  Device Name: <diskname>
  Total Number Of Keys: 0
  No keys ...
  ```

  Proceed to test the disk using the `vxfentsthdw` utility. "Testing Data Storage Disks Using vxfentsthdw" on page 116.

- If keys exist, you must remove them before you test the disk. Refer to "Removing Existing Keys From Disks" in the next section.

.
.
.

Node is Unable to Join Cluster While Another Node is Being Ejected

A cluster that is currently fencing out (ejecting) a node from the cluster prevents a new node from joining the cluster until the fencing operation is completed. The following are example messages that appear on the console for the new node:

```
...VCS FEN ERROR V-11-1-25 ... Unable to join running cluster
...VCS FEN ERROR V-11-1-25 ... since cluster is currently fencing
...VCS FEN ERROR V-11-1-25 ... a node out of the cluster.

...VCS GAB.. Port b closed
```

If you see these messages when the new node is booting, the startup script (`/etc/vxfen-startup`) on the node makes up to five attempts to join the cluster. If this is not sufficient to allow the node to join the cluster, reboot the new node or attempt to restart vxfen driver with the command:

```
# /etc/init.d/vxfen start
```

Removing Existing Keys From Disks

To remove the registration and reservation keys created by another node from a disk, use the following procedure:

1. Create a file to contain the access names of the disks:

   ```
   # vi /tmp/disklist
   ```

   For example:

   ```
   /dev/rdsk/c1t0d11s2
   ```

2. Read the existing keys:

   ```
   # vxfenadm -g all -f /tmp/disklist
   ```

   The output from this command displays the key:

   ```
   Device Name: /dev/rdsk/c1t0d11s2
   Total Number Of Keys: 1
   key[0]:
        Key Value [Numeric Format]:   65,49,45,45,45,45,45,45
        Key Value [Character Format]: A1------
   ```

3. If you know on which node the key was created, log in to that node and enter the following command:

   ```
   # vxfenadm -x -k A1 -f /tmp/disklist
   ```

   The key is removed.

4. If you do not know on which node the key was created, follow step 5 through step 7 to remove the key.

5. Register a second key "A2" temporarily with the disk:

   ```
   # vxfenadm -m -k A2 -f /tmp/disklist
   Registration completed for disk path /dev/rdsk/c1t0d11s2
   ```

6. Remove the first key from the disk by preempting it with the second key:

   ```
   # vxfenadm -p -k A2 -f /tmp/disklist -vA1
   key: A2------ prempted the key: A1------ on disk
   /dev/rdsk/c1t0d11s2
   ```

7. Remove the temporary key assigned in step 5.

   ```
   # vxfenadm -x -k A2 -f /tmp/disklist
   Deleted the key : [A2------] from device /dev/rdsk/c1t0d11s2
   ```

   No registration keys exist for the disk.

System Panics to Prevent Potential Data Corruption

When a system experiences a split brain condition and is ejected from the cluster, it panics and displays the following console message:

```
VXFEN:vxfen_plat_panic: Local cluster node ejected from cluster to
prevent potential data corruption.
```

How vxfen Driver Checks for Pre-existing Split Brain Condition

The `vxfen` driver functions to prevent an ejected node from rejoining the cluster after the failure of the private network links and before the private network links are repaired.

For example, suppose the cluster of system 1 and system 2 is functioning normally when the private network links are broken. Also suppose system 1 is the ejected system. When system 1 reboots before the private network links are restored, its membership configuration does not show system 2; however, when it attempts to register with the coordinator disks, it discovers system 2 is registered with them. Given this conflicting information about system 2, system 1 does not join the cluster and returns an error from `vxfenconfig` that resembles:

```
vxfenconfig: ERROR: There exists the potential for a preexisting
  split-brain. The coordinator disks list no nodes which are in the
  current membership. However, they also list nodes which are not
  in the current membership.

I/O Fencing Disabled!
```

Also, the following information is displayed on the console:

```
<date> <system name> vxfen: WARNING: Potentially a preexisting
<date> <system name> split-brain.
<date> <system name> Dropping out of cluster.
<date> <system name> Refer to user documentation for steps
<date> <system name> required to clear preexisting split-brain.
<date> <system name>
<date> <system name> I/O Fencing DISABLED!
<date> <system name>
<date> <system name> gab: GAB:20032: Port b closed
```

However, the same error can occur when the private network links are working and both systems go down, system 1 reboots, and system 2 fails to come back up. From the view of the cluster from system 1, system 2 may still have the registrations on the coordinator disks.

Case 1: System 2 Up, System 1 Ejected (Actual Potential Split Brain)

Determine if system 1 is up or not. If it is up and running, shut it down and repair the private network links to remove the split brain condition. Reboot system 1.

Case 2: System 2 Down, System 1 Ejected (Apparent Potential Split Brain)

1. Physically verify that system 2 is down.
2. Verify the systems currently registered with the coordinator disks. Use the following command:

   ```
   # vxfenadm -g all -f /etc/vxfentab
   ```

   The output of this command identifies the keys registered with the coordinator disks.
3. Clear the keys on the coordinator disks as well as the data disks using the command `/opt/VRTSvcs/rac/bin/vxfenclearpre`. See "Using vxfenclearpre Command to Clear Keys After Split Brain" on page 131.
4. Make any necessary repairs to system 2 and reboot.

Using vxfenclearpre Command to Clear Keys After Split Brain

When you have encountered a split brain condition, use the `vxfenclearpre` command to remove SCSI-3 registrations and reservations on the coordinator disks as well as on the data disks in all shared disk groups.

1. Shut down all other systems in the cluster that have access to the shared storage. This prevents data corruption.

2. Start the script:

```
# cd /opt/VRTSvcs/vxfen/bin
# ./vxfenclearpre
```

3. Read the script's introduction and warning. Then, you can choose to let the script run.

```
Do you still want to continue: [y/n] (default : n)
y
```

Note Informational messages resembling the following may appear on the console of one of the nodes in the cluster when a node is ejected from a disk/LUN:

```
<date> <system name> scsi: WARNING: /sbus@3,0/lpfs@0,0/sd@0,1(sd91):
<date> <system name> Error for Command: <undecoded cmd 0x5f> Error Level:
Informational
<date> <system name> scsi: Requested Block: 0 Error Block 0
<date> <system name> scsi: Vendor: <vendor> Serial Number: 0400759B006E
<date> <system name> scsi: Sense Key: Unit Attention
<date> <system name> scsi: ASC: 0x2a (<vendor unique code 0x2a>), ASCQ: 0x4,
FRU: 0x0
```

These informational messages may be ignored.

```
Cleaning up the coordinator disks...

Cleaning up the data disks for all shared disk groups...

Successfully removed SCSI-3 persistent registration and
reservations from the coordinator disks as well as the shared
data disks.

Reboot the server to proceed with normal cluster startup...
#
```

4. Reboot all systems in the cluster.

Removing or Adding Coordinator Disks

This section describes how to:

- Replace coordinator disk in the coordinator disk group
- Destroy a coordinator disk group

Note Adding or removing coordinator disks requires all services be shut down.

Note the following about the procedure:

- ✔ A coordinator disk group requires an odd number (three minimum) of disks/LUNs.
- ✔ When adding a disk, add the disk to the coordinator disk group (`vxfencoorddg`, for example) and retest the group for support of SCSI-3 persistent reservations.
- ✔ You can destroy the coordinator disk group such that no registration keys remain on the disks. The disks can then be used elsewhere.

▼ To remove and replace a disk in the coordinator disk group

1. Log in as root user on one of the cluster systems.
2. If VCS is running, shut it down:

   ```
   # hastop -all
   ```

3. Stop I/O fencing on all nodes:

   ```
   # /etc/init.d/vxfen stop
   ```

   This removes any registration keys on the disks.

4. Import the coordinator disk group. The file `/etc/vxfendg` includes the name of the disk group (for example, `vxfencoorddg`) that contains the coordinator disks, so use the command:

   ```
   # vxdg -tfC import 'cat /etc/vxfendg'
   ```

   where:

   `-t` specifies that the disk group is imported only until the system restarts.

   `-f` specifies that the import is to be done forcibly, which is necessary if one or more disks is not accessible.

   `-C` specifies that any import blocks are removed.

5. To remove disks from the disk group, use the VxVM disk administrator utility, `vxdiskadm`.

   Note You may also destroy the existing coordinator disk group. For example:
   `# vxdg destroy vxfencoorddg`

6. Add the new disk to the system, initialize it as a VxVM disk, and add it to the coordinator disk group. Refer to "Setting Up the Disk Group for Coordinator Disks" on page 121

7. Test the recreated disk group for SCSI-3 persistent reservations compliance. Refer to "Requirements for Coordinator Disks" on page 120.

8. After replacing disks in a coordinator disk group, deport the disk group:

   ```
   # vxdg deport 'cat /etc/vxfendg'
   ```

9. On each node in the cluster, start the I/O fencing driver:

   ```
   # /etc/rc.d/rc2.d/S97vxfen start
   ```

10. If VCS must be restarted, on each node, enter:

    ```
    # hastart
    ```


Additional I/O Fencing Information

This section provides additional information about I/O fencing, including an extended description of the `vxfentsthdw` command, `vxfenadm` command, and a description of I/O fencing behavior to protect data in certain scenarios.

vxfentsthdw Options

The table below describes three methods the utility provides to test storage devices.

vxfentsthdw option	Description	When to Use
`-m`	Utility runs manually, in interactive mode, prompting for systems and devices, and reporting success or failure. May be used with `-r` and `-t` options. `-m` is the default option.	For testing a few disks or for sampling disks in larger arrays.
`-f` *`filename`*	Utility tests system and device combinations listed in a text file. May be used with `-r` and `-t` options.	For testing several disks.
`-g` *`disk_group`*	Utility tests all disk devices in a specified disk group. May be used with `-r` and `-t` options.	For testing many disks and arrays of disks. Disk groups may be temporarily created for testing purposes and destroyed (ungrouped) after testing.

Using the -r Option for Non-destructive Testing

To test disk devices containing data you want to preserve, you can use the `-r` option with the `-m`, `-f`, or `-g` options, which are described in the following sections. For example, to use the `-m` option and the `-r` option, you can run the utility by entering:

```
# /opt/VRTSvcs/vxfen/bin/vxfentsthdw -rm
```

When invoked with the `-r` option, the utility does not use tests that write to the disks. Therefore, it does not test the disks for all of the usual conditions of use.

Using the -m Option

The -m option is the default option for `vxfentsthdw` and is described in detail in "Running vxfentsthdw" on page 117.

Using the -f Option: Example

Use the `-f` option to test disks that are listed in a text file. For example, you can create a file to test two disks shared by systems `north` and `south` that might resemble:

```
north /dev/rdsk/c2t2d1s2 south /dev/rdsk/c3t2d1s2
north /dev/rdsk/c2t2d2s2 south /dev/rdsk/c3t2d2s2
```

where the first disk is listed in the first line and is seen by `north` as `/dev/rdsk/c2t2d1s2` and by `south` as `/dev/rdsk/c3t2d1s2`. The other disk, in the second line, is seen as `/dev/rdsk/c2t2d2s2` from `north` and `/dev/rdsk/c3t2d2s2` from `south`. Typically, the list of disks could be extensive.

Suppose you created the file named `disks_blue`. To test the disks, you would enter:

```
# /opt/VRTSvcs/vxfen/bin/vxfentsthdw -f disks_blue
```

The utility reports the test results one disk at a time, just as for the `-m` option.

You can redirect the test results to a text file. Precede the command with "`yes`" to acknowledge that the testing destroys any data on the disks to be tested.

Caution Be advised that by redirecting the command's output to a file, a warning that the testing destroys data on the disks cannot be seen until the testing is done.

For example:

```
# yes | /opt/VRTSvcs/vxfen/bin/vxfentsthdw -f disks_blue >
blue_test.txt
```

Using the -g Option: Example

Use the `-g` option to test all disks within a disk group. For example, you create a temporary disk group consisting of all disks in a disk array and test the group.

Note Do not import the test disk group as shared; that is, do not use the `-s` option.

The utility reports the test results one disk at a time. You can redirect the test results to a text file for review.

