Veritas Storage Foundation™ for Oracle® RAC Administrator's Guide

Linux

5.1 Service Pack 1

Symantec

Veritas Storage Foundation™ for Oracle RAC Administrator's Guide

The software described in this book is furnished under a license agreement and may be used only in accordance with the terms of the agreement.

Product version: 5.1 SP1

Document version: 5.1SP1.1

Legal Notice

Copyright © 2010 Symantec Corporation. All rights reserved.

Symantec, the Symantec logo, Veritas, Veritas Storage Foundation, CommandCentral, NetBackup, Enterprise Vault, and LiveUpdate are trademarks or registered trademarks of Symantec corporation or its affiliates in the U.S. and other countries. Other names may be trademarks of their respective owners.

The product described in this document is distributed under licenses restricting its use, copying, distribution, and decompilation/reverse engineering. No part of this document may be reproduced in any form by any means without prior written authorization of Symantec Corporation and its licensors, if any.

THE DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID. SYMANTEC CORPORATION SHALL NOT BE LIABLE FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES IN CONNECTION WITH THE FURNISHING, PERFORMANCE, OR USE OF THIS DOCUMENTATION. THE INFORMATION CONTAINED IN THIS DOCUMENTATION IS SUBJECT TO CHANGE WITHOUT NOTICE.

The Licensed Software and Documentation are deemed to be commercial computer software as defined in FAR 12.212 and subject to restricted rights as defined in FAR Section 52.227-19 "Commercial Computer Software - Restricted Rights" and DFARS 227.7202, "Rights in Commercial Computer Software or Commercial Computer Software Documentation", as applicable, and any successor regulations. Any use, modification, reproduction release, performance, display or disclosure of the Licensed Software and Documentation by the U.S. Government shall be solely in accordance with the terms of this Agreement.

Symantec Corporation
350 Ellis Street
Mountain View, CA 94043

http://www.symantec.com

Technical Support

Symantec Technical Support maintains support centers globally. Technical Support's primary role is to respond to specific queries about product features and functionality. The Technical Support group also creates content for our online Knowledge Base. The Technical Support group works collaboratively with the other functional areas within Symantec to answer your questions in a timely fashion. For example, the Technical Support group works with Product Engineering and Symantec Security Response to provide alerting services and virus definition updates.

Symantec's support offerings include the following:

- A range of support options that give you the flexibility to select the right amount of service for any size organization
- Telephone and/or Web-based support that provides rapid response and up-to-the-minute information
- Upgrade assurance that delivers software upgrades
- Global support purchased on a regional business hours or 24 hours a day, 7 days a week basis
- Premium service offerings that include Account Management Services

For information about Symantec's support offerings, you can visit our Web site at the following URL:

www.symantec.com/business/support/index.jsp

All support services will be delivered in accordance with your support agreement and the then-current enterprise technical support policy.

Contacting Technical Support

Customers with a current support agreement may access Technical Support information at the following URL:

www.symantec.com/business/support/contact_techsupp_static.jsp

Before contacting Technical Support, make sure you have satisfied the system requirements that are listed in your product documentation. Also, you should be at the computer on which the problem occurred, in case it is necessary to replicate the problem.

When you contact Technical Support, please have the following information available:

- Product release level

- Hardware information
- Available memory, disk space, and NIC information
- Operating system
- Version and patch level
- Network topology
- Router, gateway, and IP address information
- Problem description:
  - Error messages and log files
  - Troubleshooting that was performed before contacting Symantec
  - Recent software configuration changes and network changes

Licensing and registration

If your Symantec product requires registration or a license key, access our technical support Web page at the following URL:

www.symantec.com/business/support/

Customer service

Customer service information is available at the following URL:

www.symantec.com/business/support/

Customer Service is available to assist with non-technical questions, such as the following types of issues:

- Questions regarding product licensing or serialization
- Product registration updates, such as address or name changes
- General product information (features, language availability, local dealers)
- Latest information about product updates and upgrades
- Information about upgrade assurance and support contracts
- Information about the Symantec Buying Programs
- Advice about Symantec's technical support options
- Nontechnical presales questions
- Issues that are related to CD-ROMs or manuals

Support agreement resources

If you want to contact Symantec regarding an existing support agreement, please contact the support agreement administration team for your region as follows:

Asia-Pacific and Japan	customercare_apac@symantec.com
Europe, Middle-East, and Africa	semea@symantec.com
North America and Latin America	supportsolutions@symantec.com

Documentation

Product guides are available on the media in PDF format. Make sure that you are using the current version of the documentation.The document version appears on page 2 of each guide. The latest product documentation is available on the Symantec website.

http://www.symantec.com/business/support/overview.jsp?pid=15107

Your feedback on product documentation is important to us. Send suggestions for improvements and reports on errors or omissions. Include the title and document version (located on the second page), and chapter and section titles of the text on which you are reporting. Send feedback to:

docs@symantec.com

About Symantec Connect

Symantec Connect is the peer-to-peer technical community site for Symantec's enterprise customers. Participants can connect and share information with other product users, including creating forum posts, articles, videos, downloads, blogs and suggesting ideas, as well as interact with Symantec product teams and Technical Support. Content is rated by the community, and members receive reward points for their contributions.

http://www.symantec.com/connect/storage-management

Contents

Technical Support 4

Section 1 SF Oracle RAC concepts and administration 15

Chapter 1 Overview of Veritas Storage Foundation for Oracle RAC 17

About Veritas Storage Foundation for Oracle RAC 17
Benefits of SF Oracle RAC 17
How SF Oracle RAC works (high-level perspective) 19
Component products and processes of SF Oracle RAC 23
Communication infrastructure 23
Cluster interconnect communication channel 25
Low-level communication: port relationship between GAB and processes 29
Cluster Volume Manager (CVM) 30
Cluster File System (CFS) 32
Oracle Disk Manager 34
Veritas Cluster Server 35
Oracle RAC 38
RAC extensions 41
About preventing data corruption with I/O fencing 42
About SCSI-3 Persistent Reservations 43
About I/O fencing operations 44
About I/O fencing communication 44
About I/O fencing components 44
About preferred fencing 46
How disk-based I/O fencing works in different event scenarios 48
About CP server 52
Periodic health evaluation of the clusters 70

Chapter 2 Administering SF Oracle RAC and its components 73

Administering SF Oracle RAC 73
Setting the environment variables 74
Starting or stopping SF Oracle RAC on each node 75
Applying Oracle patches 79
Installing Veritas Volume Manager, Veritas File System, or ODM patches on SF Oracle RAC nodes 80
Applying operating system updates 81
Adding storage to an SF Oracle RAC cluster 82
Recovering from storage failure 83
Enhancing the performance of SF Oracle RAC clusters 84
Creating snapshots for offhost processing 84
Managing database storage efficiently using SmartTier 85
Optimizing database storage using Thin Provisioning and SmartMove 85
Scheduling periodic health checks for your cluster 85
Verifying the ODM port 86
Verifying the nodes in a cluster 86
Administering VCS 89
Viewing available Veritas device drivers 90
Verifying VCS configuration 91
Starting and stopping VCS 91
Environment variables to start and stop VCS modules 91
Adding and removing LLT links 93
Configuring aggregated interfaces under LLT 97
Displaying the cluster details and LLT version for LLT links 99
Configuring destination-based load balancing for LLT 100
Enabling and disabling intelligent resource monitoring 100
Administering the AMF kernel driver 102
Administering I/O fencing 103
About administering I/O fencing 104
About the vxfentsthdw utility 104
About the vxfenadm utility 112
About the vxfenclearpre utility 117
About the vxfenswap utility 120
Enabling or disabling the preferred fencing policy 131
Administering the CP server 133
About the CP server user types and privileges 133
cpsadm command 133
About administering the coordination point server 137

Refreshing registration keys on the coordination points for server-based fencing 141
Replacing coordination points for server-based fencing in an online cluster 143
Migrating from non-secure to secure setup for CP server and SF Oracle RAC cluster communication 146
Administering CFS 148
Adding CFS file systems to VCS configuration 149
Using cfsmount to mount CFS file systems 149
Resizing CFS file systems 149
Verifying the status of CFS file systems 150
Verifying CFS port 150
Administering CVM 150
Establishing CVM cluster membership manually 151
Changing the CVM master manually 152
Importing a shared disk group manually 154
Deporting a shared disk group manually 154
Evaluating the state of CVM ports 154
Verifying if CVM is running in an SF Oracle RAC cluster 154
Verifying CVM membership state 155
Verifying the state of CVM shared disk groups 156
Verifying the activation mode 156
Administering SF Oracle RAC global clusters 156
Setting up a disaster recovery fire drill 157
Configuring the fire drill service group using the Fire Drill Setup wizard 157
Verifying a successful fire drill 159
Scheduling a fire drill 160
Sample fire drill service group configuration 160

Section 2 Performance and troubleshooting 163

Chapter 3 Troubleshooting SF Oracle RAC 165

About troubleshooting SF Oracle RAC 165
Running scripts for engineering support analysis 166
Log files 167
GAB message logging 169
Gathering LLT and GAB information for support analysis 170
About SF Oracle RAC kernel and driver messages 171
What to do if you see a licensing reminder 171
Restarting the installer after a failed connection 172
Installer cannot create UUID for the cluster 172

Troubleshooting installation and configuration check failures 172
Time synchronization checks failed with errors 172
System architecture checks failed with errors 173
Operating system and patch level synchronization checks failed with errors 173
CPU frequency checks failed with errors 174
LLT link mode checks failed with errors 174
LLT link jumbo frame setting checks failed with errors 175
Cross connection checks failed with errors 176
LLT link speed and auto negotiation checks failed with errors 176
LLT link priority checks failed with errors 177
Checks on /etc/llthosts file failed with errors 178
Checks on cluster ID failed with errors 179
Fencing configuration checks failed with errors 180
Checks on the presence of fencing keys on coordinator disks failed with errors 181
ODM configuration checks failed with errors 181
VCSMM configuration checks failed with errors 182
GAB ports configuration checks failed with errors 182
Oracle database version checks failed with errors 183
Checks on vendor clusterware failed with errors 184
Oracle Process Daemon (oprocd) checks failed with errors 184
Node ID checks for LLT and Oracle Clusterware failed with errors 185
Checks on tunables for LLT/VCSMM failed with errors 187
Oracle user checks failed with errors 188
CSSD resource checks failed with errors 189
Checks on equivalence of SF Oracle RAC cluster and RAC instances failed with errors 189
Kernel parameter checks failed with errors 190
Troubleshooting pre-installation check failures 191
Troubleshooting LLT health check warning messages 193
Troubleshooting I/O fencing 196
SCSI reservation errors during bootup 196
The vxfentsthdw utility fails when SCSI TEST UNIT READY command fails 196
Node is unable to join cluster while another node is being ejected 197
System panics to prevent potential data corruption 197
How vxfen driver checks for preexisting split-brain condition 197

Cluster ID on the I/O fencing key of coordinator disk does not match the local cluster's ID 199
Clearing keys after split-brain using vxfenclearpre command 200
Registered keys are lost on the coordinator disks 200
Replacing defective disks when the cluster is offline 200
Troubleshooting I/O fencing health check warning messages ... 2 0 3
Troubleshooting server-based I/O fencing 205
Troubleshooting server-based fencing on the SF Oracle RAC cluster nodes 207
Troubleshooting server-based I/O fencing in mixed mode 210
Understanding error messages 215
Troubleshooting CVM 215
Restoring communication between host and disks after cable disconnection 216
Shared disk group cannot be imported 216
Error importing shared disk groups 217
Unable to start CVM 217
CVM group is not online after adding a node to the cluster 217
CVMVolDg not online even though CVMCluster is online 218
Shared disks not visible 219
Troubleshooting CFS 219
Incorrect order in root user's <library> path 219
Troubleshooting interconnects 220
Example entries for mandatory devices 220
Troubleshooting Oracle 221
Error when starting an Oracle instance 221
Clearing Oracle group faults 221
Oracle log files show shutdown called even when not shutdown manually 222
DBCA fails while creating database 222
Oracle Clusterware processes fail to startup 222
Oracle Clusterware fails after restart 222
Removing Oracle Clusterware if installation fails 223
Troubleshooting the Virtual IP (VIP) Configuration 223
Loss of connectivity to OCR and voting disk causes the cluster to panic 224
OCRDUMP 224
Troubleshooting Oracle Clusterware health check warning messages 224
Troubleshooting ODM 226

File System configured incorrectly for ODM shuts down Oracle 226

Chapter 4 Prevention and recovery strategies 229

Verification of GAB ports in SF Oracle RAC cluster 229
Examining GAB seed membership 230
Manual GAB membership seeding 231
Evaluating VCS I/O fencing ports 232
Verifying normal functioning of VCS I/O fencing 233
Managing SCSI-3 PR keys in SF Oracle RAC cluster 233
Evaluating the number of SCSI-3 PR keys on a coordinator LUN, if there are multiple paths to the LUN from the hosts 234
Detecting accidental SCSI-3 PR key removal from coordinator LUNs 234
Identifying a faulty coordinator LUN 235
Starting shared volumes manually 235
Listing all the CVM shared disks 235
Failure scenarios and recovery strategies for CP server setup 235

Chapter 5 Tunable parameters 237

About SF Oracle RAC tunable parameters 237
About GAB tunable parameters 238
About GAB load-time or static tunable parameters 238
About GAB run-time or dynamic tunable parameters 240
About LLT tunable parameters 245
About LLT timer tunable parameters 246
About LLT flow control tunable parameters 250
About VXFEN tunable parameters 252
Configuring the VXFEN module parameters 253
Tuning guidelines for campus clusters 255

Section 3 Reference 257

Appendix A List of SF Oracle RAC health checks 259

LLT health checks 259
I/O fencing health checks 263
PrivNIC health checks 264
Oracle Clusterware health checks 265
CVM, CFS, and ODM health checks 266

Appendix B Error messages 269

About error messages 269
VxVM error messages 269
VXFEN driver error messages 270
VXFEN driver informational message 270
Node ejection informational messages 271

Glossary 273

Index 277

Section 1

SF Oracle RAC concepts and administration

- Chapter 1. Overview of Veritas Storage Foundation for Oracle RAC
- Chapter 2. Administering SF Oracle RAC and its components

Chapter 1

Overview of Veritas Storage Foundation for Oracle RAC

This chapter includes the following topics:

- About Veritas Storage Foundation for Oracle RAC
- How SF Oracle RAC works (high-level perspective)
- Component products and processes of SF Oracle RAC
- About preventing data corruption with I/O fencing
- Periodic health evaluation of the clusters

About Veritas Storage Foundation for Oracle RAC

Veritas Storage Foundation™ for Oracle® RAC (SF Oracle RAC) leverages proprietary storage management and high availability technologies to enable robust, manageable, and scalable deployment of Oracle RAC on UNIX platforms. The solution uses Veritas Cluster File System technology that provides the dual advantage of easy file system management as well as the use of familiar operating system tools and utilities in managing databases.

The solution stack comprises the Veritas Cluster Server (VCS), Veritas Cluster Volume Manager (CVM), Veritas Oracle Real Application Cluster Support (VRTSdbac), Veritas Oracle Disk Manager (VRTSodm), Veritas Cluster File System (CFS), and Veritas Storage Foundation, which includes the base Veritas Volume Manager (VxVM) and Veritas File System (VxFS).

Benefits of SF Oracle RAC

SF Oracle RAC provides the following benefits:

- Support for file system-based management. SF Oracle RAC provides a generic clustered file system technology for storing and managing Oracle data files as well as other application data.
- Support for high-availability of cluster interconnects.
  The PrivNIC/MultiPrivNIC agents provide maximum bandwidth as well as high availability of the cluster interconnects, including switch redundancy.
- Use of clustered file system and volume management technologies for placement of Oracle Cluster Registry (OCR) and voting disks. These technologies provide robust shared block interfaces for placement of OCR and voting disks. In the absence of SF Oracle RAC, separate LUNs need to be configured for OCR and voting disks.
- Support for a standardized approach toward application and database management. A single-vendor solution for the complete SF Oracle RAC software stack lets you devise a standardized approach toward application and database management. Further, administrators can apply existing expertise of Veritas technologies toward SF Oracle RAC.
- Increased availability and performance using dynamic multi-pathing (DMP). DMP provides wide storage array support for protection from failures and performance bottlenecks in the HBAs, SAN switches, and storage arrays.
- Easy administration and monitoring of SF Oracle RAC clusters from a single web console.
- Support for many types of applications and databases.
- Improved file system access times using Oracle Disk Manager (ODM).
- Ability to configure ASM disk groups over CVM volumes to take advantage of dynamic multi-pathing (DMP).
- Enhanced scalability and availability with access to multiple Oracle RAC instances per database in a cluster.
- Support for backup and recovery solutions using volume-level and file system-level snapshot technologies. SF Oracle RAC enables full volume-level snapshots for off-host processing and file system-level snapshots for efficient backup and rollback.
- Ability to failover applications without downtime using clustered file system technology.
- Prevention of data corruption in split-brain scenarios with robust SCSI-3 Persistent Group Reservation (PGR) based I/O fencing or Coordination Point Server-based I/O fencing. The preferred fencing feature also enables you to specify how the fencing driver determines the surviving subcluster.

- Support for sharing all types of files, in addition to Oracle database files, across nodes.
- Fast disaster recovery with minimal downtime and interruption to users. Users can transition from a local high availability site to a wide-area disaster recovery environment with primary and secondary sites. If a node fails, clients that are attached to the failed node can reconnect to a surviving node and resume access to the shared database. Recovery after failure in the SF Oracle RAC environment is far quicker than recovery for a failover database.
- Verification of disaster recovery configuration using fire drill technology without affecting production systems.
- Support for a wide range of hardware replication technologies as well as block-level replication using VVR.
- Support for campus clusters with the following capabilities:
  - Consistent reattach with Site Awareness
  - Site aware reads with VxVM mirroring
  - Monitoring of Oracle resources
  - Protection against split-brain scenarios

How SF Oracle RAC works (high-level perspective)

Oracle Real Application Clusters (RAC) is a parallel database environment that takes advantage of the processing power of multiple computers. Oracle stores data logically in the form of tablespaces and physically in the form of data files. The Oracle instance is a set of processes and shared memory that provide access to the physical database. Specifically, the instance involves server processes acting on behalf of clients to read data into shared memory and make modifications to it, and background processes that interact with each other and with the operating system to manage memory structure and do general housekeeping.

SF Oracle RAC provides the necessary infrastructure for running Oracle RAC and coordinates access to the shared data for each node to provide consistency and integrity. Each node adds its processing power to the cluster as a whole and can increase overall throughput or performance.

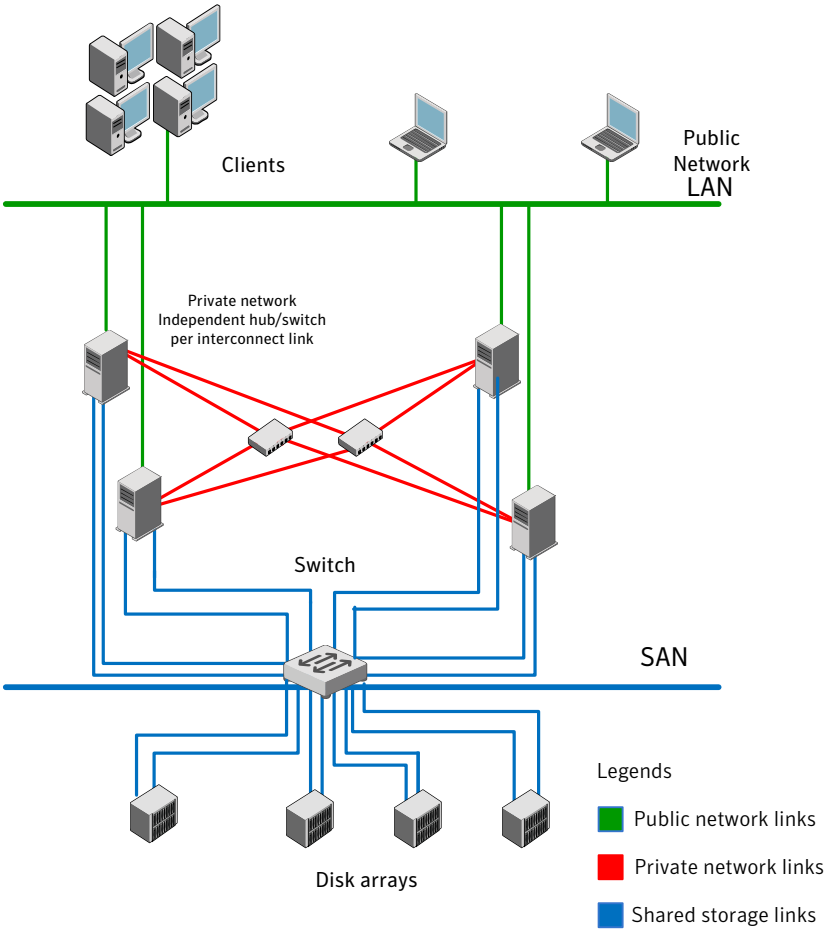
At a conceptual level, SF Oracle RAC is a cluster that manages applications (instances), networking, and storage components using resources contained in service groups. SF Oracle RAC clusters have the following properties:

- Each node runs its own operating system.
- A cluster interconnect enables cluster communications.

- A public network connects each node to a LAN for client access.
- Shared storage is accessible by each node that needs to run the application.

Figure 1-1 displays the basic layout and individual components required for a SF Oracle RAC installation.

Figure 1-1 SF Oracle RAC basic layout and components



The basic layout has the following characteristics:

- Multiple client applications that access nodes in the cluster over a public network.

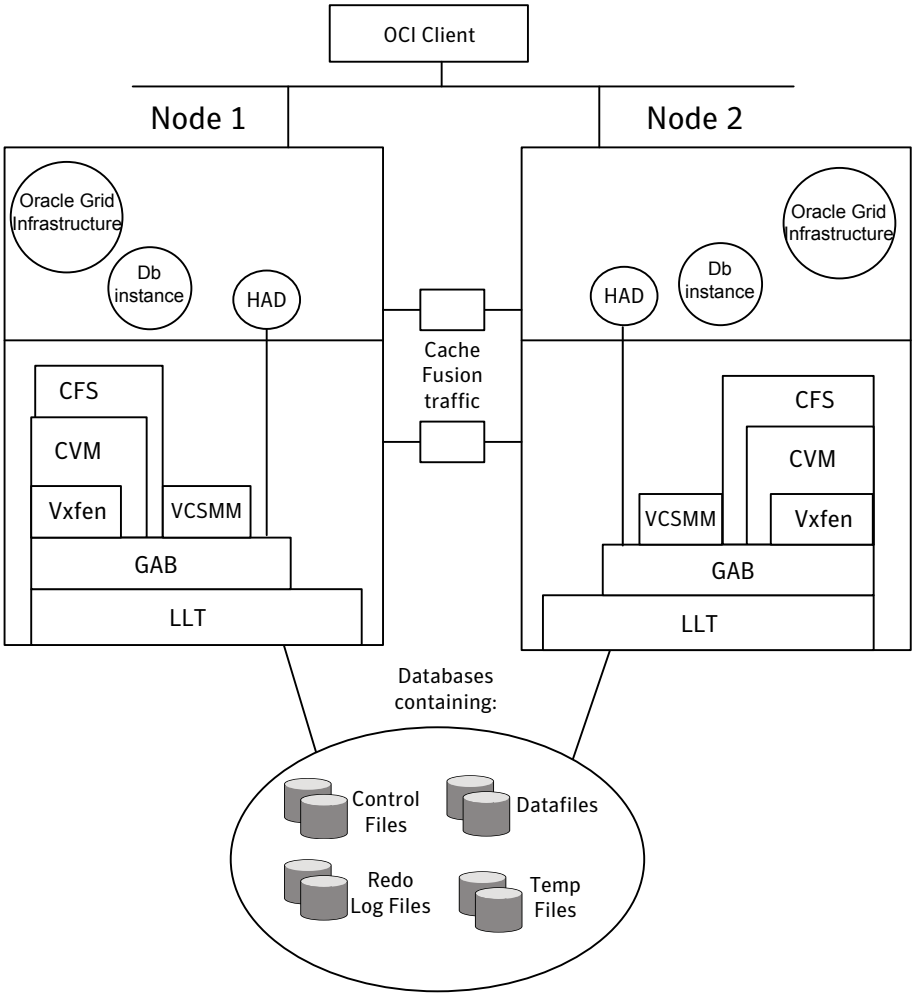
- Nodes that are connected by at least two private network links (also called cluster interconnects) using 100BaseT or gigabit Ethernet controllers on each system.
  If the private links are on a single switch, isolate them using VLAN.
- Nodes that are connected to iSCSI or Fibre Channel shared storage devices over SAN.
  All shared storage must support SCSI-3 PR.
- Nodes must be connected with private network links using similar network devices and matching port numbers.
  For example, if you use eth1 on one end of a link, the other end must also use eth1.
- Every system must have a local disk.
- The Oracle Cluster Registry, vote disks, and data files configured on the shared storage that is available to each node. The shared storage can be a cluster file system or ASM disk groups created using raw VxVM volumes.
- Three standard disks or LUNs used as coordinator disks for I/O fencing.
- VCS manages the resources that are required by Oracle RAC. The resources must run in parallel on each node.

SF Oracle RAC includes the following technologies that are engineered to improve performance, availability, and manageability of Oracle RAC environments:

- Cluster File System (CFS) and Cluster Volume Manager (CVM) technologies to manage multi-instance database access to shared storage.
- An Oracle Disk Manager (ODM) library to maximize Oracle disk I/O performance.
- Interfaces to Oracle Grid Infrastructure and RAC for managing cluster membership.

Figure 1-2 displays the technologies that make up the SF Oracle RAC internal architecture.

Figure 1-2 SF Oracle RAC architecture



SF Oracle RAC provides an environment that can tolerate failures with minimal downtime and interruption to users. If a node fails as clients access the same database on multiple nodes, clients attached to the failed node can reconnect to a surviving node and resume access. Recovery after failure in the SF Oracle RAC environment is far quicker than recovery for a single-instance database because another Oracle instance is already up and running. The recovery process involves applying outstanding redo log entries of the failed node from the surviving nodes.

Component products and processes of SF Oracle RAC

Table 1-1 lists the component products of SF Oracle RAC.

Table 1-1 SF Oracle RAC component products

Component product	Description
Cluster Volume Manager (CVM)	Enables simultaneous access to shared volumes based on technology from Veritas Volume Manager (VxVM). See "Cluster Volume Manager (CVM)" on page 30.
Cluster File System (CFS)	Enables simultaneous access to shared file systems based on technology from Veritas File System (VxFS). See "Cluster File System (CFS)" on page 32.
Veritas Cluster Server (VCS)	Manages Oracle RAC databases and infrastructure components. See "Veritas Cluster Server" on page 35.
Database Accelerator	Provides the interface with the Oracle Disk Manager (ODM) API. See "Oracle Disk Manager" on page 34.
RAC Extensions	Manages cluster membership between cluster nodes. See "RAC extensions" on page 41.

Communication infrastructure

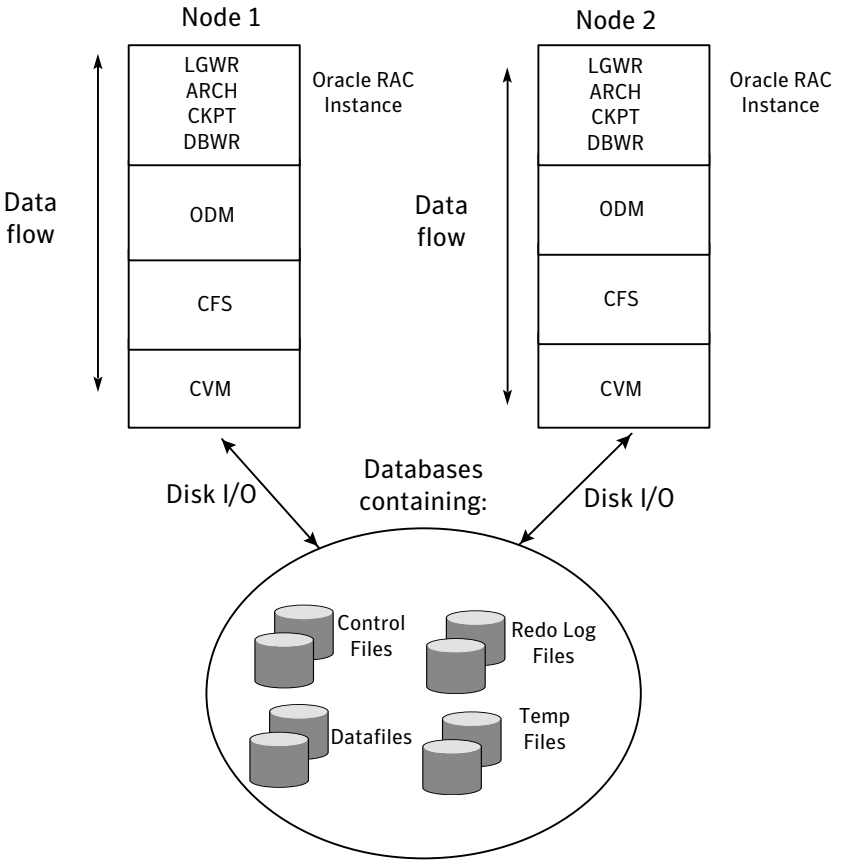
To understand the communication infrastructure, review the data flow and communication requirements.

Data flow

The CVM, CFS, ODM, and Oracle RAC elements reflect the overall data flow, or data stack, from an instance running on a server to the shared storage. The various Oracle processes composing an instance -- such as DB Writers, Log Writer, Checkpoint, and Archiver -- read and write data to the storage through the I/O stack. Oracle communicates through the ODM interface to CFS, which in turn accesses the storage through the CVM.

Figure 1-3 represents the overall data flow.

Figure 1-3 Data stack

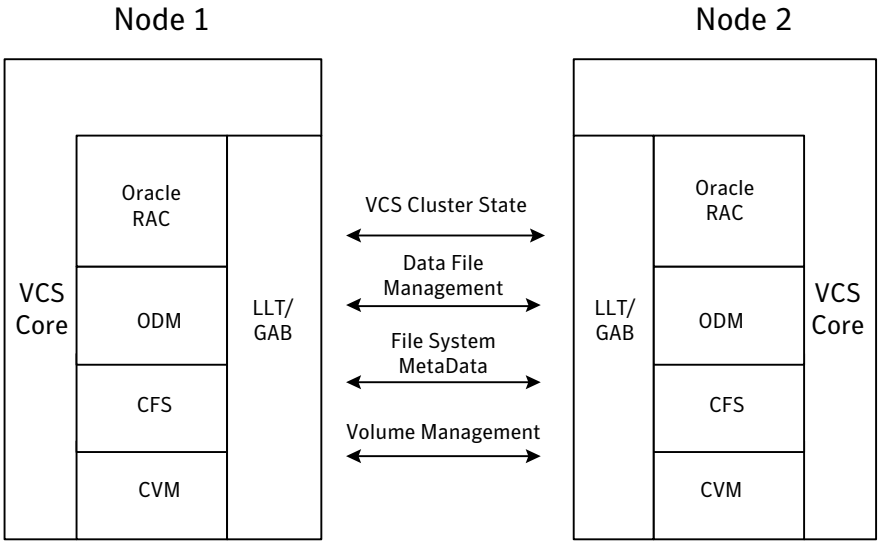


Communication requirements

End-users on a client system are unaware that they are accessing a database hosted by multiple instances. The key to performing I/O to a database accessed by multiple instances is communication between the processes. Each layer or component in the data stack must reliably communicate with its peer on other nodes to function properly. RAC instances must communicate to coordinate protection of data blocks in the database. ODM processes must communicate to coordinate data file protection and access across the cluster. CFS coordinates metadata updates for file systems, while CVM coordinates the status of logical volumes and maps.

Figure 1-4 represents the communication stack.

Figure 1-4 Communication stack



Cluster interconnect communication channel

The cluster interconnect provides an additional communication channel for all system-to-system communication, separate from the one-node communication between modules. Low Latency Transport (LLT) and Group Membership Services/Atomic Broadcast (GAB) make up the VCS communications package central to the operation of SF Oracle RAC.

In a standard operational state, significant traffic through LLT and GAB results from Lock Management, while traffic for other data is relatively sparse.

About Low Latency Transport (LLT)

The Low Latency Transport protocol is used for all cluster communications as a high-performance, low-latency replacement for the IP stack.

LLT has the following two major functions:

- Traffic distribution
  LLT provides the communications backbone for GAB. LLT distributes (load balances) inter-system communication across all configured network links. This distribution ensures all cluster communications are evenly distributed across all network links for performance and fault resilience. If a link fails,

traffic is redirected to the remaining links. A maximum of eight network links are supported.

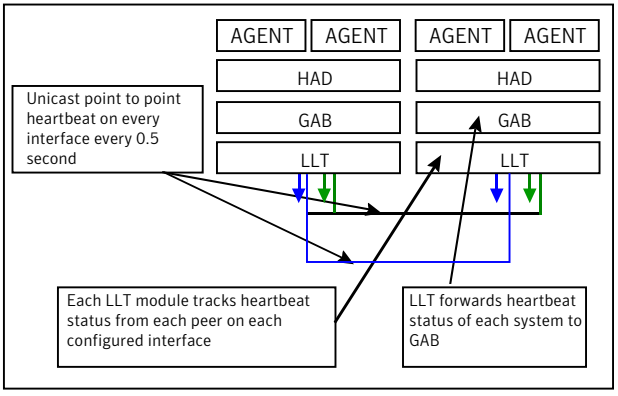
- Heartbeat
  LLT is responsible for sending and receiving heartbeat traffic over each configured network link. The heartbeat traffic is point to point unicast. LLT uses ethernet broadcast to learn the address of the nodes in the cluster. All other cluster communications, including all status and configuration traffic is point to point unicast. The heartbeat is used by the Group Membership Services to determine cluster membership.

The heartbeat signal is defined as follows:

- LLT on each system in the cluster sends heartbeat packets out on all configured LLT interfaces every half second.
- LLT on each system tracks the heartbeat status from each peer on each configured LLT interface.
- LLT on each system forwards the heartbeat status of each system in the cluster to the local Group Membership Services function of GAB.
- GAB receives the status of heartbeat from all cluster systems from LLT and makes membership determination based on this information.

Figure 1-5 shows heartbeat in the cluster.

Figure 1-5 Heartbeat in the cluster



LLT can be configured to designate specific cluster interconnect links as either high priority or low priority. High priority links are used for cluster communications to GAB as well as heartbeat signals. Low priority links, during normal operation, are used for heartbeat and link state maintenance only, and the frequency of heartbeats is reduced to 50% of normal to reduce network overhead.

If there is a failure of all configured high priority links, LLT will switch all cluster communications traffic to the first available low priority link. Communication traffic will revert back to the high priority links as soon as they become available.

While not required, best practice recommends to configure at least one low priority link, and to configure two high priority links on dedicated cluster interconnects to provide redundancy in the communications path. Low priority links are typically configured on the public or administrative network.

If you use different media speed for the private NICs, Symantec recommends that you configure the NICs with lesser speed as low-priority links to enhance LLT performance. With this setting, LLT does active-passive load balancing across the private links. At the time of configuration and failover, LLT automatically chooses the link with high-priority as the active link and uses the low-priority links only when a high-priority link fails.

LLT sends packets on all the configured links in weighted round-robin manner. LLT uses the linkburst parameter which represents the number of back-to-back packets that LLT sends on a link before the next link is chosen. In addition to the default weighted round-robin based load balancing, LLT also provides destination-based load balancing. LLT implements destination-based load balancing where the LLT link is chosen based on the destination node id and the port. With destination-based load balancing, LLT sends all the packets of a particular destination on a link. However, a potential problem with the destination-based load balancing approach is that LLT may not fully utilize the available links if the ports have dissimilar traffic. Symantec recommends destination-based load balancing when the setup has more than two cluster nodes and more active LLT ports. You must manually configure destination-based load balancing for your cluster to set up the port to LLT link mapping.

See "Configuring destination-based load balancing for LLT" on page 100.

LLT on startup sends broadcast packets with LLT node id and cluster id information onto the LAN to discover any node in the network that has same node id and cluster id pair. Each node in the network replies to this broadcast message with its cluster id, node id, and node name.

LLT on the original node does not start and gives appropriate error in the following cases:

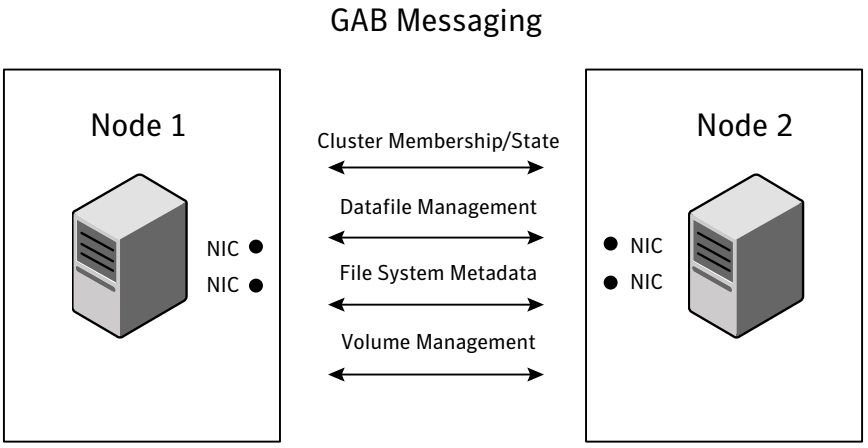
- LLT on any other node in the same network is running with the same node id and cluster id pair that it owns.
- LLT on the original node receives response from a node that does not have a node name entry in the `/etc/llthosts` file.

Group Membership Services/Atomic Broadcast

The GAB protocol is responsible for cluster membership and cluster communications.

Figure 1-6 shows the cluster communication using GAB messaging.

Figure 1-6 Cluster communication



Review the following information on cluster membership and cluster communication:

- Cluster membership
  At a high level, all nodes configured by the installer can operate as a cluster; these nodes form a cluster membership. In SF Oracle RAC, a cluster membership specifically refers to all systems configured with the same cluster ID communicating by way of a redundant cluster interconnect.
  All nodes in a distributed system, such as SF Oracle RAC, must remain constantly alert to the nodes currently participating in the cluster. Nodes can leave or join the cluster at any time because of shutting down, starting up, rebooting, powering off, or faulting processes. SF Oracle RAC uses its cluster membership capability to dynamically track the overall cluster topology.
  SF Oracle RAC uses LLT heartbeats to determine cluster membership:
  - When systems no longer receive heartbeat messages from a peer for a predetermined interval, a protocol excludes the peer from the current membership.
  - GAB informs processes on the remaining nodes that the cluster membership has changed; this action initiates recovery actions specific to each module.

For example, CVM must initiate volume recovery and CFS must perform a fast parallel file system check.

- When systems start receiving heartbeats from a peer outside of the current membership, a protocol enables the peer to join the membership.

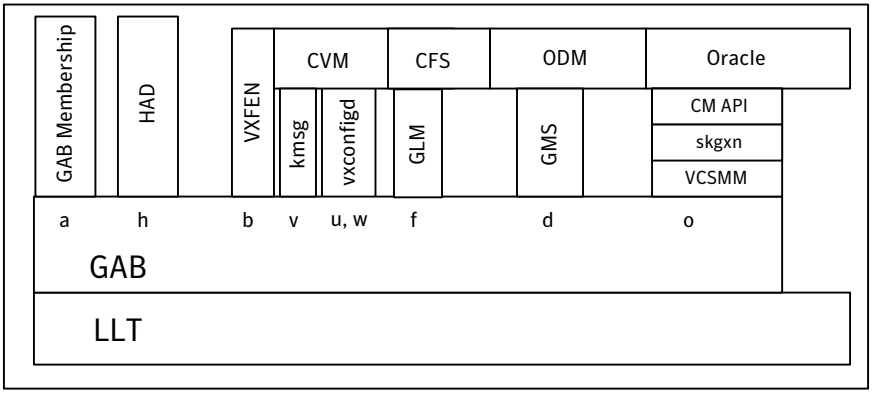
- Cluster communications
  GAB provides reliable cluster communication between SF Oracle RAC modules. GAB provides guaranteed delivery of point-to-point messages and broadcast messages to all nodes. Point-to-point messaging involves sending and acknowledging the message. Atomic-broadcast messaging ensures all systems within the cluster receive all messages. If a failure occurs while transmitting a broadcast message, GAB ensures all systems have the same information after recovery.

Low-level communication: port relationship between GAB and processes

All components in SF Oracle RAC use GAB for communication. Each process wanting to communicate with a peer process on other nodes registers with GAB on a specific port. This registration enables communication and notification of membership changes. For example, the VCS engine (HAD) registers on port h. HAD receives messages from peer had processes on port h. HAD also receives notification when a node fails or when a peer process on port h unregisters.

Some modules use multiple ports for specific communications requirements. For example, CVM uses multiple ports to allow communications by kernel and user-level functions in CVM independently.

Figure 1-7 Low-level communication



Cluster Volume Manager (CVM)

CVM is an extension of Veritas Volume Manager, the industry-standard storage virtualization platform. CVM extends the concepts of VxVM across multiple nodes. Each node recognizes the same logical volume layout, and more importantly, the same state of all volume resources.

CVM supports performance-enhancing capabilities, such as striping, mirroring, and mirror break-off (snapshot) for off-host backup. You can use standard VxVM commands from one node in the cluster to manage all storage. All other nodes immediately recognize any changes in disk group and volume configuration with no user interaction.

For detailed information, see the *Veritas Volume Manager Administrator's Guide*.

CVM architecture

CVM is designed with a "master and slave" architecture. One node in the cluster acts as the configuration master for logical volume management, and all other nodes are slaves. Any node can take over as master if the existing master fails. The CVM master exists on a per-cluster basis and uses GAB and LLT to transport its configuration data.

Just as with VxVM, the Volume Manager configuration daemon, `vxconfigd`, maintains the configuration of logical volumes. This daemon handles changes to the volumes by updating the operating system at the kernel level. For example, if a mirror of a volume fails, the mirror detaches from the volume and `vxconfigd` determines the proper course of action, updates the new volume layout, and informs the kernel of a new volume layout. CVM extends this behavior across multiple nodes and propagates volume changes to the master `vxconfigd`.

Note: You must perform operator-initiated changes on the master node.

The `vxconfigd` process on the master pushes these changes out to slave `vxconfigd` processes, each of which updates the local kernel. The kernel module for CVM is kmsg.

CVM does not impose any write locking between nodes. Each node is free to update any area of the storage. All data integrity is the responsibility of the upper application. From an application perspective, standalone systems access logical volumes in the same way as CVM systems.

CVM imposes a "Uniform Shared Storage" model. All nodes must connect to the same disk sets for a given disk group. Any node unable to detect the entire set of physical disks for a given disk group cannot import the group. If a node loses

contact with a specific disk, CVM excludes the node from participating in the use of that disk.

CVM communication

CVM communication involves various GAB ports for different types of communication. For an illustration of these ports:

CVM communication involves the following GAB ports:

- Port w
  Most CVM communication uses port w for vxconfigd communications. During any change in volume configuration, such as volume creation, plex attachment or detachment, and volume resizing, vxconfigd on the master node uses port w to share this information with slave nodes.
  When all slaves use port w to acknowledge the new configuration as the next active configuration, the master updates this record to the disk headers in the VxVM private region for the disk group as the next configuration.
- Port v
  CVM uses port v for kernel-to-kernel communication. During specific configuration events, certain actions require coordination across all nodes. An example of synchronizing events is a resize operation. CVM must ensure all nodes see the new or old size, but never a mix of size among members.
  CVM also uses this port to obtain cluster membership from GAB and determine the status of other CVM members in the cluster.
- Port u
  CVM uses the group atomic broadcast (GAB) transport mechanism of VCS to ship the commands from the slave node to the master node. CVM uses group atomic broadcast (GAB) port u.
  CVM processes one node joining the cluster at a time. If multiple nodes want to join the cluster simultaneously, each node attempts to open port u in exclusive mode. (GAB only allows one node to open a port in exclusive mode). As each node joins the cluster, GAB releases the port. The next node can then open the port and join the cluster. In a case of multiple nodes, each node continuously retries at pseudo-random intervals until it wins the port.

CVM recovery

When a node leaves a cluster, the new membership is delivered by GAB, to CVM on existing cluster nodes. The fencing driver (VXFEN) ensures that split-brain scenarios are taken care of before CVM is notified. CVM then initiates recovery of mirrors of shared volumes that might have been in an inconsistent state following the exit of the node.

For database files, when ODM is enabled with SmartSync option, Oracle Resilvering handles recovery of mirrored volumes. For non-database files, this recovery is optimized using Dirty Region Logging (DRL). The DRL is a map stored in a special purpose VxVM sub-disk and attached as an additional plex to the mirrored volume. When a DRL subdisk is created for a shared volume, the length of the sub-disk is automatically evaluated so as to cater to the number of cluster nodes. If the shared volume has Fast Mirror Resync (FlashSnap) enabled, the DCO (Data Change Object) log volume created automatically has DRL embedded in it. In the absence of DRL or DCO, CVM does a full mirror resynchronization.

Configuration differences with VxVM

CVM configuration differs from VxVM configuration in the following areas:

- Configuration commands occur on the master node.
- Disk groups are created (could be private) and imported as shared disk groups.
- Disk groups are activated per node.
- Shared disk groups are automatically imported when CVM starts.

Cluster File System (CFS)

CFS enables you to simultaneously mount the same file system on multiple nodes and is an extension of the industry-standard Veritas File System. Unlike other file systems which send data through another node to the storage, CFS is a true SAN file system. All data traffic takes place over the storage area network (SAN), and only the metadata traverses the cluster interconnect.

In addition to using the SAN fabric for reading and writing data, CFS offers storage checkpoints and rollback for backup and recovery.

Access to cluster storage in typical SF Oracle RAC configurations use CFS. Raw access to CVM volumes is also possible but not part of a common configuration.

For detailed information, see the *Veritas Storage Foundation Cluster File System Administrator's Guide*.

CFS architecture

SF Oracle RAC uses CFS to manage a file system in a large database environment. Since CFS is an extension of VxFS, it operates in a similar fashion and caches metadata and data in memory (typically called buffer cache or vnode cache). CFS uses a distributed locking mechanism called Global Lock Manager (GLM) to ensure all nodes have a consistent view of the file system. GLM provides metadata and cache coherency across multiple nodes by coordinating access to file system

metadata, such as inodes and free lists. The role of GLM is set on a per-file system basis to enable load balancing.

CFS involves a primary/secondary architecture. One of the nodes in the cluster is the primary node for a file system. Though any node can initiate an operation to create, delete, or resize data, the GLM master node carries out the actual operation. After creating a file, the GLM master node grants locks for data coherency across nodes. For example, if a node tries to modify a block in a file, it must obtain an exclusive lock to ensure other nodes that may have the same file cached have this cached copy invalidated.

SF Oracle RAC configurations minimize the use of GLM locking. Oracle RAC accesses the file system through the ODM interface and handles its own locking; only Oracle (and not GLM) buffers data and coordinates write operations to files. A single point of locking and buffering ensures maximum performance. GLM locking is only involved when metadata for a file changes, such as during create and resize operations.

CFS communication

CFS uses port f for GLM lock and metadata communication. SF Oracle RAC configurations minimize the use of GLM locking except when metadata for a file changes.

CFS file system benefits

Many features available in VxFS do not come into play in an SF Oracle RAC environment because ODM handles such features. CFS adds such features as high availability, consistency and scalability, and centralized management to VxFS. Using CFS in an SF Oracle RAC environment provides the following benefits:

- Increased manageability, including easy creation and expansion of files
  In the absence of CFS, you must provide Oracle with fixed-size partitions. With CFS, you can grow file systems dynamically to meet future requirements.
- Less prone to user error
  Raw partitions are not visible and administrators can compromise them by mistakenly putting file systems over the partitions. Nothing exists in Oracle to prevent you from making such a mistake.
- Data center consistency
  If you have raw partitions, you are limited to a RAC-specific backup strategy. CFS enables you to implement your backup strategy across the data center.

CFS recovery

The vxfsckd daemon is responsible for ensuring file system consistency when a node crashes that was a primary node for a shared file system. If the local node is a secondary node for a given file system and a reconfiguration occurs in which this node becomes the primary node, the kernel requests `vxfsckd` on the new primary node to initiate a replay of the intent log of the underlying volume. The vxfsckd daemon forks a special call to fsck that ignores the volume reservation protection normally respected by fsck and other VxFS utilities. The vxfsckd can check several volumes at once if the node takes on the primary role for multiple file systems.

After a secondary node crash, no action is required to recover file system integrity. As with any crash on a file system, internal consistency of application data for applications running at the time of the crash is the responsibility of the applications.

Comparing raw volumes and CFS for data files

Keep these points in mind about raw volumes and CFS for data files:

- If you use file-system-based data files, the file systems containing these files must be located on shared disks. Create the same file system mount point on each node.
- If you use raw devices, such as VxVM volumes, set the permissions for the volumes to be owned permanently by the database account.
  VxVM sets volume permissions on import. The VxVM volume, and any file system that is created in it, must be owned by the Oracle database user.

Oracle Disk Manager

SF Oracle RAC requires Oracle Disk Manager (ODM), a standard API published by Oracle for support of database I/O. SF Oracle RAC provides a library for Oracle to use as its I/O library.

ODM architecture

When the Veritas ODM library is linked, Oracle is able to bypass all caching and locks at the file system layer and to communicate directly with raw volumes. The SF Oracle RAC implementation of ODM generates performance equivalent to performance with raw devices while the storage uses easy-to-manage file systems.

All ODM features can operate in a cluster environment. Nodes communicate with each other before performing any operation that could potentially affect another

node. For example, before creating a new data file with a specific name, ODM checks with other nodes to see if the file name is already in use.

Veritas ODM performance enhancements

Veritas ODM enables the following performance benefits provided by Oracle Disk Manager:

- Locking for data integrity.
- Few system calls and context switches.
- Increased I/O parallelism.
- Efficient file creation and disk allocation.

Databases using file systems typically incur additional overhead:

- Extra CPU and memory usage to read data from underlying disks to the file system cache. This scenario requires copying data from the file system cache to the Oracle cache.
- File locking that allows for only a single writer at a time. Allowing Oracle to perform locking allows for finer granularity of locking at the row level.
- File systems generally go through a standard Sync I/O library when performing I/O. Oracle can make use of Kernel Async I/O libraries (KAIO) with raw devices to improve performance.

ODM communication

ODM uses port d to communicate with ODM instances on other nodes in the cluster.

Veritas Cluster Server

Veritas Cluster Server (VCS) directs SF Oracle RAC operations by controlling the startup and shutdown of components layers and providing monitoring and notification for failures.

In a typical SF Oracle RAC configuration, the Oracle RAC service groups for VCS run as "parallel" service groups rather than "failover" service groups; in the event of a failure, VCS does not attempt to migrate a failed service group. Instead, the software enables you to configure the group to restart on failure.

VCS architecture

The High Availability Daemon (HAD) is the main VCS daemon running on each node. HAD tracks changes in the cluster configuration and monitors resource

status by communicating over GAB and LLT. HAD manages all application services using agents, which are installed programs to manage resources (specific hardware or software entities).

The VCS architecture is modular for extensibility and efficiency; HAD does not need to know how to start up Oracle or any other application under VCS control. Instead, you can add agents to manage different resources with no effect on the engine (HAD). Agents only communicate with HAD on the local node, and HAD communicates status with HAD processes on other nodes. Because agents do not need to communicate across systems, VCS is able to minimize traffic on the cluster interconnect.

SF Oracle RAC provides specific agents for VCS to manage CVM, CFS, and Oracle RAC components like Oracle Grid Infrastructure and database (including instances).

VCS communication

VCS uses port h for HAD communication. Agents communicate with HAD on the local node about resources, and HAD distributes its view of resources on that node to other nodes through GAB port h. HAD also receives information from other cluster members to update its own view of the cluster.

About the IMF notification module

The Intelligent Monitoring Framework (IMF) notification module hooks into system calls and other kernel interfaces of the operating system to get notifications on various events such as when a process starts or dies, or when a block device gets mounted or unmounted from a mount point. VCS process-based and mount-based agents use Asynchronous Monitoring Framework (AMF) kernel driver that provides asynchronous event notifications to the agents that are enabled for intelligent resource monitoring.

See “About resource monitoring” on page 36.

About resource monitoring

VCS agents polls the resources periodically based on the monitor interval (in seconds) value that is defined in the MonitorInterval or in the OfflineMonitorInterval resource type attributes. After each monitor interval, VCS invokes the monitor agent function for that resource. For example, for process offline monitoring, the process agent's monitor agent function corresponding to each process resource scans the process table in each monitor interval to check whether the process has come online. For process online monitoring, the monitor agent function queries the operating system for the status of the process id that it is monitoring.. In case of the mount agent, the monitor agent function corresponding to each mount resource checks if the block device is mounted on

the mount point or not. In order to determine this, the monitor function does operations such as mount table scans or runs `statfs` equivalents.

With intelligent monitoring framework (IMF), VCS supports intelligent resource monitoring in addition to the poll-based monitoring. Process-based and Mount-based agents are IMF-aware. IMF is an extension to the VCS agent framework. You can enable or disable the intelligent monitoring functionality of the VCS agents that are IMF-aware.

See "How intelligent resource monitoring works" on page 38.

See "Enabling and disabling intelligent resource monitoring" on page 100.

Poll-based monitoring can consume a fairly large percentage of system resources such as CPU and memory on systems with a huge number of resources. This not only affects the performance of running applications, but also places a limit on how many resources an agent can monitor efficiently.

However, with IMF-based monitoring you can either eliminate poll-based monitoring completely or reduce its frequency. For example, for process offline and online monitoring, you can completely avoid the need for poll-based monitoring with IMF-based monitoring for processes. Similarly for vxfs mounts, you can eliminate the poll-based monitoring with IMF monitoring enabled. Such reduction in monitor footprint will make more system resources available for other applications to consume.

Note: Intelligent Monitoring Framework is supported for VxFS and CFS mounts only.

With IMF-enabled agents, VCS will be able to effectively monitor larger number of resources.

Thus, intelligent monitoring has the following benefits over poll-based monitoring:

- Provides faster notification of resource state changes
- Reduces VCS system utilization due to reduced monitor function footprint
- Enables VCS to effectively monitor a large number of resources

Consider enabling IMF for an agent in either of the following cases:

- You have a large number of process resources or mount resources under VCS control.
- You have any of the agents that are IMF-aware.
  See the *Veritas Cluster Server Bundled Agents Reference Guide* for details on whether your bundled agent is IMF-aware.

See the *Veritas Cluster Server Agent for Oracle Installation and Configuration Guide* for IMF-aware agents for Oracle.
See the *Veritas Storage Foundation Cluster File System Installation Guide* for IMF-aware agents in CFS environments.

How intelligent resource monitoring works

When an IMF-enabled agent starts up, the agent initializes the IMF notification module. After the resource is in a steady state, the agent registers the details that are required to monitor the resource with the IMF notification module. For example, the process agent registers the PIDs of the processes with the IMF notification module. The agent's imf_getnotification function waits for any resource state changes. When the IMF notification module notifies the imf_getnotification function about a resource state change, the agent framework runs the monitor agent function to ascertain the state of that resource. The agent notifies the state change to VCS which takes appropriate action.

See "About the IMF notification module" on page 36.

Cluster configuration files

VCS uses two configuration files in a default configuration:

- The main.cf file defines the entire cluster, including the cluster name, systems in the cluster, and definitions of service groups and resources, in addition to service group and resource dependencies.
- The types.cf file defines the resource types. Each resource in a cluster is identified by a unique name and classified according to its type. VCS includes a set of pre-defined resource types for storage, networking, and application services.

Additional files similar to types.cf may be present if you add agents. For example, SF Oracle RAC includes additional resource types files, such as OracleTypes.cf, PrivNIC.cf, and MultiPrivNIC.cf.

Oracle RAC

This section provides a brief description of Oracle Clusterware/Grid Infrastructure, the Oracle Cluster Registry, application resources, and the voting disk.

Note: Refer to the Oracle RAC documentation for additional information.

Oracle Clusterware/Grid Infrastructure

Oracle Clusterware/Grid Infrastructure manages Oracle cluster-related functions including membership, group services, global resource management, and databases. Oracle Clusterware/Grid Infrastructure is required for every Oracle RAC instance.

Oracle Clusterware/Grid Infrastructure requires the following major components:

- A cluster interconnect that allows for cluster communications
- A private virtual IP address for cluster communications over the interconnect
- A public virtual IP address for client connections
- For Oracle 11g Release 2, a public IP address as a Single Client Access Name (SCAN) address on the Domain Name Server (DNS) for round robin resolution to three addresses (recommended) or at least one address
- Shared storage accessible by each node

Co-existence with VCS

Oracle Clusterware/Grid Infrastructure supports co-existence with vendor clusterwares such as Veritas Cluster Server. When you install Oracle Clusterware/Grid Infrastructure on an SF Oracle RAC cluster, Oracle Clusterware//Grid Infrastructure detects the presence of VCS by checking the presence of the Veritas membership module (VCSMM) library. It obtains the list of nodes in the cluster from the VCSMM library at the time of installation.

When a node fails to respond across the interconnect, Oracle Clusterware/Grid Infrastructure waits before evicting another node from the cluster. This wait-time is defined by the CSS miss-count value. Oracle Clusterware/Grid Infrastructure sets the CSS miss-count parameter to a larger value (600 seconds) in the presence of VCS. This value is much higher than the LLT peer inactivity timeout interval. Thus, in the event of a network split-brain, the two clusterwares, VCS and Oracle Clusterware/Grid Infrastructure, do not interfere with each other's decisions on which nodes remain in the cluster. Veritas I/O fencing is allowed to decide on the surviving nodes first, followed by Oracle Clusterware/Grid Infrastructure.

VCS uses LLT for communicating between cluster nodes over private interconnects while Oracle Clusterware/Grid Infrastructure uses private IP addresses configured over the private interconnects to communicate between the cluster nodes. To coordinate membership changes between VCS and Oracle Clusterware/Grid Infrastructure, it is important to configure the Oracle Clusterware/Grid Infrastructure private IP address over the network interfaces used by LLT. VCS uses the CSSD agent to start, stop, and monitor Oracle Clusterware/Grid Infrastructure. The CSSD agent ensures that the OCR, the voting disk, and the private IP address resources required by Oracle Clusterware/Grid Infrastructure

are brought online by VCS before Oracle Clusterware/Grid Infrastructure starts. This prevents the premature startup of Oracle Clusterware/Grid Infrastructure, which causes cluster failures.

Oracle Cluster Registry

The Oracle Cluster Registry (OCR) contains cluster and database configuration and state information for Oracle RAC and Oracle Clusterware/Grid Infrastructure.

The information maintained in the OCR includes:

- The list of nodes
- The mapping of database instances to nodes
- Oracle Clusterware application resource profiles
- Resource profiles that define the properties of resources under Oracle Clusterware/Grid Infrastructure control
- Rules that define dependencies between the Oracle Clusterware/Grid Infrastructure resources
- The current state of the cluster

In Oracle RAC 11g Release 2, the OCR data exists on ASM or a cluster file system that is accessible to each node.

Use CVM mirrored volumes to protect OCR data from failures.Oracle Clusterware/Grid Infrastructure faults nodes if OCR is not accessible because of corruption or disk failure. Oracle automatically backs up OCR data. You can also export the OCR contents before making configuration changes in Oracle Clusterware/Grid Infrastructure. This way, if you encounter configuration problems and are unable to restart Oracle Clusterware/Grid Infrastructure, you can restore the original contents.

Consult the Oracle documentation for instructions on exporting and restoring OCR contents.

Application resources

Oracle Clusterware/Grid Infrastructure application resources are similar to VCS resources. Each component controlled by Oracle Clusterware/Grid Infrastructure is defined by an application resource, including databases, instances, services, and node applications.

Unlike VCS, Oracle Clusterware/Grid Infrastructure uses separate resources for components that run in parallel on multiple nodes.

Resource profiles

Resources are defined by profiles, which are similar to the attributes that define VCS resources. The OCR contains application resource profiles, dependencies, and status information.

Oracle Clusterware/Grid Infrastructure node applications

Oracle Clusterware/Grid Infrastructure uses these node application resources to manage Oracle components, such as databases, listeners, and virtual IP addresses. Node application resources are created during Oracle Clusterware/Grid Infrastructure installation.

Voting disk

The voting disk is a heartbeat mechanism used by Oracle Clusterware/Grid Infrastructure to maintain cluster node membership.

In Oracle RAC 11g Release 2, voting disk data exists on ASM or on a cluster file system that is accessible to each node.

The ocssd processes of Oracle Clusterware/Grid Infrastructure provides cluster node membership and group membership information to RAC instances. On each node, ocssd processes write a heartbeat to the voting disk every second. If a node is unable to access the voting disk, Oracle Clusterware/Grid Infrastructure determines the cluster is in a split-brain condition and panics the node in a way that only one sub-cluster remains.

RAC extensions

Oracle RAC relies on support services provided by VCS such as Veritas Cluster Server Membership Manager (VCSMM) to protect data integrity.

Veritas Cluster Server membership manager

To protect data integrity by coordinating locking between RAC instances, Oracle must know which instances actively access a database. Oracle provides an API called skgxn (system kernel generic interface node membership) to obtain information on membership. SF Oracle RAC implements this API as a library linked to Oracle after you install Oracle RAC. Oracle uses the linked skgxn library to make ioctl calls to VCSMM, which in turn obtains membership information for clusters and instances by communicating with GAB on port o.

Oracle and cache fusion traffic

Private IP addresses are required by Oracle for cache fusion traffic.

You must use UDP IPC for the database cache fusion traffic.

About preventing data corruption with I/O fencing

I/O fencing is a feature that prevents data corruption in the event of a communication breakdown in a cluster.

To provide high availability, the cluster must be capable of taking corrective action when a node fails. In this situation, SF Oracle RAC configures its components to reflect the altered membership.

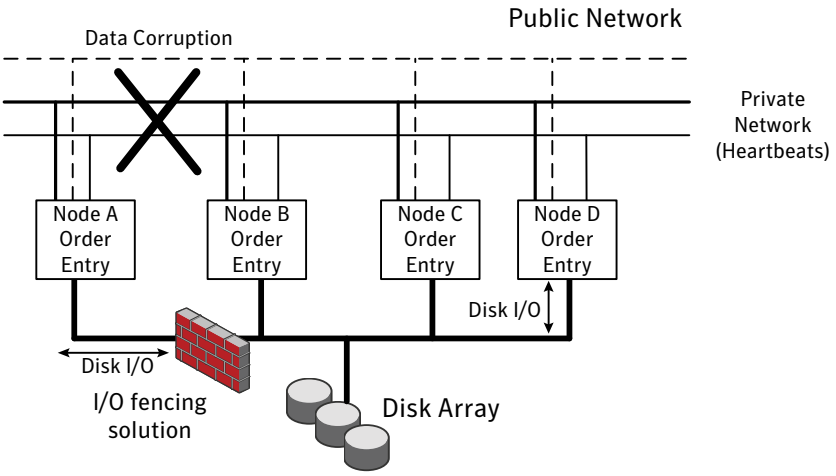
Problems arise when the mechanism that detects the failure breaks down because symptoms appear identical to those of a failed node. For example, if a system in a two-node cluster fails, the system stops sending heartbeats over the private interconnects. The remaining node then takes corrective action. The failure of the private interconnects, instead of the actual nodes, presents identical symptoms and causes each node to determine its peer has departed. This situation typically results in data corruption because both nodes try to take control of data storage in an uncoordinated manner.

In addition to a broken set of private networks, other scenarios can generate this situation. If a system is so busy that it appears to stop responding or "hang," the other nodes could declare it as dead. This declaration may also occur for the nodes that use the hardware that supports a "break" and "resume" function. When a node drops to PROM level with a break and subsequently resumes operations, the other nodes may declare the system dead. They can declare it dead even if the system later returns and begins write operations.

SF Oracle RAC uses I/O fencing to remove the risk that is associated with split-brain. I/O fencing allows write access for members of the active cluster. It blocks access to storage from non-members.

Figure 1-8 displays a schematic of a four node cluster, each node writing order entries to the connected disk array. When the private network connection between the four nodes is disrupted (between Node A and the other 3 nodes in the figure below), a split-brain situation occurs with the possibility of data corruption to the disk array. The I/O fencing process prevents split-brain and any data corruption by fencing off Node A from the cluster.

Figure 1-8 Private network disruption and I/O fencing solution



About SCSI-3 Persistent Reservations

SCSI-3 Persistent Reservations (SCSI-3 PR) are required for I/O fencing and resolve the issues of using SCSI reservations in a clustered SAN environment. SCSI-3 PR enables access for multiple nodes to a device and simultaneously blocks access for other nodes.

SCSI-3 reservations are persistent across SCSI bus resets and support multiple paths from a host to a disk. In contrast, only one host can use SCSI-2 reservations with one path. If the need arises to block access to a device because of data integrity concerns, only one host and one path remain active. The requirements for larger clusters, with multiple nodes reading and writing to storage in a controlled manner, make SCSI-2 reservations obsolete.

SCSI-3 PR uses a concept of registration and reservation. Each system registers its own "key" with a SCSI-3 device. Multiple systems registering keys form a membership and establish a reservation, typically set to "Write Exclusive Registrants Only." The WERO setting enables only registered systems to perform write operations. For a given disk, only one reservation can exist amidst numerous registrations.

With SCSI-3 PR technology, blocking write access is as easy 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. Ejecting a node is final and atomic; an ejected node cannot eject another node. In SF Oracle RAC, a node registers the same key for all paths

to the device. A single preempt and abort command ejects a node from all paths to the storage device.

About I/O fencing operations

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 tries to eject the key for departed nodes from the coordinator disks using the preempt and abort command. When the node successfully ejects the departed nodes from the coordinator disks, it also ejects the departed nodes from the data disks. In a split-brain scenario, both sides of the split would race 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 restarts the system.

About I/O fencing communication

The vxfen driver connects to GAB port b to intercept cluster membership changes (reconfiguration messages). During a membership change, the fencing driver determines which systems are members of the cluster to allow access to shared disks.

After completing fencing operations, the driver passes reconfiguration messages to higher modules. CVM handles fencing of data drives for shared disk groups. After a node successfully joins the GAB cluster and the driver determines that a preexisting split-brain does not exist, CVM can import all shared disk groups. The CVM master coordinates the order of import and the key for each disk group. As each slave joins the cluster, it accepts the CVM list of disk groups and keys, and adds its proper digit to the first byte of the key. Each slave then registers the keys with all drives in the disk groups.

About I/O fencing components

The shared storage for SF Oracle RAC must support SCSI-3 persistent reservations to enable I/O fencing. SF Oracle RAC involves two types of shared storage:

- Data disks—Store shared data
  See “About data disks” on page 45.
- Coordination points—Act as a global lock during membership changes
  See “About coordination points” on page 45.

About data disks

Data disks are standard disk devices for data storage and are either physical disks or RAID Logical Units (LUNs).

These disks must support SCSI-3 PR and must be part of standard VxVM or CVM disk groups. CVM is responsible for fencing data disks on a disk group basis. Disks that are added to a disk group and new paths that are discovered for a device are automatically fenced.

About coordination points

Coordination points provide a lock mechanism to determine which nodes get to fence off data drives from other nodes. A node must eject a peer from the coordination points before it can fence the peer from the data drives. Racing for control of the coordination points to fence data disks is the key to understand how fencing prevents split-brain.

Note: Typically, a fencing configuration for a cluster must have three coordination points. Symantec also supports server-based fencing with a single CP server as its only coordination point with a caveat that this CP server becomes a single point of failure.

The coordination points can either be disks or servers or both.

- Coordinator disks

  Disks that act as coordination points are called coordinator disks. Coordinator disks are three standard disks or LUNs set aside for I/O fencing during cluster reconfiguration. Coordinator disks do not serve any other storage purpose in the SF Oracle RAC configuration.

  You can configure coordinator disks to use Veritas Volume Manager Dynamic Multi-pathing (DMP) feature. Dynamic Multi-pathing (DMP) allows coordinator disks to take advantage of the path failover and the dynamic adding and removal capabilities of DMP. So, you can configure I/O fencing to use either DMP devices or the underlying raw character devices. I/O fencing uses SCSI-3 disk policy that is either raw or dmp based on the disk device that you use. The disk policy is dmp by default.

  See the *Veritas Volume Manager Administrator's Guide*.

- Coordination point servers

  The coordination point server (CP server) is a software solution which runs on a remote system or cluster. CP server provides arbitration functionality by allowing the SF Oracle RAC cluster nodes to perform the following tasks:

- Self-register to become a member of an active SF Oracle RAC cluster (registered with CP server) with access to the data drives
- Check which other nodes are registered as members of this activeSF Oracle RAC cluster
- Self-unregister from this active SF Oracle RAC cluster
- Forcefully unregister other nodes (preempt) as members of this active SF Oracle RAC cluster

In short, the CP server functions as another arbitration mechanism that integrates within the existing I/O fencing module.

Note: With the CP server, the fencing arbitration logic still remains on the SF Oracle RAC cluster.

Multiple SF Oracle RAC clusters running different operating systems can simultaneously access the CP server. TCP/IP based communication is used between the CP server and the SF Oracle RAC clusters.

About preferred fencing

The I/O fencing driver uses coordination points to prevent split-brain in a VCS cluster. By default, the fencing driver favors the subcluster with maximum number of nodes during the race for coordination points. With the preferred fencing feature, you can specify how the fencing driver must determine the surviving subcluster.

You can configure the preferred fencing policy using the cluster-level attribute PreferredFencingPolicy as follows:

- Enable system-based preferred fencing policy to give preference to high capacity systems.
- Enable group-based preferred fencing policy to give preference to service groups for high priority applications.
- Disable preferred fencing policy to use the default node count-based race policy.

See "How preferred fencing works" on page 46.

See "Enabling or disabling the preferred fencing policy" on page 131.

How preferred fencing works

The I/O fencing driver uses coordination points to prevent split-brain in a VCS cluster. At the time of a network partition, the fencing driver in each subcluster

races for the coordination points. The subcluster that grabs the majority of coordination points survives whereas the fencing driver causes a system panic on nodes from all other subclusters. By default, the fencing driver favors the subcluster with maximum number of nodes during the race for coordination points.

This default racing preference does not take into account the application groups that are online on any nodes or the system capacity in any subcluster. For example, consider a two-node cluster where you configured an application on one node and the other node is a standby-node. If there is a network partition and the standby-node wins the race, the node where the application runs panics and VCS has to bring the application online on the standby-node. This behavior causes disruption and takes time for the application to fail over to the surviving node and then to start up again.

The preferred fencing feature lets you specify how the fencing driver must determine the surviving subcluster. The preferred fencing solution makes use of a fencing parameter called node weight. VCS calculates the node weight based on online applications and system capacity details that you provide using specific VCS attributes, and passes to the fencing driver to influence the result of race for coordination points. At the time of a race, the racer node adds up the weights for all nodes in the local subcluster and in the leaving subcluster. If the leaving subcluster has a higher sum (of node weights) then the racer for this subcluster delays the race for the coordination points. Thus, the subcluster that has critical systems or critical applications wins the race.

The preferred fencing feature uses the cluster-level attribute PreferredFencingPolicy that takes the following race policy values:

- Disabled (default): Preferred fencing is disabled.
  When the PreferredFencingPolicy attribute value is set as Disabled, VCS sets the count based race policy and resets the value of node weight as 1.
- System: Based on the capacity of the systems in a subcluster.
  If one system is more powerful than others in terms of architecture, number of CPUs, or memory, this system is given preference in the fencing race.
  When the PreferredFencingPolicy attribute value is set as System, VCS calculates node weight based on the system-level attribute FencingWeight.
- Group: Based on the higher priority applications in a subcluster.
  The fencing driver takes into account the service groups that are online on the nodes in any subcluster. In the event of a network partition, the node with higher priority service groups is given preference in the fencing race.
  When the PreferredFencingPolicy attribute value is set as Group, VCS calculates node weight based on the group-level attribute Priority for those service groups that are active.

See the *Veritas Cluster Server Administrator's Guide* for more information on the VCS attributes.

See "Enabling or disabling the preferred fencing policy" on page 131.

How disk-based I/O fencing works in different event scenarios

Table 1-2 describes how I/O fencing works to prevent data corruption in different failure event scenarios. For each event, review the corrective operator actions.

Table 1-2 I/O fencing scenarios

Event	Node A: What happens?	Node B: What happens?	Operator action
Both private networks 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 panics and 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 networks 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.	Restart Node B after private networks are restored.
One private network 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.

Table 1-2 I/O fencing scenarios *(continued)*

Event	Node A: What happens?	Node B: What happens?	Operator action
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 panics and 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 restart Node A.

Table 1-2 I/O fencing scenarios *(continued)*

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 restart, but private networks still have no power.	Node A restarts 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 restarts 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.`	Resolve preexisting split-brain condition. See "System panics to prevent potential data corruption" on page 197.

Table 1-2 I/O fencing scenarios *(continued)*

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 restarts 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 documentation for steps required to clear preexisting split brain.`	Resolve preexisting split-brain condition. See "System panics to prevent potential data corruption" on page 197.
The disk array containing two of the three coordinator disks is powered off. No node leaves the cluster membership	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.	Power on the failed disk array so that subsequent network partition does not cause cluster shutdown, or replace coordinator disks. See "Replacing I/O fencing coordinator disks when the cluster is online" on page 121.

Table 1-2 I/O fencing scenarios *(continued)*

Event	Node A: What happens?	Node B: What happens?	Operator action
The disk array containing two of the three coordinator disks is powered off. Node B gracefully leaves the cluster and the disk array is still powered off. Graceful leave implies a clean shutdown so that vxfen is properly unconfigured.	Node A continues to operate in the cluster.	Node B has left the cluster.	Power on the failed disk array so that subsequent network partition does not cause cluster shutdown, or replace coordinator disks. See "Replacing I/O fencing coordinator disks when the cluster is online" on page 121.
The disk array containing two of the three coordinator disks is powered off. Node B abruptly crashes or a network partition occurs between node A and node B, and the disk array is still powered off.	Node A races for a majority of coordinator disks. Node A fails because only one of the three coordinator disks is available. Node A panics and removes itself from the cluster.	Node B has left cluster due to crash or network partition.	Power on the failed disk array and restart I/O fencing driver to enable Node A to register with all coordinator disks, or replace coordinator disks. See "Replacing defective disks when the cluster is offline" on page 200.

About CP server

This section discusses the CP server features.

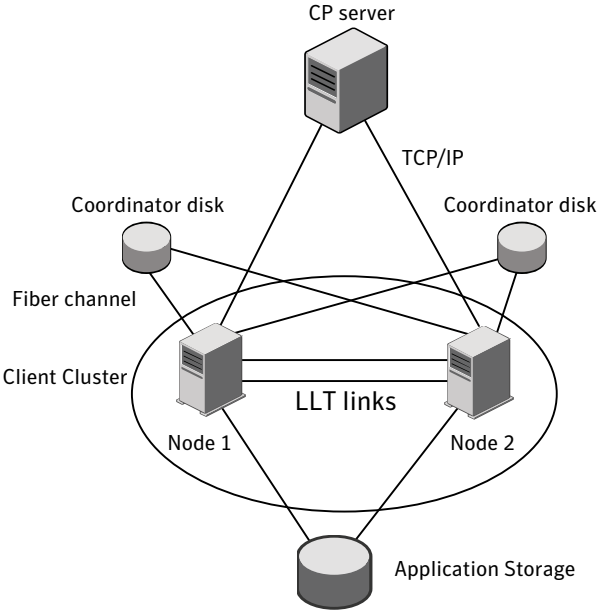
The following CP server features are described:

- SF Oracle RAC cluster configurations with server-based I/O fencing
- I/O fencing enhancements provided by the CP server
- About making CP server highly available
- Recommended CP server configurations
- About secure communication between the SF Oracle RAC cluster and CP server

Typical SF Oracle RAC cluster configuration with server-based I/O fencing

Figure 1-9 displays a configuration using a SF Oracle RAC cluster (with two nodes), a single CP server, and two coordinator disks. The nodes within the SF Oracle RAC cluster are connected to and communicate with each other using LLT links.

Figure 1-9 CP server, SF Oracle RAC cluster, and coordinator disks



I/O fencing enhancements provided by CP server

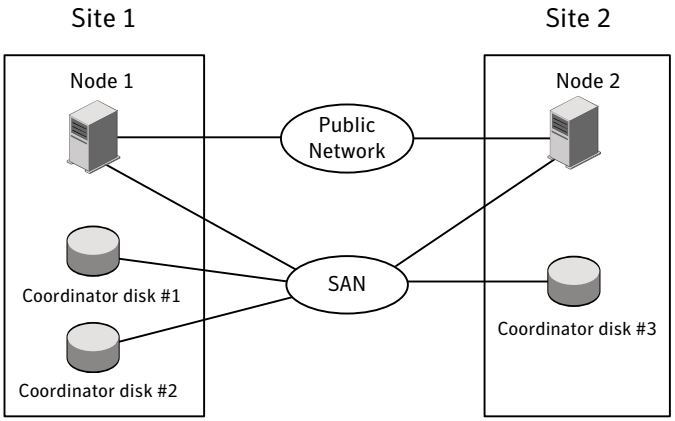
CP server configurations enhance disk-based I/O fencing by providing the following new capabilities:

- CP server configurations are scalable, and a configuration with three CP servers can provide I/O fencing for multiple SF Oracle RAC clusters. Since a single CP server configuration can serve a large number of SF Oracle RAC clusters, the cost of multiple SF Oracle RAC cluster deployments can be significantly reduced.
- Appropriately situated CP servers can eliminate any coordinator disk location bias in the I/O fencing process. For example, this location bias may occur where due to logistical restrictions two of the three coordinator disks are located at a single site, and the cost of setting up a third coordinator disk location is prohibitive.

See Figure 1-10 on page 54.
In such a configuration, if the site with two coordinator disks is inaccessible, the other site does not survive due to a lack of a majority of coordination points. I/O fencing would require extension of the SAN to the third site which may not be a suitable solution. An alternative is to place a CP server at a remote site as the third coordination point.

Note: The CP server provides an alternative arbitration mechanism without having to depend on SCSI-3 compliant coordinator disks. Data disk fencing in CVM will still require SCSI-3 I/O fencing.

Figure 1-10 Skewed placement of coordinator disks at Site 1



Defining Coordination Points

Three or more odd number of coordination points are required for I/O fencing. A coordination point can be either a CP server or a coordinator disk. A CP server provides the same functionality as a coordinator disk in an I/O fencing scenario. Therefore, it is possible to mix and match CP servers and coordinator disks for the purpose of providing arbitration.

Symantec supports the following three coordination point configurations:

- Vxfen driver based I/O fencing using SCSI-3 coordinator disks
- Customized fencing using a combination of SCSI-3 disks and CP server(s) as coordination points
- Customized fencing using only three CP servers as coordination points

Note: Symantec does not support a configuration where multiple CP servers are configured on the same machine.

Deployment and migration scenarios for CP server

Table 1-3 describes the supported deployment and migration scenarios, and the procedures you must perform on the SF Oracle RAC cluster and the CP server.

Table 1-3 CP server deployment and migration scenarios

Scenario	CP server	SF Oracle RAC cluster	Action required
Setup of CP server for a SF Oracle RAC cluster for the first time	New CP server	New SF Oracle RAC cluster using CP server as coordination point	On the designated CP server, perform the following tasks: 1 Prepare to configure the new CP server. 2 Configure the new CP server. On the SF Oracle RAC cluster nodes, configure server-based I/O fencing. See the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide* for the procedures.
Add a new SF Oracle RAC cluster to an existing and operational CP server	Existing and operational CP server	New SF Oracle RAC cluster	On the SF Oracle RAC cluster nodes, configure server-based I/O fencing. See the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide* for the procedures.
Replace the coordination point from an existing CP server to a new CP server	New CP server	Existing SF Oracle RAC cluster using CP server as coordination point	On the designated CP server, perform the following tasks: 1 Prepare to configure the new CP server. 2 Configure the new CP server. 3 Prepare the new CP server for use by the SF Oracle RAC cluster. See the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide* for the procedures. On a node in the SF Oracle RAC cluster, run the `vxfenswap` command to move to replace the CP server: See "Replacing coordination points for server-based fencing in an online cluster" on page 143.

Table 1-3 CP server deployment and migration scenarios *(continued)*

Scenario	CP server	SF Oracle RAC cluster	Action required
Replace the coordination point from an existing CP server to an operational CP server coordination point	Operational CP server	Existing SF Oracle RAC cluster using CP server as coordination point	On the designated CP server, prepare to configure the new CP server manually. See the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide* for the procedures. On a node in the SF Oracle RAC cluster, run the `vxfenswap` command to move to replace the CP server: See "Replacing coordination points for server-based fencing in an online cluster" on page 143.

Enabling fencing in a SF Oracle RAC cluster with a new CP server coordination point	New CP server	Existing SF Oracle RAC cluster with fencing configured in disabled mode	**Note:** Migrating from fencing in disabled mode to customized mode incurs application downtime on the SF Oracle RAC cluster.
			On the designated CP server, perform the following tasks:
			1 Prepare to configure the new CP server.
			2 Configure the new CP server
			3 Prepare the new CP server for use by the SF Oracle RAC cluster
			See the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide* for the procedures.
			On the SF Oracle RAC cluster nodes, perform the following:
			1 Stop all applications, VCS, and fencing on the SF Oracle RAC cluster.
			2 To stop VCS, use the following command (to be run on all the SF Oracle RAC cluster nodes):
			`# hastop -local`
			3 Stop fencing using the following command:
			`# /etc/init.d/vxfen stop`
			4 Reconfigure I/O fencing on the SF Oracle RAC cluster.
			See the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide* for the procedures.

Table 1-3 CP server deployment and migration scenarios *(continued)*

Scenario	CP server	SF Oracle RAC cluster	Action required

Enabling fencing in a SF Oracle RAC cluster with an operational CP server coordination point	Operational CP server	Existing SF Oracle RAC cluster with fencing configured in disabled mode	**Note:** Migrating from fencing in disabled mode to customized mode incurs application downtime.
			
			On the designated CP server, prepare to configure the new CP server.
			
			See the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide* for this procedure.
			
			On the SF Oracle RAC cluster nodes, perform the following tasks:
			
			1 Stop all applications, VCS, and fencing on the SF Oracle RAC cluster.
			
			2 To stop VCS, use the following command (to be run on all the SF Oracle RAC cluster nodes):
			
			`# hastop -local`
			
			3 Stop fencing using the following command:
			
			`# /etc/init.d/vxfen stop`
			
			4 Reconfigure fencing on the SF Oracle RAC cluster.
			
			See the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide* for the procedures.