```
# /opt/VRTSvcs/vxfen/bin/vxfentsthdw -g red_disks_dg > redtest.txt
```

After testing, destroy the disk group and put the disks into disk groups as you need.

Testing a Disk with Existing Keys

If the utility detects that a coordinator disk has existing keys, you see a message that resembles:

```
There are VERITAS I/O Fencing keys on the disk. Please make sure
that I/O Fencing is shut down on all nodes of the cluster before
continuing.

         ******** WARNING!!!!!!!! ********

THIS SCRIPT CAN ONLY BE USED IF THERE ARE NO OTHER ACTIVE NODES IN
THE CLUSTER!  VERIFY ALL OTHER NODES ARE POWERED OFF OR INCAPABLE
OF ACCESSING SHARED STORAGE.

If this is not the case, data corruption will result.

Do you still want to continue : [y/n] (default: n) y
```

The utility prompts you with a warning before proceeding. You may continue as long as I/O fencing is not yet configured.

How I/O Fencing Works in Different Event Scenarios

The following table describes how I/O fencing works to prevent data corruption in different failure event scenarios. For each event, corrective operator actions are indicated.

Event	Node A: What Happens?	Node B: What Happens?	Operator Action
Both private network links fail.	Node A races for majority of coordinator disks. If Node A wins race for coordinator disks, Node A ejects Node B from the shared disks and continues.	Node B races for majority of coordinator disks. If Node B loses the race for the coordinator disks, Node B removes itself from the cluster.	When Node B is ejected from cluster, repair the private networks before attempting to bring Node B back.
Both private network links function again after event above.	Node A continues to work.	Node B has crashed. It cannot start the database since it is unable to write to the data disks.	Reboot Node B after private networks are restored.
One private network link fails.	Node A prints message about an IOFENCE on the console but continues.	Node B prints message about an IOFENCE on the console but continues.	Repair private network. After network is repaired, both nodes automatically use it.
Node A hangs.	Node A is extremely busy for some reason or is in the kernel debugger. When Node A is no longer hung or in the kernel debugger, any queued writes to the data disks fail because Node A is ejected. When Node A receives message from GAB about being ejected, it removes itself from the cluster.	Node B loses heartbeats with Node A, and races for a majority of coordinator disks. Node B wins race for coordinator disks and ejects Node A from shared data disks.	Verify private networks function and reboot Node A.

Event	Node A: What Happens?	Node B: What Happens?	Operator Action
Nodes A and B and private networks lose power. Coordinator and data disks retain power. Power returns to nodes and they reboot, but private networks still have no power.	Node A reboots and I/O fencing driver (vxfen) detects Node B is registered with coordinator disks. The driver does not see Node B listed as member of cluster because private networks are down. This causes the I/O fencing device driver to prevent Node A from joining the cluster. Node A console displays: `Potentially a preexisting split brain. Dropping out of the cluster. Refer to the user documentation for steps required to clear preexisting split brain.`	Node B reboots and I/O fencing driver (vxfen) detects Node A is registered with coordinator disks. The driver does not see Node A listed as member of cluster because private networks are down. This causes the I/O fencing device driver to prevent Node B from joining the cluster. Node B console displays: `Potentially a preexisting split brain. Dropping out of the cluster. Refer to the user documentation for steps required to clear preexisting split brain.`	Refer to "System Panics to Prevent Potential Data Corruption" on page 129 for instructions on resolving preexisting split brain condition.

Event	Node A: What Happens?	Node B: What Happens?	Operator Action
Node A crashes while Node B is down. Node B comes up and Node A is still down.	Node A is crashed.	Node B reboots and detects Node A is registered with the coordinator disks. The driver does not see Node A listed as member of the cluster. The I/O fencing device driver prints message on console: `Potentially a preexisting split brain. Dropping out of the cluster. Refer to the user doc-umentation for steps required to clear preexisting split brain.`	Refer to "System Panics to Prevent Potential Data Corruption" on page 129 for instructions on resolving preexisting split brain condition.
The disk array containing two of the three coordinator disks is powered off.	Node A continues to operate as long as no nodes leave the cluster.	Node B continues to operate as long as no nodes leave the cluster.	
Node B leaves the cluster and the disk array is still powered off.	Node A races for a majority of coordinator disks. Node A fails because only one of three coordinator disks is available. Node A removes itself from the cluster.	Node B leaves the cluster.	Power on failed disk array and restart I/O fencing driver to enable Node A to register with all coordinator disks.

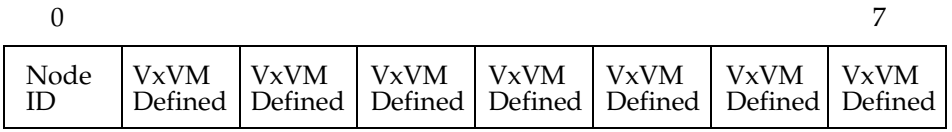
The vxfenadm Utility

Administrators can use the `vxfenadm` command to troubleshoot and test fencing configurations. The command's options for use by administrators are:

- `-g` - read and display keys
- `-i` - read SCSI inquiry information from device
- `-m` - register with disks
- `-n` - make a reservation with disks
- `-p` - remove registrations made by other systems
- `-r` - read reservations
- `-x` - remove registrations

Registration Key Formatting

The key defined by VxVM associated with a disk group consists of seven bytes maximum. This key becomes unique among the systems when the VxVM prefixes it with the ID of the system. The key used for I/O fencing, therefore, consists of eight bytes.



The keys currently assigned to disks can be displayed by using the `vxfenadm` command. For example, from the system with node ID 1, display the key for the disk `/dev/rdsk/c2t1d0s2` by entering:

```
# vxfenadm -g /dev/rdsk/c2t1d0s2
Reading SCSI Registration Keys...
Device Name: /dev/rdsk/c2t1d0s2
Total Number of Keys: 1
key[0]:
   Key Value [Numeric Format]: 65,80,71,82,48,48,48,48
   Key Value [Character Format]: APGR0000
```

The `-g` option of `vxfenadm` displays all eight bytes of a key value in two formats. In the numeric format, the first byte, representing the Node ID, contains the system ID plus 65. The remaining bytes contain the ASCII values of the letters of the key, in this case, "PGR0000." In the next line, the node ID 0 is expressed as "A;" node ID 1 would be "B."

VXFEN Tunable Parameters

On each node, edit the file `/kernel/drv/vxfen.conf` to change the value of the `vxfen` driver tunable global parameter, `max_read_coord_disk`. You must restart the system to put change into effect.

vxfen Parameter	Description	Default Value	Minimum Value	Maximum Value
max_read_coord_disk	Specifies how many times the smallest sub-cluster reads the registration keys on the coordinator disks before racing for control of the coordinator disks. The time required for the reads allows a larger sub-cluster to win the race for the coordinator disks.	25	1	1000

Example: Configuring a VXFEN Parameter

In the following example, the maximum number reads the smallest sub-cluster makes to the coordinator disk is changed from 25 to 30. To change the value for `max_read_coord_disk` to 30, edit the file and add the configuration parameter:

```
#
# vxfen configuration file
#
name="vxfen" parent="pseudo" instance=0 max_read_coord_disk=30;
```

Close and save the file. For the changes to take effect, either restart the system, or reconfigure the VXFEN module using the following steps:

1. Shut down all Oracle service groups on the system:

   ```
   # hagrp -offline oragrp -sys galaxy
   ```

2. Stop all Oracle client processes on the system, such as `sqlplus`, `svrmgrl`, and `gsd`.

3. Unconfigure the VXVFN module:

   ```
   # /sbin/vxfenconfig -U
   ```

4. Determine the VXFEN module ID:

   ```
   # /usr/sbin/modinfo | grep -i vxfen
   ```

   The module ID is the number in the first column of the output.

5. Unload the VXFEN module, using the module ID you determined:

```
# /usr/sbin/modunload -i module_ID
```

6. Configure the VXFEN module:

```
# /sbin/vxfenconfig -c
```

7. Bring the service groups back online:

```
# hagrp -online oragrp -sys galaxy
```

Manually Upgrading VCS to Release 4.1 7

You can automatically upgrade to VCS 4.1 from the 3.5 and 4.0 releases of VCS by using the `installvcs` program (see "Using the installvcs Program" on page 24 and "Using installvcs to Upgrade to VCS 4.1" on page 55).

However, if you must manually upgrade to VCS 4.1, use the procedures described in this chapter. Upgrading VCS manually entails the following activities:

- ✔ Obtaining license keys (see "Obtaining a License Key" on page 143)
- ✔ Shutting down VCS
- ✔ Removing the previous version of VCS
- ✔ Installing VCS 4.1, using procedures in "Manually Installing and Configuring VCS" on page 75
- ✔ Restoring previous configuration files to the VCS 4.1 environment
  - Include all later types definitions in `types.cf` file
  - Edit the `main.cf` file to remove obsolete attributes
  - Edit the `main.cf` file to update the `ClusterService` service group
  - Edit the `main.cf` file to update any Mount resources and add `FsckOpt` attribute
- ✔ Starting LLT, GAB, and VCS 4.1 (see "Starting LLT, GAB, and VCS" on page 152)
- ✔ Unfreezing service groups and updating user passwords (see "Unfreezing Service Groups and Updating Passwords" on page 152)
- ✔ Upgrading Cluster Manager (Java Console) to version 4.1

Obtaining a License Key

VCS 4.1 is a licensed product. When upgrading to VCS 4.1 from a release prior to VCS 2.0, you must enter a license key for each system in the cluster. A site license applies to all systems in the cluster. You can request licenses from your VERITAS Customer Support representative. See "Obtaining License Keys for VCS" on page 17.

If you are upgrading from versions 3.5 and higher, your existing VCS license is accepted and no action is required.

Shutting Down VCS

For a successful upgrade, verify that all systems in the cluster are running VCS.

▼ To shut down VCS

1. Log in to the any system as the superuser.
2. Make the VCS configuration writable. On any system, type:

   ```
   # haconf -makerw
   ```

3. List the groups in your configuration. On any system, type:

   ```
   # hagrp -list
   ```

4. Freeze all service groups. On any system, type the following command for each group name displayed in the output from step 3.

   ```
   # hagrp -freeze group_name -persistent
   ```

5. Save the configuration file (`main.cf`) with the groups frozen. On any system, type:

   ```
   # haconf -dump -makero
   ```

6. Shut down VCS. On any system, type:

   ```
   # hastop -all -force
   ```

Note Perform step 7 through step 16 on each system in the cluster.

7. Confirm that VCS has shut down. On each system, type:

   ```
   # gabconfig -a
   ```

   Output resembles:

   ```
   GAB Port Memberships
   ===================================
   Port a gen 23dc0001 membership 01
   ```

   Note that the output shows no membership for port h.

8. Confirm that GAB is not running on any disks:

```
# gabdiskhb -l
```

If it is, remove it from the disks using the `gabdiskhb` commands.

For example, if the output from `gabdiskhb -l` command resembles:

```
Port  Disk              Major   Minor   Start   Active
=======================================================
a     /dev/dsk/c1t3d0s2 37      8       16      01
h     /dev/dsk/c1t3d0s2 37      8       144     01
```

You can enter the command:

```
# gabdiskhb -d /dev/dsk/c1t3d0s2 -s 16
# gabdiskhb -d /dev/dsk/c1t3d0s2 -s 144
```

If you receive a message that says `gabdiskhb` or `gadiskx` is running, then you must reboot the systems after competing the upgrade of VCS 4.1.

9. Unconfigure GAB. On each system, type:

```
# gabconfig -U
```

If unconfiguring GAB fails and you receive messages that say the device cannot be unconfigured because it is busy, you must reboot the systems after completing the upgrade of VCS 4.1.

10. Unload the GAB module from the kernel.

    a. Determine the kernel module ID:

    ```
    # modinfo | grep gab
    ```

    b. Unload the module from the kernel:

    ```
    # modunload -i gab_id
    ```

11. Unconfigure LLT. On each system, type:

```
# lltconfig -U
```

The following message is displayed on the console:

```
lltconfig: this will attempt to stop and reset LLT.
Confirm (y/n)?
```

Note This message does *not* display if you are upgrading from version 1.0.2.

12. Type **y** on each system in response to the message.

If unconfiguring LLT fails and you receive messages that say the device cannot be unconfigured because it is busy, you must reboot the systems after completing the upgrade of VCS 4.1.

13. Unload the LLT module from the kernel.

a. Determine the kernel module ID:

```
# modinfo | grep llt
```

The module IDs are in the left-hand column of the output.

b. Unload the module from the kernel:

```
# modunload -i llt_id
```

Note If either the LLT or GAB kernel modules fails to unload, you must reboot the systems after completing the upgrade of VCS 4.1.

14. Compare your `types.cf` file with the originally installed `types.cf` file. On any system, type:

```
# diff -w /etc/VRTSvcs/conf/config/types.cf
                /etc/VRTSvcs/conf/types.cf
```

Note that if any changes have been made to the default `types.cf` file, or if any user-defined types have been added to the original `types.cf` file, the files are different.

- If the files are different, make a copy of the modified file. You require the copy for step 1 of "Restoring Previous Configuration Files to VCS 4.1" on page 150. For example:

```
# cp /etc/VRTSvcs/conf/config/types.cf
                /etc/VRTSvcs/conf/types.save
```

- If the files are the same, go to the next step.

The following example output from the `diff` command shows the minor differences you may see when VCS dumped the configuration. These minor differences can be safely ignored during the upgrade procedure.

```
60c60
<       int IfconfigTwice
---
>       int IfconfigTwice = 0
145c145
<       int OnlineNFSRestart
---
>       int OnlineNFSRestart = 0
```

15. Make a copy of the current `main.cf` configuration file for safekeeping. On any system, type:

```
# cp /etc/VRTSvcs/conf/config/main.cf
          /etc/VRTSvcs/conf/main.save
```

16. Make copies of the configuration files `/etc/llthosts`, `/etc/lltab`, and `/etc/gabtab` on each system because these files may vary by system. On each system, type:

```
# cp /etc/llthosts /etc/llthosts.save
# cp /etc/lltab /etc/lltab.save
# cp /etc/gabtab /etc/gabtab.save
```

17. Shut down the VCS CmdServer. If you are using VCS 3.5 or 4.0, determine the PID of the CmdServer. Enter:

```
# ps -ef|grep CmdServer
root   291     1  0 10:17:22 ?        0:00
/opt/VRTSvcs/bin/CmdServer

# kill -9 291
```


Removing Previous VCS Packages Using pkgrm

1. On each system, remove any previously installed patch sets in the following order: `VRTSvcs`, `VRTSgab`, and `VRTSllt`. For example:

   ```
   # showrev -p | grep VRTS
   # patchrm patch_id
   ```

2. On each system, use the `pkgrm` command to remove previous VCS packages.

   - For VCS 3.5, enter:

     ```
     # pkgrm VRTSvcsmg VRTSvcsag VRTSvcs VRTSperl VRTSgab VRTSllt
     ```

   - For VCS 4.0, enter:

     ```
     # pkgrm VRTScscw VRTScssim VRTScutil VRTSjre VRTScspro VRTSvcsag
       VRTSvcsmg VRTSvcs VRTSvxfen VRTSgab VRTSllt VRTSobgui VRTSmuob
       VRTSob VRTSperl
     ```

3. On each system, remove the VCS release 3.5 VERITAS license utility package, enter:

   ```
   # pkgrm VRTSvlic
   ```

4. Remove optional packages you might have previously installed.

   a. To remove the VCS release 3.5 and 4.0 Web Console and VCS Web GUI engine, enter:

      ```
      # pkgrm VRTSvcsw VRTSweb
      ```

   b. To remove the VCS release 3.5 and 4.0 VCS documentation package, enter:

      ```
      # pkgrm VRTSvcsmn VRTSvcsdc
      ```

   c. If you installed the VCS release 3.5 and 4.0 Java Console, enter:

      ```
      # pkgrm VRTScscm
      ```

5. For the Japanese language pack, use the `pkgrm` command to remove previous VCS release 4.0 language packages, enter:

   ```
   # pkgrm VRTSjacsu VRTSjacsp VRTSjacs VRTSmuobg VRTSmuob VRTSmulic
   ```

6. For the Japanese language pack, remove optional VCS 4.0 packages you might have previously installed.
   - For the Japanese language pack, to remove VCS Web Console and VCS Web GUI engine, enter:

     # **pkgrm VRTSjacsw VRTSjaweb**
   - For the Japanese language pack, to remove the VCS documentation package, enter:

     # **pkgrm VRTSjacsd**
   - For the Japanese language pack, to remove the Java Console, enter:

     # **pkgrm VRTSjacsj**
7. As the packages are removed, answer **Yes** when prompted.

Manually Installing VCS 4.1

Refer to "Installing VCS Software Manually" on page 76 to add the VCS software, including infrastructure packages, the required and optional VCS packages, and the required patches.

Note Do not start VCS after adding the VCS packages. Return to this chapter and go to the section, "Restoring Previous Configuration Files to VCS 4.1."

Restoring Previous Configuration Files to VCS 4.1

Note Perform step 1 through step 6 on any system in the cluster. Some steps may not apply, depending on which VCS release you are upgrading.

1. Check to see whether you need to merge any types defined in your previous installation with the newly installed types file.

   The `types.cf` file installed with VCS 4.1 contains new type definitions. Compare the saved `types.cf` file (`types.save`) created in step 14 on page 146 to the newly installed `/etc/VRTSvcs/conf/types.cf`:

   ```
   # diff -w /etc/VRTSvcs/conf/types.save
                 /etc/VRTSvcs/conf/types.cf
   ```

   a. If the only differences you see are the new types defined in the newly installed **`/etc/VRTSvcs/conf/types.cf`** file, then you don't need to restore the contents of the file `types.save`.

Note If the files are very different from each other, the output of the `diff` command may be too confusing to read. In this case, print and compare the two files manually.

   b. If the differences include any types defined in the `types.save` file, then you must edit the newly installed **`/etc/VRTSvcs/conf/types.cf`** file, adding the types used in your previous VCS configuration.

   c. Copy the appropriate `types.cf` file to `/etc/VRTSvcs/conf/config/types.cf`.

2. When upgrading from a VCS release before 2.0, run the script `updatepre20maincf` to remove obsolete attributes from `/etc/VRTSvcs/conf/config/main.cf`. The script makes a backup copy of the original `main.cf` and saves it as `/etc/VRTSvcs/conf/config/main.cf.pre20`. For example

   ```
   # cd /cdrom/cluster_server
   # ./scripts/updatepre20maincf
   ```

3. If you are upgrading from VCS 3.5, then you must edit your `main.cf` file to add a `ClusterAddress` definition and upgrade the `ClusterService` group. (Make sure you have backed up the original file; see step 15 on page 147.) You can use the "main.cf Example, for Clusters Without the GCO Option" on page 100 for reference. Using `vi` or another editor, make the following changes:

   a. In the "`cluster`" definition section, beneath the line that defines the `UserNames`, add a line that defines the cluster's virtual IP address. For example:

   ```
   ClusterAddress = "11.136.88.199"
   ```

   b. In the `ClusterService` group, under the line that defines the `OnlineRetryLimit`, add the following line:

   ```
   OnlineRetryInterval = 120
   ```

4. Examine any definitions of the Mount resource that may exist in the `main.cf` file. With VCS 3.5 and later, the `FsckOpt` attribute of the Mount resource is *required* and the definition must contain either the value `"-y"` or `"-n"`; otherwise, the resource cannot come online.

   For example, in the Mount resource shown below, the `FsckOpt` attribute is assigned `"-y"`.

   ```
   Mount Mount_home (
     MountPoint = "/export/home"
     BlockDevice = "/dev/vx/dsk/shared1/home_vol"
     FSType = vxfs
     FsckOpt = "-y"
     MountOpt = rw
     )
   ```

   Please refer to the *VERITAS Cluster Server Bundled Agents Reference Guide* for information on Mount agent and its attributes.

5. Save and close the file `main.cf`.

6. Re-verify the syntax of the file `/etc/VRTSvcs/conf/config/main.cf`:

   ```
   # cd /etc/VRTSvcs/conf/config
   # hacf -verify .
   ```

Note When upgrading manually from a VCS release prior to 2.0, you can create the `ClusterService` group. You must create it manually; refer to the *VERITAS Cluster Server User's Guide*.

Licensing VCS

Run `vxlicinst` to add VCS licenses. If necessary, see "Obtaining License Keys for VCS" on page 17.

Note If you are upgrading from versions 3.5 and higher, your existing VCS license is accepted and no action is required.

On each system, type the commands:

```
# cd /opt/VRTS/bin
# ./vxlicinst -k XXXX-XXXX-XXXX-XXXX-XXXX-XXX
```

Starting LLT, GAB, and VCS

Refer to the sections "Starting LLT" on page 92, "Starting GAB" on page 93, and "Starting VCS" on page 93.

Unfreezing Service Groups and Updating Passwords

After starting the LLT, GAB, and VCS components and verifying they are configured and running, unfreeze the service groups and update user passwords.

1. On any system, type:

   ```
   # haconf -makerw
   ```

2. For each frozen service group displayed in the output, enter the following command:

   ```
   # hagrp -unfreeze service_group -persistent
   ```

3. For VCS 3.5, you must reset passwords for each user in the cluster configuration.

   a. To list the users, enter:

   ```
   # hauser -list
   ```

   b. For each user in the output of the previous command, change the user password:

   ```
   # hauser -update user_name
   ```

   When prompted, enter the user's password and confirm it by entering it again. For example:

   ```
   # hauser -update admin
     Enter New Password:********
   Enter Again:********
   # hauser -update smith
     Enter New Password:********

     Enter Again:********
   ```

4. Save the VCS configuration.

   ```
   # haconf -dump -makero
   ```

Upgrading to the VCS 4.1 Java Console

When you upgrade to VCS release 4.1, you must also upgrade the Java Console (GUI) to version 4.1. Earlier versions of the Java Console cannot run on VCS release 4.1, although the Java Console version 3.5 can run on earlier versions of VCS.

Note The VCS 4.1 Java Console requires JRE version 1.4. If necessary, you can add it when you add the Java Console package.

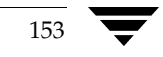
On Solaris

1. Remove the GUI from VCS 3.5 or 4.0 installations:

   ```
   # pkgrm VRTScscm
   ```

2. Create a directory for installation of the Java Console:

   ```
   # mkdir /tmp/install
   ```

3. Insert the software disc with the VCS software into a drive connected to the system. The Solaris volume-management software automatically mounts the disc as `/cdrom/cdrom0`. Type the command:

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

4. Copy the compressed package files from the software disc to the temporary directory:

   ```
   # cp -r cluster_server/pkgs/VRTScscm* /tmp/install
   ```

5. Go to the temporary directory and unzip the compressed package file:

Note If your system does not have the `gunzip` utility, copy it from the disc:
`# cp /cdrom_path/gnu/gunzip /tmp/install`

   ```
   # cd /tmp/install
   # gunzip VRTScscm.tag.gz
   ```

   The file `VRTScscm.tar` is now present in the temporary directory.

6. Extract the compressed file from the tar file:

   ```
   # tar -xvf VRTScscm.tar
   ```

7. Install the software:

   ```
   # pkgadd -d . VRTScscm
   ```

8. Answer **Yes** if prompted.

On Windows Systems

1. Remove the Java-based Cluster Manager from previous installations:

   a. From the Control Panel, double-click **Add/Remove Programs**.

   b. Select **VERITAS Cluster Manager**.

   c. Click **Add/Remove**.

   d. Follow the instructions presented by the uninstall wizard.

2. Add the new Java-based Cluster Manager. Go to "Installing the Java Console on a Windows System" on page 109.

Adding and Removing Cluster Systems 8

This chapter provides procedures for adding and removing nodes from a cluster.

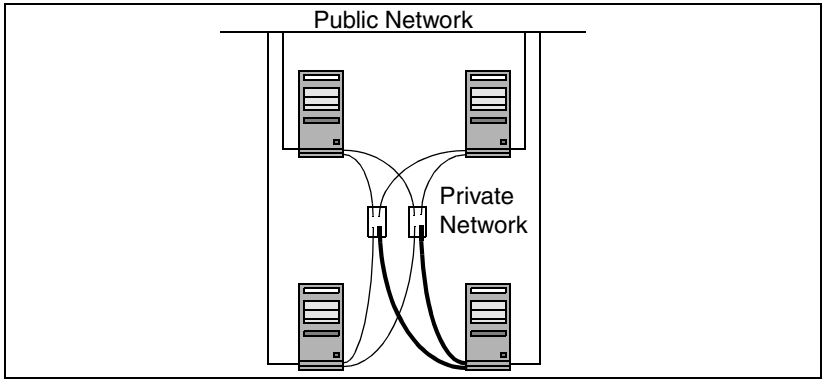
Adding a Node to a Cluster

The system you add to the cluster must meet the hardware and software requirements outlined in "Preparing to Install VCS 4.1" on page 7.

Setting up the Hardware

Before configuring a new system to an existing cluster, you must physically add the system to the cluster.

1. Connect the VCS private Ethernet controllers. If you are expanding from a two-system cluster, you need to use independent hubs for the private network connections, replacing crossover cables if they are used. If you already use independent hubs, connect the two Ethernet controllers on the new system to the independent hubs. The following illustration shows a new system being added to an existing three-system cluster using two independent hubs.



2. Connect the system to the shared storage, if required.

Installing the Software

Follow the procedures in "Installing VCS Software Manually" on page 76 to manually install the VCS 4.1 packages and "Adding a License Key" on page 81 to install the license key. Return to this sections to continue adding the node to the cluster.

Configuring LLT and GAB

1. Create the file `/etc/llthosts` on the new system. You must also update it on each of the current systems in the cluster. For example, suppose you are adding `east` to a cluster consisting of `north` and `south`:

   a. If the file on one of the existing systems resembles:

   ```
   0 north
   1 south
   ```

   b. The updated file for all systems, including the new one, would resemble:

   ```
   0 north
   1 south
   2 east
   ```

2. Create the file `/etc/lltab` on the new system, making sure that line beginning "`set-node`" specifies the new system. Refer to "/etc/lltab" on page 97; the file `/etc/lltab` on an existing system can serve as a guide. The following example describes a system where system `east` is the new system on cluster number 2:

   ```
   set-node east
   set-cluster 2
   link qfe0 /dev/qfe:0 - ether - -
   link qfe1 /dev/qfe:1 - ether - -
   ```

3. On the new system, run the command:

   ```
   # /sbin/lltconfig -c
   ```

4. Create the file `/etc/gabtab` on the new system.

   a. If the `/etc/gabtab` file on the existing systems resembles:

   ```
   /sbin/gabconfig -c
   ```

   Then the file on the new node should be the same, although VERITAS recommends to use the `-c -n`*N* option, where *N* is the number of cluster systems.

b. If the `/etc/gabtab` file on the existing systems resembles:

```
/sbin/gabconfig -c -n2
```

Then, the file on all systems, including the new system, should change to reflect the change in the number of cluster systems. For example, the new on each system should resemble:

```
/sbin/gabconfig -c -n3
```

Refer to "/etc/gabtab" on page 98. The `-n` flag indicates to VCS the number of systems required to be ready to form a cluster before VCS starts.

c. If you are adding a system to a cluster that has a heartbeat disk configured, then the new system should have access to the heartbeat disk. It requires an `/etc/gabtab` file that configures heartbeating, just as do the existing nodes. For example, the new `/etc/gabtab` file for each system may resemble:

```
/sbin/gabdiskhb -a /dev/dsk/c2t3d0s2 -s 16 -p a
/sbin/gabdiskhb -a /dev/dsk/c2t3d0s2 -s 144 -p h
/sbin/gabconfig -c -n3
```

See"Configuring Membership Heartbeat Regions on Disk (optional)" on page 86.

5. On the new system, run the command, to configure GAB:

```
# /sbin/gabconfig -c
```

6. On the new system, run the command:

```
# /sbin/gabconfig -a
```

The output should indicate that Port a membership shows all nodes including the new node. The output should resemble:

```
GAB Port Memberships
===================================
Port a gen a3640003 membership 012
```

Refer to "Verifying GAB" on page 104.

7. Run the same command on the other nodes (`north` and `south`) to verify that the Port a membership includes the new node:

```
# /sbin/gabconfig -a
  GAB Port Memberships
  ===================================
  Port a gen a3640003 membership 012
  Port h gen fd570002 membership 01
  Port h gen fd570002    visible ; 2
```


8. On one of the existing systems in the cluster,

   a. Enter the command:

      ```
      # haconf -makerw
      ```

   b. Add the new system, for example, east, to the cluster:

      ```
      # hasys -add east
      ```

   c. If necessary, modify any new system attributes.

   d. Enter the command:

      ```
      # haconf -dump -makero
      ```

9. From the new system start VCS with the new system added to the cluster:

   ```
   # hastart
   ```

10. Run the GAB configuration command on each node to verify that port a and port h include the new node in the membership:

    ```
    # /sbin/gabconfig -a
      GAB Port Memberships
      ===================================
      Port a gen a3640003 membership 012
      Port h gen fd570002 membership 012
    ```

Removing a Node from a Cluster

Removing a node from a cluster involves the following activities:

- Switching or removing any VCS service groups on that node. The node cannot be removed as long as it runs service groups on which other service groups depend.
- Deleting the system from the VCS configuration.
- Modifying the `llthosts` and `gabtab` files to reflect the change.
- Modifying startup scripts for LLT, GAB, and VCS to allow reboot of the node without affecting the cluster.

Example of Removing a Node

In the following example, the cluster consists of nodes A, B, and C; node C is to leave the cluster. Start by issuing the following commands from one of the nodes to remain, A or B:

1. Make a backup copy of the current configuration file, `main.cf`:

```
# cp -p /etc/VRTSvcs/conf/config/main.cf
    /etc/VRTSvcs/conf/config/main.cf.goodcopy
```

2. Check the status of the systems and the service groups:

```
# hastatus -summary

-- SYSTEM STATE
-- System      State          Frozen
A  A           RUNNING        0
A  B           RUNNING        0
A  C           RUNNING        0

-- GROUP STATE
-- Group    System      Probed   AutoDisabled   State
B  grp1     A           Y        N              ONLINE
B  grp1     B           Y        N              OFFLINE
B  grp2     A           Y        N              ONLINE
B  grp3     B           Y        N              OFFLINE
B  grp3     C           Y        N              ONLINE
B  grp4     C           Y        N              ONLINE
```

The example output from the `hastatus` command shows that systems A, B, and C are the nodes in the cluster. Also, service group `grp3` is configured to run on system B and system C, the leaving node. Service group `grp4` runs only on system C. Service groups `grp1` and `grp2` do not run on system C.

3. Switch failover service groups from the leaving node. You can switch `grp3` from system C to system B:

```
# hagrp -switch grp3 -to B
```

4. Check for any dependencies involving any service groups that run on the leaving node; for example, `grp4` runs only on the leaving node:

```
# hagrp -dep
```

If the service group on the leaving node requires other service groups, that is, if it is a parent to service groups on other nodes, unlink the service groups:

```
# haconf -makerw
# hagrp -unlink grp4 grp1
```

These commands enable you to edit the configuration and to remove the requirement `grp4` has for `grp1`.

5. Stop VCS on the leaving node:

```
# hastop -sys C
```

6. Check the status again. The leaving node should show a state of EXITED. Also, any service groups set up for failover should be ONLINE on other nodes:

```
# hastatus -summary

-- SYSTEM STATE
-- System      State          Frozen
A  A           RUNNING        0
A  B           RUNNING        0
A  C           EXITED         0

-- GROUP STATE
-- Group    System      Probed   AutoDisabled   State
B  grp1     A           Y        N              ONLINE
B  grp1     B           Y        N              OFFLINE
B  grp2     A           Y        N              ONLINE
B  grp3     B           Y        N              ONLINE
B  grp3     C           Y        Y              OFFLINE
B  grp4     C           Y        N              OFFLINE
```

7. Delete the leaving node from the SystemList of service groups `grp3` and `grp4`.