Table 1-3 CP server deployment and migration scenarios *(continued)*

Scenario	CP server	SF Oracle RAC cluster	Action required

Enabling fencing in a SF Oracle RAC cluster with a new CP server coordination point	New CP server	Existing SF Oracle RAC cluster with fencing configured in scsi3 mode	On the designated CP server, perform the following tasks:
			1 Prepare to configure the new CP server.
			2 Configure the new CP server
			3 Prepare the new CP server for use by the SF Oracle RAC cluster
			See the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide* for the procedures.
			Based on whether the cluster is online or offline, perform the following procedures:
			For a cluster that is online, perform the following task on the SF Oracle RAC cluster:
			◆ Run the `vxfenswap` command to migrate from disk-based fencing to the server-based fencing.
			See "Migrating from disk-based to server-based fencing in an online cluster" on page 60.
			For a cluster that is offline, perform the following tasks on the SF Oracle RAC cluster:
			1 Stop all applications, VCS, and fencing on the SF Oracle RAC cluster.
			2 To stop VCS, use the following command (to be run on all the SF Oracle RAC cluster nodes):
			`# hastop -local`
			3 Stop fencing using the following command:
			`# /etc/init.d/vxfen stop`
			4 Reconfigure I/O fencing on the SF Oracle RAC cluster.
			See the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide* for the procedures.

Table 1-3 CP server deployment and migration scenarios *(continued)*

Scenario	CP server	SF Oracle RAC cluster	Action required

Enabling fencing in a SF Oracle RAC cluster with an operational CP server coordination point	Operational CP server	Existing SF Oracle RAC cluster with fencing configured in disabled mode	On the designated CP server, prepare to configure the new CP server.
			See the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide* for this procedure.
			Based on whether the cluster is online or offline, perform the following procedures:
			For a cluster that is online, perform the following task on the SF Oracle RAC cluster:
			◆ Run the `vxfenswap` command to migrate from disk-based fencing to the server-based fencing.
			See "Migrating from disk-based to server-based fencing in an online cluster" on page 60.
			For a cluster that is offline, perform the following tasks on the SF Oracle RAC cluster:
			1 Stop all applications, VCS, and fencing on the SF Oracle RAC cluster.
			2 To stop VCS, use the following command (to be run on all the SF Oracle RAC cluster nodes):
			`# hastop -local`
			3 Stop fencing using the following command:
			`# /etc/init.d/vxfen stop`
			4 Reconfigure fencing on the SF Oracle RAC cluster.
			See the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide* for the procedures.

Table 1-3 CP server deployment and migration scenarios *(continued)*

Scenario	CP server	SF Oracle RAC cluster	Action required
Refreshing registrations of SF Oracle RAC cluster nodes on coordination points (CP servers/ coordinator disks) without incurring application downtime	Operational CP server	Existing SF Oracle RAC cluster using the CP server as coordination point	On the SF Oracle RAC cluster run the `vxfenswap` command to refresh the keys on the CP server: See "Refreshing registration keys on the coordination points for server-based fencing" on page 141.

Migrating from disk-based to server-based fencing in an online cluster

You can migrate between disk-based fencing and server-based fencing without incurring application downtime in the SF Oracle RAC clusters.

You can migrate from disk-based fencing to server-based fencing in the following cases:

- You want to leverage the benefits of server-based fencing.
- You want to replace faulty coordinator disks with CP servers.

Similarly, you can migrate from server-based fencing to disk-based fencing when you want to perform maintenance tasks on the CP server systems.

See "Migrating from server-based to disk-based fencing in an online cluster" on page 62.

To migrate from disk-based fencing to server-based fencing

1. Make sure system-to-system communication is functioning properly.
2. Make sure that the SF Oracle RAC cluster is online and uses disk-based fencing.

   ```
   # vxfenadm -d
   ```

   For example, if SF Oracle RAC cluster uses disk-based fencing:

   ```
   I/O Fencing Cluster Information:
   ================================
   Fencing Protocol Version: 201
   Fencing Mode: SCSI3
   Fencing SCSI3 Disk Policy: dmp
   Cluster Members:
     * 0 (galaxy)
     1 (nebula)
   RFSM State Information:
     node 0 in state 8 (running)
     node 1 in state 8 (running)
   ```

3. Make sure that you performed the following tasks on the designated CP server:
   - Preparing to configure the new CP server.
   - Configuring the new CP server
   - Preparing the new CP server for use by the SF Oracle RAC cluster

   See the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide* for the procedures.
4. Create a new `/etc/vxfenmode.test` file on each SF Oracle RAC cluster node with the fencing configuration changes such as the CP server information.

   Refer to the sample `vxfenmode` files in the `/etc/vxfen.d` folder.
5. From any node in the SF Oracle RAC cluster, start the vxfenswap utility:

   ```
   # vxfenswap [-n]
   ```

6. Review the message that the utility displays and confirm whether you want to commit the change.
   - If you do not want to commit the new fencing configuration changes, press Enter or answer **n** at the prompt.

     ```
     Do you wish to commit this change? [y/n] (default: n) n
     ```

The vxfenswap utility rolls back the migration operation.

- If you want to commit the new fencing configuration changes, answer **y** at the prompt.

  ```
  Do you wish to commit this change? [y/n] (default: n) y
  ```

  If the utility successfully commits, the utility moves the `/etc/vxfenmode.test` file to the `/etc/vxfenmode` file.

7 After the migration is complete, verify the change in the fencing mode.

```
# vxfenadm -d
```

For example, after the migration from disk-based fencing to server-based fencing in the SF Oracle RAC cluster:

```
I/O Fencing Cluster Information:
================================
Fencing Protocol Version: 201
Fencing Mode: Customized
Fencing Mechanism: cps
Cluster Members:
  * 0 (galaxy)
  1 (nebula)
RFSM State Information:
  node 0 in state 8 (running)
  node 1 in state 8 (running)
```

8 Verify the current coordination points that the vxfen driver uses.

```
# vxfenconfig -l
```

Migrating from server-based to disk-based fencing in an online cluster

You can migrate between disk-based fencing and server-based fencing without incurring application downtime in the SF Oracle RAC clusters.

You can migrate from server-based fencing to disk-based fencing when you want to perform maintenance tasks on the CP server systems.

Similarly, you can migrate from disk-based fencing to server-based fencing in the following cases:

- You want to leverage the benefits of server-based fencing.
- You want to replace faulty coordinator disks with CP servers.

See "Migrating from disk-based to server-based fencing in an online cluster" on page 60.

To migrate from server-based fencing to disk-based fencing

1. Make sure system-to-system communication is functioning properly.
2. Make sure that the SF Oracle RAC cluster is online.

   ```
   # vxfenadm -d
   ```

   For example, if SF Oracle RAC cluster uses server-based fencing:

   ```
   I/O Fencing Cluster Information:
   ================================
   Fencing Protocol Version: 201
   Fencing Mode: Customized
   Fencing Mechanism: cps
   Cluster Members:
     * 0 (galaxy)
     1 (nebula)
   RFSM State Information:
     node 0 in state 8 (running)
     node 1 in state 8 (running)
   ```

3. Make sure that you performed the following preparatory tasks to configure disk-based fencing:
   - Identifying disks to use as coordinator disks
   - Setting up coordinator disk group
   - Creating I/O configuration files

   See the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide* for the procedures.
4. Create a new `/etc/vxfenmode.test` file with the fencing configuration changes such as the scsi3 disk policy information.

   Refer to the sample `vxfenmode` files in the `/etc/vxfen.d` folder.
5. From any node in the SF Oracle RAC cluster, start the vxfenswap utility:

   ```
   # vxfenswap -g diskgroup [-n]
   ```

6. Review the message that the utility displays and confirm whether you want to commit the change.

- If you do not want to commit the new fencing configuration changes, press Enter or answer **n** at the prompt.

  ```
  Do you wish to commit this change? [y/n] (default: n) n
  ```

  The vxfenswap utility rolls back the migration operation.

- If you want to commit the new fencing configuration changes, answer **y** at the prompt.

  ```
  Do you wish to commit this change? [y/n] (default: n) y
  ```

  If the utility successfully commits, the utility moves the `/etc/vxfenmode.test` file to the `/etc/vxfenmode` file.

7 After the migration is complete, verify the change in the fencing mode.

```
# vxfenadm -d
```

For example, after the migration from server-based fencing to disk-based fencing in the SF Oracle RAC cluster:

```
I/O Fencing Cluster Information:
================================
Fencing Protocol Version: 201
Fencing Mode: SCSI3
Fencing SCSI3 Disk Policy: dmp
Cluster Members:
  * 0 (galaxy)
  1 (nebula)
RFSM State Information:
  node 0 in state 8 (running)
  node 1 in state 8 (running)
```

8 Verify the current coordination points that the vxfen driver uses.

```
# vxfenconfig -l
```

About making CP server highly available

If you want to configure a multi-node CP server cluster, install and configure SFHA on the CP server nodes. Otherwise, install and configure VCS on the single node.

In both the configurations, VCS provides local start and stop of the CP server process, taking care of dependencies such as NIC, IP address, etc.. Moreover, VCS also serves to restart the CP server process in case the process faults.

To make the CP server process highly available, you must perform the following tasks:

- Install and configure SFHA on the CP server systems.
- Configure the CP server process as a failover service group.
- Configure disk-based I/O fencing to protect the shared CP server database.

Note: Symantec recommends that you do not run any other applications on the single node or SFHA cluster that is used to host CP server.

A single CP server can serve multiple SF Oracle RAC clusters. A common set of CP servers can serve up to 128 SF Oracle RAC clusters.

About secure communication between the SF Oracle RAC cluster and CP server

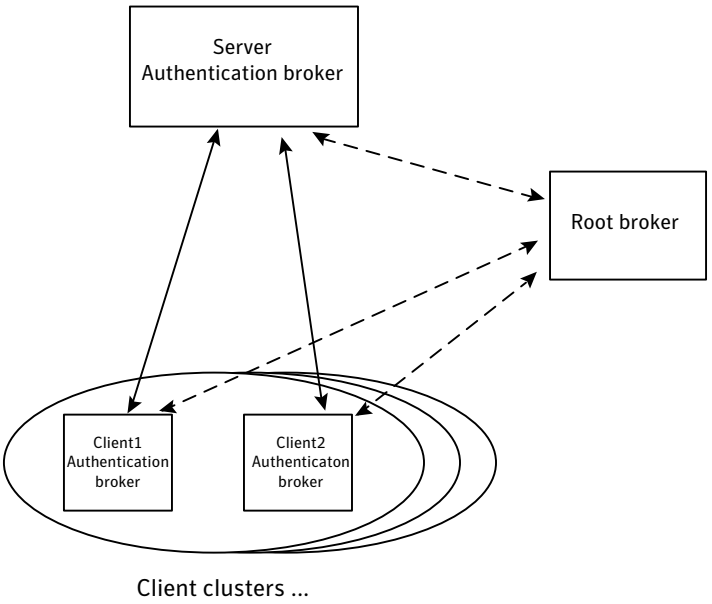
In a data center, TCP/IP communication between the SF Oracle RAC cluster and CP server must be made secure. The security of the communication channel involves encryption, authentication, and authorization.

The CP server node or cluster needs to confirm the authenticity of the SF Oracle RAC cluster nodes that communicate with it as a coordination point and only accept requests from known SF Oracle RAC cluster nodes. Requests from unknown clients are rejected as non-authenticated. Similarly, the fencing framework in SF Oracle RAC cluster must confirm that authentic users are conducting fencing operations with the CP server.

The encryption and authentication service for CP server is provided by Symantec™ Product Authentication Service. To enable Symantec™ Product Authentication Service, the VRTSat package is installed on the SF Oracle RAC clusters as well as CP server, as a part of VCS product installation.

Figure 1-11 displays a schematic of secure communication between the SF Oracle RAC cluster and CP server. An authentication broker is configured on CP server and each SF Oracle RAC cluster node which authenticates clients such as users or services, and grants them a product credential.

Figure 1-11 CP server and SF Oracle RAC clusters with authentication broker and root broker



Entities on behalf of which authentication is done, are referred to as principals. On the SF Oracle RAC cluster nodes, the current VCS installer creates the Authentication Server credentials on each node in the cluster, creates Web credentials for VCS users, and then sets up trust with the root broker. It also creates a VCS service group for the authentication broker. The installer then proceeds to start VCS in secure mode.

Typically, in an existing VCS cluster with security configured, a root broker would already have been configured and an authentication broker will be running on each cluster node.

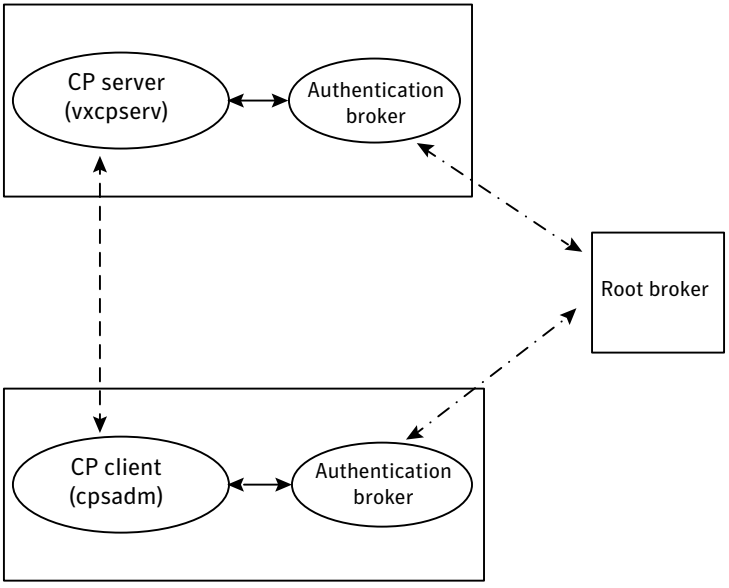
How secure communication between the CP servers and SF Oracle RAC clusters work

CP server and SF Oracle RAC cluster node communication involve the following entities:

- vxcpserv for the CP server
- cpsadm for the SF Oracle RAC cluster node

Figure 1-12 displays a schematic of the end-to-end communication flow with security enabled on CP server and SF Oracle RAC clusters.

Figure 1-12 End-To-end communication flow with security enabled on CP server and SF Oracle RAC clusters



Communication flow between CP server and SF Oracle RAC cluster nodes with security configured on them is as follows:

- Initial setup:
  Identities of authentication brokers configured on CP server, as well as SF Oracle RAC cluster nodes are configured in the root broker's authentication private domain repository.

  Note: If authentication brokers configured on CP server and SF Oracle RAC cluster nodes do not use the same root broker, then a trust should be established between the root brokers or authentication brokers, so that vxcpserv process can authenticate requests from theSF Oracle RAC cluster nodes.

  The `cpsadm` command gets the user name, domain type from the environment variables CPS_USERNAME, CPS_DOMAINTYPE. The user is expected to export these variables before running the `cpsadm` command manually. The customized fencing framework exports these environment variables internally before running the `cpsadm` commands.

The cp server process (vxcpserv) uses its own user (_CPS_SERVER_) which is added to the local authentication broker during server startup.

- Getting credentials from authentication broker:
  The `cpsadm` command tries to get the existing credentials from authentication broker running on the local node. If this fails, it tries to authenticate itself with the local authentication broker.
  The vxcpserv process tries to get the existing credentials from authentication broker running on the local node. If this fails, it tries to authenticate itself with the local authentication broker and creates a principal for itself .
- Communication between CP server and SF Oracle RAC cluster nodes:
  Once the CP server is up after establishing its credential, it becomes ready to receive data from the clients. Once authenticated with the local authentication broker, cpsadm connects to the CP server. Data is passed over to the CP server.
- Validation:
  On receiving data from a particular SF Oracle RAC cluster node, vxcpserv validates its credentials by consulting the local authentication broker. If validation fails, then the connection request data is rejected.

Security configuration details on CP server and SF Oracle RAC cluster

This section discusses the security configuration details for the CP server and SF Oracle RAC cluster.

Settings in secure mode

The following are the settings for secure communication between the CP server and SF Oracle RAC cluster:

- CP server settings:
  A user gets created in the local authentication broker during CP server startup with the following values:
  - username: _CPS_SERVER_
  - domainname: _CPS_SERVER_DOMAIN@*FQHN*
  - domaintype: vx

  where, FQHN is Fully Qualified Host Name of the client node
  Run the following command on the CP server to verify the settings:

```
# /opt/VRTScps/bin/cpsat showcred
```

Note: The CP server configuration file (/etc/vxcps.conf) must not contain a line specifying **security=0**. If there is no line specifying "security" parameter or if there is a line specifying **security=1**, CP server with security is enabled (which is the default).

- SF Oracle RAC cluster node(s) settings:
  On SF Oracle RAC cluster, a user gets created for each cluster node in the local authentication broker during VCS security configuration with the following values:
  - username: _HA_VCS_hostname
  - domainname: HA_SERVICES@*FQHN*
  - domaintype: vx

  where, FQHN is Fully Qualified Host Name of the client node
  Run the following command on the SF Oracle RAC cluster node(s) to verify the security settings:

  ```
  # /opt/VRTScps/bin/cpsat showcred
  ```

The users described above are used only for authentication for the communication between the following:

- CP server and authentication broker configured on it
- SF Oracle RAC cluster nodes and authentication brokers configured on them

For CP server's authorization, the following user gets created and used by customized fencing framework on the SF Oracle RAC cluster, if security is configured:

_HA_VCS_*hostname*@HA_SERVICES@*FQHN*

where, hostname is the client node name without qualification and FQHN is Fully Qualified Host Name of the client node.

For each SF Oracle RAC cluster node, this user must be registered on the CP server database before fencing starts on the SF Oracle RAC cluster node(s). This can be verified by issuing the following command:

```
# cpsadm -s cp_server -a list_users
```

The following is an example of the command output:

```
Username/Domain Type                             Cluster Name / UUID             Role
_HA_VCS_galaxy@HA_SERVICES@galaxy.symantec.com/vx  cluster1/ {f0735332-e3709c1c73b9} Operator
```

Note: The configuration file (/etc/vxfenmode) on each client node must not contain a line specifying **security=0**. If there is no line specifying "security" parameter or if there is a line specifying **security=1**, client node starts with security enabled (which is the default).

Settings in non-secure mode

In non-secure mode, only authorization is provided on the CP server. Passwords are not requested. Authentication and encryption are not provided. User credentials of “cpsclient@hostname” of “vx” domaintype are used by the customized fencing framework for communication between CP server or SF Oracle RAC cluster node(s).

For each SF Oracle RAC cluster node, this user must be added on the CP server database before fencing starts on the SF Oracle RAC cluster node(s). The user can be verified by issuing the following command:

```
# cpsadm -s cpserver -a list_users
```

The following is an example of the command output:

```
Username/Domain Type  Cluster Name / UUID                 Role
cpsclient@galaxy/vx   cluster1 / {f0735332-e3709c1c73b9}  Operator
```

Note: In non-secure mode, CP server configuration file (/etc/vxcps.conf) should contain a line specifying **security=0**. Similarly, on each SF Oracle RAC cluster node the configuration file (/etc/vxfenmode) should contain a line specifying **security=0**.

Periodic health evaluation of the clusters

SF Oracle RAC provides a health check utility that evaluates the components and configuration in an SF Oracle RAC cluster. The utility when invoked gathers real-time operational information on cluster components and displays the report on your system console. You must run the utility on each node in the cluster.

The utility evaluates the health of the following components:

- Low Latency Transport (LLT)
- I/O fencing
- Oracle Clusterware/Grid Infrastructure
- PrivNIC

The health check utility is installed at `/opt/VRTSvcs/rac/healthcheck/healthcheck` during the installation of SF Oracle RAC.

The utility determines the health of the components by gathering information from configuration files or by reading the threshold values set in the health check configuration file `/opt/VRTSvcs/rac/healthcheck/healthcheck.cf`. It displays a warning message when the operational state of the component violates the configured settings for the component or when the health score approaches or exceeds the threshold value. You can modify the health check configuration file to set the threshold values to the desired level.

The health checks for the I/O fencing, PrivNIC, and Oracle Clusterware components do not use threshold settings.

Note: You must set the `ORACLE_HOME` and `CRS_HOME` parameters in the configuration file as appropriate for your setup.

Table 1-4 provides guidelines on changing the threshold values.

Table 1-4 Setting threshold values

Requirement	Setting the threshold
To detect warnings early	Reduce the corresponding threshold value.
To suppress warnings	Increase the corresponding threshold value. **Caution:** Using very high threshold values can prevent the health check utility from forecasting potential problems in the cluster. Exercise caution with high values.

Note: You can schedule periodic health evaluation of your clusters, by scheduling the utility to run as a cron job.

See "Scheduling periodic health checks for your cluster" on page 85.

For detailed information on the list of health checks performed for each component, see the appendix *List of SF Oracle RAC health checks*.

Chapter 2

Administering SF Oracle RAC and its components

This chapter includes the following topics:

- Administering SF Oracle RAC
- Administering VCS
- Administering I/O fencing
- Administering the CP server
- Administering CFS
- Administering CVM
- Administering SF Oracle RAC global clusters

Administering SF Oracle RAC

This section provides instructions for the following SF Oracle RAC administration tasks:

- Setting the environment variables
  See "Setting the environment variables" on page 74.
- Starting or stopping SF Oracle RAC on each node
  See "Starting or stopping SF Oracle RAC on each node" on page 75.
- Applying Oracle patches
  See "Applying Oracle patches" on page 79.
- Installing Veritas Volume Manager or Veritas File System patches

See "Installing Veritas Volume Manager, Veritas File System, or ODM patches on SF Oracle RAC nodes" on page 80.

- Applying operating system updates
  See "Applying operating system updates" on page 81.
- Adding storage to an SF Oracle RAC cluster
  See "Adding storage to an SF Oracle RAC cluster" on page 82.
- Recovering from storage failure
  See "Recovering from storage failure" on page 83.
- Enhancing the performance of SF Oracle RAC clusters
  See "Enhancing the performance of SF Oracle RAC clusters" on page 84.
- Creating snapshots for offhost processing
  See "Creating snapshots for offhost processing" on page 84.
- Managing database storage efficiently using SmartTier
  See "Managing database storage efficiently using SmartTier" on page 85.
- Optimizing database storage using Thin Provisioning and SmartMove
  See "Optimizing database storage using Thin Provisioning and SmartMove" on page 85.
- Scheduling periodic health checks for your cluster
  See "Scheduling periodic health checks for your cluster" on page 85.
- Verifying the ODM port
  See "Verifying the ODM port" on page 86.
- Verifying the nodes in a cluster
  See "Verifying the nodes in a cluster" on page 86.

If you encounter issues while administering SF Oracle RAC, refer to the troubleshooting section for assistance.

Setting the environment variables

Set the MANPATH variable in the .profile file (or other appropriate shell setup file for your system) to enable viewing of manual pages.

Based on the shell you use, type one of the following:

For sh, ksh, or bash

```
# export MANPATH=$MANPATH:\
/opt/VRTS/man
```

For csh

```
# setenv MANPATH $MANPATH:/opt/VRTS/man
```

Some terminal programs may display garbage characters when you view the man pages. Set the environment variable LC_ALL=C to resolve this issue.

Set the PATH environment variable in the .profile file (or other appropriate shell setup file for your system) on each system to include installation and other commands.

Based on the shell you use, type one of the following:

For sh, ksh, or bash

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

Starting or stopping SF Oracle RAC on each node

You can start or stop SF Oracle RAC on each node in the cluster using the SF Oracle RAC installer or manually.

To start SF Oracle RAC	Using installer: See "Starting SF Oracle RAC using the SF Oracle RAC installer" on page 75. Manual: See "Starting SF Oracle RAC manually on each node" on page 76.
To stop SF Oracle RAC	Using installer: See "Stopping SF Oracle RAC using the SF Oracle RAC installer" on page 77. Manual: See "Stopping SF Oracle RAC manually on each node" on page 77.

Starting SF Oracle RAC using the SF Oracle RAC installer

Run the SF Oracle RAC installer with the `-start` option to start SF Oracle RAC on each node.

To start SF Oracle RAC using the SF Oracle RAC installer

1. Log into one of the nodes in the cluster as the root user.
2. Start SF Oracle RAC:

```
# /opt/VRTS/install/installsfrac -start galaxy nebula
```

Starting SF Oracle RAC manually on each node

Perform the steps in the following procedures to start SF Oracle RAC manually on each node.

To start SF Oracle RAC manually on each node

1. Log into each node as the root user.
2. Start LLT:

```
# /etc/init.d/llt start
```

3. Start GAB:

```
# /etc/init.d/gab start
```

4. Start fencing:

```
# /etc/init.d/vxfen start
```

5. Start GLM:

```
# /etc/init.d/vxglm start
```

6. Start GMS:

```
# /etc/init.d/vxgms start
```

7. Start VCSMM:

```
# /etc/init.d/vcsmm start
```

8. Start ODM:

```
# /etc/init.d/vxodm start
```

9 Start VCS, CVM, and CFS:

```
# hastart
```

10 Verify that all GAB ports are up and running:

```
# gabconfig -a
GAB Port Memberships
===============================================================
Port a gen   3c4c05 membership 01
Port b gen   3c4c0b membership 01
Port d gen   3c4c06 membership 01
Port f gen   3c4c15 membership 01
Port h gen   3c4c19 membership 01
Port o gen   3c4c0d membership 01
Port u gen   3c4c17 membership 01
Port v gen   3c4c11 membership 01
Port w gen   3c4c13 membership 01
```

Stopping SF Oracle RAC using the SF Oracle RAC installer

Run the SF Oracle RAC installer with the `-stop` option to stop SF Oracle RAC on each node.

To stop SF Oracle RAC using the SF Oracle RAC installer

1 Log into one of the nodes in the cluster as the root user.

2 Stop VCS:

```
# hastop -all
```

3 Stop SF Oracle RAC:

```
# /opt/VRTS/install/installsfrac -stop galaxy nebula
```

Stopping SF Oracle RAC manually on each node

Perform the steps in the following procedures to stop SF Oracle RAC manually on each node.

To stop SF Oracle RAC manually on each node

1. Stop the Oracle database.

   If the Oracle RAC instance is managed by VCS, log in as the root user and take the service group offline:

   ```
   # hagrp -offline oracle_group -sys node_name
   ```

   If the Oracle database instance is not managed by VCS, log in as the Oracle user on one of the nodes and shut down the instance:

   ```
   $ srvctl stop instance -d db_name \
   -i instance_name
   ```

2. Stop all applications that are not configured under VCS. Use native application commands to stop the application.

3. Unmount the VxFS file systems that are not managed by VCS.

   Make sure that no processes are running, which make use of mounted shared file system or shared volumes:

   ```
   # mount -v | grep vxfs
   # fuser -cu /mount_point
   ```

   Unmount the VxFS file system:

   ```
   # umount /mount_point
   ```

4. Take the VCS service groups offline:

   ```
   # hagrp -offline group_name -sys node_name
   ```

   Verify that the VCS service groups are offline:

   ```
   # hagrp -state group_name
   ```

5 Stop VCS, CVM and CFS:

```
# hastop -local
```

Verify that the ports 'f', 'u', 'v', 'w' and 'h' are closed:

```
# gabconfig -a
GAB Port Memberships
===============================================================
Port a gen   761f03 membership 01
Port b gen   761f08 membership 01
Port d gen   761f02 membership 01
Port o gen   761f01 membership 01
```

6 Stop ODM:

```
# /etc/init.d/vxodm stop
```

7 Stop GMS:

```
# /etc/init.d/vxgms stop
```

8 Stop GLM:

```
# /etc/init.d/vxglm stop
```

9 Stop VCSMM:

```
# /etc/init.d/vcsmm stop
```

10 Stop fencing:

```
# /etc/init.d/vxfen stop
```

11 Stop GAB:

```
# /etc/init.d/gab stop
```

12 Stop LLT:

```
# /etc/init.d/llt stop
```

Applying Oracle patches

Before installing any Oracle RAC patch or patchset software:

- Review the latest information on supported Oracle RAC patches and patchsets: http://entsupport.symantec.com/docs/280186
- You must have installed the base version of the Oracle RAC software.

To apply Oracle patches

1. Stop the Oracle database.

   If the database instances are not managed by VCS, run the following on one of the nodes in the cluster:

   ```
   $ srvctl stop database -d db_name
   ```

   If the database instances are managed by VCS, take the corresponding VCS service groups offline. As superuser, enter:

   ```
   # hagrp -offline group_name -any
   ```

2. Install the patches or patchsets required for your Oracle RAC installation.

   For instructions, see the Oracle documentation that accompanies the patch or patchset.

3. Relink the SF Oracle RAC libraries with Oracle libraries.

   For instructions, see the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide*.

4. Start the Oracle database.

   If the database instances are not managed by VCS, run the following on one of the nodes in the cluster:

   ```
   $ srvctl start database -d db_name
   ```

   If the database instances are managed by VCS, bring the corresponding VCS service groups online. As superuser, enter:

   ```
   # hagrp -online group_name -any
   ```

Installing Veritas Volume Manager, Veritas File System, or ODM patches on SF Oracle RAC nodes

Perform the steps on each node in the cluster to install Veritas Volume Manager, Veritas File System, or ODM patches.

To install Veritas Volume Manager, Veritas File System, or ODM patches on SF Oracle RAC nodes

1. Log in as the root user.
2. Stop SF Oracle RAC manually on each node.

   See "Stopping SF Oracle RAC manually on each node" on page 77.
3. Install the VxVM, VxFS, or ODM patch:

   ```
   # cd patch_dir
   ```

   where *patch_dir* is the directory that contains the VxVM or VxFS patch you want to install.

   ```
   # rpm -Uvh rpm_name
   ```
4. If there are applications that are not managed by VCS, start the applications manually using native application commands.

Applying operating system updates

If you need to apply updates to the base version of the operating system, perform the steps in this section on each node of the cluster, one node at a time.

To apply operating system updates

1. Log in to the node as the root user and change to `/opt/VRTS/install` directory:

   ```
   # cd /opt/VRTS/install
   ```
2. Take the VCS service groups offline:

   ```
   # hagrp -offline grp_name -sys node_name
   ```
3. Stop SF Oracle RAC:

   ```
   # ./installsfrac -stop
   ```
4. Upgrade the operating system.

   For instructions, see the operating system documentation.

5 Reboot the node:

```
# shutdown -r now
```

6 Repeat all the steps on each node in the cluster.

Adding storage to an SF Oracle RAC cluster

You can add storage to an SF Oracle RAC cluster in the following ways:

Extend the volume space on a disk group	Use the `vxresize` command to extend the volume space on a disk group. It automatically locates available disk space on the specified volume and frees up unused space to the disk group for later use. See the `vxresize` (1M) manual page for information on various options. See "To extend the volume space on a disk group" on page 83.
Add a disk to a disk group	Use the `vxdg` command to add a disk to a disk group. See the `vxdg` (1M) manual page for information on various options. See "To add a disk to a disk group" on page 83.

To extend the volume space on a disk group

1 Determine the length by which you can increase an existing volume.

```
# vxassist [-g diskgroup] maxgrow volume_name
```

For example, to determine the maximum size the volume `oradatavol` in the disk group `oradatadg` can grow, given its attributes and free storage available:

```
# vxassist -g oradatadg maxgrow oradatavol
```

2 Extend the volume, as required. You can extend an existing volume to a certain length by specifying the new size of the volume (the new size must include the additional space you plan to add). You can also extend a volume by a certain length by specifying the additional amount of space you want to add to the existing volume.

To extend a volume to a certain length	For example, to extend the volume `oradata_vol` of size 10 GB in the disk group `oradata_dg` to 30 GB: `# vxresize -g oradata_dg \ oradata_vol 30g`
To extend a volume by a certain length	For example, to extend the volume `oradata_vol` of size 10 GB in the disk group `oradata_dg` by 10 GB: `# vxresize -g oradata_dg \ oradata_vol +10g`

To add a disk to a disk group

1 Determine the CVM master node in the cluster:

```
# vxdctl -c mode
```

2 Add the disk to the disk group:

```
# vxdg -g dg_name adddisk disk_name
```

Recovering from storage failure

Veritas Volume Manager (VxVM) protects systems from disk and other hardware failures and helps you to recover from such events. Recovery procedures help you prevent loss of data or system access due to disk and other hardware failures.

For information on various failure and recovery scenarios, see the *Veritas Volume Manager Troubleshooting Guide*.

Enhancing the performance of SF Oracle RAC clusters

The main components of clustering that impact the performance of an SF Oracle RAC cluster are:

- Kernel components, specifically LLT and GAB
- VCS engine (had)
- VCS agents

Each VCS agent process has two components—the agent framework and the agent functions. The agent framework provides common functionality, such as communication with the HAD, multithreading for multiple resources, scheduling threads, and invoking functions. Agent functions implement functionality that is particular to an agent. For various options provided by the clustering components to monitor and enhance performance, see the chapter "VCS performance considerations" in the *Veritas Cluster Server Administrator's Guide*.

Veritas Volume Manager can improve system performance by optimizing the layout of data storage on the available hardware. For more information on tuning Veritas Volume Manager for better performance, see the chapter "Performance monitoring and tuning" in the *Veritas Volume Manager Administrator's Guide*.

Veritas Volume Replicator Advisor (VRAdvisor) is a planning tool that helps you determine an optimum Veritas Volume Replicator (VVR) configuration. For installing VRAdvisor and evaluating various parameters using the data collection and data analysis process, see the *Veritas Volume Replicator Advisor User's Guide*.

Mounting a snapshot file system for backups increases the load on the system as it involves high resource consumption to perform copy-on-writes and to read data blocks from the snapshot. In such situations, cluster snapshots can be used to do off-host backups. Off-host backups reduce the load of a backup application from the primary server. Overhead from remote snapshots is small when compared to overall snapshot overhead. Therefore, running a backup application by mounting a snapshot from a relatively less loaded node is beneficial to overall cluster performance.

Creating snapshots for offhost processing

You can capture a point-in-time copy of actively changing data at a given instant. You can then perform system backup, upgrade, and other maintenance tasks on the point-in-time copies while providing continuous availability of your critical

data. If required, you can offload processing of the point-in-time copies onto another host to avoid contention for system resources on your production server.

For instructions on creating snapshots for offhost processing, see the *Veritas Storage Foundation: Storage and Availability Management for Oracle Databases* guide.

Managing database storage efficiently using SmartTier

SmartTier matches data storage with data usage requirements. After data matching, the data can be relocated based upon data usage and other requirements determined by the storage or database administrator (DBA). The technology enables the database administrator to manage data so that less frequently used data can be moved to slower, less expensive disks. This also permits the frequently accessed data to be stored on faster disks for quicker retrieval.

For instructions on managing database storage efficiently using SmartTier, see the *Veritas Storage Foundation: Storage and Availability Management for Oracle Databases* guide.

Optimizing database storage using Thin Provisioning and SmartMove

Thin Provisioning is a storage array feature that optimizes storage use by automating storage provisioning. With Storage Foundation's SmartMove feature, Veritas File System (VxFS) lets Veritas Volume Manager (VxVM) know which blocks have data. VxVM, which is the copy engine for migration, copies only the used blocks and avoids the empty spaces, thus optimizing thin storage utilization.

For instructions on managing database storage efficiently using Thin Provisioning and SmartMove, see the *Veritas Storage Foundation Advanced Features Administrator's Guide*.

Scheduling periodic health checks for your cluster

You can manually invoke the health check utility at any time or schedule the utility to run as a cron job.

If the health check completes without any warnings, the following message is displayed:

```
Success: The SF Oracle RAC components are working fine on this node.
```

If you manually invoke the health check, run the utility on each node in the cluster. The results of the check are displayed on the console.

Log in as the root user on each node in the cluster and run the following commands:

```
# cd  /opt/VRTSvcs/rac/healthcheck/
# ./healthcheck
```

If you want to schedule periodic health checks for your cluster, create a cron job that runs on each node in the cluster. Redirect the health check report to a file.

Verifying the ODM port

It is recommended to enable ODM in SF Oracle RAC. Run the following command to verify that ODM is running:

```
# gabconfig -a | grep "Port d"
```

Verifying the nodes in a cluster

Table 2-1 lists the various options that you can use to periodically verify the nodes in your cluster.

Table 2-1 Options for verifying the nodes in a cluster

Type of check	Description
Veritas Operations Services (VOS) checks	Use the Veritas Operations Services (VOS) to evaluate your systems before and after any installation, configuration, upgrade, patch updates, or other routine administrative activities. The utility performs a number of compatibility and operational checks on the cluster that enable you to diagnose and troubleshoot issues in the cluster. The utility is periodically updated with new features and enhancements. For more information and to download the utility, visit http://go.symantec.com/vos.

SF Oracle RAC installation and configuration checks	Use the option "SF Oracle RAC Installation and Configuration Checks" in the SF Oracle RAC installer to evaluate your systems before and after any installation, configuration, upgrade, patch updates, or other routine administrative activities. The utility is typically updated when a new version of SF Oracle RAC is released.
	To check the nodes in a cluster:
	`# cd /opt/VRTS/install`
	`# ./installsfrac -configure galaxy nebula`
	For more information:
	See "Verifying the cluster" on page 87.

Table 2-1 Options for verifying the nodes in a cluster *(continued)*

Type of check	Description
SF Oracle RAC health checks	SF Oracle RAC provides a health check utility that examines the functional health of the components in an SF Oracle RAC cluster. The utility when invoked gathers real-time operational information on cluster components and displays the report on your system console. You must run the utility on each node in the cluster. To schedule periodic health checks: See "Periodic health evaluation of the clusters" on page 70.
Using VRTSexplorer	VRTSexplorer, also known as Veritas Explorer, is a tool provided by Symantec to gather system and configuration information from a node to diagnose or analyze issues in the cluster. The utility is located at `/opt/VRTSspt/VRTSexplorer`. For more information: See "Running VRTSexplorer to diagnose issues in the cluster" on page 89.

Verifying the cluster

The installer performs the following checks on each node during the process:

- Verifies that SF Oracle RAC is installed.
- Verifies whether the Oracle process daemon (oprocd) is running or not.
- Verifies that the Veritas libraries (ODM and VCSMM) are linked with the Oracle RAC libraries.
- Verifies whether the tunables for LLT/VCSMM tunables are set correctly.
- Verifies that the CSSD resource is configured under the CVM service group.
- Verifies whether the specified Oracle user has read permissions on the /etc/llttab and /etc/llthosts files.
- Verifies that LLT and Oracle Clusterware/Grid Infrastructure have the same view of the number of nodes in the cluster.
- Verifies that the Oracle RAC database version is the same on all nodes in the cluster. The check fails if the version or patchset information varies across nodes.

For information on resolving issues that you may encounter during the checks:

See "Troubleshooting installation and configuration check failures" on page 172.

The installer skips the following checks when LLT is configured over UDP:

- LLT links' full duplex setting
- LLT link jumbo frame setting (MTU)
- LLT links' cross connection
- LLT links' speed and auto negotiation settings

Run the SF Oracle RAC installer to verify the functional health of the cluster.

To verify the cluster

1 Start the SF Oracle RAC installer:

```
# cd /opt/VRTS/install

# ./installsfrac -configure galaxy nebula
```

2 Select the option **SF Oracle RAC Installation and Configuration Checks**.

```
1)  Configure SF Oracle RAC sub-components
2)  SF Oracle RAC Installation and Configuration Checks
3)  Prepare to Install Oracle
4)  Install Oracle Clusterware/Grid Infrastructure and Database
5)  Post Oracle Installation Tasks
6)  Exit SF Oracle RAC Configuration
Choose option: [1-6,q] (1) 2
```

3 Enter **y** to confirm that Oracle RAC is installed on the nodes.

4 Enter the full path of the Oracle Clusterware/Grid Infrastructure home directory.

Note: The installer tries to discover the location of the Oracle Clusterware/Grid Infrastructure home directory from the Oracle inventory. If the installer discovers the information, you are not prompted for this information.

The installer verifies the path information and re-prompts if the path is incorrect.

5 Enter the full path of the Oracle RAC database home directory.

Note: The installer tries to discover the location of the Oracle RAC database home directory from the Oracle inventory. If the installer discovers the information, you are not prompted for this information.

The installer verifies the path information.

6 Enter the Oracle user name.

The installer displays the location of the temporary log file and starts the verification process.

Running VRTSexplorer to diagnose issues in the cluster

Perform this step if you encounter issues in your cluster environment that require professional support.

VRTSexplorer, also known as Veritas Explorer, is a tool provided by Symantec to gather system and configuration information from a node to diagnose or analyze issues in the cluster. The utility is located at `/opt/VRTSspt/VRTSexplorer`.

Note: The utility collects only technical information such as the server configuration and installed Veritas products information required to troubleshoot issues. No user data is collected by the utility. For more details on the information that is collected by the utility, refer to the README located in the `/opt/VRTSspt/VRTSexplorer` directory.

To run the utility:

```
# cd /opt/VRTSspt/VRTSexplorer
# ./VRTSexplorer
```

For information on various command options, see `VRTSexplorer` (1M).

If you find issues in the cluster that require professional help, run the utility and send the compressed tar file output to Symantec Technical Support to resolve the issue.

Administering VCS

This section provides instructions for the following VCS administration tasks:

- Viewing available Veritas devices and drivers
  See "Viewing available Veritas device drivers" on page 90.

- Verifying VCS configuration
  See "Verifying VCS configuration" on page 91.
- Starting and stopping VCS
  See "Starting and stopping VCS" on page 91.
- Environment variables to start and stop VCS modules
  See "Environment variables to start and stop VCS modules" on page 91.
- Adding and removing LLT links
  See "Adding and removing LLT links" on page 93.
- Displaying the cluster details and LLT version for LLT links
  See "Displaying the cluster details and LLT version for LLT links" on page 99.
- Configuring aggregated interfaces under LLT
  See "Configuring aggregated interfaces under LLT" on page 97.
- Configuring destination-based load balancing for LLT
  See "Configuring destination-based load balancing for LLT" on page 100.
- Enabling and disabling intelligent resource monitoring
  See "Enabling and disabling intelligent resource monitoring" on page 100.
- Administering the AMF kernel driver
  See "Administering the AMF kernel driver" on page 102.

If you encounter issues while administering VCS, see the troubleshooting section in the *Veritas Cluster Server Administrator's Guide* for assistance.

Viewing available Veritas device drivers

To view the drivers that are loaded in memory, run the `lsmod` command as shown in the following examples.

For example:

If you want to view whether or not the driver 'gab' is loaded in memory:

```
# lsmod |grep "^gab"
gab      246420  11 vxfen,vcsmm
```

If you want to view whether or not the 'vx' drivers are loaded in memory:

```
# lsmod |grep "^vx"
vxfen                  260888  1
vxodm                  159820  1
vxglm                  267400  2
vxgms                  285692  2
vxspec                   4360  4
```

```
vxio                 3090616  3 vxspec
vxdmp                 341064  92 vxspec,vxio
vxportal                7820  2
vxfs                 2326872  11 vxportal,fdd
```

Verifying VCS configuration

To verify the VCS configuration:

```
# cd /etc/VRTSvcs/conf/config
# hacf -verify .
```

Starting and stopping VCS

To start VCS on each node:

```
# hastart
```

To stop VCS on each node:

```
# hastop -local
```

You can also use the command `hastop -all`; however, make sure that you wait for port 'h' to close before restarting VCS.

Environment variables to start and stop VCS modules

The start and stop environment variables for AMF, LLT, GAB, VxFEN, and VCS engine define the default VCS behavior to start these modules during system restart or stop these modules during system shutdown.

Note: The startup and shutdown of AMF, LLT, GAB, VxFEN, and VCS engine are inter-dependent. For a clean startup or shutdown of VCS, you must either enable or disable the startup and shutdown modes for all these modules.

Table 2-2 describes the start and stop variables for VCS.

Table 2-2 Start and stop environment variables for VCS

Environment variable	Definition and default value
AMF_START	Startup mode for the AMF driver. By default, the AMF driver is disabled to start up after a system reboot. This environment variable is defined in the following file: `/etc/sysconfig/amf` Default: 0
AMF_STOP	Shutdown mode for the AMF driver. By default, the AMF driver is enabled to stop during a system shutdown. This environment variable is defined in the following file: `/etc/sysconfig/amf` Default: 1
LLT_START	Startup mode for LLT. By default, LLT is enabled to start up after a system reboot. This environment variable is defined in the following file: `/etc/sysconfig/llt` Default: 1
LLT_STOP	Shutdown mode for LLT. By default, LLT is enabled to stop during a system shutdown. This environment variable is defined in the following file: `/etc/sysconfig/llt` Default: 1
GAB_START	Startup mode for GAB. By default, GAB is enabled to start up after a system reboot. This environment variable is defined in the following file: `/etc/sysconfig/gab` Default: 1
GAB_STOP	Shutdown mode for GAB. By default, GAB is enabled to stop during a system shutdown. This environment variable is defined in the following file: `/etc/sysconfig/gab` Default: 1

Table 2-2 Start and stop environment variables for VCS *(continued)*

Environment variable	Definition and default value
VXFEN_START	Startup mode for VxFEN. By default, VxFEN is enabled to start up after a system reboot. This environment variable is defined in the following file: `/etc/sysconfig/vxfen` Default: 1
VXFEN_STOP	Shutdown mode for VxFEN. By default, VxFEN is enabled to stop during a system shutdown. This environment variable is defined in the following file: `/etc/sysconfig/vxfen` Default: 1
VCS_START	Startup mode for VCS engine. By default, VCS engine is enabled to start up after a system reboot. This environment variable is defined in the following file: `/etc/sysconfig/vcs` Default: 1
VCS_STOP	Shutdown mode for VCS engine. By default, VCS engine is enabled to stop during a system shutdown. This environment variable is defined in the following file: `/etc/sysconfig/vcs` Default: 1

Adding and removing LLT links

In an SF Oracle RAC cluster, Oracle Clusterware heartbeat link must be configured as an LLT link. If Oracle Clusterware and LLT use different links for their communication, then the membership change between VCS and Oracle Clusterware is not coordinated correctly. For example, if only the Oracle Clusterware links are down, Oracle Clusterware kills one set of nodes after the expiry of the css-misscount interval and initiates the Oracle Clusterware and database recovery, even before CVM and CFS detect the node failures. This uncoordinated recovery may cause data corruption.

If you need additional capacity for Oracle communication on your private interconnects, you can add LLT links. The network IDs of the interfaces connected to the same physical network must match. The interfaces specified in the PrivNIC or MultiPrivNIC configuration must be exactly the same in name and total number as those which have been used for LLT configuration.

You can use the `lltconfig` command to add or remove LLT links when LLT is running.

LLT links can be added or removed while clients are connected.

See the `lltconfig`(1M) manual page for more details.

Note: When you add or remove LLT links, you need not shut down GAB or the high availability daemon, `had`. Your changes take effect immediately, but are lost on the next restart. For changes to persist, you must also update the `/etc/lltab` file.

To add LLT links

1 Depending on the LLT link type, run the following command to add an LLT link:

- For ether link type:

```
# lltconfig -t devtag -d device
[-b ether ] [-s SAP] [-m mtu]
```

- For UDP link type:

```
# lltconfig -t devtag -d device
-b udp [-s port] [-m mtu]
-I IPaddr -B bcast
```

- For UDP6 link type:

```
# lltconfig -t devtag -d device
-b udp6 [-s port] [-m mtu]
-I IPaddr -B mcast
```

Where:

devtag Tag to identify the link

device	Device name of the interface. For link type ether, you can specify the device name as an interface name. For example, eth0. Preferably, specify the device name as eth-macaddress. For example, eth- xx:xx:xx:xx:xx:xx. For link types udp and udp6, the device is the udp and udp6 device name respectively.
bcast	Broadcast address for the link type udp
mcast	Multicast address for the link type udp6
IPaddr	IP address for link types udp and udp6
SAP	SAP to bind on the network links for link type ether
port	Port for link types udp and udp6
mtu	Maximum transmission unit to send packets on network links

For example:

- For ether link type:

```
# lltconfig -t eth4 -d eth4 -s 0xcafe -m 1500
```

- For UDP link type:

```
# lltconfig -t link1 -d udp -b udp
-I 192.1.2.255 -B 192.1.2.255
```

- For UDP6 link type:

```
# lltconfig -t link1 -d udp6 -b udp6 -I 2000::1
```

2 If you want to configure the link under PrivNIC or MultiPrivNIC as a failover target in the case of link failure, modify the PrivNIC or MultiPrivNIC configuration as follows:

```
# haconf -makerw
# hares -modify resource_name Device device
    device_id [-sys hostname]
# haconf -dump -makero
```

The following is an example of configuring the link under PrivNIC.

Assuming that you have two LLT links configured under PrivNIC as follows:

```
PrivNIC ora_priv (
        Critical = 0
        Device@galaxy = { eth1 = 0, eth2 = 1 }
        Device@nebula = { eth1 = 0, eth2 = 1 }
        Address@galaxy = "192.168.12.1"
        Address@nebula = "192.168.12.2"
        NetMask = "255.255.255.0"
        )
```

To configure the new LLT link under PrivNIC, run the following commands:

```
# haconf -makerw
# hares -modify ora_priv Device eth1 0 eth2 1 eth3 2 -sys galaxy
# hares -modify ora_priv Device eth1 0 eth2 1 eth3 2 -sys nebula
# haconf -dump -makero
```

The updated PrivNIC resource now resembles:

```
PrivNIC ora_priv (
        Critical = 0
        Device@galaxy = { eth1 = 0, eth2 = 1, eth3 = 2 }
        Device@nebula = { eth1 = 0, eth2 = 1, eth3 = 2 }
        Address@galaxy = "192.168.12.1"
        Address@nebula = "192.168.12.2"
        NetMask = "255.255.255.0"
        )
```

To remove an LLT link

1 Run the following command to remove a network link that is configured under LLT:

```
# lltconfig -u devtag
```

2 If the link you removed is configured as a PrivNIC or MultiPrivNIC resource, you also need to modify the resource configuration after removing the link.

If you have configured the link under PrivNIC or MultiPrivNIC as a failover target in the case of link failure, modify the PrivNIC or MultiPrivNIC configuration as follows:

```
# haconf -makerw
# hares -modify resource_name Device link_name \
device_id [-sys hostname]
# haconf -dump -makero
```

For example, if the links eth1, eth2, and eth3 were configured as PrivNIC resources, and you want to remove eth3:

```
# haconf -makerw
# hares -modify ora_priv Device eth1
   0 \
eth2
   1
```

where eth1 and eth2 are the links that you want to retain in your cluster.

```
# haconf -dump -makero
```

Configuring aggregated interfaces under LLT

If you want to configure LLT to use aggregated interfaces after installing and configuring VCS, you can use one of the following approaches:

- Edit the /etc/llttab file
  This approach requires you to stop LLT. The aggregated interface configuration is persistent across reboots.
- Run the `lltconfig` command
  This approach lets you configure aggregated interfaces on the fly. However, the changes are not persistent across reboots.

To configure aggregated interfaces under LLT by editing the /etc/llttab file

1. If LLT is running, run the following command to stop LLT:
2. Add the following entry to the /etc/llttab file to configure an aggregated interface.

   link *tag device_name systemid_range link_type sap mtu_size*

tag	Tag to identify the link
device_name	Device name of the aggregated interface.
systemid_range	Range of systems for which the command is valid. If the link command is valid for all systems, specify a dash (-).
link_type	The link type must be ether.
sap	SAP to bind on the network links. Default is 0xcafe.
mtu_size	Maximum transmission unit to send packets on network links

3. Restart LLT for the changes to take effect.

To configure aggregated interfaces under LLT using the lltconfig command

- When LLT is running, use the following command to configure an aggregated interface:

  ```
  lltconfig -t devtag -d device
  [-b linktype ] [-s SAP] [-m mtu]
  ```

devtag	Tag to identify the link
device	Device name of the aggregated interface.
link_type	The link type must be ether.
sap	SAP to bind on the network links. Default is 0xcafe.
mtu_size	Maximum transmission unit to send packets on network links

  See the `lltconfig`(1M) manual page for more details.

  You need not reboot after you make this change. However, to make these changes persistent across reboot, you must update the /etc/lltab file.

  See "To configure aggregated interfaces under LLT by editing the /etc/lltab file" on page 98.

Displaying the cluster details and LLT version for LLT links

You can use the `lltdump` command to display the LLT version for a specific LLT link. You can also display the cluster ID and node ID details.

See the `lltdump`(1M) manual page for more details.

To display the cluster details and LLT version for LLT links

- Run the following command to display the details:

  ```
  # /opt/VRTSllt/lltdump -D -f link
  ```

  For example, if eth2 is connected to galaxy, then the command displays a list of all cluster IDs and node IDs present on the network link eth2.

  ```
  # /opt/VRTSllt/lltdump -D -f eth2

  lltdump : Configuration:

  device : eth2

  sap : 0xcafe
  promisc sap : 0
  promisc mac : 0
  cidsnoop : 1
  === Listening for LLT packets ===
  cid nid vmaj vmin
  3456 1 5 0
  3456 3 5 0
  83 0 4 0
  27 1 3 7
  3456 2 5 0
  ```

Configuring destination-based load balancing for LLT

Destination-based load balancing for LLT is turned off by default. Symantec recommends destination-based load balancing when the cluster setup has more than two nodes and more active LLT ports.

See "About Low Latency Transport (LLT)" on page 25.

To configure destination-based load balancing for LLT

- Run the following command to configure destination-based load balancing:

  ```
  lltconfig -F linkburst: 0
  ```

Enabling and disabling intelligent resource monitoring

Review the following procedures to enable or disable intelligent resource monitoring. The intelligent resource monitoring feature is disabled by default.

The IMF resource type attribute determines whether an IMF-aware agent must perform intelligent resource monitoring.

See "About resource monitoring" on page 36.

To enable intelligent resource monitoring

1 Make the VCS configuration writable.

```
# haconf -makerw
```

2 Run the following command to enable intelligent resource monitoring.

- To enable intelligent monitoring of offline resources:

```
hatype -modify resource_type IMF -update Mode 1
```

- To enable intelligent monitoring of online resources:

```
hatype -modify resource_type IMF -update Mode 2
```

- To enable intelligent monitoring of both online and offline resources:

```
hatype -modify resource_type IMF -update Mode 3
```

3 Change the values of the MonitorFreq key and the RegisterRetryLimit key of the IMF attribute.

See the *Veritas Cluster Server Bundled Agents Reference Guide* for agent-specific recommendations to set these attributes.

Review the agent-specific recommendations in the attribute definition tables to set these attribute key values.

4 Save the VCS configuration.

```
# haconf -dump -makero
```

5 Make sure that the AMF kernel driver is configured on all nodes in the cluster.

```
/etc/init.d/amf status
```

Configure the AMF driver if the command output returns that the AMF driver is not loaded or not configured.

See "Administering the AMF kernel driver" on page 102.

6 Restart the agent. Run the following commands on each node.

```
haagent -stop agent_name -force -sys sys_name
haagent -start agent_name -sys sys_name
```

To disable intelligent resource monitoring

1 Make the VCS configuration writable.

```
# haconf -makerw
```

2 To disable intelligent resource monitoring for all the resources of a certain type, run the following command:

```
hatype -modify resource_type IMF -update Mode 0
```

3 To disable intelligent resource monitoring for a specific resource, run the following command:

```
hares -override resource_name IMF
hares -modify resource_name IMF -update Mode 0
```

4 Save the VCS configuration.

```
# haconf -dump -makero
```

Administering the AMF kernel driver

Review the following procedures to start or stop the AMF kernel driver.

See "About the IMF notification module" on page 36.

See "Environment variables to start and stop VCS modules" on page 91.

To start the AMF kernel driver

1 Set the value of the AMF_START variable to 1 in the following file:

```
/etc/sysconfig/amf
```

2 Start the AMF kernel driver. Run the following command:

```
/etc/init.d/amf start
```

To stop the AMF kernel driver

1 Stop the AMF kernel driver. Run the following command:

```
/etc/init.d/amf stop
```

2 Set the value of the AMF_START variable to 0 in the following file:

```
/etc/sysconfig/amf
```

Administering I/O fencing

This section describes I/O fencing and provides instructions for common I/O fencing administration tasks.

- About administering I/O fencing
  See "About administering I/O fencing" on page 104.
- About vxfentsthdw utility
  See "About the vxfentsthdw utility" on page 104.
- About vxfenadm utility
  See "About the vxfenadm utility" on page 112.
- About vxfenclearpre utility
  See "About the vxfenclearpre utility" on page 117.
- About vxfenswap utility
  See "About the vxfenswap utility" on page 120.

If you encounter issues while administering I/O fencing, refer to the troubleshooting section for assistance.

See "Troubleshooting I/O fencing" on page 196.

About administering I/O fencing

The I/O fencing feature provides the following utilities that are available through the VRTSvxfen RPM:

vxfentsthdw	Tests hardware for I/O fencing See "About the vxfentsthdw utility" on page 104.
vxfenconfig	Configures and unconfigures I/O fencing Checks the list of coordinator disks used by the vxfen driver.
vxfenadm	Displays information on I/O fencing operations and manages SCSI-3 disk registrations and reservations for I/O fencing See "About the vxfenadm utility" on page 112.
vxfenclearpre	Removes SCSI-3 registrations and reservations from disks See "About the vxfenclearpre utility" on page 117.
vxfenswap	Replaces coordinator points without stopping I/O fencing See "About the vxfenswap utility" on page 120.
vxfendisk	Generates the list of paths of disks in the diskgroup. This utility requires that Veritas Volume Manager is installed and configured.

The I/O fencing commands reside in the /opt/VRTS/bin folder. Make sure you added this folder path to the PATH environment variable.

Refer to the corresponding manual page for more information on the commands.

About the vxfentsthdw utility

You can use the vxfentsthdw utility to verify that shared storage arrays to be used for data support SCSI-3 persistent reservations and I/O fencing. During the I/O fencing configuration, the testing utility is used to test a single disk. The utility has other options that may be more suitable for testing storage devices in other configurations. You also need to test coordinator disk groups.

See *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide* to set up I/O fencing.

The utility, which you can run from one system in the cluster, tests the storage used for data by setting and verifying SCSI-3 registrations on the disk or disks you specify, setting and verifying persistent reservations on the disks, writing data to the disks and reading it, and removing the registrations from the disks.

Refer also to the vxfentsthdw(1M) manual page.

About general guidelines for using vxfentsthdw utility

Review the following guidelines to use the vxfentsthdw utility:

- The utility requires two systems connected to the shared storage.

  Caution: The tests overwrite and destroy data on the disks, unless you use the -r option.

- The two nodes must have ssh (default) or rsh communication. If you use rsh, launch the vxfentsthdw utility with the -n option.

  After completing the testing process, you can remove permissions for communication and restore public network connections.

- To ensure both systems are connected to the same disk during the testing, you can use the vxfenadm -i *diskpath* command to verify a disk's serial number.

  See "Verifying that the nodes see the same disk" on page 116.

- For disk arrays with many disks, use the -m option to sample a few disks before creating a disk group and using the -g option to test them all.

- The utility indicates a disk can be used for I/O fencing with a message resembling:

  ```
  The disk /dev/sdx is ready to be configured for
  I/O Fencing on node galaxy
  ```

  If the utility does not show a message stating a disk is ready, verification has failed.

- If the disk you intend to test has existing SCSI-3 registration keys, the test issues a warning before proceeding.

About the vxfentsthdw command options

Table 2-3 describes the methods that the utility provides to test storage devices.

Table 2-3 vxfentsthdw options

vxfentsthdw option	Description	When to use
-n	Utility uses rsh for communication.	Use when rsh is used for communication.

Table 2-3 vxfentsthdw options *(continued)*

vxfentsthdw option	Description	When to use
-r	Non-destructive testing. Testing of the disks for SCSI-3 persistent reservations occurs in a non-destructive way; that is, there is only testing for reads, not writes. May be used with -m, -f, or -g options.	Use during non-destructive testing. See "Performing non-destructive testing on the disks using the -r option" on page 108.
-t	Testing of the return value of SCSI TEST UNIT (TUR) command under SCSI-3 reservations. A warning is printed on failure of TUR testing.	When you want to perform TUR testing.
-d	Use DMP devices. May be used with -c or -g options.	By default, the script picks up the DMP paths for disks in the disk group. If you want the script to use the raw paths for disks in the disk group, use the -w option.
-w	Use raw devices. May be used with -c or -g options.	With the -w option, the script picks the operating system paths for disks in the disk group. By default, the script uses the -d option to pick up the DMP paths for disks in the disk group.
-c	Utility tests the coordinator disk group prompting for systems and devices, and reporting success or failure.	For testing disks in coordinator disk group. See "Testing the coordinator disk group using vxfentsthdw -c option" on page 107.
-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. See "Testing the shared disks using the vxfentsthdw -m option" on page 109.

Table 2-3 vxfentsthdw options *(continued)*

vxfentsthdw option	Description	When to use
-f *filename*	Utility tests system/device combinations listed in a text file. May be used with -r and -t options.	For testing several disks. See "Testing the shared disks listed in a file using the vxfentsthdw -f option" on page 111.
-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. See "Testing all the disks in a disk group using the vxfentsthdw -g option" on page 111.

Testing the coordinator disk group using vxfentsthdw -c option

Use the vxfentsthdw utility to verify disks are configured to support I/O fencing. In this procedure, the vxfentsthdw utility tests the three disks one disk at a time from each node.

The procedure in this section uses the following disks for example:

- From the node galaxy, the disks are /dev/sdg, /dev/sdh, and /dev/sdi.
- From the node nebula, the disks are /dev/sdx, /dev/sdy, and /dev/sdz.

Note: To test the coordinator disk group using the vxfentsthdw utility, the utility requires that the coordinator disk group, vxfencoorddg, be accessible from two nodes.

To test the coordinator disk group using vxfentsthdw -c

1 Use the vxfentsthdw command with the -c option. For example:

```
# vxfentsthdw -c vxfencoorddg
```

2 Enter the nodes you are using to test the coordinator disks:

```
Enter the first node of the cluster: galaxy
Enter the second node of the cluster: nebula
```

3 Review the output of the testing process for both nodes for all disks in the coordinator disk group. Each disk should display output that resembles:

```
ALL tests on the disk /dev/sdg have PASSED.
The disk is now ready to be configured for I/O Fencing on node
galaxy as a COORDINATOR DISK.

ALL tests on the disk /dev/sdx have PASSED.
The disk is now ready to be configured for I/O Fencing on node
nebula as a COORDINATOR DISK.
```

4 After you test all disks in the disk group, the vxfencoorddg disk group is ready for use.

Removing and replacing a failed disk

If a disk in the coordinator disk group fails verification, remove the failed disk or LUN from the vxfencoorddg disk group, replace it with another, and retest the disk group.

To remove and replace a failed disk

1 Use the vxdiskadm utility to remove the failed disk from the disk group.

Refer to the *Veritas Volume Manager Administrator's Guide*.

2 Add a new disk to the node, initialize it, and add it to the coordinator disk group.

See the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide* for instructions to initialize disks for I/O fencing and to set up coordinator disk groups.

If necessary, start the disk group.

See the *Veritas Volume Manager Administrator's Guide* for instructions to start the disk group.

3 Retest the disk group.

See "Testing the coordinator disk group using vxfentsthdw -c option" on page 107.

Performing non-destructive testing on the disks using the -r option

You can perform non-destructive testing on the disk devices when you want to preserve the data.

To perform non-destructive testing on disks

- To test disk devices containing data you want to preserve, you can use the -r option with the -m, -f, or -g options.

  For example, to use the -m option and the -r option, you can run the utility as follows:

  ```
  # 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.

Testing the shared disks using the vxfentsthdw -m option

Review the procedure to test the shared disks. By default, the utility uses the -m option.

This procedure uses the /dev/sdx disk in the steps.

If the utility does not show a message stating a disk is ready, verification has failed. Failure of verification can be the result of an improperly configured disk array. It can also be caused by a bad disk.

If the failure is due to a bad disk, remove and replace it. The vxfentsthdw utility indicates a disk can be used for I/O fencing with a message resembling:

```
The disk /dev/sdx is ready to be configured for
I/O Fencing on node galaxy
```

Note: For A/P arrays, run the `vxfentsthdw` command only on active enabled paths.

To test disks using vxfentsthdw script

1. Make sure system-to-system communication is functioning properly.
2. From one node, start the utility.

   ```
   # vxfentsthdw [-n]
   ```

3 After reviewing the overview and warning that the tests overwrite data on the disks, confirm to continue the process and enter the node names.

```
******** 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: galaxy
Enter the second node of the cluster: nebula
```

4 Enter the names of the disks you are checking. For each node, the disk may be known by the same name:

```
Enter the disk name to be checked for SCSI-3 PGR on node
galaxy in the format:
        for dmp: /dev/vx/rdmp/sdx
        for raw: /dev/sdx
Make sure it's the same disk as seen by nodes galaxy and nebula
/dev/sdr

Enter the disk name to be checked for SCSI-3 PGR on node
nebula in the format:
        for dmp: /dev/vx/rdmp/sdx
        for raw: /dev/sdx
Make sure it's the same disk as seen by nodes galaxy and nebula
/dev/sdr
```

If the serial numbers of the disks are not identical, then the test terminates.

5 Review the output as the utility performs the checks and report its activities.

6 If a disk is ready for I/O fencing on each node, the utility reports success:

```
ALL tests on the disk /dev/sdx have PASSED
The disk is now ready to be configured for I/O Fencing on node
galaxy
...
Removing test keys and temporary files, if any ...
.
.
```

7 Run the vxfentsthdw utility for each disk you intend to verify.

Testing the shared disks listed in a file using the vxfentsthdw -f option

Use the -f option to test disks that are listed in a text file. Review the following example procedure.

To test the shared disks listed in a file

1 Create a text file disks_test to test two disks shared by systems galaxy and nebula that might resemble:

```
galaxy /dev/sdz nebula /dev/sdy
galaxy /dev/sdu nebula /dev/sdw
```

Where the first disk is listed in the first line and is seen by galaxy as /dev/sdz and by nebula as /dev/sdy. The other disk, in the second line, is seen as /dev/sdu from galaxy and /dev/sdw from nebula. Typically, the list of disks could be extensive.

2 To test the disks, enter the following command:

```
# vxfentsthdw -f disks_test
```

The utility reports the test results one disk at a time, just as for the -m option.

Testing all the disks in a disk group using the vxfentsthdw -g option

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.

After testing, destroy the disk group and put the disks into disk groups as you need.

To test all the disks in a diskgroup

1 Create a diskgroup for the disks that you want to test.

2 Enter the following command to test the diskgroup test_disks_dg:

```
# vxfentsthdw -g test_disks_dg
```

The utility reports the test results one disk at a time.

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.

About the vxfenadm utility

Administrators can use the vxfenadm command to troubleshoot and test fencing configurations.

The command's options for use by administrators are as follows:

-s	read the keys on a disk and display the keys in numeric, character, and node format **Note:** The -g and -G options are deprecated. Use the -s option.
-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

Refer to the `vxfenadm`(1m) manual page for a complete list of the command options.

About the I/O fencing registration key format

The keys that the vxfen driver registers on the data disks and the coordinator disks consist of eight bytes. The key format is different for the coordinator disks and data disks.

The key format of the coordinator disks is as follows:

Byte	0	1	2	3	4	5	6	7
Value	V	F	cID 0x	cID 0x	cID 0x	cID 0x	nID 0x	nID 0x

where:

- VF is the unique identifier that carves out a namespace for the keys (consumes two bytes)
- cID 0x is the LLT cluster ID in hexadecimal (consumes four bytes)
- nID 0x is the LLT node ID in hexadecimal (consumes two bytes)

The vxfen driver uses this key format in both scsi3 mode and customized mode of I/O fencing.

The key format of the data disks that are configured as failover disk groups under VCS is as follows:

Byte	0	1	2	3	4	5	6	7
Value	A+nID	V	C	S				

where nID is the LLT node ID

For example: If the node ID is 1, then the first byte has the value as B ('A' + 1 = B).

The key format of the data disks configured as parallel disk groups under CVM is as follows:

Byte	0	1	2	3	4	5	6	7
Value	A+nID	P	G	R	DGcount	DGcount	DGcount	DGcount

where DGcount is the count of disk group in the configuration

Displaying the I/O fencing registration keys

You can display the keys that are currently assigned to the disks using the vxfenadm command.

The variables such as *disk_7*, *disk_8*, and *disk_9* in the following procedure represent the disk names in your setup.

To display the I/O fencing registration keys

1 To display the key for the disks, run the following command:

```
# vxfenadm -s disk_name
```

For example:

- To display the key for the coordinator disk /dev/sdx from the system with node ID 1, enter the following command:

```
# vxfenadm -s /dev/sdx
key[1]:
  [Numeric Format]:  86,70,68,69,69,68,48,48
  [Character Format]: VFDEED00
* [Node Format]: Cluster ID: 57069 Node ID: 0 Node Name: galaxy
```

The -s option of vxfenadm displays all eight bytes of a key value in three formats. In the numeric format,

  - The first two bytes, represent the identifier VF, contains the ASCII value 86, 70.
  - The next four bytes contain the ASCII value of the cluster ID 57069 encoded in hex (0xDEED) which are 68, 69, 69, 68.
  - The remaining bytes contain the ASCII value of the node ID 0 (0x00) which are 48, 48. Node ID 1 would be 01 and node ID 10 would be 0A.

An asterisk before the Node Format indicates that the vxfenadm command is run from the node of a cluster where LLT is configured and running.

- To display the keys on a CVM parallel disk group:

```
# vxfenadm -s /dev/vx/rdmp/disk_7

Reading SCSI Registration Keys...

Device Name: /dev/vx/rdmp/disk_7
Total Number Of Keys: 1
key[0]:
  [Numeric Format]:  66,80,71,82,48,48,48,48
  [Character Format]: BPGR0001
  [Node Format]: Cluster ID: unknown  Node ID: 1 Node Name: nebula
```

- To display the keys on a VCS failover disk group:

```
# vxfenadm -s /dev/vx/rdmp/disk_8

Reading SCSI Registration Keys...

Device Name: /dev/vx/rdmp/disk_8
Total Number Of Keys: 1
key[0]:
  [Numeric Format]:  65,86,67,83,0,0,0,0
  [Character Format]: AVCS
  [Node Format]: Cluster ID: unknown  Node ID: 0 Node Name: galaxy
```

2 To display the keys that are registered in all the disks specified in a disk file:

```
# vxfenadm -s all -f disk_filename
```

For example:

To display all the keys on coordinator disks:

```
# vxfenadm -s all -f /etc/vxfentab

Device Name: /dev/vx/rdmp/disk_9
Total Number Of Keys: 2
key[0]:
  [Numeric Format]:  86,70,66,69,65,68,48,50
  [Character Format]: VFBEAD02
  [Node Format]: Cluster ID: 48813 Node ID: 2  Node Name: unknown
key[1]:
  [Numeric Format]:  86,70,68,69,69,68,48,48
  [Character Format]: VFDEED00
* [Node Format]: Cluster ID: 57069 Node ID: 0  Node Name: galaxy
```

You can verify the cluster ID using the lltstat -C command, and the node ID using the lltstat -N command. For example:

```
# lltstat -C
57069
```

If the disk has keys which do not belong to a specific cluster, then the vxfenadm command cannot look up the node name for the node ID and hence prints the node name as unknown. For example:

```
Device Name: /dev/vx/rdmp/disk_7
Total Number Of Keys: 1
key[0]:
  [Numeric Format]:  86,70,45,45,45,45,48,49
```

```
[Character Format]: VF----01
[Node Format]: Cluster ID: unknown  Node ID: 1  Node Name: nebula
```

For disks with arbitrary format of keys, the vxfenadm command prints all the fields as unknown. For example:

```
[Numeric Format]:  65,66,67,68,49,50,51,45
[Character Format]: ABCD123-
[Node Format]: Cluster ID: unknown Node ID: unknown
       Node Name: unknown
```

Verifying that the nodes see the same disk

To confirm whether a disk (or LUN) supports SCSI-3 persistent reservations, two nodes must simultaneously have access to the same disks. Because a shared disk is likely to have a different name on each node, check the serial number to verify the identity of the disk. Use the vxfenadm command with the -i option to verify that the same serial number for the LUN is returned on all paths to the LUN.

For example, an EMC disk is accessible by the /dev/sdr path on node A and the /dev/sdt path on node B.

To verify that the nodes see the same disks

1 Verify the connection of the shared storage for data to two of the nodes on which you installed SF Oracle RAC.

2 From node A, enter the following command:

```
# vxfenadm -i /dev/sdr

Vendor id       : EMC
Product id      : SYMMETRIX
Revision        : 5567
Serial Number   : 42031000a
```

The same serial number information should appear when you enter the equivalent command on node B using the /dev/sdt path.

On a disk from another manufacturer, Hitachi Data Systems, the output is different and may resemble:

```
# vxfenadm -i /dev/sdt

Vendor id       : HITACHI
Product id      : OPEN-3
Revision        : 0117
Serial Number   : 0401EB6F0002
```

Refer to the `vxfenadm`(1M) manual page for more information.

About the vxfenclearpre utility

You can use the vxfenclearpre utility to remove SCSI-3 registrations and reservations on the disks.

See "Removing preexisting keys" on page 117.

This utility currently does not support server-based fencing. You must manually resolve any preexisting split-brain with server-based fencing configuration.

See "Issues during fencing startup on SF Oracle RAC cluster nodes set up for server-based fencing" on page 207.

Removing preexisting keys

If you encountered a split-brain condition, use the vxfenclearpre utility to remove SCSI-3 registrations and reservations on the coordinator disks as well as on the data disks in all shared disk groups.

You can also use this procedure to remove the registration and reservation keys created by another node from a disk.

To clear keys after split-brain

1. Stop VCS on all nodes.

   ```
   # hastop -all
   ```

2. Make sure that the port h is closed on all the nodes. Run the following command on each node to verify that the port h is closed:

   ```
   # gabconfig -a
   ```

   Port h must not appear in the output.

3. Stop I/O fencing on all nodes. Enter the following command on each node:

   ```
   # /etc/init.d/vxfen stop
   ```

4. If you have any applications that run outside of VCS control that have access to the shared storage, then shut down all other nodes in the cluster that have access to the shared storage. This prevents data corruption.

5. Start the vxfenclearpre script:

   ```
   # vxfenclearpre
   ```

6 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
```

In some cases, 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. You can ignore these informational messages.

```
<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
```

The script cleans up the disks and displays the following status messages.

```
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.

You can retry starting fencing module. In order to
restart the whole product, you might want to
reboot the system.
```

7 Start the fencing module.

```
# /etc/init.d/vxfen start
```

8 Start VCS on all nodes.

```
# hastart
```

About the vxfenswap utility

The vxfenswap utility allows you to replace coordinator disks in a cluster that is online. The utility verifies that the serial number of the new disks are identical on all the nodes and the new disks can support I/O fencing.

This utility also supports server-based fencing.

Refer to the `vxfenswap`(1M) manual page.

See the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide* for details on the I/O fencing requirements.

You can replace the coordinator disks without stopping I/O fencing in the following cases:

- The disk becomes defective or inoperable and you want to switch to a new diskgroup.
  See "Replacing I/O fencing coordinator disks when the cluster is online" on page 121.
  See "Replacing the coordinator diskgroup in a cluster that is online" on page 123.
  If you want to replace the coordinator disks when the cluster is offline, you cannot use the vxfenswap utility. You must manually perform the steps that the utility does to replace the coordinator disks.
  See "Replacing defective disks when the cluster is offline" on page 200.
- You want to switch the disk interface between raw devices and DMP devices.
  See "Changing the disk interaction policy in a cluster that is online" on page 126.
- The keys that are registered on the coordinator disks are lost.
  In such a case, the cluster might panic when a network partition occurs. You can replace the coordinator disks with the same disks using the vxfenswap command. During the disk replacement, the missing keys register again without any risk of data corruption.
  See "Refreshing lost keys on coordinator disks" on page 129.

In server-based fencing configuration, you can use the vxfenswap utility to perform the following tasks:

- Perform a planned replacement of customized coordination points (CP servers or SCSI-3 disks).
  See "Replacing coordination points for server-based fencing in an online cluster" on page 143.
- Refresh the I/O fencing keys that are registered on the coordination points.
  See "Refreshing registration keys on the coordination points for server-based fencing" on page 141.

You can also use the vxfenswap utility to migrate between the disk-based and the server-based fencing without incurring application downtime in the SF Oracle RAC cluster.

See "Migrating from disk-based to server-based fencing in an online cluster" on page 60.

See "Migrating from server-based to disk-based fencing in an online cluster" on page 62.

If the vxfenswap operation is unsuccessful, then you can use the `-a cancel` of the `vxfenswap` command to manually roll back the changes that the vxfenswap utility does.

- For disk-based fencing, use the `vxfenswap -g diskgroup -a cancel` command to cancel the vxfenswap operation.
  You must run this command if a node fails during the process of disk replacement, or if you aborted the disk replacement.
- For server-based fencing, use the `vxfenswap -a cancel` command to cancel the vxfenswap operation.

Replacing I/O fencing coordinator disks when the cluster is online

Review the procedures to add, remove, or replace one or more coordinator disks in a cluster that is operational.

Warning: The cluster might panic if any node leaves the cluster membership before the vxfenswap script replaces the set of coordinator disks.

To replace a disk in a coordinator diskgroup when the cluster is online

1. Make sure system-to-system communication is functioning properly.
2. Make sure that the cluster is online.

```
# vxfenadm -d

I/O Fencing Cluster Information:
================================
Fencing Protocol Version: 201
Fencing Mode: SCSI3
Fencing SCSI3 Disk Policy: dmp
Cluster Members:
  * 0 (galaxy)
  1 (nebula)
RFSM State Information:
  node 0 in state 8 (running)
  node 1 in state 8 (running)
```

3. Import the coordinator disk group.

   The file /etc/vxfendg includes the name of the disk group (typically, 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 node 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 locks are removed.

4. If your setup uses VRTSvxvm version 5.1 RP2 (or later) or 5.1 SP1 (or later), then skip to step 5. You need not set `coordinator=off` to add or remove disks. For other VxVM versions, perform this step:

   Turn off the coordinator attribute value for the coordinator disk group.

```
# vxdg -g vxfencoorddg set coordinator=off
```

5. To remove disks from the coordinator disk group, use the VxVM disk administrator utility vxdiskadm.
6. Perform the following steps to add new disks to the coordinator disk group:

   - Add new disks to the node.

- Initialize the new disks as VxVM disks.
- Check the disks for I/O fencing compliance.
- Add the new disks to the coordinator disk group and set the coordinator attribute value as "on" for the coordinator disk group.

See the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide* for detailed instructions.

Note that though the disk group content changes, the I/O fencing remains in the same state.

7 From one node, start the vxfenswap utility. You must specify the diskgroup to the utility.

The utility performs the following tasks:

- Backs up the existing /etc/vxfentab file.
- Creates a test file /etc/vxfentab.test for the diskgroup that is modified on each node.
- Reads the diskgroup you specified in the vxfenswap command and adds the diskgroup to the /etc/vxfentab.test file on each node.
- Verifies that the serial number of the new disks are identical on all the nodes. The script terminates if the check fails.
- Verifies that the new disks can support I/O fencing on each node.

8 If the disk verification passes, the utility reports success and asks if you want to commit the new set of coordinator disks.

9 Review the message that the utility displays and confirm that you want to commit the new set of coordinator disks. Else skip to step 10.

```
Do you wish to commit this change? [y/n] (default: n) y
```

If the utility successfully commits, the utility moves the /etc/vxfentab.test file to the /etc/vxfentab file.

10 If you do not want to commit the new set of coordinator disks, answer n.

The vxfenswap utility rolls back the disk replacement operation.

Replacing the coordinator diskgroup in a cluster that is online

You can also replace the coordinator diskgroup using the vxfenswap utility. The following example replaces the coordinator disk group vxfencoorddg with a new disk group vxfendg.

To replace the coordinator diskgroup

1. Make sure system-to-system communication is functioning properly.
2. Make sure that the cluster is online.

```
# vxfenadm -d


I/O Fencing Cluster Information:
================================
Fencing Protocol Version: 201
Fencing Mode: SCSI3
Fencing SCSI3 Disk Policy: dmp
Cluster Members:
  * 0 (galaxy)
  1 (nebula)
RFSM State Information:
  node 0 in state 8 (running)
  node 1 in state 8 (running)
```

3. Find the name of the current coordinator diskgroup (typically vxfencoorddg) that is in the /etc/vxfendg file.

```
# cat /etc/vxfendg
vxfencoorddg
```

4. Find the alternative disk groups available to replace the current coordinator diskgroup.

```
# vxdisk -o alldgs list

DEVICE    TYPE    DISK  GROUP    STATUS
sda    auto:cdsdisk    -  (vxfendg)    online
sdb    auto:cdsdisk    -  (vxfendg)    online
sdc    auto:cdsdisk    -  (vxfendg)    online
sdx    auto:cdsdisk    -  (vxfencoorddg)    online
sdy    auto:cdsdisk    -  (vxfencoorddg)    online
sdz    auto:cdsdisk    -  (vxfencoorddg)    online
```

5 Validate the new disk group for I/O fencing compliance. Run the following command:

```
# vxfentsthdw -c vxfendg
```

See "Testing the coordinator disk group using vxfentsthdw -c option" on page 107.

6 If the new disk group is not already deported, run the following command to deport the disk group:

```
# vxdg deport vxfendg
```

7 Make sure that the /etc/vxfenmode file is updated to specify the correct disk policy.

See the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide* for more information.

8 From any node, start the vxfenswap utility. For example, if vxfendg is the new diskgroup that you want to use as the coordinator diskgroup:

```
# vxfenswap -g vxfendg [-n]
```

The utility performs the following tasks:

- Backs up the existing /etc/vxfentab file.
- Creates a test file /etc/vxfentab.test for the diskgroup that is modified on each node.
- Reads the diskgroup you specified in the vxfenswap command and adds the diskgroup to the /etc/vxfentab.test file on each node.
- Verifies that the serial number of the new disks are identical on all the nodes. The script terminates if the check fails.
- Verifies that the new disk group can support I/O fencing on each node.

9 If the disk verification passes, the utility reports success and asks if you want to replace the coordinator disk group.

10 Review the message that the utility displays and confirm that you want to replace the coordinator disk group. Else skip to step 13.

```
Do you wish to commit this change? [y/n] (default: n) y
```

If the utility successfully commits, the utility moves the /etc/vxfentab.test file to the /etc/vxfentab file.

The utility also updates the /etc/vxfendg file with this new diskgroup.

11 Set the coordinator attribute value as "on" for the new coordinator disk group.

```
# vxdg -g vxfendg set coordinator=on
```

Set the coordinator attribute value as "off" for the old disk group.

```
# vxdg -g vxfencoorddg set coordinator=off
```

12 Verify that the coordinator disk group has changed.

```
# cat /etc/vxfendg
vxfendg
```

The swap operation for the coordinator disk group is complete now.

13 If you do not want to replace the coordinator disk group, answer n at the prompt.

The vxfenswap utility rolls back any changes to the coordinator diskgroup.

Changing the disk interaction policy in a cluster that is online

In a cluster that is online, you can change the disk interaction policy from dmp to raw using the vxfenswap utility.