```
# hagrp -modify grp3 SystemList -delete C
# hagrp -modify grp4 SystemList -delete C
```

8. For service groups that run only on the leaving node, delete the resources from the group before deleting the group:

```
# hagrp -resources grp4
  processx_grp4
  processy_grp4
# hares -delete processx_grp4
# hares -delete processy_grp4
```

9. Delete the service group:

```
# hagrp -delete grp4
```

10. Check the status:

```
# hastatus -summary
-- SYSTEM STATE
-- System      State          Frozen
A  A           RUNNING        0
A  B           RUNNING        0
A  C           EXITED         0

-- GROUP STATE
-- Group    System      Probed   AutoDisabled   State
B  grp1     A           Y        N              ONLINE
B  grp1     B           Y        N              OFFLINE
B  grp2     A           Y        N              ONLINE
B  grp3     B           Y        N              ONLINE
```

11. Delete the node from the cluster:

```
hasys -delete C
```

12. Save the configuration, making it read only:

```
haconf -dump -makero
```


Modifying Configuration Files On Each Remaining Node

1. If necessary, modify the `/etc/gabtab` file. No change is required to this file if the `/sbin/gabconfig` command has only the argument `-c`, although it is recommended to use the `-n`*N* option, where *N* is the number of cluster systems. If the command has the form `/sbin/gabconfig -c -n`*N*, where *N* is the number of cluster systems, then make sure that *N* is not greater than the actual number of nodes in the cluster, or GAB does not automatically seed.

   Note VERITAS does not recommend the use of the `-c -x` option for `/sbin/gabconfig`. Using `-c -x` dramatically increases configuration time for the Gigabit Ethernet controller and can lead to a split-brain condition.

2. Modify `/etc/llthosts` file on each remaining system to remove the entry of the leaving node. For example, change:

   ```
   0 A
   1 B
   2 C
   ```

   to:

   ```
   0 A
   1 B
   ```

Unloading LLT and GAB and Removing VCS On the Node

On the node leaving the cluster:

1. Unconfigure GAB and LLT:

   ```
   # /sbin/gabconfig -U
   # /sbin/lltconfig -U
   ```

2. Unload the GAB and LLT modules from the kernel.

   a. Determine the kernel module IDs:

   ```
   # modinfo | grep gab
   # modinfo | grep llt
   ```

   The module IDs are in the left-hand column of the output.

   b. Unload the module from the kernel:

   ```
   # modunload -i gab_id
   # modunload -i llt_id
   ```

3. Rename the startup files to prevent LLT, GAB, or VCS from starting up in the future:

```
# mv /etc/rc2.d/S70llt /etc/rc2.d/s70llt
# mv /etc/rc2.d/S92gab /etc/rc2.d/s92gab
# mv /etc/rc3.d/S99vcs /etc/rc3.d/s99vcs
```

4. To determine the packages to remove, enter:

```
# pkginfo | grep VRTS
```

5. To permanently remove the VCS packages from the system, use the `pkgrm` command. Start by removing the following packages, which may have been optionally installed, in the order shown:

```
# pkgrm VRTScscm
# pkgrm VRTSvcsw
# pkgrm VRTSweb
# pkgrm VRTScscw
# pkgrm VRTScssim
# pkgrm VRTScutil
# pkgrm VRTSjre
# pkgrm VRTSvcsdc
# pkgrm VRTSvcsmn
# pkgrm VRTSvcsag
# pkgrm VRTSvcsmg
# pkgrm VRTSvcs
# pkgrm VRTSvxfen
# pkgrm VRTSgab
# pkgrm VRTSllt
# pkgrm VRTSat
# pkgrm VRTSperl
```

6. Optionally, you can remove the infrastructure packages using the order shown:

```
# pkgrm VRTSvlic
# pkgrm VRTScpi
```

7. Remove the VCS patch.

```
# patchrm 117499-02
```

8. Remove the LLT and GAB configuration files.

```
# rm /etc/lltab
# rm /etc/gabtab
# rm /etc/llthosts
```


Installing VCS on a Single System 9

You can install VCS 4.1 on a single system. This chapter describes how to create a single-system cluster, add a node, and create a multinode cluster.

Creating a Single-System Cluster

The installation involves the following tasks:

- ✔ Install the software using the Solaris utility, `pkgadd`.
- ✔ Remove any LLT or GAB configuration files and rename LLT and GAB startup files. A single-node cluster does not require the node-to-node communication service, LLT, or the membership communication service, GAB.
- ✔ Create and modify the VCS configuration files.
- ✔ Start VCS and verify single-node operation.

Setting the PATH Variable

The installation and other commands are located in the `/sbin`, `/usr/sbin`, and `/opt/VRTSvcs/bin` directories. Add these directories to your PATH environment variable.

If you are using the Bourne Shell (`sh` or `ksh`), use the following command:

```
$ PATH=/usr/sbin:/sbin:/opt/VRTSvcs/bin:$PATH; export PATH
```

If you are using the C Shell (`csh` or `tcsh`), use the following command:

```
% setenv PATH /usr/sbin:/sbin:/opt/VRTSvcs/bin:$PATH
```


Installing the Software

Follow the procedures in "Installing VCS Software Manually" on page 76 to manually install the VCS 4.1 packages and "Adding a License Key" on page 81 to install the license key. Return to this sections to continue creating the single-system cluster.

Renaming the LLT and GAB Startup Files

Rename the LLT and GAB startup files. If you need to upgrade the single-system cluster to a multiple system cluster at a later time, you may need them.

```
# mv /etc/rc2.d/S70llt /etc/rc2.d/X70llt
# mv /etc/rc2.d/S92gab /etc/rc2.d/X92gab
```

Setting Up Configuration Files

This section describes setting up the configuration files `main.cf` and `types.cf` for your single-node VCS installation.

main.cf File

VCS requires the configuration file, `main.cf`, to exist in the directory `/etc/VRTSvcs/conf/config`. The `main.cf` configuration file has the following essential elements:

- An "include" statement that specifies the file, `types.cf`, which defines the VCS bundled agent resources
- The name of the cluster
- The name of the system that make up the single-system cluster

An example `main.cf` for a single-system cluster:

```
include "types.cf"
cluster VCSCluster1 ( )
system sn1
```

types.cf File

Note that the "include" statement in `main.cf` refers to a file named `types.cf`. This text file describes the VCS bundled agent resource type definitions. During new installations, the `types.cf` file is automatically copied in to the `/etc/VRTSvcs/conf/config` directory.

Editing the main.cf File

Refer to the *VERITAS Cluster Server User's Guide* for a full description of the `main.cf` file, how to edit it and verify it.

Verifying Single-Node Operation

1. Bring up VCS manually as a single-node cluster using `hastart`(1M) with the `-onenode` option:

   ```
   # hastart -onenode
   ```

2. Verify that the `had` and `hashadow` daemons are running in single-node mode:

   ```
   # ps -ef | grep ha
   root  285   1  0 14:49:31 ?  0:02 /opt/VRTSvcs/bin/had -onenode
   root  288   1  0 14:49:33 ?  0:00 /opt/VRTSvcs/bin/hashadow
   ```

Adding a System to a Single-System Cluster

Adding systems to a single-system cluster involves the activities described below. All systems in the new cluster must run the same version of VCS. For our discussion, we refer to the existing single-node VCS system as System A. We refer to the system that is to join System A to form a multiple-node cluster as System B. The activities include:

- ✔ Setting up System B to be compatible with System A

  The steps to set up System B include shutting down and uninstalling VCS if VCS is present on the system, and, if necessary, adding VxVM and VxFS software that is compatible with that running on System A.
- ✔ Adding Ethernet cards for private heartbeat network for System B
- ✔ Preparing System A by adding, if necessary, an Ethernet card for the private heartbeat network, and making the Ethernet cable connections between the two systems
- ✔ Connecting both systems to shared storage
- ✔ Bringing up VCS on System A and editing the configuration file
- ✔ Installing VCS on System B, if necessary, and updating the configuration files
- ✔ Editing the configuration files on the System A, starting LLT and GAB, restarting VCS, and modifying service groups for two systems
- ✔ Starting VCS on the System B
- ✔ Checking the new two-system cluster

Setting Up a System to Join the Single System Cluster

The new system to join the existing single system running VCS must run the same version of Solaris and have the same patch level.

- If VCS is not currently running on System B, proceed to "Installing VxVM, VxFS if Necessary" on page 168.
- If the system you plan to add as System B is currently part of an existing cluster, remove the system from the cluster, referring to "Removing a Node from a Cluster" on page 159. After removing the node from the cluster, remove the VCS packages and configuration files as described in that section.
- If the system you plan to add as System B is also currently a single VCS system, uninstall VCS (refer to "Removing VCS Packages Using pkgrm" on page 95), omitting the steps to unconfigure and unload GAB and LLT. If you renamed the LLT and GAB startup files (see "Renaming the LLT and GAB Startup Files" on page 166), remove them. Proceed to "Installing VxVM, VxFS if Necessary."

Installing VxVM, VxFS if Necessary

If VxVM with the cluster option or VxFS with the cluster option is installed on the existing system in the cluster, then the same versions must also be installed on the new system.

Refer to the appropriate documentation for VxVM and VxFS to verify the versions of the installed products and make sure the same version is running on all systems that are to use any shared storage.

Installing and Configuring Ethernet Cards for Private Network

Both systems require Ethernet cards (NICs) that enable the private network. If both System A and System B have Ethernet cards installed, you can ignore this step.

For high availability, two separate NICs on each system should be used, such that the failure of one NIC doesn't prevent heartbeating from continuing.

Note The following procedure shows highlights of the procedure described in "Setting Up the Private Network" on page 10

1. Shut down VCS on System A:

   ```
   # hastop -local
   ```

2. Shut down the system to get to the `OK` prompt:

   ```
   # sync;sync;init 0
   ```

3. Install the Ethernet card on System A.

4. Install the Ethernet card on System B

5. Configure the Ethernet card on both systems.

6. Make the two Ethernet cable connections from System A to System B for the private networks.

7. Restart the systems.

Configuring the Shared Storage

Use the procedures described in "Preparing to Install VCS 4.1" for setting up shared storage ("Setting Up Shared Storage" on page 11) to make the connection to shared storage from System B. Configure VxVM on System B and reboot the system when you are prompted.

Bringing Up the Existing System

1. On System A, enter the command:

   ```
   ok boot -r
   ```

2. Log in as root user.

3. Make the VCS configuration writable:

   ```
   # haconf -makerw
   ```

4. Display the service groups currently configured:

   ```
   # hagrp -list
   ```

5. Freeze the service groups:

   ```
   # hagrp -freeze group -persistent
   ```

   Repeat this command for each service group listed in step 4.

6. Make the configuration read-only:

   ```
   # haconf -dump -makero
   ```

7. Stop VCS on System A:

   ```
   # hastop -local -force
   ```

8. Rename the GAB and LLT startup files to their original names so VCS can use them:

   ```
   # mv /etc/rc2.d/X92gab /etc/rc2.d/S92gab
   # mv /etc/rc2.d/X70llt /etc/rc2.d/S70llt
   ```

Installing VCS on the New System

System B must be running the same version of VCS as the version on System A.

Follow the procedures in "Installing VCS Software Manually" on page 76 to manually install the VCS 4.1 packages and "Adding a License Key" on page 81 to install the license key. Return to this sections to continue adding a system to the single-system cluster.

Create Configuration Files on New System

1. Create the file `/etc/lltab` that lists both systems. Refer to "Setting Up /etc/lltab" on page 83.

2. Create the file `/etc/llthosts`. Refer to "Setting Up /etc/llthosts" on page 83. Set up `/etc/llthosts` for a two-system cluster.

3. Create the file `/etc/gabtab`. Refer to "Configuring Group Membership and Atomic Broadcast (GAB)" on page 86.

4. Start LLT on System B:

   ```
   # /etc/rc2.d/S70llt start
   ```

5. Start GAB on System B:

   ```
   # /etc/rc2.d/S92gab start
   ```

Reconfiguring VCS on the Existing System

1. On System A, create the files `/etc/lltab`, `/etc/llthosts`, and `/etc/gabtab`. Use the files created on System B as a guide, customizing the `/etc/lltab` for System A.

2. Start LLT on System A:

   ```
   # /etc/rc2.d/S70llt start
   ```

3. Start GAB on System A:

   ```
   # /etc/rc2.d/S92gab start
   ```


4. Check the membership of the cluster:

   ```
   # gabconfig -a
   ```

5. Start VCS on System A:

   ```
   # hastart
   ```

6. Make the VCS configuration writable:

   ```
   # haconf -makerw
   ```

7. Add System B to the cluster:

   ```
   # hasys -add sysB
   ```

8. Add System B to the system list of each service group:

   a. List the service groups:

      ```
      # hagrp -list
      ```

   b. For each service group listed, add the system:

      ```
      # hagrp -modify group SystemList -add sysB 1
      ```

Verifying Configuration on Both Systems

1. On System B, check the cluster membership:

   # **gabconfig -a**

2. Start the VCS on System B:

   # **hastart**

3. Verify that VCS is up on both systems:

   # **hastatus**

4. List the service groups:

   # **hagrp -list**

5. Unfreeze the service groups:

   # **hagrp -unfreeze *group* -persistent**

6. Implement the new two-system configuration:

   # **haconf -dump -makero**

Advanced Topics Related to Installing VCS A

This appendix contains procedures that may not be necessary for all users. It covers:

Reconciling Major/Minor Numbers for NFS Shared Disks

Your configuration may include disks on the shared bus that support NFS. File systems exported by NFS can be configured on disk partitions or on VERITAS Volume Manager volumes. An example disk partition name is `/dev/dsk/c1t1d0s3`. An example volume name is `/dev/vx/dsk/shareddg/vol3`. Each name represents the block device on which the file system is to be mounted.

In a VCS cluster, block devices providing NFS service must have the same major and minor numbers on each system. Major numbers identify the drivers required by the devices (such as a Solaris partition or a VxVM volume), and minor number identify the specific devices themselves. NFS also uses them to identify the exported file system.

Major and minor numbers must be checked to ensure that the NFS identity for the file system is the same when exported from each system.

Checking Major and Minor Numbers for Disk Partitions

The following sections describe checking and changing, if necessary, the major and minor numbers for disk partitions used by cluster systems.

▼ To check major and minor numbers on disk partitions

1. Use the following command on all systems exporting an NFS file system. This command displays the major and minor numbers for the block device.

   ```
   # ls -lL block_device
   ```

   The variable *block_device* refers to a partition on which a file system is mounted for export via NFS. Use this command on each NFS file system. For example, type:

   ```
   # ls -lL /dev/dsk/c1t1d0s3
   ```

   Output on System A resembles:

   ```
   crw-r-----  1 root  sys  32,1 Dec 3 11:50 /dev/dsk/c1t1d0s3
   ```

   Output on System B resembles:

   ```
   crw-r-----  1 root  sys  32,1 Dec 3 11:55 /dev/dsk/c1t1d0s3
   ```

   Note that the major numbers (32) and the minor numbers (1) match, satisfactorily meeting the requirement for NFS file systems.

▼ To reconcile major numbers that do not match on disk partitions

If the major and minor do not match, do the following. For example, suppose the output in the previous section was:

Output on System A resembles:

```
crw-r-----  1 root  sys  32,1 Dec 3 11:50 /dev/dsk/c1t1d0s3
```

Output on System B resembles:

```
crw-r-----  1 root  sys  36,1 Dec 3 11:55 /dev/dsk/c1t1d0s3
```

1. Place the VCS command directory in your path. For example:

   ```
   # export PATH=$PATH:/usr/sbin:/sbin:/opt/VRTS/bin
   ```

2. Attempt to change the major number on System B (now 36) to match that of System A (32). Use the command:

   ```
   haremajor -sd major_number
   ```

   For example, on System B, enter:

   ```
   # haremajor -sd 32
   ```

3. If the command succeeds, go to step 7.

4. If the command fails, you may receive a message resembling:

```
Error: Preexisiting major number 32
These are available numbers on this system: 128...
Check /etc/name_to_major on all systems for
available numbers.
```

5. Notice that the number 36 (the major number on System A) is not available on System B. Run the `haremajor` command on System B and change it to 128,

```
# haremajor -sd 128
```

6. Run the same command on System A. If the command fails on System A, the output lists the available numbers. Rerun the command on both systems, setting the major number to one available to both.

7. Reboot each system on which the command succeeds.

8. Proceed to reconcile the major numbers for your next partition.

▼ To reconcile minor numbers that do not match on disk partitions

If the minor numbers for NFS file systems in disk partitions do not match, you can use the following procedure:

1. Complete step a through step d, below. In this example, the minor numbers are `1` and `3` and the minor numbers are reconciled by setting each to 30.

   a. Type the following command on both systems using the name of your block device:

   ```
   # ls -1 /dev/dsk/c1t1d0s3
   ```

   Output from this command resembles the following on System A:

   ```
   lrwxrwxrwx  1 root  root  83 Dec 3 11:50
     /dev/dsk/c1t1d0s3 -> ../../
     devices/sbus@1f,0/QLGC,isp@0,10000/sd@1,0:d,raw
   ```

   The `device name` (in bold, above) includes the slash following the word `devices`, and continues to, but does not include, the colon.

b. Type the following command on both systems to determine the instance numbers used by the SCSI driver:

```
# grep sd /etc/path_to_inst | sort -n -k 2,2
```

Output from this command resembles the following on System A:

```
"/sbus@1f,0/QLGC,isp@0,10000/sd@0,0" 0 "sd"
"/sbus@1f,0/QLGC,isp@0,10000/sd@1,0" 1 "sd"
"/sbus@1f,0/QLGC,isp@0,10000/sd@2,0" 2 "sd"
"/sbus@1f,0/QLGC,isp@0,10000/sd@3,0" 3 "sd"
.
.
"/sbus@1f,0/SUNW,fas@e,8800000/sd@d,0" 27 "sd"
"/sbus@1f,0/SUNW,fas@e,8800000/sd@e,0" 28 "sd"
"/sbus@1f,0/SUNW,fas@e,8800000/sd@f,0" 29 "sd"
```

In the output, the instance numbers are in the second field. The instance number associated with the device name that matches the name for System A displayed in step a, is "1."

c. Compare instance numbers the device in the output on each system.

- If the instance number from one system is not used on the other (that is, it does not appear in the output of step b), edit `/etc/path_to_inst` to make the second system's instance number equal to that of the first system.
- If the instance numbers are being used on both systems, edit `/etc/path_to_inst` on both systems. Change the instance number associated with the device name to an unused number greater than the highest number used by other devices. For example, the output of step b shows the instance numbers used by all devices (from 0 to 29), so edit the file `/etc/path_to_inst` on each system and reset the instance numbers to 30.

d. Type the following command to reboot each system on which `/etc/path_to_inst` was modified:

```
# reboot -- -rv
```

Checking the Major and Minor Number for VxVM Volumes

The following sections describe checking and changing, if necessary, the major and minor numbers for VxVM volumes used by cluster systems.

▼ To check major and minor numbers on VxVM volumes

1. Place the VCS command directory in your path. For example:

   ```
   # export PATH=$PATH:/usr/sbin:/sbin:/opt/VRTS/bin
   ```

2. To list the devices, use the `ls -lL` *`block_device`* command on each system:

   ```
   # ls -lL /dev/vx/dsk/shareddg/vol3
   ```

   On System A, the output may resemble:

   ```
   brw-------   1 root  root  32,43000 Mar 22 16:41
   /dev/vx/dsk/shareddg/vol3
   ```

   On System B, the output may resemble:

   ```
   brw-------   1 root  root  36,43000 Mar 22 16:41
   /dev/vx/dsk/shareddg/vol3
   ```

3. Import the associated shared disk group on each system.

4. Use the following command on each system exporting an NFS file system. The command displays the major numbers for `vxio` and `vxspec` used by VERITAS Volume Manager (other major numbers are also displayed, but only `vxio` and `vxspec` are of concern for reconciliation):

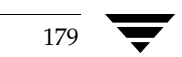
   ```
   # grep vx /etc/name_to_major
   ```

   Output on System A would resemble:

   ```
   vxdmp 30
   vxio 32
   vxspec 33
   vxfen 87
   vxglm 91
   ```

   On System B:

   ```
   vxdmp 30
   vxio 36
   vxspec 37
   vxfen 87
   vxglm 91
   ```

5. To change System B's major numbers for `vxio` and `vxspec` to match those of System A, use the command:

   ```
   haremajor -vx major_number_vxio major_number_vxspec
   ```

   For example, enter:

   ```
   # haremajor -vx 32 33
   ```

   If the command succeeds, proceed to step 8. If this command fails, you receive a report similar to the following:

   ```
   Error: Preexisiting major number 32
   These are available numbers on this system: 128...
   Check /etc/name_to_major on all systems for
   available numbers.
   ```

6. If you receive this report, use the `haremajor` command on System A to change the major number (`32/33`) to match that of System B (`36/37`). For example, enter:

   ```
   # haremajor -vx 36 37
   ```

   If the command fails again, you receive a report similar to the following:

   ```
   Error: Preexisiting major number 36
   These are available numbers on this node: 126...
   Check /etc/name_to_major on all systems for
   available numbers.
   ```

7. If you receive the second report, choose the larger of the two available numbers (in this example, `128`), and use this number in the `haremajor` command to reconcile the major numbers. Type the following command on both systems:

   ```
   # haremajor -vx 128 129
   ```

8. Reboot each system on which `haremajor` was successful.

9. If the minor numbers match, proceed to reconcile the major and minor numbers of your next NFS block device.

10. If the block device on which the minor number does not match is a volume, consult the `vxdg`(1M) manual page for instructions on reconciling the VERITAS Volume Manager minor numbers, with specific reference to the `reminor` option.

Systems where the vxio driver number have been changed require rebooting.

Upgrading Solaris Versions

The operating system upgrade may take hours to finish. We recommend coordinating with your system administrator to plan the outage time of the other system. This helps reduce downtime and ensures availability of services for your customers.

When you upgrade the operating system, you must remove the GAB and LLT packages before upgrading the operating system, and reinstall GAB and LLT after upgrading the operating system.

Note Be sure that you have the VERITAS software disc with the VCS software, including the GAB and LLT packages, on hand before you begin.

▼ To stop VCS

1. Make the VCS configuration writable. On the first system, type:

   ```
   # haconf -makerw
   ```

2. Move all service groups from the system you are upgrading to another system and keep services from failing over to this server. On the system you are upgrading, type:

   ```
   # hasys -freeze -persistent -evacuate upgrade_server
   ```

3. Check if all service groups and resources are OFFLINE on the this system and ONLINE on the other system. Type:

   ```
   # hastatus -summary
   ```

4. Close the configuration and unload the VCS services on the system you are upgrading. On the upgrading system, type:

   ```
   # haconf -dump -makero
   # hastop -local
   ```

5. Confirm that VCS has shut down. On the upgrading system, type:

   ```
   # gabconfig -a
   ```

   Output resembles:

   ```
   GAB Port Memberships
   ===================================
   Port a gen 23dc0001 membership 01
   ```

   Note that the output shows no membership for port h.

▼ To stop GAB and LLT, unload the kernel modules, and remove packages

1. Unconfigure GAB. Type:

   ```
   # gabconfig -U
   ```

2. Unload the GAB module from the kernel.

   a. Determine the kernel module ID:

      ```
      # modinfo | grep gab
      ```

   b. Unload the module from the kernel:

      ```
      # modunload -i gab_id
      ```

3. Unconfigure LLT. On each system, type:

   ```
   # lltconfig -U
   ```

   The following message is displayed on the console:

   ```
   lltconfig: this will attempt to stop and reset LLT.
   Confirm (y/n)?
   ```

4. Type **Y** on each system in response to the message.

5. Unload the LLT module from the kernel.

   a. Determine the kernel module ID:

      ```
      # modinfo | grep llt
      ```

      The module IDs are in the left-hand column of the output.

   b. Unload the module from the kernel:

      ```
      # modunload -i llt_id
      ```

6. On each system, use the `pkgrm` command to remove the GAB and LLT packages:

   ```
   # pkgrm VRTSgab VRTSllt
   ```

▼ To upgrade Solaris

1. Follow the Sun installation guide to upgrade the operating system kernel to the new version of Solaris.

2. As the system comes up, enter single-user mode.

▼ To reinstall GAB, LLT from the software disc and restart

1. In single-user mode, log in as root user on the system you are upgrading.

2. Check whether the `/tmp` directory is mounted.

   ```
   # mount
   ```

   If the `/tmp` directory is not mounted, then enter:

   ```
   # mount /tmp
   ```

3. Create a directory for installation:

   ```
   # mkdir /tmp/install
   ```

4. Insert the software disc with the VCS software into a drive connected to the system you are upgrading. The Solaris volume-management software automatically mounts the disc as `/cdrom/cdrom0`. Type the command:

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

5. Copy the compressed package files from the software disc to the temporary directory:

   ```
   # cp -r cluster_server/pkgs/VRTSllt.tar.gz /tmp/install
   # cp -r cluster_server/pkgs/VRTSgab.tar.gz /tmp/install
   ```

6. Go to the temporary directory and unzip the compressed package files:

   Note If your system does not have the `gunzip` utility, copy it from the disc:
   `# cp /cdrom_path/gnu/gunzip /tmp/install`

   ```
   # cd /tmp/install
   # gunzip VRTSllt.tar.gz
   # gunzip VRTSgab.tar.gz
   ```

   The following files are now present in the temporary directory:

   ```
   VRTSgab.tar
   VRTSllt.tar
   ```

7. Extract the required VCS files from the compressed files:

   ```
   # tar -xvf VRTSllt.tar
   # tar -xvf VRTSgab.tar
   ```

8. Install the LLT and GAB packages. As you enter the command, be sure to list the packages in the order shown in the following example:

   ```
   # pkgadd -d . VRTSllt VRTSgab
   ```


9. Bring system up in multi-user mode:

```
# cd /
# init 3
```

10. Verify VCS services are running on the upgraded server. On the upgraded server, type:

```
# ps -ef | grep ha
root   576   1  0 16:54:12 ?     0:02 /opt/VRTSvcs/bin/had
root   578   1  0 16:54:13 ?     0:00 /opt/VRTSvcs/bin/hashadow
```

If they are not running, reload the VCS services. Type:

```
# hastart
```

11. Unfreeze the upgraded server and save the configuration. On the upgraded server, type:

```
# hasys -unfreeze -persistent upgraded_server
# haconf -dump -makero
```

▼ **To complete the Solaris operating system upgrade on other systems**

Beginning with "To stop VCS" on page 181, repeat the procedure to upgrade the OS on each other system in the cluster.

LLT Over UDP

VCS 4.1 provides the option of using LLT over the UDP (User Datagram Protocol) layer for clusters using wide-area networks and routers. UDP makes LLT packets routable and thus able to span longer distances more economically.

Note LLT over UDP is not supported on IPV6.

When to Use LLT Over UDP

Use LLT over UDP when:

- LLT must be used over WANs
- When hardware, such as blade servers, do not support LLT over Ethernet

Performance Considerations

Because LLT over UDP is slower that LLT over Ethernet, LLT over UDP should only be used when the hardware configuration makes it necessary.

Configuring LLT over UDP

The following is a checklist for configuring LLT over UDP. Examples are provided in the sections that follow.

- ✔ Make sure that each NIC has an IP address configured before configuring LLT. Each link must be in a different subnet. See the examples in the following sections.
- ✔ Make sure that each link has a unique non-well known UDP port; see "Selecting UDP Ports" on page 186.
- ✔ Set the broadcast address correctly for direct-attached (non-routed) links.
- ✔ For links that cross an IP router, disable broadcast features and specify the IP address of each link manually in the `/etc/lltab` file. See "Sample Configuration: Links Crossing IP Routers" on page 189.

The link Command in the /etc/lltab File

The following table describes the fields of the `link` command shown in the `/etc/lltab` file examples that follow; see "Sample Configuration: Direct-Attached Links" on page 188, and "Sample Configuration: Links Crossing IP Routers" on page 189. Note that some of these fields differ from the command for standard LLT links.

`<tag-name>`	A unique string that is used as a tag by LLT; for example `link1`, `link2`,...
`<device>`	The device path of the UDP protocol; for example `/dev/udp`
`<node-range>`	Nodes using the link. "`-`" indicates *all* cluster nodes are to be configured for this link.
`<link-type>`	Type of link; must be "`udp`" for LLT over UDP
`<udp-port>`	Unique UDP port in range of 49152-65535 for the link; see "Selecting UDP Ports" on page 186.
`<MTU>`	"`-`" is the default, which has a value of 8192. The value may be increased or decreased depending on the configuration. The `lltstat -l` command can display the current value.

`<IP address>`	IP address of the link on the local node.
`<bcast-address>`	• for clusters having broadcasts enabled, specify the value of the subnet broadcast address • "-" is the default for clusters spanning routers

The set-addr Command in the /etc/lltab File

The `set-addr` command in the `/etc/lltab` file is required when the broadcast feature of LLT is disabled, such as when LLT must cross IP routers. The following table describes the fields of the `set-addr` command; see "Sample Configuration: Links Crossing IP Routers" on page 189.

`<node-id>`	The ID of the cluster node; for example, 0.
`<link tag-name>`	The string used by LLT to identify the link; for example `link1`, `link2`,...
`<address>`	IP address assigned to the link on the peer node.

Selecting UDP Ports

When selecting a UDP port, select an available 16-bit integer from the range described below.

- Use available ports (that is, ports that are not in use)] in the private range 49152 to 65535
- Do not use:
  - Ports from the range of well-known ports, 0 to 1023
  - Ports from the range of registered ports, 1024 to 49151

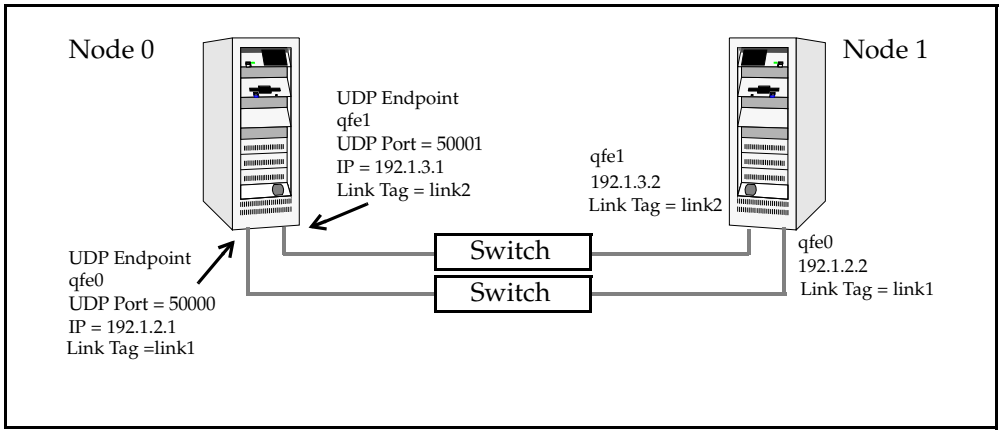
To check which ports are defined as defaults for a node, examine the file `/etc/services`. You should also use the `netstat` command to list the ports currently in use. For example:

```
# netstat -a | more
UDP
   Local Address         Remote Address      State
-------------------- -------------------- -------
      *.sunrpc                            Idle
      *.*                                 Unbound
      *.32771                             Idle
      *.32776                             Idle
      *.32777                             Idle
      *.name                              Idle
      *.biff                              Idle
      *.talk                              Idle
      *.32779                             Idle
.
.
.
      *.55098                             Idle
      *.syslog                            Idle
      *.58702                             Idle
      *.*                                 Unbound
```

Look in the UDP section of the output; UDP ports listed under `Local Address` are already in use. If a port is listed in the `/etc/services` file, its associated name is displayed rather than the port number in the output of `netstat -a`.

Sample Configuration: Direct-Attached Links

The following illustration depicts a typical configuration of direct-attached links employing LLT over UDP.



The configuration represented by the following `/etc/lltab` file for Node 0 has directly attached crossover links or links connected through a hub or switch. These links do not cross routers.

Because LLT broadcasts requests to peer nodes to discover their addresses, the addresses of peer nodes do not need to be specified in the `/etc/lltab` file using the `set-addr` command. For direct attached links, you do need to set the broadcast address of the links in the `/etc/lltab` file. Verify that the IP addresses and broadcast addresses are set correctly by using the `ifconfig -a` command.

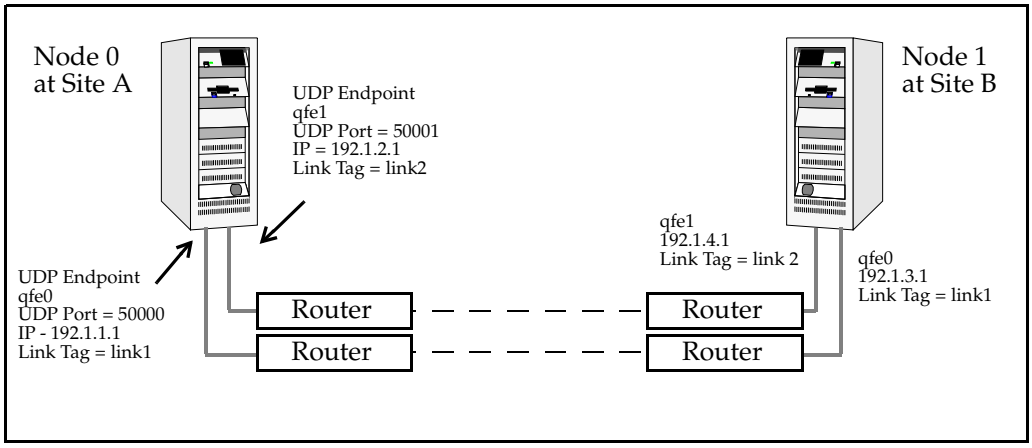
```
set-node Node0
set-cluster 1
#configure Links
#link <tag-name> <device> <node-range> <link-type> <udp port> <MTU>
<IP-address> <bcast-address>
link link1 /dev/udp - udp 50000 - 192.1.2.1 192.1.2.255
link link2 /dev/udp - udp 50001 - 192.1.3.1 192.1.3.255
```

The file for Node 1 would resemble:

```
set-node Node1
set-cluster 1
#configure Links
#link <tag-name> <device> <node-range> <link-type> <udp port> <MTU>
<IP-address> <bcast-address>
link link1 /dev/udp - udp 50000 - 192.1.2.2 192.1.2.255
link link2 /dev/udp - udp 50001 - 192.1.3.2 192.1.3.255
```

Sample Configuration: Links Crossing IP Routers

The following illustration depicts a typical configuration of links crossing an IP router employing LLT over UDP. The illustration shows just two nodes of a four-node cluster.



The configuration represented by the following `/etc/lltab` file for Node 1 has links crossing IP routers. Notice that IP addresses are shown for each link on each peer node. The broadcast features are disabled because LLT is unable to broadcast requests for addresses across routers. Since broadcasts are disabled, the broadcast address does not need to be set in the in the `link` command of the `/etc/lltab` file.

```
set-node Node1
set-cluster 1

link link1 /dev/udp - udp 50000 - 192.1.3.1 -
link link2  /dev/udp - udp 50001 - 192.1.4.1 -

#set address of each link for all peer nodes in the cluster
#format: set-addr <node-id> <link tag-name> <address>
set-addr        0 link1 192.1.1.1
set-addr        0 link2 192.1.2.1
set-addr        2 link1 192.1.5.2
set-addr        2 link2 192.1.6.2
set-addr        3 link1 192.1.7.3
set-addr        3 link2 192.1.8.3

#disable LLT broadcasts
set-bcasthb     0
set-arp         0
```


The `/etc/lltab` file on Node 0 would resemble:

```
set-node Node0
set-cluster 1

link link1 /dev/udp - udp 50000 - 192.1.1.1 -
link link2 /dev/udp - udp 50001 - 192.1.2.1 -

#set address of each link for all peer nodes in the cluster
#format: set-addr <node-id> <link tag-name> <address>
set-addr        1 link1 192.1.3.1
set-addr        1 link2 192.1.4.1
set-addr        2 link1 192.1.5.2
set-addr        2 link2 192.1.6.2
set-addr        3 link1 192.1.7.3
set-addr        3 link2 192.1.8.3

#disable LLT broadcasts
set-bcasthb     0
set-arp         0
```

Upgrading From VCS QuickStart B

This appendix describes procedures to upgrade from VCS QuickStart 3.5 to VCS 4.1.

Upgrading From VCS QuickStart 3.5

Highlights of the upgrade procedure are:

- ✔ Uninstall VCS QuickStart 3.5 from all cluster systems using the VCS QuickStart 3.5 `uninstallvcs` program.
- ✔ Save previous configuration files
- ✔ Install VCS 4.1 using `installvcs` with the `-installonly` option.
- ✔ Restore the previous configuration files for use in the VCS 4.1 environment
- ✔ Start LLT, GAB, and VCS manually.
- ✔ Update user passwords.

Uninstall VCS QuickStart 3.5

Uninstall VCS QuickStart 3.5 using the `uninstallvcs` program. When the program completes, the VCS QuickStart packages are removed from the system, but the VCS configuration files remain. This enables you to use the existing cluster configuration for VCS 4.1.

```
# cd /opt/VRTS/install
# ./uninstallvcs
```

Use the program to remove VCS QuickStart from each cluster node.

Saving the Existing Configuration Files

Save the configuration files used with VCS QuickStart 3.5.

1. Copy the `types.cf` file to a file named `types.save`. Put the copy in the directory `/etc/VRTSvcs/conf`.

```
# cp /etc/VRTSvcs/conf/config/types.cf
        /etc/VRTSvcs/conf/types.save
```

2. Copy the `main.cf` file to a file named `main.save`. Put the copy in the directory `/etc/VRTSvcs/conf`.

```
# cp /etc/VRTSvcs/conf/config/main.cf
        /etc/VRTSvcs/conf/main.save
```

Install VCS 4.1 Using -installonly Option

▼ To mount the disc

1. Log in as root user on one of the systems where VCS is to be installed.

2. Insert the software disc with the VCS software into a drive connected to the system. The Solaris volume-management software automatically mounts the disc as `/cdrom/cdrom0`. Type the command:

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

▼ To install VCS 4.1 using -installonly option

1. Change to the directory where you can start the `installvcs` program:

```
# cd cluster_server
```

2. Start the VCS installation program by entering:

```
# ./installvcs -installonly
```

The installer starts by discovering the existing configuration files. For example:

```
VCS configuration files exist on this system with the following
information:

  Cluster Name: qs_clus1
  Cluster ID: 14
  Systems: pacific atlantic
  Service Groups: ClusterService
```

3. The installer continues, making initial checks for system-to-system communications capability and file system space. The installer sets up logs at:

   `/var/tmp/inmstallvcs`*`date_and_time`*

4. The installer checks for the existence of infrastructure packages, `VRTSvlic` and `VRTScpi`. It install them, replacing older versions if they exist.

5. While verifying licenses that exist on the systems, the installer alerts you that VCS QuickStart is not supported with VCS 4.1.

   ```
   VCS QuickStart is not supported with VCS 4.1.

   To upgrade VCS QuickStart to VCS 4.1, uninstall VCS QuickStart
   from your systems and install VCS using VCS license keys.
   ```

   If you have already uninstalled VCS QuickStart (see "Uninstall VCS QuickStart 3.5" on page 191), you can proceed to add the VCS 4.1 licenses.

6. The installer prompts you to enter VCS license keys for each system. Enter them and continue the installation.

7. After the installer completes VCS licensing, the installer:

   - Prompts you to indicate whether or not to install optional packages
   - Lists the required and selected optional packages it is to install
   - Checks the systems for existing versions of the packages and verifies the file system space
   - Installs the packages after you indicate whether to install them consecutively or simultaneously
   - Indicates the names and locations for summary, log, and response files it creates at the conclusion of installation

   Note Ignore the instruction about running `installvcs` with the `-configure` option.

Restoring QuickStart 3.5 Configuration for use with VCS 4.1

Note Perform the following step 1 and step 2 on any system in the cluster.

1. Check to see whether you need to merge any types defined in your previous installation with the newly installed types file.

   The `types.cf` file installed with VCS 4.1 contains new type definitions. Compare the saved `types.cf` file (`types.save`) created in step 1 on page 192 to the newly installed `/etc/VRTSvcs/conf/types.cf`:

   ```
   # diff -w /etc/VRTSvcs/conf/types.save
                 /etc/VRTSvcs/conf/types.cf
   ```

   a. If the only differences you see are the new types defined in the newly installed **`/etc/VRTSvcs/conf/types.cf`** file, then you don't need to restore the contents of the file `types.save`.

   Note If the files are very different from each other, the output of the `diff` command may be too confusing to read. In this case, print and compare the two files manually.

   b. If the differences include any types defined in the `types.save` file, then you must edit the newly installed **`/etc/VRTSvcs/conf/types.cf`** file, adding the types used in your previous VCS configuration.

   c. Copy the appropriate `types.cf` file to `/etc/VRTSvcs/conf/config/types.cf`.

2. Edit your `main.cf` file to add a `ClusterAddress` definition and upgrade the `ClusterService` group. (Make sure you have backed up the original file; see step 2 on page 192). You can use the "main.cf Example, for Clusters Without the GCO Option" on page 100 for reference. Using `vi` or another editor, make the following changes:

   a. In the "`cluster`" definition section, beneath the line that defines the `UserNames`, add a line that defines the cluster's virtual IP address. For example:

   ```
   ClusterAddress = "11.136.88.199"
   ```

   b. In the `ClusterService` group, under the line that defines the `OnlineRetryLimit`, add the following line:

   ```
   OnlineRetryInterval = 120
   ```

c. In the `ClusterService` group, change the application name (`AppName`) from `vcsqs` to `vcs`. For example:

```
VRTSWebApp VCSweb (
        Critical =0
        AppName = vcs
        InstallDir = "/opt/VRTSweb/VERITAS"
        TimeForOnline = 5
        )
```

d. Re-verify the syntax of the `main.cf` file:

```
# cd /etc/VRTSvcs/conf/config
# hacf -verify .
```

Starting LLT, GAB, and VCS

After installing the VCS 4.1 packages, refer to the appropriate sections in this document to start LLT (see "Starting LLT" on page 92), start GAB (see "Starting GAB" on page 93), and to start VCS (see "Starting VCS" on page 93).

Note Start VCS on the system where the `types.cf` and `main.cf` files are restored first. VCS can then be started on the other nodes.

Updating User Passwords

When VCS is running, you must update your user passwords.

1. Make the VCS configuration writable:

   ```
   # haconf -makerw
   ```

2. Update the passwords for all users in the cluster. For each user, use the `hauser` command:

   ```
   hauser -update username
   ```

   For example:

   ```
   # hauser -update admin
   Enter New Password:********

   Enter Again:********
   # hauser -update smith
   Enter New Password:********

   Enter Again:********
   ```


3. Save the VCS configuration:

```
# haconf -dump -makero
```

Index