To change the disk interaction policy

1 Make sure system-to-system communication is functioning properly.

2 Make sure that the cluster is online.

```
# vxfenadm -d


I/O Fencing Cluster Information:
================================
Fencing Protocol Version: 201
Fencing Mode: SCSI3
Fencing SCSI3 Disk Policy: dmp
Cluster Members:
  * 0 (galaxy)
  1 (nebula)
RFSM State Information:
  node 0 in state 8 (running)
  node 1 in state 8 (running)
```

3 On each node in the cluster, edit the /etc/vxfenmode file to change the disk policy.

```
# cat /etc/vxfenmode
vxfen_mode=scsi3
scsi3_disk_policy=raw
```

4 From any node, start the vxfenswap utility:

```
# vxfenswap -g vxfencoordg [-n]
```

5 Verify the change in the disk policy.

```
# vxfenadm -d


I/O Fencing Cluster Information:
================================
Fencing Protocol Version: 201
Fencing Mode: SCSI3
Fencing SCSI3 Disk Policy: raw
```

Adding disks from a recovered site to the coordinator diskgroup

In a campus cluster environment, consider a case where the primary site goes down and the secondary site comes online with a limited set of disks. When the primary site restores, the primary site's disks are also available to act as coordinator disks. You can use the vxfenswap utility to add these disks to the coordinator diskgroup.

To add new disks from a recovered site to the coordinator diskgroup

1. Make sure system-to-system communication is functioning properly.
2. Make sure that the cluster is online.

```
# vxfenadm -d


I/O Fencing Cluster Information:
================================
Fencing Protocol Version: 201
Fencing Mode: SCSI3
Fencing SCSI3 Disk Policy: dmp
Cluster Members:
  * 0 (galaxy)
  1 (nebula)
RFSM State Information:
  node 0 in state 8 (running)
  node 1 in state 8 (running)
```

3. Verify the name of the coordinator diskgroup.

```
# cat /etc/vxfendg
vxfencoorddg
```

4. Run the following command:

```
# vxdisk -o alldgs list

DEVICE    TYPE    DISK  GROUP    STATUS
sdx    auto:cdsdisk    -  (vxfencoorddg)    online
sdy    auto    -  -    offline
sdz    auto    -  -    offline
```

5. Verify the number of disks used in the coordinator diskgroup.

```
# vxfenconfig -l
I/O Fencing Configuration Information:
======================================
Count                : 1
Disk List
Disk Name        Major  Minor  Serial Number     Policy

/dev/vx/rdmp/sdx       32  48  R450 00013154 0312     dmp
```

6 When the primary site comes online, start the vxfenswap utility on any node in the cluster:

```
# vxfenswap -g vxfencoorddg [-n]
```

7 Verify the count of the coordinator disks.

```
# vxfenconfig -l
I/O Fencing Configuration Information:
======================================
Single Disk Flag      : 0
Count                 : 3
Disk List
Disk Name         Major  Minor  Serial Number     Policy

/dev/vx/rdmp/sdx       32  48  R450 00013154 0312     dmp
/dev/vx/rdmp/sdy       32  32  R450 00013154 0313     dmp
/dev/vx/rdmp/sdz       32  16  R450 00013154 0314     dmp
```

Refreshing lost keys on coordinator disks

If the coordinator disks lose the keys that are registered, the cluster might panic when a network partition occurs.

You can use the vxfenswap utility to replace the coordinator disks with the same disks. The vxfenswap utility registers the missing keys during the disk replacement.

To refresh lost keys on coordinator disks

1. Make sure system-to-system communication is functioning properly.
2. Make sure that the cluster is online.

```
# vxfenadm -d


I/O Fencing Cluster Information:
================================
Fencing Protocol Version: 201
Fencing Mode: SCSI3
Fencing SCSI3 Disk Policy: dmp
Cluster Members:
  * 0 (galaxy)
  1 (nebula)
RFSM State Information:
  node 0 in state 8 (running)
  node 1 in state 8 (running)
```

3. Run the following command to view the coordinator disks that do not have keys:

```
# vxfenadm -s all -f /etc/vxfentab


Device Name: /dev/vx/rdmp/sdx
Total Number of Keys: 0
No keys...
...
```

4. On any node, run the following command to start the vxfenswap utility:

```
# vxfenswap -g vxfencoorddg [-n]
```

5. Verify that the keys are atomically placed on the coordinator disks.

```
# vxfenadm -s all -f /etc/vxfentab


Device Name: /dev/vx/rdmp/sdx
Total Number of Keys: 4
...
```

Enabling or disabling the preferred fencing policy

You can enable or disable the preferred fencing feature for your I/O fencing configuration.

You can enable preferred fencing to use system-based race policy or group-based race policy. If you disable preferred fencing, the I/O fencing configuration uses the default count-based race policy.

See "About preferred fencing" on page 46.

See "How preferred fencing works" on page 46.

To enable preferred fencing for the I/O fencing configuration

1 Make sure that the cluster is running with I/O fencing set up.

```
# vxfenadm -d
```

2 Make sure that the cluster-level attribute UseFence has the value set to SCSI3.

```
# haclus -value UseFence
```

3 To enable system-based race policy, perform the following steps:

- Make the VCS configuration writable.

  ```
  # haconf -makerw
  ```

- Set the value of the cluster-level attribute PreferredFencingPolicy as System.

  ```
  # haclus -modify PreferredFencingPolicy System
  ```

- Set the value of the system-level attribute FencingWeight for each node in the cluster.
  For example, in a two-node cluster, where you want to assign galaxy five times more weight compared to nebula, run the following commands:

  ```
  # hasys -modify galaxy FencingWeight 50
  # hasys -modify nebula FencingWeight 10
  ```

- Save the VCS configuration.

  ```
  # haconf -dump -makero
  ```

4 To enable group-based race policy, perform the following steps:

- Make the VCS configuration writable.

```
# haconf -makerw
```

- Set the value of the cluster-level attribute PreferredFencingPolicy as Group.

  ```
  # haclus -modify PreferredFencingPolicy Group
  ```

- Set the value of the group-level attribute Priority for each service group. For example, run the following command:

  ```
  # hagrp -modify service_group Priority 1
  ```

  Make sure that you assign a parent service group an equal or lower priority than its child service group. In case the parent and the child service groups are hosted in different subclusters, then the subcluster that hosts the child service group gets higher preference.

- Save the VCS configuration.

  ```
  # haconf -dump -makero
  ```

5 To view the fencing node weights that are currently set in the fencing driver, run the following command:

```
# vxfenconfig -a
```

To disable preferred fencing for the I/O fencing configuration

1 Make sure that the cluster is running with I/O fencing set up.

```
# vxfenadm -d
```

2 Make sure that the cluster-level attribute UseFence has the value set to SCSI3.

```
# haclus -value UseFence
```

3 To disable preferred fencing and use the default race policy, set the value of the cluster-level attribute PreferredFencingPolicy as Disabled.

```
# haconf -makerw
# haclus -modify PreferredFencingPolicy Disabled
# haconf -dump -makero
```

Administering the CP server

This section provides the following CP server administration information:

- CP server administration user types and privileges
- CP server administration command (cpsadm)

This section also provides instructions for the following CP server administration tasks:

- Refreshing registration keys on the coordination points for server-based fencing
- Coordination Point replacement for an online cluster
- Migrating from non-secure to secure setup for CP server and SF Oracle RAC cluster communication

About the CP server user types and privileges

The CP server supports the following user types, each with a different access level privilege:

- CP server administrator (admin)
- CP server operator

Different access level privileges permit the user to issue different commands. If a user is neither a CP server admin nor a CP server operator user, then the user has guest status and can issue limited commands.

The user types and their access level privileges are assigned to individual users during SF Oracle RAC cluster configuration for fencing. During the installation process, you are prompted for a user name, password, and access level privilege (CP server admin or CP server operator).

To administer and operate a CP server, there must be at least one CP server admin.

A root user on a CP server is given all the administrator privileges, and these administrator privileges can be used to perform all the CP server specific operations.

cpsadm command

The `cpsadm` command is used to perform CP server administrative and maintenance tasks.

cpsadm command usage

The `cpsadmn` command has the following usage:

```
# cpsadm -s cp server -a action [-p port num] [-c cluster name] [-u
uuid] [-e username@domain.com] [-f user role] [-g domain type] [-h
hostname] [-n nodeid] [-o victim's hostname] [-v victim's nodeid]
```

where -s *cp server* is used to specify the CP server by its virtual IP address or virtual hostname and -a *action* is used to specify the CP server action.

For descriptions of the `cpsadm` command parameters, see Table 2-4.

For descriptions of specific types of `cpsadm` command actions, see Table 2-5.

Table 2-4 cpsadm command parameters

Parameter	Name	Description
-s	cp server	This parameter specifies the CP server by its virtual IP address or virtual hostname.
-a	action	This parameter specifies the action to be taken by CP server (see Table 2-5).
-c	cluster name	This parameter specifies the SF Oracle RAC cluster name.
-u	uuid	This parameter specifies the UUID (universally unique ID) of the SF Oracle RAC cluster.
-n	node id	This parameter specifies nodeid of SF Oracle RAC cluster node.
-v	victim node id	This parameter specifies a victim node's node ID.
-p	port	This parameter specifies the port number to connect to the CP server.
-e	user name	This parameter specifies the user to be added to the CP server.
-f	user role	This parameter specifies the user role (either cps_admin or cps_operator).
-g	domain type	This parameter specifies the domain type (e.g. vx, unixpwd, nis etc).
-h	host	This parameter specifies the hostname.

Table 2-4 cpsadm command parameters *(continued)*

Parameter	Name	Description
-o	victim node host name	This parameter specifies the victim node's host name.

Table 2-5 describes the specific `cpsadm` command action types. This command's action data (-a) may contain some of the above `cpsadm` command parameters.

Table 2-5 cpsadm command action types

Action	Description	User type
add_clus	Adds a SF Oracle RAC cluster.	CP server admin
add_clus_to_user	Adds a cluster to a specific user.	CP server admin
add_node	Adds a node to a cluster.	CP server admin CP server operator
add_user	Adds a user. The user added can be a user added to the cluster or a global-admin type of user.	CP server admin
db_snapshot	Takes a snapshot of the database. By default, the CP server database snapshot is located at: /etc/VRTScps/db	CP server admin
halt_cps	Stops the CP server(s).	CP server admin
list_membership	Lists the membership.	CP server admin CP server operator
list_nodes	Lists all nodes in the current cluster. To issue this command, the user needs to know either the SF Oracle RAC cluster name or UUID of the SF Oracle RAC cluster.	CP server admin CP server operator CP server guest
list_users	Lists all users.	CP server admin

Table 2-5 cpsadm command action types *(continued)*

Action	Description	User type
ping_cps	Pings a CP server. This ping command process is not performed at the network level using ICMP "echo request" packets. This ping_cps command initiates the sending of packets from the SF Oracle RAC cluster node to the CP server, seeking a response back from the CP server to the SF Oracle RAC cluster node.	CP server admin CP server operator CP server guest
preempt_node	Preempts a node by removing its registration from the CP server.	CP server admin CP server operator
reg_node	Registers a node.	CP server admin CP server operator
rm_clus	Removes a SF Oracle RAC cluster.	CP server admin
rm_clus_from_user	Removes a cluster from a specific user.	CP server admin
rm_node	Removes a node from the cluster.	CP server admin CP server operator
rm_user	Removes a user.	CP server admin
unreg_node	Unregisters a node.	CP server admin CP server operator

Environment variables associated with the coordination point server

Table 2-6 describes the environment variables that are required for the `cpsadm` command. The `cpsadm` command detects these environment variables and uses their value when communicating with the CP server. They are used to authenticate and authorize the user.

Note: The environment variables are not required when the `cpsadm` command is run on the CP server. The environment variables are required when the `cpsadm` command is run on the SF Oracle RAC cluster nodes.

Table 2-6 cpsadm command environment variables

Environment variable	Description
CPS_USERNAME	This is the fully qualified username as configured in VxSS (vssat showcred).
CPS_DOMAINTYPE	One of the following values: ■ vx ■ unixpwd ■ nis ■ nisplus ■ ldap

The environment variables must be exported directly on the shell before running the `cpsadm` command. For example,

```
# export CPS_USERNAME=
# export CPS_DOMAINTYPE=
```

Additionally, the username and domaintype values are the same as those added onto the CP server. To view these values run the following `cpsadm` command:

```
# cpsadm -s cp_server -a list_users
```

Where *cp_server* is the CP server's virtual IP address or virtual hostname.

About administering the coordination point server

This section describes how to perform administrative and maintenance tasks on the coordination point server (CP server).

For more information about the `cpsadm` command and the associated command options, see the `cpsadm`(1M) manual page.

Adding and removing SF Oracle RAC cluster entries from the CP server database

- To add a SF Oracle RAC cluster to the CP server database
  Type the following command:

```
# cpsadm -s cp_server -a add_clus -c cluster_name -u uuid
```

- To remove a SF Oracle RAC cluster from the CP server database
  Type the following command:

```
# cpsadm -s cp_server -a rm_clus -u uuid
```

cp_server	The CP server's virtual IP address or virtual hostname.
cluster_name	The SF Oracle RAC cluster name.
uuid	The UUID (Universally Unique ID) of the SF Oracle RAC cluster.

Adding and removing a SF Oracle RAC cluster node from the CP server database

- To add a SF Oracle RAC cluster node from the CP server database
  Type the following command:

```
# cpsadm -s cp_server -a add_node -u uuid -n nodeid
-h host
```

- To remove a SF Oracle RAC cluster node from the CP server database
  Type the following command:

```
# cpsadm -s cp_server -a rm_node -u uuid -n nodeid
```

cp_server	The CP server's virtual IP address or virtual hostname.
uuid	The UUID (Universally Unique ID) of the SF Oracle RAC cluster.
nodeid	The node id of the SF Oracle RAC cluster node.
host	Hostname

Adding or removing CP server users

- To add a user
  Type the following command:

```
# cpsadm -s cp_server -a add_user -e user_name -f user_role
-g domain_type -u uuid
```

- To remove a user

Type the following command:

```
# cpsadm -s cp_server -a rm_user -e user_name -g domain_type
```

cp_server	The CP server's virtual IP address or virtual hostname.
user_name	The user to be added to the CP server configuration.
user_role	The user role, either cps_admin or cps_operator.
domain_type	The domain type, for example vx, unixpwd, nis, etc.
uuid	The UUID (Universally Unique ID) of the SF Oracle RAC cluster.

Listing the CP server users

To list the CP server users

Type the following command:

```
# cpsadm -s cp_server -a list_users
```

Listing the nodes in all the SF Oracle RAC clusters

To list the nodes in all the SF Oracle RAC cluster

Type the following command:

```
# cpsadm -s cp_server -a list_nodes
```

Listing the membership of nodes in the SF Oracle RAC cluster

To list the membership of nodes in SF Oracle RAC cluster

Type the following command:

```
# cpsadm -s cp_server -a list_membership -c cluster_name
```

cp_server	The CP server's virtual IP address or virtual hostname.
cluster_name	The SF Oracle RAC cluster name.

Preempting a node

To preempt a node

Type the following command:

```
# cpsadm -s cp_server -a preempt_node -u uuid -n nodeid
-v victim_node id
```

cp_server	The CP server's virtual IP address or virtual hostname.
uuid	The UUID (Universally Unique ID) of the SF Oracle RAC cluster.
nodeid	The node id of the SF Oracle RAC cluster node.
victim_node id	The victim node's node id.

Registering and unregistering a node

- To register a node
  Type the following command:

  ```
  # cpsadm -s cp_server -a reg_node -u uuid -n nodeid
  ```

- To unregister a node
  Type the following command:

  ```
  # cpsadm -s cp_server -a unreg_node -u uuid -n nodeid
  ```

cp_server	The CP server's virtual IP address or virtual hostname.
uuid	The UUID (Universally Unique ID) of the SF Oracle RAC cluster.
nodeid	The nodeid of the SF Oracle RAC cluster node.

Enable and disable access for a user to a SF Oracle RAC cluster

- To enable access for a user to a SF Oracle RAC cluster
  Type the following command:

  ```
  # cpsadm -s cp_server -a add_clus_to_user -e user
  -f user_role -g domain_type -u uuid
  ```

- To disable access for a user to a SF Oracle RAC cluster
  Type the following command:

  ```
  # cpsadm -s cp_server -a rm_clus_from_user -e user_name
  -f user_role -g domain_type -u uuid
  ```

cp_server	The CP server's virtual IP address or virtual hostname.

user_name	The user name to be added to the CP server.
user_role	The user role, either cps_admin or cps_operator.
domain_type	The domain type, for example vx, unixpwd, nis, etc.
uuid	The UUID (Universally Unique ID) of the SF Oracle RAC cluster

Stopping the CP server

To stop the CP server

Type the following command:

```
# cpsadm -s cp_server -a halt_cps
```

Checking the connectivity of CP servers

To check the connectivity of a CP server

Type the following command:

```
# cpsadm -s cp_server -a ping_cps
```

Taking a CP server database snapshot

To take a CP server database snapshot

Type the following command:

```
# cpsadm -s cp_server -a db_snapshot
```

Refreshing registration keys on the coordination points for server-based fencing

Replacing keys on a coordination point (CP server) when the SF Oracle RAC cluster is online involves refreshing that coordination point's registrations. You can perform a planned refresh of registrations on a CP server without incurring application downtime on the SF Oracle RAC cluster. You must refresh registrations on a CP server if the CP server agent issues an alert on the loss of such registrations on the CP server database.

The following procedure describes how to refresh the coordination point registrations.

To refresh the registration keys on the coordination points for server-based fencing

1 Ensure that the SF Oracle RAC cluster nodes and users have been added to the new CP server(s). Run the following commands:

```
# cpsadm -s cp_server -a list_nodes

# cpsadm -s cp_server -a list_users
```

If the SF Oracle RAC cluster nodes are not present here, prepare the new CP server(s) for use by the SF Oracle RAC cluster.

See the Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide for instructions.

2 Ensure that fencing is running on the cluster in customized mode using the coordination points mentioned in the `/etc/vxfenmode` file.

For example, enter the following command:

```
# vxfenadm -d

================================
Fencing Protocol Version: 201
Fencing Mode: CUSTOMIZED
Cluster Members:
* 0 (galaxy)
1 (nebula)
RFSM State Information:
node 0 in state 8 (running)
node 1 in state 8 (running)
```

3 List the coordination points currently used by I/O fencing :

```
# vxfenconfig -l
```

4 Run the `vxfenswap` utility from one of the nodes of the cluster.

The `vxfenswap` utility requires secure ssh connection to all the cluster nodes. Use `-n` to use rsh instead of default ssh.

For example:

```
# vxfenswap [-n]
```

The command returns:

```
VERITAS vxfenswap version <version> <platform>
The logfile generated for vxfenswap is
/var/VRTSvcs/log/vxfen/vxfenswap.log.
19156
Please Wait...
VXFEN vxfenconfig NOTICE Driver will use customized fencing
- mechanism cps
Validation of coordination points change has succeeded on
all nodes.
You may commit the changes now.
WARNING: This may cause the whole cluster to panic
if a node leaves membership before the change is complete.
```

5 You are then prompted to commit the change. Enter **y** for yes.

The command returns a confirmation of successful coordination point replacement.

6 Confirm the successful execution of the vxfenswap utility. If CP agent is configured, it should report ONLINE as it succeeds to find the registrations on coordination points. The registrations on the CP server and coordinator disks can be viewed using the `cpsadm` and `vxfenadm` utilities respectively.

Note that a running online coordination point refreshment operation can be canceled at any time using the command:

```
# vxfenswap -a cancel
```

Replacing coordination points for server-based fencing in an online cluster

Use the following procedure to perform a planned replacement of customized coordination points (CP servers or SCSI-3 disks) without incurring application downtime on an online SF Oracle RAC cluster.

Note: If multiple clusters share the same CP server, you must perform this replacement procedure in each cluster.

You can use the vxfenswap utility to replace coordination points when fencing is running in customized mode in an online cluster, with vxfen_mechanism=cps. The utility also supports migration from server-based fencing (vxfen_mode=customized) to disk-based fencing (vxfen_mode=scsi3) and vice versa in an online cluster.

However, if the SF Oracle RAC cluster has fencing disabled (vxfen_mode=disabled), then you must take the cluster offline to configure disk-based or server-based fencing.

See "Deployment and migration scenarios for CP server" on page 55.

You can cancel the coordination point replacement operation at any time using the `vxfenswap -a cancel` command.

See "About the vxfenswap utility" on page 120.

To replace coordination points for an online cluster

1 Ensure that the SF Oracle RAC cluster nodes and users have been added to the new CP server(s). Run the following commands:

```
# cpsadm -s cpserver -a list_nodes
# cpsadm -s cpserver -a list_users
```

If the SF Oracle RAC cluster nodes are not present here, prepare the new CP server(s) for use by the SF Oracle RAC cluster.

See the Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide for instructions.

2 Ensure that fencing is running on the cluster using the old set of coordination points and in customized mode.

For example, enter the following command:

```
# vxfenadm -d
```

The command returns:

```
I/O Fencing Cluster Information:
================================
Fencing Protocol Version: <version>
Fencing Mode: Customized
Cluster Members:
* 0 (galaxy)
1 (nebula)
RFSM State Information:
node 0 in state 8 (running)
node 1 in state 8 (running)
```

3 Back up the `/etc/vxfenmode` file on each of the client cluster nodes.

4 Use a text editor to access `/etc/vxfenmode` and update the values to the new CP server (coordination points).

The values of the `/etc/vxfenmode` file have to be updated on all the nodes in the SF Oracle RAC cluster.

Review and if necessary, update the vxfenmode parameters for security, the coordination points, and if applicable to your configuration, vxfendg.

Refer to the text information within the `vxfenmode` file for additional information about these parameters and their new possible values.

5 Run the `vxfenswap` utility from one of the nodes of the cluster.

The `vxfenswap` utility requires secure ssh connection to all the cluster nodes. Use -n to use rsh instead of default ssh.

```
# vxfenswap [-n]
```

If validation of coordination points from all the nodes fails, the `vxfenswap` utility rollbacks the coordination point replacement operation. Proceed to restore `/etc/vxfenmode` with the backed up file on all the VCS cluster nodes.

You are then prompted to commit the change. Enter **y** for yes.

Confirm the successful execution of the `vxfenswap` utility by checking the coordination points currently used by the vxfen driver.

For example, run the following command:

```
# vxfenconfig -l
```

Migrating from non-secure to secure setup for CP server and SF Oracle RAC cluster communication

The following procedure describes how to migrate from a non-secure to secure set up for the CP server and SF Oracle RAC cluster.

To migrate from non-secure to secure setup for CP server and SF Oracle RAC cluster

1 Stop fencing on all the SF Oracle RAC cluster nodes of all the clusters (which are using the CP servers).

```
# /etc/init.d/vxfen stop
```

2 Stop all the CP servers using the following command on each CP server:

```
# hagrp -offline CPSSG -any
```

3 Ensure that security is configured for communication between CP servers and SF Oracle RAC cluster nodes.

See "About secure communication between the SF Oracle RAC cluster and CP server" on page 65.

4 Modify /etc/vxcps.conf on each CP server to set security=1.

5 Start CP servers using the following command on all of them:

```
# hagrp -online CPSSG -any
```

6 Add the following user for each client node on each CP server:

_HA_VCS_*hostname*@HA_SERVICES@*FQHN*

where, hostname is the client node name without qualification, and FQHN is Fully Qualified Host Name of the client node.

Add the users to the CP server database.

For example, issue the following commands on the CP server (mycps.symantecexample.com):

```
# cpsadm -s mycps.symantecexample.com -a add_user -e\
_HA_VCS_galaxy@HA_SERVICES@galaxy.symantec.com\
-g vx

User _HA_VCS_galaxy@HA_SERVICES@galaxy.symantec.com successfully added

# cpsadm -s mycps.symantecexample.com -a add_user -e\
_HA_VCS_nebula@HA_SERVICES@nebula.symantec.com\
-g vx

User _HA_VCS_nebula@HA_SERVICES@nebula.symantec.com successfully added
```

7 Authorize the user to administer the cluster.

For example, issue the following command on the CP server (mycps.symantecexample.com):

```
# cpsadm -s mycps.symantecexample.com -a\
add_clus_to_user -c cpcluster\
-u {f0735332-1dd1-11b2-a3cb-e3709c1c73b9}\
-e _HA_VCS_galaxy@HA_SERVICES@galaxy.symantec.com\
-f cps_operator -g vx

Cluster successfully added to user

_HA_VCS_galaxy@HA_SERVICES@galaxy.symantec.com privileges.
```

8 Modify /etc/vxfenmode file on each SF Oracle RAC cluster node to set security=1.

9 After modifying the /etc/vxfenmode file, run the vxfen init script on the SF Oracle RAC cluster node to start fencing.

For an example:

```
# /etc/init.d/vxfen start
```

Administering CFS

This section describes some of the major aspects of cluster file system administration.

This section provides instructions for the following CFS administration tasks:

- Adding CFS file systems to VCS configuration
  See "Adding CFS file systems to VCS configuration" on page 149.
- Using cfsmount to mount CFS file systems
  See "Using cfsmount to mount CFS file systems" on page 149.
- Resizing CFS file systems
  See "Resizing CFS file systems" on page 149.
- Verifying the status of CFS file systems
  See "Verifying the status of CFS file systems" on page 150.
- Verifying CFS port
  See "Verifying CFS port" on page 150.

If you encounter issues while administering CFS, refer to the troubleshooting section for assistance.

See "Troubleshooting CFS" on page 219.

Adding CFS file systems to VCS configuration

To add a CFS file system to the VCS main.cf file without using an editor:

```
# cfsmntadm add oradatadg oradatavol \
/oradata1 cvm all=suid,rw

Mount Point is being added...
  /oradata1 added to the cluster-configuration
```

See the *Veritas Storage Foundation Cluster File System Administrator's Guide* for more information on the command.

Using cfsmount to mount CFS file systems

To mount a CFS file system using cfsmount:

```
# cfsmount /oradata1
Mounting...
[/dev/vx/dsk/oradatadg/oradatavol]
mounted successfully at /oradata1 on galaxy
[/dev/vx/dsk/oradatadg/oradatavol]
mounted successfully at /oradata1 on nebula
```

See the *Veritas Storage Foundation Cluster File System Administrator's Guide* for more information on the command.

Resizing CFS file systems

If you see a message on the console indicating that a CFS file system is full, you may want to resize the file system. The `vxresize` command lets you resize a CFS file system. It extends the file system and the underlying volume.

See the `vxresize` (1M) manual page for information on various options.

The following command resizes an Oracle data CFS file system (the Oracle data volume is CFS mounted):

```
# vxresize -g oradatadg oradatavol +2G
```

where oradatadg is the CVM disk group, oradatavol is the volume, and +2G indicates the increase in volume size by 2 Gigabytes.

Verifying the status of CFS file systems

Run the "cfscluster status" command to see the status of the nodes and their mount points:

```
# cfscluster status

Node             :  galaxy
Cluster Manager  :  running
CVM state        :  running

MOUNT POINT           SHARED VOLUME  DISK GROUP        STATUS
/app/crshome          crsbinvol      bindg             MOUNTED
/ocrvote              ocrvotevol     ocrvotedg         MOUNTED
/app/oracle/orahome   orabinvol bindg                  MOUNTED
/oradata1             oradatavol     oradatadg         MOUNTED
/arch                 archvol        oradatadg         MOUNTED

Node             :  nebula
Cluster Manager  :  running
CVM state        :  running

MOUNT POINT           SHARED VOLUME  DISK GROUP        STATUS
/app/crshome          crsbinvol      bindg             MOUNTED
/ocrvote              ocrvotevol     ocrvotedg         MOUNTED
/app/oracle/orahome   orabinvol bindg                  MOUNTED
/oradata1             oradatavol     oradatadg         MOUNTED
/arch                 archvol        oradatadg         MOUNTED
```

Verifying CFS port

CFS uses port ‘f’ for communication between nodes. The CFS port state can be verified as follows:

```
# gabconfig -a | grep "Port f"
```

Administering CVM

This section provides instructions for the following CVM administration tasks:

- Listing all the CVM shared disks
  See “Listing all the CVM shared disks” on page 235.
- Establishing CVM cluster membership manually
  See “Establishing CVM cluster membership manually” on page 151.

- Importing a shared disk group manually
  See "Importing a shared disk group manually" on page 154.
- Deporting a shared disk group manually
  See "Deporting a shared disk group manually " on page 154.
- Starting shared volumes manually
  See "Starting shared volumes manually" on page 235.
- Evaluating the state of CVM ports
  See "Evaluating the state of CVM ports" on page 154.
- Verifying if CVM is running in an SF Oracle RAC cluster
  See "Verifying if CVM is running in an SF Oracle RAC cluster" on page 154.
- Verifying CVM membership state
  See "Verifying CVM membership state" on page 155.
- Verifying the state of CVM shared disk groups
  See "Verifying the state of CVM shared disk groups" on page 156.
- Verifying the activation mode
  See "Verifying the activation mode" on page 156.

If you encounter issues while administering CVM, refer to the troubleshooting section for assistance.

See "Troubleshooting CVM" on page 215.

Establishing CVM cluster membership manually

In most cases you do not have to start CVM manually; it normally starts when VCS is started.

Run the following command to start CVM manually:

```
# vxclustadm -m vcs -t gab startnode

vxclustadm: initialization completed
```

Note that `vxclustadm` reads main.cf for cluster configuration information and is therefore not dependent upon VCS to be running. You do not need to run the `vxclustadm startnode` command as normally the `hastart` (VCS start) command starts CVM automatically.

To verify whether CVM is started properly:

```
# vxclustadm nidmap
Name          CVM Nid     CM Nid      State
```

```
galaxy       0          0        Joined: Master
nebula       1          1        Joined: Slave
```

Changing the CVM master manually

You can change the CVM master manually from one node in the cluster to another node, while the cluster is online. CVM migrates the master node, and reconfigures the cluster.

Symantec recommends that you switch the master when the cluster is not handling VxVM configuration changes or cluster reconfiguration operations. In most cases, CVM aborts the operation to change the master, if CVM detects that any configuration changes are occurring in the VxVM or the cluster. After the master change operation starts reconfiguring the cluster, other commands that require configuration changes will fail.

See "Errors during CVM master switching" on page 153.

To change the master online, the cluster must be cluster protocol version 100 or greater.

To change the CVM master manually

1 To view the current master, use one of the following commands:

```
# vxclustadm nidmap
Name              CVM Nid     CM Nid     State
galaxy            0           0          Joined: Slave
nebula            1           1          Joined: Master

# vxdctl -c mode
mode: enabled: cluster active - MASTER
master: nebula
```

In this example, the CVM master is nebula.

2 From any node on the cluster, run the following command to change the CVM master:

```
# vxclustadm setmaster nodename
```

where *nodename* specifies the name of the new CVM master.

The following example shows changing the master on a cluster from nebula to galaxy:

```
# vxclustadm setmaster galaxy
```

3 To monitor the master switching, use the following command:

```
# vxclustadm -v nodestate
 state: cluster member
        nodeId=0
        masterId=0
        neighborId=1
        members[0]=0xf
        joiners[0]=0x0
        leavers[0]=0x0
        members[1]=0x0
        joiners[1]=0x0
        leavers[1]=0x0
        reconfig_seqnum=0x9f9767
        vxfen=off
state: master switching in progress
reconfig: vxconfigd in join
```

In this example, the state indicates that master is being changed.

4 To verify whether the master has successfully changed, use one of the following commands:

```
# vxclustadm nidmap
Name              CVM Nid    CM Nid    State
galaxy            0          0         Joined: Master
nebula            1          1         Joined: Slave
```

```
# vxdctl -c mode
mode: enabled: cluster active - MASTER
master: galaxy
```

Errors during CVM master switching

Symantec recommends that you switch the master when the cluster is not handling VxVM or cluster configuration changes.

In most cases, CVM aborts the operation to change the master, if CVM detects any configuration changes in progress. CVM logs the reason for the failure into the system logs. In some cases, the failure is displayed in the `vxclustadm setmaster` output as follows:

```
# vxclustadm setmaster galaxy
VxVM vxclustadm ERROR V-5-1-0 Master switching, a reconfiguration or
```

```
a transaction is in progress.
Try again
```

In some cases, if the master switching operation is interrupted with another reconfiguration operation, the master change fails. In this case, the existing master remains the master of the cluster. After the reconfiguration is complete, reissue the `vxclustadm setmaster` command to change the master.

If the master switching operation has started the reconfiguration, any command that initiates a configuration change fails with the following error:

```
Node processing a master-switch request. Retry operation.
```

If you see this message, retry the command after the master switching has completed.

Importing a shared disk group manually

You can use the following command to manually import a shared disk group:

```
# vxdg -s import dg_name
```

Deporting a shared disk group manually

You can use the following command to manually deport a shared disk group:

```
# vxdg deport dg_name
```

Note that the deport of a shared disk group removes the SCSI-3 PGR keys on the disks. It also removes the ‘shared’ flag on the disks.

Evaluating the state of CVM ports

CVM kernel (vxio driver) uses port ‘v’ for kernel messaging, port ‘w’ for vxconfigd communication between the cluster nodes, and port 'u' for shipping commands from the slave node to the master node. The following command displays the state of CVM ports:

```
# gabconfig -a | egrep "Port [vwu]"
```

Verifying if CVM is running in an SF Oracle RAC cluster

You can use the following options to verify whether CVM is up or not in an SF Oracle RAC cluster.

The following output is displayed on a node that is not a member of the cluster:

```
# vxdctl -c mode
mode: enabled: cluster inactive
# vxclustadm -v nodestate
state: out of cluster
```

On the master node, the following output is displayed:

```
# vxdctl -c mode

mode: enabled: cluster active - MASTER
master: galaxy
```

On the slave nodes, the following output is displayed:

```
# vxdctl -c mode

mode: enabled: cluster active - SLAVE
master: nebula
```

The following command lets you view all the CVM nodes at the same time:

```
# vxclustadm nidmap

Name      CVM Nid    CM Nid     State
galaxy          0          0     Joined: Master
nebula          1          1     Joined: Slave
```

Verifying CVM membership state

The state of CVM can be verified as follows:

```
# vxclustadm -v nodestate

state: joining
        nodeId=0
        masterId=0
        neighborId=0
        members=0x1
        joiners=0x0
        leavers=0x0
        reconfig_seqnum=0x0
        reconfig: vxconfigd in join
```

The state indicates that CVM has completed its kernel level join and is in the middle of vxconfigd level join.

The `vxdctl -c mode` command indicates whether a node is a CVM master or CVM slave.

Verifying the state of CVM shared disk groups

You can use the following command to list the shared disk groups currently imported in the SF Oracle RAC cluster:

```
# vxdg list |grep shared

orabinvol_dg enabled,shared 1052685125.1485.csha3
```

Verifying the activation mode

In an SF Oracle RAC cluster, the activation of shared disk group should be set to "shared-write" on each of the cluster nodes.

To verify whether the "shared-write"activation is set:

```
# vxdg list dg_name |grep activation

local-activation: shared-write
```

If "shared-write" activation is not set, run the following command:

```
# vxdg -g dg_name set activation=sw
```

Administering SF Oracle RAC global clusters

This section provides instructions for the following global cluster administration tasks:

- About setting up a fire drill
  See "Setting up a disaster recovery fire drill" on page 157.
- Configuring the fire drill service group using the wizard
  See "Configuring the fire drill service group using the Fire Drill Setup wizard" on page 157.
- Verifying a successful fire drill
  See "Verifying a successful fire drill" on page 159.
- Scheduling a fire drill
  See "Scheduling a fire drill" on page 160.

For a sample fire drill service group configuration:

See "Sample fire drill service group configuration" on page 160.

Setting up a disaster recovery fire drill

The Disaster Recovery Fire Drill procedure tests the fault-readiness of a configuration by mimicking a failover from the primary site to the secondary site. This procedure is done without stopping the application at the primary site and disrupting user access, interrupting the flow of replicated data, or causing the secondary site to need resynchronization.

The initial steps to create a fire drill service group on the secondary site that closely follows the configuration of the original application service group and contains a point-in-time copy of the production data in the Replicated Volume Group (RVG). Bringing the fire drill service group online on the secondary site demonstrates the ability of the application service group to fail over and come online at the secondary site, should the need arise. Fire drill service groups do not interact with outside clients or with other instances of resources, so they can safely come online even when the application service group is online.

You must conduct a fire drill only at the secondary site; do not bring the fire drill service group online on the node hosting the original application.

Before you perform a fire drill in a disaster recovery setup that uses VVR, perform the following steps:

- Set the value of the ReuseMntPt attribute to 1 for all Mount resources.
- Configure the fire drill service group.
  See "Configuring the fire drill service group using the Fire Drill Setup wizard" on page 157.
- After the fire drill service group is taken offline, reset the value of the ReuseMntPt attribute to 0 for all Mount resources.

VCS also supports HA fire drills to verify a resource can fail over to another node in the cluster.

Note: You can conduct fire drills only on regular VxVM volumes; volume sets (vset) are not supported.

VCS provides hardware replication agents for array-based solutions, such as Hitachi Truecopy, EMC SRDF, and so on . If you are using hardware replication agents to monitor the replicated data clusters, refer to the VCS replication agent documentation for details on setting up and configuring fire drill.

Configuring the fire drill service group using the Fire Drill Setup wizard

Use the Fire Drill Setup Wizard to set up the fire drill configuration.

The wizard performs the following specific tasks:

- Creates a Cache object to store changed blocks during the fire drill, which minimizes disk space and disk spindles required to perform the fire drill.
- Configures a VCS service group that resembles the real application group.

The wizard works only with application groups that contain one disk group. The wizard sets up the first RVG in an application. If the application has more than one RVG, you must create space-optimized snapshots and configure VCS manually, using the first RVG as reference.

You can schedule the fire drill for the service group using the `fdsched` script.

See "Scheduling a fire drill" on page 160.

Running the fire drill setup wizard

To run the wizard

1. Start the RVG Secondary Fire Drill wizard on the VVR secondary site, where the application service group is offline and the replication group is online as a secondary:

   ```
   # /opt/VRTSvcs/bin/fdsetup
   ```

2. Read the information on the Welcome screen and press the **Enter** key.
3. The wizard identifies the global service groups. Enter the name of the service group for the fire drill.
4. Review the list of volumes in disk group that could be used for a spaceoptimized snapshot. Enter the volumes to be selected for the snapshot. Typically, all volumes used by the application, whether replicated or not, should be prepared, otherwise a snapshot might not succeed.

   Press the **Enter** key when prompted.

5 Enter the cache size to store writes when the snapshot exists. The size of the cache must be large enough to store the expected number of changed blocks during the fire drill. However, the cache is configured to grow automatically if it fills up. Enter disks on which to create the cache.

Press the **Enter** key when prompted.

6 The wizard starts running commands to create the fire drill setup.

Press the **Enter** key when prompted.

The wizard creates the application group with its associated resources. It also creates a fire drill group with resources for the application (Oracle, for example), the CFSMount, and the RVGSnapshot types.

The application resources in both service groups define the same application, the same database in this example. The wizard sets the FireDrill attribute for the application resource to 1 to prevent the agent from reporting a concurrency violation when the actual application instance and the fire drill service group are online at the same time.

Configuring local attributes in the fire drill service group

The fire drill setup wizard does not recognize localized attribute values for resources. If the application service group has resources with local (per-system) attribute values, you must manually set these attributes after running the wizard.

Verifying a successful fire drill

Bring the fire drill service group online on a node that does not have the application running. Verify that the fire drill service group comes online. This action validates that your disaster recovery solution is configured correctly and the production service group will fail over to the secondary site in the event of an actual failure (disaster) at the primary site.

If the fire drill service group does not come online, review the VCS engine log to troubleshoot the issues so that corrective action can be taken as necessary in the production service group.

You can also view the fire drill log, located at `/tmp/fd-servicegroup.pid`

Remember to take the fire drill offline once its functioning has been validated. Failing to take the fire drill offline could cause failures in your environment. For example, if the application service group were to fail over to the node hosting the fire drill service group, there would be resource conflicts, resulting in both service groups faulting.

Scheduling a fire drill

You can schedule the fire drill for the service group using the fdsched script. The fdsched script is designed to run only on the lowest numbered node that is currently running in the cluster. The scheduler runs the command `hagrp -online firedrill_group -any` at periodic intervals.

To schedule a fire drill

1. Add the file `/opt/VRTSvcs/bin/fdsched` to your crontab.
2. To make fire drills highly available, add the `fdsched` file to each node in the cluster.

Sample fire drill service group configuration

The sample configuration file, `/etc/VRTSvcs/conf/sample_rac/sfrac15_main.cf`, describes a fire drill service group configuration on the secondary site. The configuration uses VVR for replicating data between the sites.

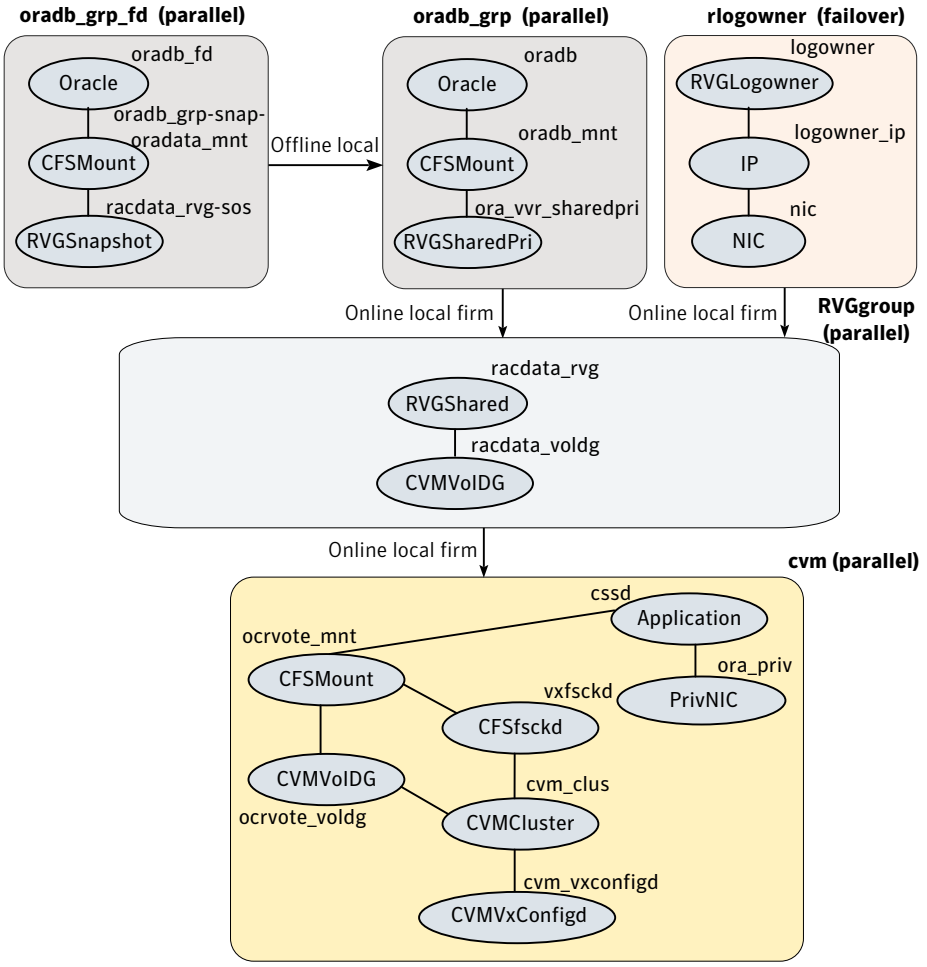
This sample file describes the following configuration:

- Two SF Oracle RAC clusters, comprising two nodes each, hosted at different geographical locations.
- A single Oracle database that is stored on CFS.
- The database is managed by the VCS agent for Oracle.
  The agent starts, stops, and monitors the database.
- The database uses the Oracle UDP IPC for database cache fusion.
- A common IP address is used by Oracle Clusterware and database cache fusion.
  The private IP address is managed by the PrivNIC agent for high availability.
- One virtual IP address must be configured under the `ClusterService` group on each site for inter-cluster communication.
- The Oracle Cluster Registry (OCR) and voting disk are stored on CFS.
- Veritas Volume Replicator (VVR) is used to replicate data between the sites.
- The shared volumes replicated across the sites are configured under the RVG group.
- The replication link used by VVR for communicating log information between sites are configured under the `rlogowner` group. This is a failover group that will be online on only one of the nodes in the cluster at each site.
- The database group is configured as a global group by specifying the clusters on the primary and secondary sites as values for the ClusterList group attribute.

- The fire drill service group `oradb_grp_fd` creates a snapshot of the replicated data on the secondary site and starts the database using the snapshot. An offline local dependency is set between the fire drill service group and the application service group to make sure a fire drill does not block an application failover in case a disaster strikes the primary site.

Figure 2-1 illustrates the configuration.

Figure 2-1 Service group configuration for sfrac15_main.cf file



Section 2

Performance and troubleshooting

- Chapter 3. Troubleshooting SF Oracle RAC
- Chapter 4. Prevention and recovery strategies
- Chapter 5. Tunable parameters

Chapter 3

Troubleshooting SF Oracle RAC

This chapter includes the following topics:

- About troubleshooting SF Oracle RAC
- What to do if you see a licensing reminder
- Restarting the installer after a failed connection
- Installer cannot create UUID for the cluster
- Troubleshooting installation and configuration check failures
- Troubleshooting LLT health check warning messages
- Troubleshooting I/O fencing
- Troubleshooting CVM
- Troubleshooting CFS
- Troubleshooting interconnects
- Troubleshooting Oracle
- Troubleshooting ODM

About troubleshooting SF Oracle RAC

SF Oracle RAC contains several component products, and as a result can be affected by any issue with component products. The first step in case of trouble should be to identify the source of the problem. It is rare to encounter problems in SF Oracle

RAC itself; more commonly the problem can be traced to setup issues or problems in component products.

Use the information in this chapter to diagnose the source of problems. Indications may point to SF Oracle RAC set up or configuration issues, in which case solutions are provided wherever possible. In cases where indications point to a component product or to Oracle as the source of a problem, it may be necessary to refer to the appropriate documentation to resolve it.

Running scripts for engineering support analysis

Troubleshooting scripts gather information about the configuration and status of your cluster and its modules. The scripts identify package information, debugging messages, console messages, and information about disk groups and volumes. Forwarding the output of these scripts to Symantec Tech Support can assist with analyzing and solving any problems.

getdbac

The getdbac script gathers information about the SF Oracle RAC modules. The file /tmp/vcsopslog.*time_stamp*.tar.Z contains the script's output.

To use the getdbac script, on each system enter:

```
# /opt/VRTSvcs/bin/getdbac -local
```

getcomms

The getcomms script gathers information about the GAB and LLT modules. The file /tmp/commslog.*time_stamp*.tar contains the script's output.

To use the getcomms script, on each system enter:

```
# /opt/VRTSgab/getcomms -local
```

hagetcf

The hagetcf script gathers information about the VCS cluster and the status of resources. The output from this script is placed in a tar file, /tmp/vcsconf.*sys_name*.tar.gz, on each cluster system.

To use the hagetcf script, on each system enter:

```
# /opt/VRTSvcs/bin/hagetcf
```

Log files

Table 3-1 lists the various log files and their location. The log files contain useful information for identifying issues and resolving them.

Table 3-1 List of log files

Log file	Location	Description
Oracle installation error log	*$ORACLE_BASE*\ /oraInventory/logs/\ installActions*date_time*.log	Contains errors that occurred during Oracle RAC installation. It clarifies the nature of the error and when it occurred during the installation. **Note:** Verify if there are any installation errors logged in this file, since they may prove to be critical errors. If there are any installation problems, send this file to Tech Support for debugging the issue.
Oracle alert log	*$ORACLE_BASE*/diag/rdbms/*db_name*/\ *instance_name*/trace/alert_*instance_name*.log The log path is configurable.	Contains messages and errors reported by database operations.
VCS engine log file	/var/VRTSvcs/log/engine_A.log	Contains all actions performed by the high availability daemon had. **Note:** Verify if there are any CVM or PrivNIC errors logged in this file, since they may prove to be critical errors.
CVM log files	/var/adm/vx/cmdlog /var/adm/vx/ddl.log /var/adm/vx/translog /var/adm/vx/dmpevents.log /var/VRTSvcs/log/engine_A.log	The cmdlog file contains the list of CVM commands. For more information on collecting important CVM logs: See "Collecting important CVM logs" on page 168.

Table 3-1 List of log files *(continued)*

Log file	Location	Description

Agent log file for CVM	`/var/VRTSvcs/log/engine_A.log`
	`/var/VRTSvcs/log/CVMVxconfigd_A.log`
	`/var/VRTSvcs/log/CVMCluster_A.log`
	`/var/VRTSvcs/log/CVMVolDg_A.log`
		Contains messages and errors related to CVM agent functions.
		Search for "cvm" in the engine_A.log for debug information.
		For more information, see the *Veritas Volume Manager Administrator's Guide*.

Agent log files for CFS	`/var/VRTSvcs/log/engine_A.log`
	`/var/VRTSvcs/log/CFSfsckd_A.log`
	`/var/VRTSvcs/log/CFSMount_A.log`
		Contains messages and errors related to CFS agent functions.
		Search for "cfs" in the engine_A.log for debug information.

Agent log files for Oracle	`/var/VRTSvcs/log/Oracle_A.log`	Contains messages and errors related to agent functions.
OS system log	`/var/log/messages`	Contains messages and errors arising from operating system modules and drivers.

I/O fencing kernel logs	`/var/VRTSvcs/log/vxfen/vxfen.log`
	Obtain the logs by running the following command:
	`# /opt/VRTSvcs/vxfen/bin/\`
	`vxfendebug -p`
		Contains messages, errors, or diagnostic information for I/O fencing.

VCSMM log files	`/var/VRTSvcs/log/vcsmmconfig.log`	Contains messages, errors, or diagnostic information for VCSMM.

Collecting important CVM logs

You need to stop and restart the cluster to collect detailed CVM TIME_JOIN messages.

To collect detailed CVM TIME_JOIN messages

1 On all the nodes in the cluster, perform the following steps.

- Edit the `/opt/VRTSvcs/bin/CVMCluster/online` script.
  Insert the '-T' option to the following string.

Original string: `clust_run=`$VXCLUSTADM -m vcs -t $TRANSPORT startnode 2> $CVM_ERR_FILE``

Modified string: **`clust_run=`$VXCLUSTADM -m vcs -t $TRANSPORT -T startnode 2> $CVM_ERR_FILE``**

2 Stop the cluster.

```
# hastop -all
```

3 Start the cluster

```
# hastart
```

At this point, CVM TIME_JOIN messages display in the `/var/adm/messages`file and on the console.

You can also enable vxconfigd daemon logging as follows:

```
# vxdctl debug 9 /var/adm/vx/vxconfigd_debug.out
```

The debug information that is enabled is accumulated in the system console log and in the text file `/var/adm/vx/vxconfigd_debug.out`. '9' represents the level of debugging. '1' represents minimal debugging. '9' represents verbose output.

Caution: Turning on vxconfigd debugging degrades VxVM performance. Use vxconfigd debugging with discretion in a production environment.

To disable vxconfigd debugging:

```
# vxdctl debug 0
```

The CVM kernel message dump can be collected on a live node as follows:

```
# /etc/vx/diag.d/kmsgdump32 -d 2000 > \
/var/adm/vx/kmsgdump.out
```

GAB message logging

If GAB encounters some problem, then First Failure Data Capture (FFDC) logs are also generated and dumped along with other core dumps and stack traces.

When you have configured GAB, GAB also starts a GAB logging daemon (`/opt/VRTSgab/gablogd`). GAB logging daemon is enabled by default. You can change the value of the GAB tunable parameter `gab_ibuf_count` to disable the GAB logging daemon.

See "About GAB load-time or static tunable parameters" on page 238.

This GAB logging daemon collects the GAB related logs when a critical events such as an iofence or failure of the master of any GAB port occur, and stores the data in a compact binary form. You can use the `gabread_ffdc` utility as follows to read the GAB binary log files:

```
/opt/VRTSgab/gabread_ffdc-kernel_version  binary_logs_files_location
```

You can change the values of the following environment variables that control the GAB binary log files:

- GAB_FFDC_MAX_INDX: Defines the maximum number of GAB binary log files
  The GAB logging daemon collects the defined number of log files each of eight MB size. The default value is 20, and the files are named `gablog.1` through `gablog.20`. At any point in time, the most recent file is the `gablog.1` file.
- GAB_FFDC_LOGDIR: Defines the log directory location for GAB binary log files
  The default location is:
  `/var/log/gab_ffdc`
  Note that the gablog daemon writes its log to the `glgd_A.log` and `glgd_B.log` files in the same directory.

You can either define these variables in the following GAB startup file or use the `export` command. You must restart GAB for the changes to take effect.

```
/etc/sysconfig/gab
```

Gathering LLT and GAB information for support analysis

You must run the `getcomms` script to gather LLT and GAB information when you encounter issues with LLT and GAB.

To gather LLT and GAB information for support analysis

1. If you had changed the default value of the GAB_FFDC_LOGDIR parameter, you must again export the same variable before you run the `getcomms` script.

   See "GAB message logging" on page 169.

2. Run the following command on each node to gather information:

   ```
   # /opt/VRTSgab/getcomms -local
   ```

   Troubleshoot and fix the issue.

   If the issue cannot be fixed, then contact Symantec technical support with the file `/tmp/commslog.time_stamp.tar` that the `getcomms` script generates.

About SF Oracle RAC kernel and driver messages

SF Oracle RAC drivers such as GAB print messages to the console if the kernel and driver messages are configured to be displayed on the console. Make sure that the kernel and driver messages are logged to the console.

For details on how to configure console messages, see the `syslog` and `/etc/syslog.conf` files. For more information, see the operating system documentation.

What to do if you see a licensing reminder

In this release, you can install without a license key. In order to comply with the End User License Agreement, you must either install a license key or make the host managed by a Management Server. If you do not comply with these terms within 60 days, the following warning messages result:

```
WARNING V-365-1-1 This host is not entitled to run Veritas Storage
Foundation/Veritas Cluster Server.As set forth in the End User
License Agreement (EULA) you must complete one of the two options
set forth below. To comply with this condition of the EULA and
stop logging of this message, you have <nn> days to either:
- make this host managed by a Management Server (see
  http://go.symantec.com/sfhakeyless for details and free download),
  or
- add a valid license key matching the functionality in use on this host
  using the command 'vxlicinst'
```

To comply with the terms of the EULA, and remove these messages, you must do one of the following within 60 days:

- Install a valid license key corresponding to the functionality in use on the host. After you install the license key, you must validate the license key using the following command:

  ```
  # vxkeyless display
  ```

- Continue with keyless licensing by managing the server or cluster with a management server.
  For more information about keyless licensing, see the following URL:
  http://go.symantec.com/sfhakeyless

Restarting the installer after a failed connection

If an installation is killed because of a failed connection, you can restart the installer to resume the installation. The installer detects the existing installation. The installer prompts you whether you want to resume the installation. If you resume the installation, the installation proceeds from the point where the installation failed.

Installer cannot create UUID for the cluster

The installer displays the following error message if the installer cannot find the uuidconfig.pl script before it configures the UUID for the cluster:

```
Couldn't find uuidconfig.pl for uuid configuration,
please create uuid manually before start vcs
```

You may see the error message during SF Oracle RAC configuration, upgrade, or when you add a node to the cluster using the installer.

Workaround: To start SF Oracle RAC, you must run the uuidconfig.pl script manually to configure the UUID on each cluster node.

See the *Veritas Cluster Server Administrator's Guide*.

Troubleshooting installation and configuration check failures

This section provides guidelines for resolving failures that may be reported when you run cluster verification checks using the **SF Oracle RAC Installation and Configuration Checks** option with the SF Oracle RAC installation program.

Note: In addition to the documented check failures, the logs may contain more information about the failures. Review the log files at `/opt/VRTS/install/logs`.

Time synchronization checks failed with errors

The time synchronization check may fail with the following error:

```
Checking time synchronization...Failed
```

Cause

Time difference exceeds five seconds.

Resolution

Synchronize the time settings across the nodes.

Note: For Oracle RAC 11g Release 2, it is mandatory to configure NTP for synchronizing time on all nodes in the cluster.

System architecture checks failed with errors

The system architecture check may fail with the following error:

```
Checking system architecture information...Failed
```

You may see the following messages in the logs:

```
Combinations with mismatched system architecture
types: <list of systems>
```

Cause

All the nodes in the cluster do not have the same number of processors or CPU architecture.

Resolution

Make sure that the nodes have the same architecture and the same number and type of processors.

Operating system and patch level synchronization checks failed with errors

The operating system and patch level checks may fail with the following error:

```
Checking OS and patch level synchronization... Failed
```

You may see the following message in the logs:

```
Combinations with mismatched OS and
patch level: <list of systems>
```

Cause

All the nodes in the cluster do not have the same operating system version and patch levels.

Resolution

Install the same version of the operating system and patch levels on all nodes in the cluster.

CPU frequency checks failed with errors

The CPU frequency checks across nodes may fail with the following error:

```
Checking for CPU frequency match...Failed
```

You may see the following message in the logs:

```
Combinations with mismatched frequencies: <list of systems>
```

Cause

The speed of the processors varies across the nodes in the cluster.

Resolution

Make sure that the nodes have the same type of processor.

LLT link mode checks failed with errors

The LLT link check may fail with the following error:

```
Checking LLT Links' Full Duplex setting for
sys_name...Failed
```

You may see the following messages in the logs:

- ```
  Error: /etc/llttab does not exist on sys_name.
  LLT link duplexity check can not proceed for sys_name
     Error: Link link_name is Half-Duplex on sys_name
  ```

Table 3-2 lists the messages displayed for the check.
  ```

Table 3-2 LLT Links' full duplex checks - messages

Error message	Cause	Resolution

`Checking LLT Links' Full Duplex setting for <sys name>...Failed`
 	
`Error: /etc/llttab does not exist on sys_name. LLT link duplexity check can not proceed for sys_name`
`Link link_name is Half-Duplex on sys_name`
	The link mode is not set to full duplex.	Set the link mode to full duplex.
		
		For instructions, see the section "Preparing the private interconnects for LLT" in the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide.*

`Link link_name is not Full-Duplex on sys_name.`
 	
`Link link_name is Half-Duplex on sys_name.`
`Link link_name is UNKNOWN on sys_name.`
 	
`Link link_name is Half-Duplex on sys_name.`
	Sometimes the check can not proceed if the `/etc/llttab` file does not exist on the node. The file may have been accidentally deleted or the SF Oracle RAC configuration is incomplete.	Reconfigure SF Oracle RAC.
		
		For instructions, see the chapter "Configuring SF Oracle RAC" in the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide.*

LLT link jumbo frame setting checks failed with errors

The LLT link check may fail with the following error:

```
Checking LLT Link Jumbo Frame setting (MTU)...Failed
```

You may see the following messages in the logs:

- `Error: /etc/llttab does not exist on sys_name.`
  `LLT link jumbo frame check can not proceed for sys_name`

- `Error: Link link_name has suspicious value of`
  `MTU: frame_size Bytes on sys_name`
  `The NIC: link_name couldn't be probed for Jumbo`
  `Frame setting on sys_name. Skipping.`

Table 3-3 lists the messages displayed for the check.

Table 3-3 LLT link jumbo frame setting (MTU) checks - messages

Error message	Cause	Resolution
`Error: /etc/llttab does not exist on sys_name. LLT link jumbo frame check can not proceed for sys_name`	Sometimes the check can not proceed if the `/etc/llttab` file does not exist on the node. The file may have been accidentally deleted or the SF Oracle RAC configuration is incomplete.	Reconfigure SF Oracle RAC. For instructions, see the chapter "Configuring SF Oracle RAC" in the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide*.
`Error: Link link_name has suspicious value of MTU: frame_size Bytes on sys_name` `The NIC: link_name couldn't be probed for Jumbo Frame setting on sys_name. Skipping.`	The MTU size exceeds the acceptable range of 1500 to 9000 bytes.	Set the MTU size within the acceptable range. The instructions vary from device to device. For instructions, see the device documentation.

Cross connection checks failed with errors

The cross connection checks may fail with the following error:

```
Cross connection checks...Failed
```

Cause

The nodes in the cluster are not connected using similar network devices. For example, if you use eth1 as the first private link on one node in the cluster, all other nodes must also use eth1 as the first private link.

Resolution

Verify that the nodes in the cluster are connected with network links using similar network devices.

LLT link speed and auto negotiation checks failed with errors

The LLT link checks may fail with the following error:

```
Checking LLT Links' speed and auto negotiation
settings for sys_name...Failed
```

You may see the following message in the logs:

```
Error: /etc/llttab does not exist on sys_name.
LLT Private Link check can not proceed for sys_name.
```

Table 3-4 lists the messages displayed for the check.

Table 3-4 LLT links' speed and auto negotiation checks - messages

Error message	Cause	Resolution
`Checking LLT Links' speed and auto negotiation settings for sys_name... Failed`	The check fails if the media speed settings vary across the nodes and if auto-negotiation is enabled for the interface on any node in the cluster.	Set the media speed to the same value across all the nodes and disable auto-negotiation. For instructions, see the section "Preparing the private interconnects for LLT" in the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide*.
`Error: /etc/llttab does not exist on sys_name. LLT Private Link check can not proceed for sys_name.`	The SF Oracle RAC configuration has failed or is incomplete.	Reconfigure SF Oracle RAC. For instructions, see the chapter "Configuring SF Oracle RAC" in the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide*.

LLT link priority checks failed with errors

The LLT link checks may fail with the following error:

```
Checking LLT links configured on private link
and as high priority on sys_name...Failed
```

You may see the following messages in the logs:

- `Error: There are only no_of_links High Priority links present on sys_name (less than 2)`
- `Error: At least one of the LLT links is on public network on sys_name.`

Table 3-5 lists the messages displayed for the check.

Table 3-5 LLT link priority checks - Messages

Error message	Cause	Resolution
`Error: There are only` *`no_of_links`* `High Priority links present on` *`sys_name`* `(less than 2)`	The number of high priority LLT links in the cluster are less than the two.	Configure at least two high-priority links. For instructions on adding high-priority LLT links:
`Error: At least one of the LLT links is on public network on` *`sys_name`*`.`	One of the LLT links is configured on a public network.	All the high-priority LLT links must be configured on private networks. Remove the high-priority LLT link configured on the public network. For instructions on removing the high-priority link configured on the public network:

Checks on /etc/llthosts file failed with errors

The check on /etc/llthosts file may fail with the following error:

```
Checking for /etc/llthosts file match...Failed
```

You may see the following message in the logs:

```
Error: /etc/llthosts different on sys_name1 & sys_name2.
```

Cause

Check for one of the following causes:

- The nodes listed in the `/etc/llthosts` file varies across nodes.
- The `/etc/llthosts` file has been incorrectly modified manually.
- The cluster has been reconfigured with a different set of nodes that contain the same cluster ID but different nodes.
- A node has been removed from the cluster but the `/etc/llthosts` file has not been updated.

Resolution

Update the `/etc/llthosts` file to correctly reflect the number of nodes in the cluster. Restart the cluster.

Checks on cluster ID failed with errors

The cluster ID checks may fail with the following error:

```
Checking for cluster-ID match...Failed
```

You may see the following messages in the logs:

- `Error: /etc/llttab file is not present on some nodes, while it is present on other nodes.`
- `Error: /etc/llttab  file is not present on any node. Skipping cluster-ID match check.`
- `Error: Cluster-ID differences between: <list of nodes>.`

Table 3-6 lists the messages displayed for the check.

Table 3-6 Checking cluster-ID match - messages

Error message	Cause	Resolution
`Error: /etc/llttab file is not present on some nodes, while it is present on other nodes.`	Some of the nodes in the cluster are reconfigured with different cluster ID.	Update the `/etc/llttab` file to correctly reflect the cluster IDs of the nodes in the cluster. Restart the cluster.
`Error: /etc/llttab  file is not present on any node. Skipping cluster-ID match check.`	The LLT configuration has failed or is incomplete.	Reconfigure SF Oracle RAC. For instructions, see the chapter "Configuring SF Oracle RAC" in the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide*.
`Error: Cluster-ID differences between: <list of nodes>.`	The `/etc/llttab` file has been incorrectly modified manually.	Update the `/etc/llttab` file to correctly reflect the cluster IDs of the nodes in the cluster. Restart the cluster.

Fencing configuration checks failed with errors

This check is valid only for disk-based I/O fencing.

The fencing configuration checks may fail with the following error:

`Checking Fencing configuration on sys_name...Failed`

You may see the following messages in the logs:

- `Error: Symantec I/O Fencing module is not configured on node: sys_name`
- `Error: Failed (Configured in disabled mode)`
  `Configure with diskbased fencing or CPS.`

Table 3-7 lists the messages displayed for the check.

Table 3-7 Checking fencing configuration - messages

Error message	Cause	Resolution
`Symantec I/O Fencing module is not configured on node: sys_name`	Fencing is not configured on the cluster. Fencing is configured using CPS.	Configure I/O fencing. For instructions to configure I/O fencing, see the chapter "Configuring SF Oracle RAC" in the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide.* Ignore the check failure if server-based fencing is configured.
`Failed (Configured in disabled mode)`	Fencing is configured in disabled mode.	Reconfigure disk-based fencing. For instructions, see the chapter "Configuring SF Oracle RAC clusters for data integrity" in the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide.*

Checks on the presence of fencing keys on coordinator disks failed with errors

The fencing configuration checks may fail with the following error:

```
Checking the presence of fencing keys on coordinator disks
on sys_name...Failed
```

Cause

The SCSI-3 PGR keys are not present on the coordinator disks for some of the nodes in the cluster.

Resolution

Refresh the lost keys on the coordinator disks as described in the following procedure.

To refresh the lost keys on the coordinator disks

1 Determine the node from which the keys are missing. Check the output. The output will not display the nodes from which keys are missing.

```
# vxfenadm -s all -f /etc/vxfentab
```

2 Refresh the lost keys.

See "Refreshing lost keys on coordinator disks" on page 129.

ODM configuration checks failed with errors

The ODM configuration check may fail with the following error:

```
Checking ODM configuration on sys_name...Failed
```

Cause

Check for one of the following causes:

- ODM is not running in the cluster.
- ODM is running in standalone mode.

Resolution

To resolve the issue

1. Start ODM:
2. Run the following command to verify whether ODM is running in clustered or standalone mode:

```
# cat /dev/odm/cluster
cluster status: enabled
```

If the status is enabled, ODM is running in clustered mode; any other status indicates that ODM is running in standalone mode.

If ODM is running in standalone mode, change the mode as follows:

```
# /etc/init.d/vxgms start
# /etc/init.d/vxodm restart
```

VCSMM configuration checks failed with errors

The VCSMM configuration checks may fail with the following error:

```
Checking VCSMM configuration on sys_name...Failed
```

Cause

Port o is not up. VCSMM is not running.

Resolution

Start VCSMM.

GAB ports configuration checks failed with errors

The GAB ports configuration checks may fail with the following error:

```
Checking GAB ports configuration on sys_name...Failed
```

You may see the following message in the logs:

```
GAB is not configured on node: sys_name
```

Table 3-8 lists the messages displayed for the check.

Table 3-8 Checking GAB ports configuration - messages

Cause	Resolution
LLT is not set up correctly.	Verify that the LLT checks have completed successfully. If not, resolve the issues using the information in the following section: See "Troubleshooting installation and configuration check failures" on page 172.
GAB has become unregistered because LLT is not set up correctly.	GAB has become unregistered because LLT is not set up correctly. Verify that the configuration is correct in the `/etc/llttab` file. If the LLT configuration is incorrect, make the appropriate changes and reboot.
SF Oracle RAC configuration has failed or is incomplete.	Reconfigure SF Oracle RAC. For instructions, see the chapter "Configuring SF Oracle RAC" in the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide*.

Oracle database version checks failed with errors

The Oracle database checks may fail with the following error:

```
Checking Oracle DB level...Failed
```

You may see the following message in the logs:

```
Oracle DB level different on: sys_name1 (version1)
and sys_name2 (version2).

Problem running 'opatch' utility on sys_name.
Error info: error_info
```

Table 3-9 lists the messages displayed for the check.

Table 3-9 Oracle database version checks - messages

Error message	Cause	Resolution
`Oracle DB level different on: sys_name1 version1 and sys_name2 version2.`	The Oracle database version or patchset information varies across nodes.	Install the same version of the database and patchsets across the nodes in the cluster.

Table 3-9 Oracle database version checks - messages *(continued)*

Error message	Cause	Resolution

`Problem running 'opatch' utility on sys_name. Error info: error_info`	The permissions are not set correctly on the oraInventory directory.	Run the `orainstRoot.sh` script to set the permissions correctly.
		`# cd $ORACLE_BASE/\ oraInventory`
		`# ./orainstRoot.sh`

Checks on vendor clusterware failed with errors

The init.cssd patch checks may fail with the following error:

```
Checking vendor clusterware on sys_name...Failed
```

Cause

This check fails if Oracle Clusterware/Grid Infrastructure fails to detect the presence of Veritas Cluster Server. The SF Oracle RAC installation has failed or is incomplete.

Resolution

Relink the Oracle libraries with SF Oracle RAC.

For instructions, see the chapter "Performing Oracle RAC post-installation tasks" in the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide*.

Oracle Process Daemon (oprocd) checks failed with errors

The Oracle process daemon (oprocd) check may fail with the following error:

```
Checking presence of Oracle Process Daemon (oprocd)
on sys_name...Failed
```

Table 3-10 lists the messages displayed for the check.

Table 3-10 Oracle Process Daemon checks - causes and resolutions

Cause	Resolution
The Oracle process daemon (oprocd) is running on the system.	Verify that *$CRS_HOME/lib/libskgxn2.so* is a symbolic link to */opt/ORCLcluster/lib/libskgxn2.so*. Verify that */opt/ORCLcluster/lib/libskgxn2.so* is a symbolic link to */opt/VRTSvcs/rac/lib/libskgxn2_64.so*.
The SF Oracle RAC installation did not complete successfully.	Verify the installer logs at */opt/VRTS/install/logs*.

Node ID checks for LLT and Oracle Clusterware failed with errors

The node ID checks may fail with the following error:

```
Checking Node id mismatch for LLT & Oracle
Clusterware on sys_name...Failed
```

You may see the following messages in the logs:

- ```
  Error: Failure in running olsnodes
  command on sys_name. Node ID match check can not
  proceed.
  ```

- ```
  Error: There is no node <nodename>
  present in Oracle Clusterware membership, which is listed
  in /etc/llthosts.
  ```

- ```
  Error: Node numbering of LLT and Oracle
  Clusterware is different. This step is mandatory for SF Oracle RAC
  to function.
  ```

- ```
  Error: /etc/llthosts file is missing on
  sys_name. Node ID match check can not proceed. Skipping.
  ```

Table 3-11 lists the messages displayed for the check.

Table 3-11 Node ID checks for LLT and Oracle Clusterware - messages

Error message	Cause	Resolution
`Error: Failure in running olsnodes command on sys_name. Node ID match check can not proceed.`	The Oracle libskgxn2.so library is not linked with the VCSMM library.	Relink the Oracle libskgxn2 library with the VCSMM library. For instructions, see the section "Completing Oracle RAC post-installation tasks" in the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide.*

`Error: There is no node node_name present in Oracle Clusterware membership, which is listed in /etc/llthosts.`	Oracle Clusterware is installed only on a subset of the nodes in the SF Oracle RAC cluster.	Run the checks on the subset of nodes where Oracle Clusterware is installed.
		`# cd /opt/VRTS/install`
		`# ./installsfrac -configure \`
		`node_names`

`Error: Node numbering of LLT and Oracle Clusterware is different.`	The check fails if the LLT configuration has been incorrectly modified after Oracle RAC installation.	Make sure that the node ID information in the LLT configuration file corresponds with that of Oracle Clusterware.
		Check the node information displayed by the following command:
		`# lltstat -nvv|more`
		Create the file and populate it in the following format using the information displayed by the lltstat command:
		`node_id node name`

Table 3-11 Node ID checks for LLT and Oracle Clusterware - messages *(continued)*

Error message	Cause	Resolution

`Error: /etc/llthosts file is missing on sys_name. Node ID match check can not proceed. Skipping.`	The SF Oracle RAC file is accidentally removed or the installation is incomplete.	If the installation is incomplete or failed, reinstall SF Oracle RAC.
		For instructions, see the chapter "Installing SF Oracle RAC" in the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide.*
		Create the /etc/llthosts file and populate it as follows:
		Check the node information displayed by the following command:
		`# lltstat -nvv|more`
		Create the file and populate it in the following format using the information displayed by the lltstat command:
		`node_id node name`

Checks on tunables for LLT/VCSMM failed with errors

The checks on LLT and VCSMM tunables may fail with the following error:

```
Checking Tunables for LLT/VCSMM...Failed
```

Cause

The tunables for the module may not be set correctly.

Resolution

Set the correct tunable settings.

For more information:

Oracle user checks failed with errors

The permission checks on the Oracle user may fail with the following error:

```
Checking permissions for ORACLE_USER:oracle_user
on sys_name...Failed
```

You may see the following messages in the logs:

- `Error:  No Oracle user detected. Skipping check for oracle user permission for /etc/llthosts and /etc/lltab files.`
- `Error: Either $vcs->{llthosts} or $vcs->{lltab} or both are not available on sys_name. Skipping this check.`
- `Error: Permission denied for ORACLE_USER:$ora_user for files $vcs->{llthosts} and $vcs->{lltab} on sys_name`

Table 3-12 lists the messages displayed for the check.

Table 3-12 Checking permissions for ORACLE_USER - messages

Error message	Cause	Resolution
`No Oracle user detected. Skipping check for oracle user permission for /etc/llthosts and /etc/lltab files.`	The specified Oracle user does not exist.	Specify the correct Oracle user and rerun the checks.
`Either $vcs->{llthosts} or $vcs->{lltab} or both are not available on sys_name. Skipping this check.`	Either one or both of the /etc/llthosts file or the /etc/lltab file are missing. SF Oracle RAC is not configured properly.	Reconfigure SF Oracle RAC. For instructions, see the chapter "Configuring SF Oracle RAC" in the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide*.

`Permission denied for ORACLE_USER:oracle_user for files $vcs->{llthosts} and $vcs->{lltab} on sys_name`	The files do not have read permissions for "other" users.	Grant the read permission for the files to "other" users.
		`# chmod o+r /etc/lltab`
		`# chmod o+r /etc/llthosts`

CSSD resource checks failed with errors

The cssd check fails with the following error:

```
Checking whether cssd resource exists in
CVM service group...Failed
```

Cause

The CSSD resource is not configured under VCS.

Resolution

Configure the CSSD resource under VCS.

For instructions, see the section "Completing Oracle RAC post-installation tasks" in the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide*.

Checks on equivalence of SF Oracle RAC cluster and RAC instances failed with errors

The equivalence checks may fail with the following error:

```
Checking equivalence of SF Oracle RAC cluster
and RAC instances from system_name...Failed
```

You may see the following messages in the logs:

- `Error:  No Oracle user detected. Skipping checking if number of nodes configured under SF Oracle RAC and those under RAC are same.`
- `Error: Either $vcs->{llthosts} or $vcs->{llttab} or both are not available on sys_name. Skipping this check.`
- `Error: Oracle user: $prod->{orauser} does not have permission to access $vcs->{llthosts} and $vcs->{llttab}.`

Table 3-13 lists the messages displayed for the check.

Table 3-13 Checks on equivalence of SF Oracle RAC cluster and RAC instances - messages

Error message	Causes	Resolutions
`No Oracle user detected. Skipping checking if number of nodes configured under SF Oracle RAC and those under RAC are same.`	The specified Oracle user does not exist.	Specify the correct Oracle user and rerun the checks.
`Either $vcs->{llthosts} or $vcs->{llttab} or both are not available on sys_name. Skipping this check.`	Either one or both of the /etc/llthosts file or the /etc/llttab file are missing. SF Oracle RAC is not configured properly.	Reconfigure SF Oracle RAC. For instructions, see the chapter "Configuring SF Oracle RAC" in the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide.*

`Oracle user: $prod->{orauser} does not have permission to access $vcs->{llthosts} and $vcs->{llttab}.`	The files do not have read permissions for "other" users.	Grant the read permission for the files to "other" users.
		`# chmod o+r /etc/llttab`
		`# chmod o+r /etc/llthosts`

Kernel parameter checks failed with errors

The kernel parameter checks may fail with the following error:

`Checking Kernel Parameters on system_name...Failed`

Cause

The kernel parameter settings are incorrect.

Resolution

Check the logs at `/opt/VRTS/install/logs` to determine which kernel messages have failed. See the Oracle RAC documentation for instructions on setting the kernel parameters correctly.

Troubleshooting pre-installation check failures

Table 3-14 provides guidelines for resolving failures that may be reported when you start the SF Oracle RAC installation program.

Table 3-14 Troubleshooting pre-installation check failures

Error message	Cause	Resolution
Checking system communication...Failed	Passwordless SSH or RSH communication is not set up between the systems in the cluster.	Set up passwordless SSH or RSH communication between the systems in the cluster. For instructions, see the section "Setting up inter-system communication" in the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide*.
Checking release compatibility...Failed	The current system architecture or operating system version is not supported. For example, 32-bit architectures are not supported in this release.	Make sure that the systems meet the hardware and software criteria required for the release. For information, see the chapter "Requirements and planning" in the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide*.
Checking installed product	For information on the messages displayed for this check and the appropriate resolutions: See Table 3-15 on page 192.	For information on the messages displayed for this check and the appropriate resolutions: See Table 3-15 on page 192.
Checking platform version	The version of the operating system installed on the system is not supported.	Make sure that the systems meet the software criteria required for the release. For information, see the chapter "Requirements and planning" in the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide*.

Table 3-14 Troubleshooting pre-installation check failures *(continued)*

Error message	Cause	Resolution
Performing product prechecks	One of the following checks failed: ■ Time synchronization ■ CPU speed checks ■ System architecture checks ■ OS patch level checks	For information on resolving these issues: See "Troubleshooting installation and configuration check failures" on page 172.

Table 3-15 Checking installed product - messages

Message	Resolution
`Entered systems have different products installed: prod_name-prod_ver-sys_name Systems running different products must be upgraded independently.`	Two or more systems specified have different products installed. For example, SFCFS is installed on some systems while SF Oracle RAC is installed on other systems. You need to upgrade the systems separately. For instructions on upgrading SF Oracle RAC, see the chapter "Upgrading SF Oracle RAC" in the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide*. For instructions on upgrading other installed products, see the appropriate product documentation.
`Entered systems have different versions of $cprod_name installed: prod_name-prod_ver-sys_name. Systems running different product versions must be upgraded independently.`	Two or more systems specified have different versions of SF Oracle RAC installed. For example, SF Oracle RAC 5.0 is installed on some systems while SF Oracle RAC 5.0 MP3 is installed on other systems. You need to upgrade the systems separately. For instructions on upgrading SF Oracle RAC, see the chapter "Upgrading SF Oracle RAC" in the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide*.

Table 3-15 Checking installed product - messages *(continued)*

Message	Resolution
prod_name is installed. Upgrading *prod_name* directly to *another_prod_name* is not supported.	A product other than SF Oracle RAC is installed. For example, SFCFS 5.0 MP3 is installed on the systems. The SF Oracle RAC installer does not support direct upgrade from other products to SF Oracle RAC. You need to first upgrade to version 5.1 Service Pack 1 of the installed product, then upgrade to SF Oracle RAC 5.1 SP1. For instructions on upgrading to SF Oracle RAC, see the section "Upgrading from Storage Foundation products to SF Oracle RAC" in the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide*.
prod_name is installed. *component_prod_name* is part of *prod_name*. Upgrading only *component_prod_name version* may partially upgrade the installed RAC stack.	If a previous version of SF Oracle RAC is installed, the installer supports partial product upgrade. For example, you can upgrade VCS in the stack to version 5.1 Service Pack 1. If you want to upgrade the complete SF Oracle RAC stack later, you can run the `installsfrac` program.

Troubleshooting LLT health check warning messages

Table 3-16 lists the warning messages displayed during the health check and corresponding recommendations for resolving the issues.

Table 3-16 Troubleshooting LLT warning messages

Warning	Possible causes	Recommendation
`Warning: OS timer is not called for` *num* `seconds`	CPU and memory consumption on the node is high.	Check for applications that may be throttling the CPU and memory resources on the node.
`Warning: Kernel failed to allocate memory` *num* `time(s)`	The available memory on the node is insufficient.	Reduce memory consumption on the node and free up allocated memory.

Table 3-16 Troubleshooting LLT warning messages *(continued)*

Warning	Possible causes	Recommendation
Flow-control occurred *num* time(s)and back-enabled *num* time(s) on port *port number* for node *node number*	■ The network bandwidth between the local node and the peer node (*node number*) is insufficient. ■ The CPU and memory consumption on the peer node (*node number*) is very high.	■ Allocate more bandwidth for communication between the local node and the peer node. ■ Check for applications that may be throttling the CPU and memory resources on the node.
`Warning: Connectivity with node node id on link link id is flaky num time(s).` `The distribution is: (0-4 s) <num> (4-8 s) <num> (8-12 s) <num> (12-16 s) <num> (>=16 s)`	The private interconnect between the local node and the peer node is unstable.	Check the connectivity between the local node and the peer node. Replace the link, if needed.
`One or more link connectivity with peer node(s) node name is in trouble.`	The private interconnects between the local node and peer node may not have sufficient bandwidth.	Check the private interconnects between the local node and the peer node.
`Link connectivity with node id is on only one link. Symantec recommends configuring a minimum of 2 links.`	■ One of the configured private interconnects is non-operational. ■ Only one private interconnect has been configured under LLT.	■ If the private interconnect is faulty, replace the link. ■ Configure a minimum of two private interconnects.
`Only one link is configured under LLT. Symantec recommends configuring a minimum of 2 links.`	■ One of the configured private interconnects is non-operational. ■ Only one private interconnect has been configured under LLT.	■ If the private interconnect is faulty, replace the link. ■ Configure a minimum of two private interconnects.

Table 3-16 Troubleshooting LLT warning messages *(continued)*

Warning	Possible causes	Recommendation

`Retransmitted % percentage of total transmitted packets.`
`Sent % percentage of total transmitted packet when no link is up.`
`% percentage of total received packets are with bad checksum.`
`% percentage of total received packets are out of window.`
`% percentage of total received packets are misaligned.`
	■ The network interface card or the network links are faulty.	■ Verify the network connectivity between nodes.
		Check the network interface card for anomalies.

`% percentage of total received packets are with DLPI error.`	The DLPI driver or NIC may be faulty or corrupted.	Check the DLPI driver and NIC for anomalies.
`% per of total transmitted packets are with large xmit latency (>16ms) for port port id` `%per received packets are with large recv latency (>16ms) for port port id.`	The CPU and memory consumption on the node may be high or the network bandwidth is insufficient.	■ Check for applications that may be throttling CPU and memory usage. ■ Make sure that the network bandwidth is adequate for facilitating low-latency data transmission.

Table 3-16 Troubleshooting LLT warning messages *(continued)*

Warning	Possible causes	Recommendation
`LLT is not running.`	SF Oracle RAC is not configured properly.	Reconfigure SF Oracle RAC. For instructions, see the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide*.

Troubleshooting I/O fencing

The following sections discuss troubleshooting the I/O fencing problems. Review the symptoms and recommended solutions.

SCSI reservation errors during bootup

When restarting a node of an SF Oracle RAC cluster, SCSI reservation errors may be observed such as:

```
date system name kernel: scsi3 (0,0,6) : RESERVATION CONFLICT
```

This message is printed for each disk that is a member of any shared disk group which is protected by SCSI-3 PR I/O fencing. This message may be safely ignored.

The vxfentsthdw utility fails when SCSI TEST UNIT READY command fails

While running the vxfentsthdw utility, you may see a message that resembles as follows:

```
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 the Hitachi Data Systems 99XX arrays if bit 186 of the system mode option is not enabled.

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:

```
...VxFEN ERROR V-11-1-25 ... Unable to join running cluster
since cluster is currently fencing
a node out of the cluster.
```

If you see these messages when the new node is booting, the vxfen startup script on the node makes up to five attempts to join the cluster.

To manually join the node to the cluster when I/O fencing attempts fail

- If the vxfen script fails in the attempts to allow the node to join the cluster, restart vxfen driver with the command:

  ```
  # /etc/init.d/vxfen stop
  # /etc/init.d/vxfen start
  ```

  If the command fails, restart the new node.

System panics to prevent potential data corruption

When a node 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.
```

See "How vxfen driver checks for preexisting split-brain condition" on page 197.

How vxfen driver checks for preexisting 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 restarts 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 restarts, 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.

To resolve actual and apparent potential split-brain conditions

- Depending on the split-brain condition that you encountered, do the following:

Actual potential split-brain condition—system 2 is up and system 1 is ejected	1 Determine if system1 is up or not. 2 If system 1 is up and running, shut it down and repair the private network links to remove the split-brain condition. 3 Restart system 1.

Apparent potential split-brain condition—system 2 is down and system 1 is ejected	1 Physically verify that system 2 is down. Verify the systems currently registered with the coordinator disks. Use the following command: `# vxfenadm -s all -f /etc/vxfentab` The output of this command identifies the keys registered with the coordinator disks. 2 Clear the keys on the coordinator disks as well as the data disks using the `vxfenclearpre` command. See "Clearing keys after split-brain using vxfenclearpre command" on page 200. 3 Make any necessary repairs to system 2. 4 Restart system 2.

Cluster ID on the I/O fencing key of coordinator disk does not match the local cluster's ID

If you accidentally assign coordinator disks of a cluster to another cluster, then the fencing driver displays an error message similar to the following when you start I/O fencing:

```
000068 06:37:33 2bdd5845  0 ... 3066 0 VXFEN WARNING V-11-1-56
 Coordinator disk has key with cluster id 48813
 which does not match local cluster id 57069
```

The warning implies that the local cluster with the cluster ID 57069 has keys. However, the disk also has keys for cluster with ID 48813 which indicates that nodes from the cluster with cluster id 48813 potentially use the same coordinator disk.

You can run the following commands to verify whether these disks are used by another cluster. Run the following commands on one of the nodes in the local cluster. For example, on galaxy:

```
galaxy> # lltstat -C
57069

galaxy> # cat /etc/vxfentab
/dev/vx/rdmp/disk_7
/dev/vx/rdmp/disk_8
```

```
/dev/vx/rdmp/disk_9

galaxy> # vxfenadm -s /dev/vx/rdmp/disk_7
Reading SCSI Registration Keys...
Device Name: /dev/vx/rdmp/disk_7
Total Number Of Keys: 1
key[0]:
 [Numeric Format]:  86,70,48,49,52,66,48,48
 [Character Format]: VFBEAD00
 [Node Format]: Cluster ID: 48813 Node ID: 0 Node Name: unknown
```

Where *disk_7*, *disk_8*, and *disk_9* represent the disk names in your setup.

Recommended action: You must use a unique set of coordinator disks for each cluster. If the other cluster does not use these coordinator disks, then clear the keys using the vxfenclearpre command before you use them as coordinator disks in the local cluster.

See "About the vxfenclearpre utility" on page 117.

Clearing keys after split-brain using vxfenclearpre command

If you have encountered a preexisting 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.

See "About the vxfenclearpre utility" on page 117.

Registered keys are lost on the coordinator disks

If the coordinator disks lose the keys that are registered, the cluster might panic when a cluster reconfiguration occurs.

To refresh the missing keys

- Use the vxfenswap utility to replace the coordinator disks with the same disks. The vxfenswap utility registers the missing keys during the disk replacement.

  See "Refreshing lost keys on coordinator disks" on page 129.

Replacing defective disks when the cluster is offline

If the disk becomes defective or inoperable and you want to switch to a new diskgroup in a cluster that is offline, then perform the following procedure.

In a cluster that is online, you can replace the disks using the vxfenswap utility.

See "About the vxfenswap utility" on page 120.

Review the following information to replace coordinator disk in the coordinator disk group, or to destroy a coordinator disk group.

Note the following about the procedure:

- When you add a disk, add the disk to the disk group vxfencoorddg 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 replace a disk in the coordinator disk group when the cluster is offline

1. Log in as superuser on one of the cluster nodes.
2. If VCS is running, shut it down:

   ```
   # hastop -all
   ```

   Make sure that the port h is closed on all the nodes. Run the following command to verify that the port h is closed:

   ```
   # gabconfig -a
   ```

3. Stop the VCSMM driver on each node:

   ```
   # /etc/init.d/vcsmm stop
   ```

4. Stop I/O fencing on each node:

   ```
   # /etc/init.d/vxfen stop
   ```

   This removes any registration keys on the disks.

5. Import the coordinator disk group. The file /etc/vxfendg includes the name of the disk group (typically, 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 node 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 locks are removed.

6 To remove disks from the disk group, use the VxVM disk administrator utility, vxdiskadm.

You may also destroy the existing coordinator disk group. For example:

- Verify whether the coordinator attribute is set to on.

```
# vxdg list vxfencoorddg |  grep flags: | grep coordinator
```

- Destroy the coordinator disk group.

```
# vxdg -o coordinator destroy vxfencoorddg
```

7 Add the new disk to the node and initialize it as a VxVM disk.

Then, add the new disk to the vxfencoorddg disk group:

- If you destroyed the disk group in step 6, then create the disk group again and add the new disk to it.
  See the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide* for detailed instructions.
- If the diskgroup already exists, then add the new disk to it.

```
# vxdg -g vxfencoorddg -o coordinator adddisk disk_name
```

8 Test the recreated disk group for SCSI-3 persistent reservations compliance.

See "Testing the coordinator disk group using vxfentsthdw -c option" on page 107.

9 After replacing disks in a coordinator disk group, deport the disk group:

```
# vxdg deport `cat /etc/vxfendg`
```

10 On each node, start the I/O fencing driver:

```
# /etc/init.d/vxfen start
```

11 On each node, start the VCSMM driver:

```
# /etc/init.d/vcsmm start
```

12 Verify that the I/O fencing module has started and is enabled.

```
# gabconfig -a
```

Make sure that port b and port o memberships exist in the output for all nodes in the cluster.

```
# vxfenadm -d
```

Make sure that I/O fencing mode is not disabled in the output.

13 If necessary, restart VCS on each node:

```
# hastart
```

Troubleshooting I/O fencing health check warning messages

Table 3-17 lists the warning messages displayed during the health check and corresponding recommendations for resolving the issues.

Table 3-17 Troubleshooting I/O fencing warning messages

Warning	Possible causes	Recommendation
`VxFEN is not running on all nodes in the cluster.`	I/O fencing is not enabled in the cluster.	Configure fencing. For instructions, see the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide.*
`VxFEN is not configured in SCSI3 mode.`	■ If you are using disk-based I/O fencing, it is not configured in the SCSI3 mode.	Start fencing in SCSI3 mode on all nodes in the cluster.

Table 3-17 Troubleshooting I/O fencing warning messages *(continued)*

Warning	Possible causes	Recommendation

`VxFEN is running with only one coordinator disk. Loss of this disk will prevent cluster reconfiguration on loss of a node. Symantec recommends configuring a minimum of 3 coordinator disks.`
 	
`VxFEN is running with even number of coordinator disks. There must be odd number of coordinator disks.`
 	
`Replace the disk` *`disk name`* `using OCDR procedure.`
	■ I/O fencing is enabled with only one coordinator disk.
	■ The number of disks configured for I/O fencing are even.
		■ Configure a minimum of three coordinator disks.
		■ Add or remove coordinators disks to meet the odd coordinator disk criterion. Use the `vxfenswap` utility to add or remove disks.
		See "About the vxfenswap utility" on page 120.
		■ If the coordinator disk is faulty, replace the disk using the `vxfenswap` utility.
		See "About the vxfenswap utility" on page 120.

`The coordinator disk (`*`disk name`*`) does not have the required key for the local node.`	The key may have been accidentally removed.	Refresh the key on the coordinator disk. For instructions: See "Refreshing lost keys on coordinator disks" on page 129.

Table 3-17 Troubleshooting I/O fencing warning messages *(continued)*

Warning	Possible causes	Recommendation

`SCSI3 write-exclusive reservation is missing on shared disk (disk_name)`	The SCSI3 reservation is accidentally removed.	Register and reserve the key again.
		
		To register and reserve the key
		
		1 Determine the key name by checking the keys on the other shared disks on that node:
		
		`# vxfenadm -g`
		
		2 Register the key:
		
		`# vxfenadm -m -k \ key_name`
		
		3 Reserve the key:
		
		`# vxfenadm -n -k \ key_name`
		
		4 Verify that the reservation is set to write-exclusive mode:
		
		`# vxfenadm -R`

Troubleshooting server-based I/O fencing

All CP server operations and messages are logged in the /var/VRTScps/log directory in a detailed and easy to read format. The entries are sorted by date and time. The logs can be used for troubleshooting purposes or to review for any possible security issue on the system that hosts the CP server.

The following files contain logs and text files that may be useful in understanding and troubleshooting a CP server:

- /var/VRTScps/log/cpserver_[ABC].log
- /var/VRTSat/vrtsat_broker.txt (Security related)

- If the vxcpserv process fails on the CP server, then review the following diagnostic files:
  - /var/VRTScps/diag/FFDC_CPS_*pid*_vxcpserv.log
  - /var/VRTScps/diag/stack_*pid*_vxcpserv.txt

Note: If the vxcpserv process fails on the CP server, these files are present in addition to a core file. VCS restarts vxcpserv process automatically in such situations.

The file /var/VRTSvcs/log/vxfen/vxfend_[ABC].log contains logs and text files that may be useful in understanding and troubleshooting fencing-related issues on a SF Oracle RAC cluster (client cluster) node.

See "Troubleshooting issues related to the CP server service group" on page 206.

See "Checking the connectivity of CP server" on page 206.

See "Issues during fencing startup on SF Oracle RAC cluster nodes set up for server-based fencing" on page 207.

See "Issues during online migration of coordination points" on page 209.

See "Troubleshooting server-based I/O fencing in mixed mode" on page 210.

See "Checking keys on coordination points when vxfen_mechanism value is set to cps" on page 214.

Troubleshooting issues related to the CP server service group

If you cannot bring up the CPSSG service group after the CP server configuration, perform the following steps:

- Verify that the CPSSG service group and its resources are valid and properly configured in the VCS configuration.
- Check the VCS engine log (`/var/VRTSvcs/log/engine_[ABC].log`) to see if any of the CPSSG service group resources are FAULTED.
- Review the sample dependency graphs to make sure the required resources are configured correctly.

Checking the connectivity of CP server

You can test the connectivity of CP server using the `cpsadm` command.

You must have set the environment variables CPS_USERNAME and CPS_DOMAINTYPE to run the `cpsadm` command on the SF Oracle RAC cluster (client cluster) nodes.

To check the connectivity of CP server

- Run the following command to check whether a CP server is up and running at a process level:

  ```
  # cpsadm -s cp_server -a ping_cps
  ```

  where *cp_server* is the virtual IP address or virtual hostname on which the CP server is listening.

Troubleshooting server-based fencing on the SF Oracle RAC cluster nodes

The file /var/VRTSvcs/log/vxfen/vxfend_[ABC].log contains logs and text files that may be useful in understanding and troubleshooting fencing-related issues on a SF Oracle RAC cluster (client cluster) node.

Issues during fencing startup on SF Oracle RAC cluster nodes set up for server-based fencing

Table 3-18 Fencing startup issues on SF Oracle RAC cluster (client cluster) nodes

Issue	Description and resolution
`cpsadm` command on the SF Oracle RAC cluster gives connection error	If you receive a connection error message after issuing the `cpsadm` command on the SF Oracle RAC cluster, perform the following actions: ■ Ensure that the CP server is reachable from all the SF Oracle RAC cluster nodes. ■ Check that the SF Oracle RAC cluster nodes use the correct CP server virtual IP or virtual hostname and the correct port number. Check the /etc/vxfenmode file. ■ Ensure that the running CP server is using the same virtual IP/virtual hostname and port number.
Authorization failure	Authorization failure occurs when the CP server's nodes or users are not added in the CP server configuration. Therefore, fencing on the SF Oracle RAC cluster (client cluster) node is not allowed to access the CP server and register itself on the CP server. Fencing fails to come up if it fails to register with a majority of the coordination points. To resolve this issue, add the CP server node and user in the CP server configuration and restart fencing.

Table 3-18 Fencing startup issues on SF Oracle RAC cluster (client cluster) nodes *(continued)*

Issue	Description and resolution
Authentication failure	If you had configured secure communication between the CP server and the SF Oracle RAC cluster (client cluster) nodes, authentication failure can occur due to the following causes: ■ Symantec Product Authentication Services (AT) is not properly configured on the CP server and/or the SF Oracle RAC cluster. ■ The CP server and the SF Oracle RAC cluster nodes use the same root broker but the certificate hash of the root broker is not same on the SF Oracle RAC cluster and the CP server. Run the following command on both the CP server and the SF Oracle RAC cluster to see the certificate hash: `# cpsat showalltrustedcreds` ■ The CP server and the SF Oracle RAC cluster nodes use different root brokers, and trust is not established between the authentication brokers: See "About secure communication between the SF Oracle RAC cluster and CP server" on page 65. ■ The hostname of the SF Oracle RAC cluster nodes is not the same hostname used when configuring AT. The hostname of the SF Oracle RAC cluster nodes must be set to the hostname used when configuring AT. You can view the fully qualified hostname registered with AT using the `cpsat showcred` command. After entering this command, the hostname appears in the User Name field. ■ The CP server and SF Oracle RAC cluster do not have the same security setting. In order to configure secure communication, both the CP server and the SF Oracle RAC cluster must have same security setting. In order to have the same security setting, the security parameter must have same value in the `/etc/vxcps.conf` file on CP server and in the `/etc/vxfenmode` file on the SF Oracle RAC cluster (client cluster) nodes.

Table 3-18 Fencing startup issues on SF Oracle RAC cluster (client cluster) nodes *(continued)*

Issue	Description and resolution
Preexisting split-brain	Assume the following situations to understand preexisting split-brain in server-based fencing: ■ There are three CP servers acting as coordination points. One of the three CP servers then becomes inaccessible. While in this state, also one client node leaves the cluster. When the inaccessible CP server restarts, it has a stale registration from the node which left the SF Oracle RAC cluster. In this case, no new nodes can join the cluster. Each node that attempts to join the cluster gets a list of registrations from the CP server. One CP server includes an extra registration (of the node which left earlier). This makes the joiner node conclude that there exists a preexisting split-brain between the joiner node and the node which is represented by the stale registration. ■ All the client nodes have crashed simultaneously, due to which fencing keys are not cleared from the CP servers. Consequently, when the nodes restart, the vxfen configuration fails reporting preexisting split brain. These situations are similar to that of preexisting split-brain with coordinator disks, where the problem is solved by the administrator running the `vxfenclearpre` command. A similar solution is required in server-based fencing using the `cpsadm` command. Run the `cpsadm` command to clear a registration on a CP server: `# cpsadm -s cp_server -a unreg_node -c cluster_name -n nodeid` where *cp_server* is the virtual IP address or virtual hostname on which the CP server is listening, *cluster_name* is the VCS name for the SF Oracle RAC cluster, and *nodeid* specifies the node id of SF Oracle RAC cluster node. Ensure that fencing is not already running on a node before clearing its registration on the CP server. After removing all stale registrations, the joiner node will be able to join the cluster.

Issues during online migration of coordination points

During online migration of coordination points using the vxfenswap utility, the operation is automatically rolled back if a failure is encountered during validation of coordination points from all the cluster nodes.

Validation failure of the new set of coordination points can occur in the following circumstances:

- The /etc/vxfenmode file is not updated on all the SF Oracle RAC cluster nodes, because new coordination points on the node were being picked up from an old /etc/vxfenmode file.

- The coordination points listed in the /etc/vxfenmode file on the different SF Oracle RAC cluster nodes are not the same. If different coordination points are listed in the /etc/vxfenmode file on the cluster nodes, then the operation fails due to failure during the coordination point snapshot check.
- There is no network connectivity from one or more SF Oracle RAC cluster nodes to the CP server(s).
- Cluster, nodes, or users for the SF Oracle RAC cluster nodes have not been added on the new CP servers, thereby causing authorization failure.

Vxfen service group activity after issuing the vxfenswap command

After issuing the `vxfenswap` command, the Coordination Point agent reads the details of coordination points from the `vxfenconfig -l` output and starts monitoring the registrations on them.

During vxfenswap, when the vxfenmode file is being changed by the user, the Coordination Point agent does not move to FAULTED state but continues monitoring the old set of coordination points.

As long as the changes to vxfenmode file are not committed or the new set of coordination points are not re-elected in `vxfenconfig -l` output, the Coordination Point agent continues monitoring the old set of coordination points it read from `vxfenconfig -l` output in every monitor cycle.

The status of the Coordination Point agent (either ONLINE or FAULTED) depends upon the accessibility of the coordination points, the registrations on these coordination points, and the fault tolerance value.

When the changes to vxfenmode file are committed and reflected in the `vxfenconfig -l` output, then the Coordination Point agent reads the new set of coordination points and proceeds to monitor them in its new monitor cycle.

Troubleshooting server-based I/O fencing in mixed mode

Use the following procedure to troubleshoot a mixed I/O fencing configuration (configuration which uses both coordinator disks and CP server for I/O fencing).

This procedure uses the following commands to obtain I/O fencing information:

- To obtain I/O fencing cluster information on the coordinator disks, run the following command on one of the cluster nodes:

  ```
  # vxfenadm -s diskname
  ```

  Any keys other than the valid keys used by the cluster nodes that appear in the command output are spurious keys.

- To obtain I/O fencing cluster information on the CP server, run the following command on one of the cluster nodes:

  ```
  # cpsadm -s cp_server -a list_membership -c cluster_name
  ```

  where *cp server* is the virtual IP address or virtual hostname on which the CP server is listening, and *cluster name* is the VCS name for the SF Oracle RAC cluster.

  Nodes which are not in GAB membership, but registered with CP server indicate a pre-existing network partition.

  Note that when running this command on the SF Oracle RAC cluster nodes, you need to first export the CPS_USERNAME and CPS_DOMAINTYPE variables.

  The CPS_USERNAME value is the user name which is added for this node on the CP server.

- To obtain the user name, run the following command on the CP server:

  ```
  # cpsadm -s cp_server -a list_users
  ```

  where *cp server* is the virtual IP address or virtual hostname on which the CP server is listening.

  The CPS_DOMAINTYPE value is vx.

  The following are export variable command examples:

```
# export CPS_USERNAME=_HA_VCS_test-system@HA_SERVICES@test-system.symantec.com
```

```
# export CPS_DOMAINTYPE=vx
```

Once a pre-existing network partition is detected using the above commands, all spurious keys on the coordinator disks or CP server must be removed by the administrator.

To troubleshoot server-based I/O fencing configuration in mixed mode

1 Review the current I/O fencing configuration by accessing and viewing the information in the vxfenmode file.

Enter the following command on one of the SF Oracle RAC cluster nodes:

```
# cat /etc/vxfenmode

vxfen_mode=customized
vxfen_mechanism=cps
scsi3_disk_policy=dmp
security=0
cps1=[10.140.94.101]:14250
vxfendg=vxfencoorddg
```

2 Review the I/O fencing cluster information.

Enter the `vxfenadm -d` command on one of the cluster nodes:

```
# vxfenadm -d

I/O Fencing Cluster Information:
================================

 Fencing Protocol Version: 201
 Fencing Mode: Customized
 Fencing Mechanism: cps
 Cluster Members:

        * 0 (galaxy)
          1 (nebula)

 RFSM State Information:
        node   0 in state  8 (running)
        node   1 in state  8 (running)
```

3 Review the SCSI registration keys for the coordinator disks used in the I/O fencing configuration.

The variables *disk_7* and *disk_8* in the following commands represent the disk names in your setup.

Enter the `vxfenadm -s` command on each of the SF Oracle RAC cluster nodes.

```
# vxfenadm -s /dev/vx/rdmp/disk_7
```

```
Device Name: /dev/vx/rdmp/disk_7
Total Number Of Keys: 2
key[0]:
         [Numeric Format]:  86,70,66,69,65,68,48,48
         [Character Format]: VFBEAD00
         [Node Format]: Cluster ID: 57069 Node ID: 0   Node Name: galaxy
key[1]:
         [Numeric Format]:  86,70,66,69,65,68,48,49
         [Character Format]: VFBEAD01
*        [Node Format]: Cluster ID: 57069 Node ID: 1   Node Name: nebula
```

Run the command on the other node:

```
# vxfenadm -s /dev/vx/rdmp/disk_8
```

```
Device Name: /dev/vx/rdmp/disk_8
Total Number Of Keys: 2
key[0]:
         [Numeric Format]:  86,70,66,69,65,68,48,48
         [Character Format]: VFBEAD00
         [Node Format]: Cluster ID: 57069 Node ID: 0   Node Name: galaxy
key[1]:
         [Numeric Format]:  86,70,66,69,65,68,48,49
         [Character Format]: VFBEAD01
*        [Node Format]: Cluster ID: 57069 Node ID: 1   Node Name: nebula
```

4 Review the CP server information about the cluster nodes. On the CP server, run the `cpsadm list nodes` command to review a list of nodes in the cluster.

```
# cpsadm -s cp_server -a list_nodes
```

where *cp server* is the virtual IP address or virtual hostname on which the CP server is listening.

5 Review the CP server list membership. On the CP server, run the following command to review the list membership.

```
# cpsadm -s cp_server -a list_membership -c cluster_name
```

where *cp_server* is the virtual IP address or virtual hostname on which the CP server is listening, and *cluster_name* is the VCS name for the SF Oracle RAC cluster.

For example:

```
# cpsadm -s 10.140.94.101 -a list_membership -c gl-ss2

List of registered nodes: 0 1
```

Checking keys on coordination points when vxfen_mechanism value is set to cps

When I/O fencing is configured in customized mode and the vxfen_mechanism value is set to cps, the recommended way of reading keys from the coordination points (coordinator disks and CP servers) is as follows:

- For coordinator disks, the disks can be put in a file and then information about them supplied to the `vxfenadm` command.
  For example:

  ```
  # vxfenadm -s all -f file_name
  ```

- For CP servers, the `cpsadm` command can be used to obtain the membership of the SF Oracle RAC cluster.
  For example:

  ```
  # cpsadm -s cp_server -a list_membership -c cluster_name
  ```

Where *cp_server* is the virtual IP address or virtual hostname on which CP server is configured, and *cluster_name* is the VCS name for the SF Oracle RAC cluster.

Understanding error messages

VCS generates two error message logs: the engine log and the agent log.

Log file names are appended by letters. Letter A indicates the first log file, B the second, C the third, and so on.

The engine log is located at /var/VRTSvcs/log/engine_A.log. The format of engine log messages is:

Timestamp (Year/MM/DD) | Mnemonic | Severity | UMI| Message Text

- Timestamp: The date and time the message was generated.
- Mnemonic: The string ID that represents the product (for example, VCS).
- Severity: Levels include CRITICAL, ERROR, WARNING, NOTICE, and INFO (most to least severe, respectively).
- UMI: A unique message ID.
- Message Text: The actual message that was generated by VCS.

A typical engine log resembles:

```
2003/02/10 16:08:09 VCS INFO V-16-1-10077 received new cluster membership.
```

The agent log is located at /var/VRTSvcs/log/agent_A.log.

The format of agent log messages resembles:

```
Timestamp (Year/MM/DD) | Mnemonic | Severity | UMI | Agent Type |
Resource Name | Entry Point | Message Text
```

A typical agent log resembles:

```
2003/02/23 10:38:23 VCS WARNING V-16-2-23331
Oracle:VRT:monitor:Open for ora_lgwr failed, setting cookie to
null.
```

Note that the logs on all nodes may not be identical because of the following circumstances:

- VCS logs local events on the local nodes.
- All nodes may not be running when an event occurs.

Troubleshooting CVM

This section discusses troubleshooting CVM problems.

Restoring communication between host and disks after cable disconnection

If a fiber cable is inadvertently disconnected between the host and a disk, you can restore communication between the host and the disk without restarting.

To restore lost cable communication between host and disk

1. Reconnect the cable.
2. On all nodes, use the `fdisk -l` command to scan for new disks.

   It may take a few minutes before the host is capable of seeing the disk.
3. On all nodes, issue the following command to rescan the disks:

   ```
   # vxdisk scandisks
   ```

4. On the master node, reattach the disks to the disk group they were in and retain the same media name:

   ```
   # vxreattach
   ```

   This may take some time. For more details, see `vxreattach` (1M) manual page.

Shared disk group cannot be imported

If you see a message resembling:

```
vxvm:vxconfigd:ERROR:vold_pgr_register(/dev/vx/rdmp/disk_name):
local_node_id<0
Please make sure that CVM and vxfen are configured
and operating correctly
```

First, make sure that CVM is running. You can see the CVM nodes in the cluster by running the vxclustadm nidmap command.

```
# vxclustadm nidmap
Name          CVM Nid     CM Nid      State
galaxy         1           0           Joined: Master
nebula         0           1           Joined: Slave
```

This above ouput shows that CVM is healthy, with system galaxy as the CVM master. If CVM is functioning correctly, then the output above is displayed when CVM cannot retrieve the node ID of the local system from the `vxfen` driver. This usually happens when port b is not configured.

To verify vxfen driver is configured

- Check the GAB ports with the command:

```
# gabconfig -a
```

Port b must exist on the local system.

Error importing shared disk groups

The following message may appear when importing shared disk group:

```
VxVM vxdg ERROR V-5-1-587 Disk group disk group name: import
failed: No valid disk found containing disk group
```

You may need to remove keys written to the disk.

For information about removing keys written to the disk:

See "Removing preexisting keys" on page 117.

Unable to start CVM

If you cannot start CVM, check the consistency between the `/etc/llthosts` and `main.cf` files for node IDs.

You may need to remove keys written to the disk.

For information about removing keys written to the disk:

See "Removing preexisting keys" on page 117.

CVM group is not online after adding a node to the cluster

The possible causes for the CVM group being offline after adding a node to the cluster are as follows:

- The cssd resource is configured as a critical resource in the cvm group.
- Other resources configured in the cvm group as critical resources are not online.

To resolve the issue if cssd is configured as a critical resource

1. Log onto one of the nodes in the existing cluster as the root user.
2. Configure the cssd resource as a non-critical resource in the cvm group:

```
# haconf -makerw
# hares -modify cssd Critical 0
# haconf -dump -makero
```

To resolve the issue if other resources in the group are not online

1. Log onto one of the nodes in the existing cluster as the root user.
2. Bring the resource online:

```
# hares -online resource_name -sys system_name
```

3. Verify the status of the resource:

```
# hastatus -resource  resource_name
```

4. If the resource is not online, configure the resource as a non-critical resource:

```
# haconf -makerw
# hares -modify resource_name Critical 0
# haconf -dump -makero
```

CVMVolDg not online even though CVMCluster is online

When the CVMCluster resource goes online, then all shared disk groups that have the auto-import flag set are automatically imported. If the disk group import fails for some reason, the CVMVolDg resources fault. Clearing and taking the CVMVolDg type resources offline does not resolve the problem.

To resolve the resource issue

1. Fix the problem causing the import of the shared disk group to fail.
2. Offline the cvm group containing the resource of type CVMVolDg as well as the service group containing the CVMCluster resource type.
3. Bring the cvm group containing the CVMCluster resource online.
4. Bring the cvm group containing the CVMVolDg resource online.

Shared disks not visible

If the shared disks in `/dev/rdsk` are not visible, perform the following tasks:

Make sure that all shared LUNs are discovered by the HBA and SCSI layer. This can be verified by running the `ls -ltr` command on any of the disks under `/dev/rdsk/*`.

For example:

```
# ls -ltr /dev/rdsk/disk_name
lrwxrwxrwx   1 root     root          81 Aug 18 11:58
c2t5006016141E02D28d4s2
-> ../../devices/pci@7c0/pci@0/pci@8/SUNW,qlc@0/fp@0,
0/ssd@w5006016141e02d28,4:c,raw
lrwxrwxrwx   1 root     root          81 Aug 18 11:58
c2t5006016141E02D28d3s2
-> ../../devices/pci@7c0/pci@0/pci@8/SUNW,qlc@0/fp@0,
0/ssd@w5006016141e02d28,3:c,raw
lrwxrwxrwx   1 root     root          81 Aug 18 11:58
c2t5006016141E02D28d2s2
-> ../../devices/pci@7c0/pci@0/pci@8/SUNW,qlc@0/fp@0,
0/ssd@w5006016141e02d28,2:c,raw
lrwxrwxrwx   1 root     root          81 Aug 18 11:58
c2t5006016141E02D28d1s2
-> ../../devices/pci@7c0/pci@0/pci@8/SUNW,qlc@0/fp@0,
0/ssd@w5006016141e02d28,1:c,raw
```

If all LUNs are not discovered by SCSI, the problem might be corrected by specifying `dev_flags` or `default_dev_flags` and *max_luns* parameters for the SCSI driver.

If the LUNs are not visible in `/dev/rdsk/*` files, it may indicate a problem with SAN configuration or zoning.

Troubleshooting CFS

This section discusses troubleshooting CFS problems.

Incorrect order in root user's <library> path

An incorrect order in the root user's <library> path can cause the system to hang while changing the primary node in the Cluster File System or the RAC cluster.

If the <library> path of the root user contains an entry pointing to a Cluster File System (CFS) file system before the /usr/lib entry, the system may hang when trying to perform one of the following tasks:

- Changing the primary node for the CFS file system
- Unmounting the CFS files system on the primary node
- Stopping the cluster or the service group on the primary node

This configuration issue occurs primarily in a RAC environment with Oracle binaries installed on a shared CFS file system.

The following is an example of a <library path> that may cause the system to hang:

```
LIBPATH=/app/oracle/orahome/lib:/usr/lib:/usr/ccs/lib
```

In the above example, /app/oracle is a CFS file system, and if the user tries to change the primary node for this file system, the system will hang. The user is still able to ping and telnet to the system, but simple commands such as `ls` will not respond. One of the first steps required during the changing of the primary node is freezing the file system cluster wide, followed by a quick issuing of the `fsck` command to replay the intent log.

Since the initial entry in <library> path is pointing to the frozen file system itself, the `fsck` command goes into a deadlock situation. In fact, all commands (including `ls`) which rely on the <library> path will hang from now on.

The recommended procedure to correct for this problem is as follows: Move any entries pointing to a CFS file system in any user's (especially root) <library> path towards the end of the list after the entry for /usr/lib

Therefore, the above example of a <library path> would be changed to the following:

```
LIBPATH=/usr/lib:/usr/ccs/lib:/app/oracle/orahome/lib
```

Troubleshooting interconnects

This section discusses troubleshooting interconnect problems.

Example entries for mandatory devices

If you are using eth2 and eth3 for interconnectivity, use the following procedure examples to set mandatory devices.

To set mandatory devices entry in the /etc/sysconfig/network/config

Enter:

```
MANDATORY_DEVICES="eth2-00:04:23:AD:4A:4C
eth3-00:04:23:AD:4A:4D"
```

To set a persistent name entry in an interface file

In file: /etc/sysconfig/network/ifcfg-eth-id-00:09:3d:00:cd:22 (Name of the eth0 Interface file), enter:

```
BOOTPROTO='static'
BROADCAST='10.212.255.255'
IPADDR='10.212.88.22'
MTU=' '
NETMASK='255.255.254.0'
NETWORK='10.212.88.0'
REMOTE_IP=' '
STARTMODE='onboot'
UNIQUE='RFE1.bBSepP2NetB'
_nm_name='bus-pci-0000:06:07.0'
PERSISTENT_NAME=eth0
```

Troubleshooting Oracle

This section discusses troubleshooting Oracle.

Error when starting an Oracle instance

If the VCSMM driver (the membership module) is not configured, an error displays while starting the Oracle instance that resembles:

```
ORA-29702: error occurred in Cluster Group Operation
```

To start the VCSMM driver:

```
# /etc/init.d/vcsmm start
```

The command included in the `/etc/vcsmmtab` file enables the VCSMM driver to be started at system boot.

Clearing Oracle group faults

If the Oracle group faults, you can clear the faults and bring the group online by running the following commands:

```
# hagrp -clear oracle_grp -sys galaxy

# hagrp -clear oracle_grp -sys nebula

# hagrp -online oracle_grp -sys galaxy

# hagrp -online oracle_grp -sys nebula
```

Oracle log files show shutdown called even when not shutdown manually

The Oracle enterprise agent calls shutdown if monitoring of the Oracle resources fails. On all cluster nodes, review the following VCS and Oracle agent log files for any errors or status:

```
/var/VRTSvcs/log/engine_A.log
/var/VRTSvcs/log/Oracle_A.log
```

DBCA fails while creating database

Verify that the `hostname -i` command returns the public IP address of the current node. This command is used by the installer and the output is stored in the OCR. If `hostname -i` returns 127.0.0.1, it causes the DBCA to fail.

Oracle Clusterware processes fail to startup

Verify that the correct private IP address is configured on the private link using the PrivNIC or MultiPrivNIC agent. Check the CSS log files to learn more.

You can find the CSS log files at $CRS_HOME/log/*node_name*/cssd/*

You can find the CSS log files at $GRID_HOME/log/*node_name*/cssd/*

Consult the Oracle RAC documentation for more information.

Oracle Clusterware fails after restart

If the Oracle Clusterware fails to start after boot up, check for the occurrence of the following strings in the `/var/adm/messages` file.

String value in the file: `Oracle CSSD failure. Rebooting for cluster integrity`	Oracle Clusterware may fail due to Oracle CSSD failure. The Oracle CSSD failure may be caused by one of the following events: ■ Communication failure occurred and Oracle Clusterware fenced out the node. ■ OCR and Vote disk became unavailable. ■ `ocssd` was killed manually. ■ Killing the `init.cssd` script.
String value in the file: `Waiting for file system containing`	The Oracle Clusterware installation is on a shared disk and the `init` script is waiting for that file system to be made available.
String value in the file: `Oracle Cluster Ready Services disabled by corrupt install`	The following file is not available or has corrupt entries: `/etc/oracle/scls_scr/\ hostname/root/crsstart.`
String value in the file: `OCR initialization failed accessing OCR device`	The shared file system containing the OCR is not available and Oracle Clusterware is waiting for it to become available.

Removing Oracle Clusterware if installation fails

For instructions on removing Oracle Clusterware, see the Oracle documentation.

Troubleshooting the Virtual IP (VIP) Configuration

When troubleshooting issues with the VIP configuration, use the following commands and files:

- Check for network problems on all nodes:

  `/etc/ifconfig -a`

- Make sure the virtual host name is registered with the DNS server:
- Verify the `/etc/hosts` file on each node.
- Verify the virtual host name on each node.

```
# ping virtual_host_name
```

- Check the output of the following command:

```
$GRID_HOME/bin/crsctl stat res -t
```

- On the problem node, use the command:

```
$ srvctl start nodeapps -n node_name
```

Loss of connectivity to OCR and voting disk causes the cluster to panic

If the CVM master node looses connectivity to the SAN, the default settings for the disk detach policy (`global`) and the disk group fail policy (`dgdisable`) causes the Oracle Cluster Registry (OCR) and voting disk to be disabled from all nodes, thereby panicking the cluster.

Update the default settings of the disk detach policy to `global` and the disk group fail policy to `leave` for OCR and voting disk on all SF Oracle RAC disk groups.

To set the disk policies for OCR and voting disk

1. Log in to one of the nodes as the root user.
2. Set the policies on all SF Oracle RAC disk groups as follows:

```
# vxdg -g dg_name set diskdetpolicy=global
 # vxdg -g dg_name set dgfailpolicy=leave
```

OCRDUMP

Executing the `$ORA_GRID_HOME/bin/ocrdump` creates a `OCRDUMPFILE` in the working directory. This text file contains a dump of all the parameters stored in the cluster registry, which is useful in case of errors.

Check if the following variable occurs in the `OCRDUMPFILE`: SYSTEM.css.misscount. This variable is the timeout value in seconds that will be used by CRS to fence off the nodes in case of communication failure. Verify that the timeout value in OCRDUMP is 150 seconds. During Oracle installation, SF Oracle RAC software updates this value to 150 seconds.

Troubleshooting Oracle Clusterware health check warning messages

Table 3-19 lists the warning messages displayed during the health check and corresponding recommendations for resolving the issues.

Table 3-19 Troubleshooting Oracle Clusterware warning messages

Warning	Possible causes	Recommendation
`Oracle Clusterware is not running.`	Oracle Clusterware is not started. Oracle Clusterware is waiting for dependencies such as OCR or voting disk or private IP addresses to be available.	■ Bring the cvm group online to start Oracle Clusterware: `# hagrp -online cvm \ -sys system_name` ■ Check the VCS log file `/var/VRTSvcs/log/engine_A.log` to determine the cause and fix the issue.
`No CSSD resource is configured under VCS.`	The CSSD resource is not configured under VCS.	Configure the CSSD resource under VCS and bring the resource online. For instructions, see the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide*.
`The CSSD resource name is not running.`	■ VCS is not running. ■ The dependent resources, such as the CFSMount or CVMVolDg resource for OCR and voting disk are not online.	■ Start VCS: `# hastart` ■ Check the VCS log file `/var/VRTSvcs/log/engine_A.log` to determine the cause and fix the issue.
`Mismatch between LLT links llt nics and Oracle Clusterware links crs nics.`	The private interconnects used by Oracle Clusterware are not configured over LLT interfaces.	The private interconnects for Oracle Clusterware must use LLT links. Configure the private IP addresses on one of the LLT links. For instructions, see the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide*.

Table 3-19 Troubleshooting Oracle Clusterware warning messages *(continued)*

Warning	Possible causes	Recommendation
`Mismatch between Oracle Clusterware links crs nics and PrivNIC links private nics.`	The private IP addresses used by Oracle Clusterware are not configured under PrivNIC.	Configure the PrivNIC resource to monitor the private IP address used by Oracle Clusterware. For instructions, see the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide*.
`Mismatch between CRS nodes crs nodes and LLT nodes llt nodes.`	The host names configured during the Oracle Clusterware installation are not the same as the host names configured under LLT.	Make sure that the host names configured during the Oracle Clusterware installation are the same as the host names configured under LLT.

Troubleshooting ODM

This section discusses troubleshooting ODM.

File System configured incorrectly for ODM shuts down Oracle

Linking Oracle RAC with the Veritas ODM libraries provides the best file system performance.

Review the instructions on creating the link and confirming that Oracle uses the libraries. Shared file systems in RAC clusters without ODM libraries linked to Oracle RAC may exhibit slow performance and are not supported.

If ODM cannot find the resources it needs to provide support for cluster file systems, it does not allow Oracle to identify cluster files and causes Oracle to fail at startup.

To verify cluster status, run the following command and review the output:

```
# cat /dev/odm/cluster

cluster status: enabled
```

If the status is "enabled," ODM is supporting cluster files. Any other cluster status indicates that ODM is not supporting cluster files. Other possible values include:

`pending`	ODM cannot yet communicate with its peers, but anticipates being able to eventually.
`failed`	ODM cluster support has failed to initialize properly. Check console logs.
`disabled`	ODM is not supporting cluster files. If you think ODM should be supporting the cluster files: ■ Check `/dev/odm` mount options in `/etc/vfstab`. If the `nocluster` option is being used, it can force the `disabled` cluster support state. ■ Make sure that the `VRTSgms` (group messaging service) RPM is installed. Run the following command:

If `/dev/odm` is not mounted, no status can be reported.

Chapter 4

Prevention and recovery strategies

This chapter includes the following topics:

- Verification of GAB ports in SF Oracle RAC cluster
- Examining GAB seed membership
- Manual GAB membership seeding
- Evaluating VCS I/O fencing ports
- Verifying normal functioning of VCS I/O fencing
- Managing SCSI-3 PR keys in SF Oracle RAC cluster
- Identifying a faulty coordinator LUN
- Starting shared volumes manually
- Listing all the CVM shared disks
- Failure scenarios and recovery strategies for CP server setup

Verification of GAB ports in SF Oracle RAC cluster

The following ports need to be up on all the nodes of SF Oracle RAC cluster:

- GAB
- I/O fencing
- ODM
- CFS

- VCS (‘had’)
- vcsmm (membership module for SF Oracle RAC)
- CVM (kernel messaging)
- CVM (vxconfigd)
- CVM (to ship commands from slave node to master node)

The following command can be used to verify the state of GAB ports:

```
# gabconfig -a
```

GAB Port Memberships

```
Port a gen 7e6e7e05 membership 01
Port b gen 58039502 membership 01
Port d gen 588a7d02 membership 01
Port f gen 1ea84702 membership 01
Port h gen cf430b02 membership 01
Port o gen de8f0202 membership 01
Port u gen de4f0203 membership 01
Port v gen db411702 membership 01
Port w gen cf430b02 membership 01
```

The data indicates that all the GAB ports are up on the cluster having nodes 0 and 1.

For more information on the GAB ports in SF Oracle RAC cluster, see the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide*.

Examining GAB seed membership

The number of systems that participate in the cluster is specified as an argument to the `gabconfig` command in /etc/gabtab. In the following example, two nodes are expected to form a cluster:

```
# cat /etc/gabtab
```

```
/sbin/gabconfig -c -n2
```

GAB waits until the specified number of nodes becomes available to automatically create the port “a” membership. Port “a” indicates GAB membership for an SF Oracle RAC cluster node. Every GAB reconfiguration, such as a node joining or leaving increments or decrements this seed membership in every cluster member node.

A sample port ‘a’ membership as seen in `gabconfig -a` is shown:

```
Port a gen 7e6e7e01 membership 01
```

In this case, 7e6e7e01 indicates the “membership generation number” and 01 corresponds to the cluster’s “node map”. All nodes present in the node map reflects the same membership ID as seen by the following command:

```
# gabconfig -a | grep "Port a"
```

The semi-colon is used as a placeholder for a node that has left the cluster. In the following example, node 0 has left the cluster:

```
# gabconfig -a | grep "Port a"

Port a gen 7e6e7e04 membership ;1
```

When the last node exits the port “a” membership, there are no other nodes to increment the membership ID. Thus the port “a” membership ceases to exist on any node in the cluster.

When the last and the final system is brought back up from a complete cluster cold shutdown state, the cluster will seed automatically and form port “a” membership on all systems. Systems can then be brought down and restarted in any combination so long as at least one node remains active at any given time.

The fact that all nodes share the same membership ID and node map certifies that all nodes in the node map participates in the same port “a” membership. This consistency check is used to detect “split-brain” and “pre-existing split-brain” scenarios.

Split-brain occurs when a running cluster is segregated into two or more partitions that have no knowledge of the other partitions. The pre-existing network partition is detected when the "cold" nodes (not previously participating in cluster) start and are allowed to form a membership that might not include all nodes (multiple sub-clusters), thus resulting in a potential split-brain.

Note: Symantec I/O fencing prevents data corruption resulting from any split-brain scenarios.

Manual GAB membership seeding

It is possible that one of the nodes does not come up when all the nodes in the cluster are restarted, due to the "minimum seed requirement" safety that is enforced by GAB. Human intervention is needed to safely determine that the other node is in fact not participating in its own mini-cluster.

The following should be carefully validated before manual seeding, to prevent introducing split-brain and subsequent data corruption:

- Verify that none of the other nodes in the cluster have a port “a” membership
- Verify that none of the other nodes have any shared disk groups imported
- Determine why any node that is still running does not have a port “a” membership

Run the following command to manually seed GAB membership:

```
# gabconfig -cx
```

Refer to gabconfig (1M) for more details.

Evaluating VCS I/O fencing ports

I/O Fencing (VxFEN) uses a dedicated port that GAB provides for communication across nodes in the cluster. You can see this port as port ‘b’ when `gabconfig -a` runs on any node in the cluster. The entry corresponding to port ‘b’ in this membership indicates the existing members in the cluster as viewed by I/O Fencing.

GAB uses port "a" for maintaining the cluster membership and must be active for I/O Fencing to start.

To check whether fencing is enabled in a cluster, the ‘-d’ option can be used with vxfenadm (1M) to display the I/O Fencing mode on each cluster node. Port “b” membership should be present in the output of `gabconfig -a` and the output should list all the nodes in the cluster.

If the GAB ports that are needed for I/O fencing are not up, that is, if port “a” is not visible in the output of `gabconfig -a` command, LLT and GAB must be started on the node.

The following commands can be used to start LLT and GAB respectively:

To start LLT on each node:

```
# /etc/init.d/llt start
```

If LLT is configured correctly on each node, the console output displays:

```
LLT INFO V-14-1-10009 LLT Protocol available
```

To start GAB, on each node:

```
# /etc/init.d/gab start
```

If GAB is configured correctly on each node, the console output displays:

```
GAB INFO V-15-1-20021 GAB available
```

```
GAB INFO V-15-1-20026 Port a registration waiting for seed port
membership
```

Verifying normal functioning of VCS I/O fencing

It is mandatory to have VCS I/O fencing enabled in SF Oracle RAC cluster to protect against split-brain scenarios. VCS I/O fencing can be assumed to be running normally in the following cases:

- Fencing port 'b' enabled on the nodes

  ```
  # gabconfig -a
  ```

  To verify that fencing is enabled on the nodes:

  ```
  # vxfenadm -d
  ```

- Registered keys present on the coordinator disks

  ```
  # vxfenadm -g all -f /etc/vxfentab
  ```

Managing SCSI-3 PR keys in SF Oracle RAC cluster

I/O Fencing places the SCSI-3 PR keys on coordinator LUNs. The format of the key follows the naming convention wherein ASCII “A” is prefixed to the LLT ID of the system that is followed by 7 dash characters.

For example:

node 0 uses A-------

node 1 uses B-------

In an SF Oracle RAC/SF CFS/SF HA environment, VxVM/CVM registers the keys on data disks, the format of which is ASCII “A” prefixed to the LLT ID of the system followed by the characters “PGRxxxx” where ‘xxxx’ = i such that the disk group is the ith shared group to be imported.

For example: node 0 uses APGR0001 (for the first imported shared group).

In addition to the registration keys, VCS/CVM also installs a reservation key on the data LUN. There is one reservation key per cluster as only one node can reserve the LUN.

See “About SCSI-3 Persistent Reservations” on page 43.

The following command lists the keys on a data disk group:

```
# vxdg list |grep data

galaxy_data1 enabled,shared,cds 1201715530.28.pushover
```

Select the data disk belonging to galaxy_data1:

```
# vxdisk -o alldgs list |grep galaxy_data1
```

The following command lists the PR keys:

```
# vxdisk -o listreserve list sdh

.....

.....
```

Alternatively, the PR keys can be listed using `vxfenadm` command:

```
#echo "/dev/vx/dmp/sdh" > /tmp/disk71

# vxfenadm -g all -f /tmp/disk71

Device Name: /dev/vx/dmp/sdh
Total Number Of Keys: 2
key[0]:
        Key Value [Numeric Format]:  66,80,71,82,48,48,48,52
        Key Value [Character Format]: BPGR0004
key[1]:
        Key Value [Numeric Format]:  65,80,71,82,48,48,48,52
        Key Value [Character Format]: APGR0004
```

Evaluating the number of SCSI-3 PR keys on a coordinator LUN, if there are multiple paths to the LUN from the hosts

The utility vxfenadm (1M) can be used to display the keys on the coordinator LUN. The key value identifies the node that corresponds to each key. Each node installs a registration key on all the available paths to the LUN. Thus, the total number of registration keys is the sum of the keys that are installed by each node in the above manner.

See “About the vxfenadm utility” on page 112.

Detecting accidental SCSI-3 PR key removal from coordinator LUNs

The keys currently installed on the coordinator disks can be read using the following command:

```
# vxfenadm -s all -f /etc/vxfentab
```

There should be a key for each node in the operating cluster on each of the coordinator disks for normal cluster operation. There will be two keys for every node if you have a two-path DMP configuration.

Identifying a faulty coordinator LUN

The utility vxfentsthdw (1M) provided with I/O fencing can be used to identify faulty coordinator LUNs. This utility must be run from any node in the cluster. The coordinator LUN, which needs to be checked, should be supplied to the utility.

See "About the vxfentsthdw utility" on page 104.

Starting shared volumes manually

Following a manual CVM shared disk group import, the volumes in the disk group need to be started manually, as follows:

```
# vxvol -g dg_name startall
```

To verify that the volumes are started, run the following command:

```
# vxprint -htrg dg_name | grep ^v
```

Listing all the CVM shared disks

You can use the following command to list all the CVM shared disks:

```
# vxdisk -o alldgs list |grep shared
```

Failure scenarios and recovery strategies for CP server setup

Table 4-1 describes the various failure scenarios and recovery strategies that should be considered when setting up and maintaining I/O fencing using CP servers.

Table 4-1 Failure scenarios and recovery strategy considerations

CP server issue	Description
CP server planned replacement	After you configure server-based I/O fencing, you may need to replace the CP servers. As an administrator, you can perform a planned replacement of a CP server with either another CP server or a SCSI-3 disk without incurring application downtime on the SF Oracle RAC cluster. **Note:** If multiple clusters are sharing the CP server, all of the clusters have to initiate online replacement independently of each other.
CP server planned registration refreshing	After setting up I/O fencing using CP servers, you may need to refresh a CP server registration. As an administrator, you can perform a planned refreshing of registrations on a CP server without incurring application downtime on the SF Oracle RAC cluster: See "Refreshing registration keys on the coordination points for server-based fencing" on page 141. **Note:** Registration refreshing on a CP server would be performed in case the CP server agent alerts on loss of such registrations on the CP server database.
CP server process failure	The applications and high availability of applications on the SF Oracle RAC clusters sharing a CP server are immune to the failure of the CP server process or an unresponsive CP server process, unless a split-brain occurs after a failure or unresponsive process and before the CP server process can be restarted.
CP server failure	The applications and high availability of applications on the SF Oracle RAC clusters sharing a CP server are immune to failure of a CP server, unless a split-brain occurs after the failure and before the CP server can be restarted.

Chapter 5

Tunable parameters

This chapter includes the following topics:

- About SF Oracle RAC tunable parameters
- About GAB tunable parameters
- About LLT tunable parameters
- About VXFEN tunable parameters
- Tuning guidelines for campus clusters

About SF Oracle RAC tunable parameters

Tunable parameters can be configured to enhance the performance of specific SF Oracle RAC features. This chapter discusses how to configure the following SF Oracle RAC tunables:

- GAB
- LLT
- VXFEN

Symantec recommends that the user not change the tunable kernel parameters without assistance from Symantec support personnel. Several of the tunable parameters preallocate memory for critical data structures, and a change in their values could increase memory use or degrade performance.

Warning: Do not adjust the SF Oracle RAC tunable parameters for VXFEN as described below to enhance performance without assistance from Symantec support personnel.

About GAB tunable parameters

GAB provides various configuration and tunable parameters to modify and control the behavior of the GAB module.

These tunable parameters not only provide control of the configurations like maximum possible number of nodes in the cluster, but also provide control on how GAB behaves when it encounters a fault or a failure condition. Some of these tunable parameters are needed when the GAB module is loaded into the system. Any changes to these load-time tunable parameter require either unload followed by reload of GAB module or system reboot. Other tunable parameters (run-time) can be changed while GAB module is loaded, configured, and cluster is running. Any changes to such a tunable parameter will have immediate effect on the tunable parameter values and GAB behavior.

These tunable parameters are defined in the following file:

`/etc/sysconfig/gab`

See "About GAB load-time or static tunable parameters" on page 238.

See "About GAB run-time or dynamic tunable parameters" on page 240.

About GAB load-time or static tunable parameters

Table 5-1 lists the static tunable parameters in GAB that are used during module load time. Use the `gabconfig -e` command to list all such GAB tunable parameters.

You can modify these tunable parameters only by adding new values in the GAB configuration file. The changes take effect only on reboot or on reload of the GAB module.

Table 5-1 GAB static tunable parameters

GAB parameter	Description	Values (default and range)
numnids	Maximum number of nodes in the cluster	Default: 64 Range: 1-64
numports	Maximum number of ports in the cluster	Default: 32 Range: 1-32

Table 5-1 GAB static tunable parameters *(continued)*

GAB parameter	Description	Values (default and range)
flowctrl	Number of pending messages in GAB queues (send or receive) before GAB hits flow control. This can be overwritten while cluster is up and running with the `gabconfig -Q` option. Use the `gabconfig` command to control value of this tunable.	Default: 128 Range: 1-1024
logbufsize	GAB internal log buffer size in bytes	Default: 48100 Range: 8100-65400
msglogsize	Maximum messages in internal message log	Default: 256 Range: 128-4096
isolate_time	Maximum time to wait for isolated client Can be overridden at runtime See "About GAB run-time or dynamic tunable parameters" on page 240.	Default: 120000 msec (2 minutes) Range: 160000-240000 (in msec)
gab_kill_ntries	Number of times to attempt to kill client Can be overridden at runtime See "About GAB run-time or dynamic tunable parameters" on page 240.	Default: 5 Range: 3-10
gab_conn_wait	Maximum number of wait periods (as defined in the stable timeout parameter) before GAB disconnects the node from the cluster during cluster reconfiguration	Default: 12 Range: 1-256

Table 5-1 GAB static tunable parameters *(continued)*

GAB parameter	Description	Values (default and range)
gab_ibuf_count	Determines whether the GAB logging daemon is enabled or disabled The GAB logging daemon is enabled by default. To disable, change the value of gab_ibuf_count to 0. This can be overwritten while cluster is up and running with the `gabconfig -K` option. Use the `gabconfig` command to control value of this tunable.	Default: 8 Range: 0-32
gab_kstat_size	Number of system statistics to maintain in GAB	Default: 60 Range: 0 - 240

About GAB run-time or dynamic tunable parameters

You can change the GAB dynamic tunable parameters while GAB is configured and while the cluster is running. The changes take effect immediately on running the `gabconfig` command. Note that some of these parameters also control how GAB behaves when it encounters a fault or a failure condition. Some of these conditions can trigger a PANIC which is aimed at preventing data corruption.

You can display the default values using the `gabconfig -l` command. To make changes to these values persistent across reboots, you can append the appropriate command options to the `/etc/gabtab` file along with any existing options. For example, you can add the `-k` option to an existing `/etc/gabtab` file that might read as follows:

```
gabconfig -c -n 4
```

After adding the option, the `/etc/gabtab` file looks similar to the following:

```
gabconfig -c -n4 -k
```

Table 5-2 describes the GAB dynamic tunable parameters as seen with the `gabconfig -l` command, and specifies the command to modify them.

Table 5-2 GAB dynamic tunable parameters

GAB parameter	Description and command

Control Seed port	Default: Disabled
	
	This option defines the minimum number of nodes that can form the cluster. This option controls the forming of the cluster. If the number of nodes in the cluster is less than the number specified in the `gabtab` file, then the cluster will not form. For example: if you type `gabconfig -c -n4`, then the cluster will not form until all four nodes join the cluster. If this option is enabled using the `gabconfig -x` command then the node will join the cluster even if the other nodes in the cluster are not yet part of the membership.
	
	Use the following command to set the number of nodes that can form the cluster:
	
	`gabconfig -n count`
	
	Use the following command to enable control seed port. Node can form the cluster without waiting for other nodes for membership:
	
	`gabconfig -x`

Halt on process death	Default: Disabled
	
	This option controls our ability to halt (panic) the system on user process death. If someone kills _had and _hashadow using kill -9, the system can potentially loose high availability. If you enable this option, then the GAB will PANIC the system on detecting the death of the client process. The default behavior is to disable this option.
	
	Use the following command to enable halt system on process death:
	
	`gabconfig -p`
	
	Use the following command to disable halt system on process death:
	
	`gabconfig -P`

Table 5-2 GAB dynamic tunable parameters *(continued)*

GAB parameter	Description and command

Missed heartbeat halt	Default: Disabled
	
	If this option is enabled then the system will panic on missing the first heartbeat from the VCS engine or the `vxconfigd` daemon in a CVM environment. The default option is to disable the immediate panic.
	
	This GAB option controls whether GAB can panic the node or not when the VCS engine or the vxconfigd daemon miss to heartbeat with GAB. If the VCS engine experiences a hang and is unable to heartbeat with GAB, then GAB will NOT PANIC the system immediately. GAB will first try to abort the process by sending SIGABRT (kill_ntries - default value 5 times) times after an interval of "iofence_timeout" (default value 15 seconds). If this fails, then GAB will wait for the "isolate timeout" period which is controlled by a global tunable called isolate_time (default value 2 minutes). If the process is still alive, then GAB will PANIC the system.
	
	If this option is enabled GAB will immediately HALT the system in case of missed heartbeat from client.
	
	Use the following command to enable system halt when process heartbeat fails:
	
	`gabconfig -b`
	
	Use the following command to disable system halt when process heartbeat fails:
	
	`gabconfig -B`

Table 5-2 GAB dynamic tunable parameters *(continued)*

GAB parameter	Description and command

Halt on rejoin	Default: Disabled
	
	This option allows the user to configure the behavior of the VCS engine or any other user process when one or more nodes rejoin a cluster after a network partition. By default GAB will not PANIC the node running the VCS engine. GAB kills the userland process (the VCS engine or the vxconfigd process). This recycles the user port (port h in case of the VCS engine) and clears up messages with the old generation number programmatically. Restart of the process, if required, must be handled outside of GAB control, e.g., for hashadow process restarts _had.
	
	When GAB has kernel clients (such as fencing, VxVM, or VxFS), then the node will always PANIC when it rejoins the cluster after a network partition. The PANIC is mandatory since this is the only way GAB can clear ports and remove old messages.
	
	Use the following command to enable system halt on rejoin:
	
	`gabconfig -j`
	
	Use the following command to disable system halt on rejoin:
	
	`gabconfig -J`

Keep on killing	Default: Disabled If this option is enabled, then GAB prevents the system from PANICKING when the VCS engine or the vxconfigd process fail to heartbeat with GAB and GAB fails to kill the VCS engine or the vxconfigd process. GAB will try to continuously kill the VCS engine and will not panic if the kill fails. Repeat attempts to kill process if it does not die `gabconfig -k`

Table 5-2 GAB dynamic tunable parameters *(continued)*

GAB parameter	Description and command

Quorum flag	Default: Disabled
	This is an option in GAB which allows a node to IOFENCE (resulting in a PANIC) if the new membership set is < 50% of the old membership set. This option is typically disabled and is used when integrating with other products
	Enable iofence quorum
	`gabconfig -q`
	Disable iofence quorum
	`gabconfig -d`

GAB queue limit	Default: Send queue limit: 128
	Default: Receive queue limit: 128
	GAB queue limit option controls the number of pending message before which GAB sets flow. Send queue limit controls the number of pending message in GAB send queue. Once GAB reaches this limit it will set flow control for the sender process of the GAB client. GAB receive queue limit controls the number of pending message in GAB receive queue before GAB send flow control for the receive side.
	Set the send queue limit to specified value
	`gabconfig -Q sendq:value`
	Set the receive queue limit to specified value
	`gabconfig -Q recvq:value`

IOFENCE timeout	Default: 15000(ms) This parameter specifies the timeout (in milliseconds) for which GAB will wait for the clients to respond to an IOFENCE message before taking next action. Based on the value of kill_ntries , GAB will attempt to kill client process by sending SIGABRT signal. If the client process is still registered after GAB attempted to kill client process for the value of kill_ntries times, GAB will halt the system after waiting for additional isolate_timeout value. Set the iofence timeout value to specified value in milliseconds. `gabconfig -f value`

Table 5-2 GAB dynamic tunable parameters *(continued)*

GAB parameter	Description and command
Stable timeout	Default: 5000(ms) Specifies the time GAB waits to reconfigure membership after the last report from LLT of a change in the state of local node connections for a given port. Any change in the state of connections will restart GAB waiting period. Set the stable timeout to specified value `gabconfig -t stable`
Isolate timeout	Default: 120000(ms) This tunable specifies the timeout value for which GAB will wait for client process to unregister in response to GAB sending SIGKILL signal. If the process still exists after isolate timeout GAB will halt the system `gabconfig -S isolate_time:value`
Kill_ntries	Default: 5 This tunable specifies the number of attempts GAB will make to kill the process by sending SIGABRT signal. `gabconfig -S kill_ntries:value`

About LLT tunable parameters

LLT provides various configuration and tunable parameters to modify and control the behavior of the LLT module. This section describes some of the LLT tunable parameters that can be changed at run-time and at LLT start-time.

See "About Low Latency Transport (LLT)" on page 25.

The tunable parameters are classified into two categories:

- LLT timer tunable parameters
  See "About LLT timer tunable parameters" on page 246.
- LLT flow control tunable parameters
  See "About LLT flow control tunable parameters" on page 250.

You can set the LLT tunable parameters either with the `lltconfig` command or in the `/etc/lltab` file. You can use the `lltconfig` command to change a parameter on the local node at run time. Symantec recommends you run the

command on all the nodes in the cluster to change the values of the parameters. To set an LLT parameter across system reboots, you must include the parameter definition in the `/etc/lltab` file. Default values of the parameters are taken if nothing is specified in /etc/lltab. The parameters values specified in the /etc/lltab file come into effect at LLT start-time only. Symantec recommends that you specify the same definition of the tunable parameters in the `/etc/lltab` file of each node.

To set a timer tunable:

- To set a timer tunable parameter using the `lltconfig` command:

  ```
  # lltconfig -T timer tunable:value
  ```

- To set a timer tunable parameter in the `/etc/lltab` file:

  ```
  set-timer  timer tunable:value
  ```

To set a flow control tunable

- To set a flow control tunable parameter using the `lltconfig` command:

  ```
  # lltconfig -F flowcontrol tunable:value
  ```

- To set a flow control tunable parameter in the `/etc/lltab` file:

  ```
  set-flow  flowcontrol tunable:value
  ```

See the `lltconfig`(1M) and `lltab`(1M) manual pages.

About LLT timer tunable parameters

Table 5-3 lists the LLT timer tunable parameters. The timer values are set in .01 sec units. lltconfig –T query can be used to display current timer values.

Table 5-3 LLT timer tunable parameters

LLT parameter	Description	Default	When to change	Dependency with other LLT tunable parameters
peerinact	LLT marks a link of a peer node as “inactive," if it does not receive any packet on that link for this timer interval. Once a link is marked as "inactive," LLT will not send any data on that link.	1600	■ Change this value for delaying or speeding up node/link inactive notification mechanism as per client’s notification processing logic. ■ Increase the value for planned replacement of faulty network cable /switch. ■ In some circumstances, when the private networks links are very slow or the network traffic becomes very bursty, increase this value so as to avoid false notifications of peer death. Set the value to 0 for planned replacement of faulty network cable or faulty switch.	The timer value should always be higher than the peertrouble timer value.
peertrouble	LLT marks a high-pri link of a peer node as "troubled", if it does not receive any packet on that link for this timer interval. Once a link is marked as "troubled", LLT will not send any data on that link till the link is up.		■ In some circumstances, when the private networks links are very slow or nodes in the cluster are very busy, increase the value. ■ Increase the value for planned replacement of faulty network cable /faulty switch.	This timer value should always be lower than peerinact timer value. Also, It should be close to its default value.

Table 5-3 LLT timer tunable parameters *(continued)*

LLT parameter	Description	Default	When to change	Dependency with other LLT tunable parameters
peertroublelo	LLT marks a low-pri link of a peer node as "troubled", if it does not receive any packet on that link for this timer interval. Once a link is marked as "troubled", LLT will not send any data on that link till the link is available.	400	■ In some circumstances, when the private networks links are very slow or nodes in the cluster are very busy, increase the value. ■ Increase the value for planned replacement of faulty network cable /faulty switch.	This timer value should always be lower than peerinact timer value. Also, It should be close to its default value.
heartbeat	LLT sends heartbeat packets repeatedly to peer nodes after every heartbeat timer interval on each high link.	50	In some circumstances, when the private networks links are very slow (or congested) or nodes in the cluster are very busy, increase the value.	This timer value should be lower than peertrouble timer value. Also, it should not be close to peertrouble timer value.
heartbeatlo	LLT sends heartbeat packets repeatedly to peer nodes after every heartbeatlo timer interval on each low pri link.	100	In some circumstances, when the networks links are very slow or nodes in the cluster are very busy, increase the value.	This timer value should be lower than peertroublelo timer value. Also, it should not be close to peertroublelo timer value.
timetoreqhb	LLT attempts to request heartbeat mechanism (explicitly sends 5 hbreqs, special kind of heartbeat requests to the peer node on this link) if it did not receive any packets on this link from that peer node for this timer interval. If the peer node does not respond to the special heartbeat requests, LLT marks this link as “expired” for that peer node. Value can be set within range starting from 0 to (peerinact – 200). If this timer is set to 0, the request heartbeat mechanism is disabled.	1400	Decrease the value of this tunable for speeding up node/link inactive notification mechanism as per client’s notification processing logic. Disable the request heartbeat mechanism by setting the value of this timer to 0 for planned replacement of faulty network cable /switch. In some circumstances, when the private networks links are very slow or the network traffic becomes very bursty, don’t change the value of this timer tunable.	This timer is set to ‘peerinact - 200’ automatically every time when the peerinact timer is changed.

Table 5-3 LLT timer tunable parameters *(continued)*

LLT parameter	Description	Default	When to change	Dependency with other LLT tunable parameters
timetosendhb	LLT sends out of timer context heartbeats to keep the node alive when LLT timer does not run at regular interval. This option specifies the amount of time to wait before sending a heartbeat in case of timer not running. If this timer tunable is set to 0, the out of timer context heartbeating mechanism is disabled.	200	Disable the out of timer context heart-beating mechanism by setting the value of this timer to 0 for planned replacement of faulty network cable /switch. In some circumstances, when the private networks links are very slow or nodes in the cluster are very busy, increase the value	This timer value should not be more than peerinact timer value. Also, it should not be close to the peerinact timer value.
sendhbcap	This value specifies the maximum time for which LLT will send contiguous out of timer context heartbeats.	18000		
oos	If the out-of-sequence timer has expired for a node, LLT sends an appropriate NAK to that node. LLT does not send a NAK as soon as it receives an oos packet. It waits for the oos timer value before sending the NAK.	10	Do not change this value for performance reasons. Lowering the value can result in unnecessary retransmissions/negative acknowledgement traffic. You can increase the value of oos if the round trip time is large in the cluster (for example, campus cluster).	Not applicable
retrans	LLT retransmits a packet if it does not receive its acknowledgement for this timer interval value.	10	Do not change this value. Lowering the value can result in unnecessary retransmissions. You can increase the value of retrans if the round trip time is large in the cluster (for example, campus cluster).	Not applicable

Table 5-3 LLT timer tunable parameters *(continued)*

LLT parameter	Description	Default	When to change	Dependency with other LLT tunable parameters
service	LLT calls its service routine (which delivers messages to LLT clients) after every service timer interval.	100	Do not change this value for performance reasons.	Not applicable
arp	LLT flushes stored address of peer nodes when this timer expires and relearns the addresses.	30000	Do not change this value for performance reasons.	Not applicable
arpreq	LLT sends an arp request when this timer expires to detect other peer nodes in the cluster.	3000	Do not change this value for performance reasons.	Not applicable

About LLT flow control tunable parameters

Table 5-4 lists the LLT flow control tunable parameters. The flow control values are set in number of packets. The `lltconfig -F` query can be used to display current flow control settings.

Table 5-4 LLT flow control tunable parameters

LLT parameter	Description	Default	When to change	Dependency with other LLT tunable parameters
highwater	When the number of packets in transmit queue for a node reaches highwater, LLT is flow controlled.	200	If a client generates data in bursty manner, increase this value to match the incoming data rate. Note that increasing the value means more memory consumption so set an appropriate value to avoid wasting memory unnecessarily. Lowering the value can result in unnecessary flow controlling the client.	This flow control value should always be higher than the lowwater flow control value.

Table 5-4 LLT flow control tunable parameters *(continued)*

LLT parameter	Description	Default	When to change	Dependency with other LLT tunable parameters
lowwater	When LLT has flow controlled the client, it will not start accepting packets again till the number of packets in the port transmit queue for a node drops to lowwater.	100	Not applicable	This flow control value should be lower than the highwater flow control value. The value should not be close the highwater flow control value.
rporthighwater	When the number of packets in the receive queue for a port reaches highwater, LLT is flow controlled.	200	If a client generates data in bursty manner, increase this value to match the incoming data rate. Note that increasing the value means more memory consumption so set an appropriate value to avoid wasting memory unnecessarily. Lowering the value can result in unnecessary flow controlling the client.	This flow control value should always be higher than the rportlowwater flow control value.
rportlowwater	When LLT has flow controlled the client, it will not start accepting packets for that client again till the number of packets in the port receive queue for the port drops to rportlowwater.	100	Not applicable	This flow control value should be lower than the rpothighwater flow control value. The value should not be close the rporthighwater flow control value.
window	This is the maximum number of un-ACKed packets LLT will put in flight.	50	Change the value as per the private networks speed. Lowering the value irrespective of network speed may result in unnecessary retransmission of out of window sequence packets.	This flow control value should not be higher than the difference between the highwater flow control value and the lowwater flow control value. The value of this parameter (window) should be aligned with the value of the bandwidth delay product.

Table 5-4 LLT flow control tunable parameters *(continued)*

LLT parameter	Description	Default	When to change	Dependency with other LLT tunable parameters
linkburst	This flow control value should not be higher than the difference between the highwater flow control value and the lowwater flow control value.	32	For performance reasons, its value should be either 0 or at least 32.	Not applicable
ackval	LLT sends acknowledgement of a packet by piggybacking an ACK packet on the next outbound data packet to the sender node. If there are no data packets on which to piggyback the ACK packet, LLT waits for ackval number of packets before sending an explicit ACK to the sender.	10	Do not change this value for performance reasons. Increasing the value can result in unnecessary retransmissions.	Not applicable
sws	To avoid Silly Window Syndrome, LLT transmits more packets only when the count of un-acked packet goes to below of this tunable value.	40	For performance reason, its value should be changed whenever the value of the window tunable is changed as per the formula given below: sws = window *4/5.	Its value should be lower than that of window. Its value should be close to the value of window tunable.

About VXFEN tunable parameters

The section describes the VXFEN tunable parameters and how to reconfigure the VXFEN module.

Table 5-5 describes the tunable parameters for the VXFEN driver.

Table 5-5 VXFEN tunable parameters

vxfen Parameter	Description and Values: Default, Minimum, and Maximum
dbg_log_size	Size of debug log in bytes ■ Values Default: 131072 (128 KB) Minimum: 65536 (64 KB) Maximum: 524288 (512 KB)

Table 5-5 VXFEN tunable parameters *(continued)*

vxfen Parameter	Description and Values: Default, Minimum, and Maximum
vxfen_max_delay	Specifies the maximum number of seconds that the smaller sub-cluster waits before racing with larger sub-clusters for control of the coordinator disks when a split-brain occurs. This value must be greater than the vxfen_max_delay value. ■ Values Default: 60 Minimum: 1 Maximum: 600
vxfen_min_delay	Specifies the minimum number of seconds that the smaller sub-cluster waits before racing with larger sub-clusters for control of the coordinator disks when a split-brain occurs. This value must be smaller than the vxfen_max_delay value. ■ Values Default: 1 Minimum: 1 Maximum: 600
vxfen_vxfnd_tmt	Specifies the time in seconds that the I/O fencing driver VxFEN waits for the I/O fencing daemon VXFEND to return after completing a given task. ■ Values Default: 60 Minimum: 10 Maximum: 600

In the event of a network partition, the smaller sub-cluster delays before racing for the coordinator disks. The time delay allows a larger sub-cluster to win the race for the coordinator disks. The vxfen_max_delay and vxfen_min_delay parameters define the delay in seconds.

Configuring the VXFEN module parameters

After adjusting the tunable kernel driver parameters, you must reconfigure the VXFEN module for the parameter changes to take effect.

The following example procedure changes the value of the vxfen_min_delay parameter.

On each Linux node, edit the file /etc/sysconfig/vxfen to change the value of the vxfen driver tunable global parameters, vxfen_max_delay and vxfen_min_delay.

Note: You must restart the VXFEN module to put any parameter change into effect.

To configure the VxFEN parameters and reconfigure the VxFEN module

1. Shut down all Oracle service groups on the node.

   ```
   # hagrp -offline oragrp -sys galaxy
   ```

2. Stop all Oracle client processes, such as sqlplus, svrmgrl, and gsd, on the node.
3. Stop all the applications that are not configured under VCS. Use native application commands to stop the application.
4. Stop VCS on all the nodes. Run the following command on each node:

   ```
   # hastop -local
   ```

5. Stop the VxFEN driver.

   ```
   # /etc/init.d/vxfen stop
   ```

6. Edit the /etc/sysconfig/vxfen file.

   For example, change the entry from:

   ```
   vxfen_min_delay=1
   ```

   to:

   ```
   vxfen_min_delay=30
   ```

7. Start the VXFEN module.

   ```
   # /etc/init.d/vxfen start
   ```

8. Bring the service groups online.

   ```
   # hagrp -online oragrp -sys galaxy
   ```

9. Start all the applications that are not configured under VCS. Use native application commands to start the applications.

Tuning guidelines for campus clusters

An important consideration while tuning an SF Oracle RAC campus cluster is setting the LLT peerinact time. Follow the guidelines below to determine the optimum value of peerinact time:

- Calculate the roundtrip time using lltping (1M).
- Evaluate LLT heartbeat time as half of the round trip time.
- Set the LLT peer trouble time as 2-4 times the heartbeat time.
- LLT peerinact time should be set to be more than 4 times the heart beat time.

Section 3

Reference

- Appendix A. List of SF Oracle RAC health checks
- Appendix B. Error messages

Appendix A

List of SF Oracle RAC health checks

This appendix includes the following topics:

- LLT health checks
- I/O fencing health checks
- PrivNIC health checks
- Oracle Clusterware health checks
- CVM, CFS, and ODM health checks

LLT health checks

This section lists the health checks performed for LLT, the messages displayed for each check, and a brief description of the check.

Note: Warning messages indicate issues in the components or the general health of the cluster.

For recommendations on resolving the issues, see the troubleshooting chapter in this document.

Table A-1 lists the health checks performed for LLT.

Table A-1 List of health checks for LLT

List of health checks	Message	Description
LLT timer subsystem scheduling check	`Warning: OS timer is not called for num seconds`	Checks whether the LLT module runs in accordance with the scheduled operating system timer. The message indicates that the operating system timer is not called for the specified interval. The parameter `timer_threshold` contains the optimum threshold for this check.
OS memory availability check for the packet transmission	`Warning: Kernel failed to allocate memory num time(s)`	Checks whether the kernel has allocated sufficient memory to LLT for cluster communication. The message indicates that the kernel attempts at allocating the requisite memory has failed (*num*) times. The parameter `no_mem_to_xmit_allow` contains the optimum threshold for this check.
Flow control status monitoring	Flow-control occurred *num* time(s)and back-enabled *num* time(s) on port *port number* for node *node number*	Checks whether LLT has sufficient bandwidth to accept incoming data packets and transmit the data packets. Flow control also depends on the peer nodes' ability to accept incoming data traffic. The message indicates that packet transmission and reception was controlled (*num*) and normalized (*num*) times. The parameter `max_canput` contains the optimum threshold for this check.

Table A-1 List of health checks for LLT *(continued)*

List of health checks	Message	Description
Flaky link monitoring	`Warning: Connectivity with node node id on link link id is flaky num time(s).` `The distribution is: (0-4 s) <num> (4-8 s) <num> (8-12 s) <num> (12-16 s) <num> (>=16 s)`	Checks whether the private interconnects are stable. The message indicates that connectivity (*link*) with the peer node (*node id*) is monitored (*num*) times within a stipulated duration (for example, 0-4 seconds). The parameter `flakylink_allow` contains the optimum threshold for this check.
Node status check	`One or more link connectivity with peer node(s) node name is in trouble.`	Checks the current node membership. The message indicates connectivity issues with the peer node (*node name*). This check does not require a threshold.
Link status check	`Link connectivity with node id is on only one link. Symantec recommends configuring a minimum of 2 links.`	Checks the number of private interconnects that are currently active. The message indicates that only one private interconnect exists or is operational between the local node and the peer node *node id*. This check does not require a threshold.
Connectivity check	`Only one link is configured under LLT. Symantec recommends configuring a minimum of 2 links.`	Checks the number of private interconnects that were configured under LLT during configuration. The message indicates that only one private interconnect exists or is configured under LLT. This check does not require a threshold.

Table A-1 List of health checks for LLT *(continued)*

List of health checks	Message	Description

LLT packet related checks	`Retransmitted % percentage of total transmitted packets.`
	`Sent % percentage of total transmitted packet when no link is up.`
	`% percentage of total received packets are with bad checksum.`
	`% percentage of total received packets are out of window.`
	`% percentage of total received packets are misaligned.`
		Checks whether the data packets transmitted by LLT reach peer nodes without any error. If there is an error in packet transmission, it indicates an error in the private interconnects.
		The message indicates the percentage of data packets transmitted or received and the associated transmission errors, such as bad checksum and failed link.
		The following parameters contain the optimum thresholds for this check: `retrans_pct`, `send_nolinkup_pct`, `recv_oow_pct`, `recv_misaligned_pct`, `recv_badcksum_pct`

DLPI layer related checks	`% percentage of total received packets are with DLPI error.`	Checks whether the data link layer (DLP) is causing errors in packet transmission. The message indicates that some of the received packets (%) contain DLPI error. You can set a desired percentage value in the configuration file. The parameter `recv_dlpierror_pct` contains the optimum threshold for this check.

Table A-1 List of health checks for LLT *(continued)*

List of health checks	Message	Description
Traffic distribution over the links	`Traffic distribution over links: %% Send data on link`*`num percentage`* `%% Recv data on link`*`num percentage`*	Checks the distribution of traffic over all the links configured under LLT. The message displays the percentage of data (%) sent and recd on a particular link (*num*) This check does not require a threshold.
LLT Ports status check	*`%per`* `of total transmitted packets are with large xmit latency (>16ms) for port` *`port id`* *`%per`* `received packets are with large recv latency (>16ms) for port` *`port id`*.	Checks the latency period for transmitting or receiving packets. The message indicates that some percentage (%) of the transmitted/received packets exceed the stipulated latency time. The following parameters contain the optimum thresholds for this check: `hirecvlatencycnt_pct`, `hixmitlatencycnt_pct`.
System load monitoring	`Load Information: Average :` *`num`*,***num***,***num***	Monitors the system workload at the stipulated periodicity (1 second, 5 seconds, 15 seconds) This check does not require a threshold.

I/O fencing health checks

This section lists the health checks performed for I/O fencing, the messages displayed for each check, and a brief description of the check.

Note: Warning messages indicate issues in the components or the general health of the cluster.

For recommendations on resolving the issues, see the troubleshooting chapter in this document.

Table A-2 lists the health checks performed for I/O fencing.

Table A-2 List of health checks for I/O fencing

List of health checks	Message	Description
Checks related to the working status of VxFEN	`VxFEN is not configured in SCSI3 mode.` `VxFEN is not running on all nodes in the cluster.`	Checks whether Symantec fencing is configured in SCSI3 mode and running on all nodes in the cluster.

Checks related to coordinator disk health	`VxFEN is running with only one coordinator disk. Loss of this disk will prevent cluster reconfiguration on loss of a node. Symantec recommends configuring a minimum of 3 coordinator disks.`
	`VxFEN is running with even number of coordinator disks. There must be odd number of coordinator disks.`
	`Replace the disk` *`disk name`* `using OCDR procedure.`
		Checks how many coordinator disks are configured for Symantec fencing and the status of the disks.

Verify the keys on the coordinator disks	`The coordinator disk (`*`disk name`*`) does not have the required key for the local node.`	Checks whether the coordinator disks configured under Symantec fencing has the key of the current node.
Verify SCSI3 write-exclusive reservation on shared disk	`SCSI3 write-exclusive reservation is missing on shared disk (`*`disk_name`*`)`	Checks if the shared disk is accessible to the node for write operations.

PrivNIC health checks

This section lists the health checks performed for PrivNIC, the messages displayed for each check, and a brief description of the check.

Note: Warning messages indicate issues in the components or the general health of the cluster.

For recommendations on resolving the issues, see the troubleshooting chapter in this document.

Table A-3 lists the health checks performed for PrivNIC.

Table A-3 List of health checks for PrivNIC

List of health checks	Message	Description
Checks related to the working status of VCS	`VCS is not running on the local system.`	Checks whether VCS is running or not.
Checks related to the configuration of the PrivNIC resources	`No PrivNIC resource is configured under VCS.`	Checks whether the PrivNIC resource is configured under VCS.
Checks related to the status of the PrivNIC resources	If the PrivNIC resource is not online: `The PrivNIC resource` *`name`* `is not online.`	Checks whether the PrivNIC resources configured under VCS are online.
Compare the NICs used by PrivNIC with the NICs configured under LLT	For PrivNIC resources: `Mismatch between LLT links` *`llt nics`* `and PrivNIC links` *`private nics`*`.`	Checks whether the NICs used by the PrivNIC resource have the same interface (*private nics*) as those configured as LLT links (*llt nics*).
Cross check NICs used by PrivNIC with the NICs used by Oracle Clusterware.	For PrivNIC resources: `Mismatch between Oracle Clusterware links` *`crs nics`* `and PrivNIC links` *`private nics`*`.`	Checks whether the private interconnect used for Oracle Clusterware is the same as the NIC configured under the PrivNIC resource.

Oracle Clusterware health checks

This section lists the health checks performed for Oracle Clusterware, the messages displayed for each check, and a brief description of the check.

Note: Warning messages indicate issues in the components or the general health of the cluster.

For recommendations on resolving the issues, see the troubleshooting chapter in this document.

Table A-4 lists the health checks performed for the Oracle Clusterware module.

Table A-4 List of health checks for the Oracle Clusterware module

List of health checks	Message	Description
Verify the working status of CRS	`Oracle Clusterware is not running.`	Checks whether Oracle Clusterware is up and running.
Verify the working status of CSSD resource	`No CSSD resource is configured under VCS.` `The CSSD resource` *`name`* `is not running.`	Checks whether the CSSD resource is configured under VCS and if the resource is online.
Compare the NICs used by CRS with the NICs configured under LLT	`Mismatch between LLT links` *`llt_nic1, llt_nic2`* `and Oracle Clusterware links` *`crs_nic1, crs_nic2`*`.`	Checks whether the private interconnects used by Oracle Clusterware are the same as the LLT links (*llt nics*).
Compare the NICs used by CRS with the NICs monitored by PrivNIC	`Mismatch between Oracle Clusterware links` *`crs nics`* `and PrivNIC links` *`private nics`*`.`	Checks whether the private interconnects (*crs nics*) used by Oracle Clusterware are monitored by the PrivNIC resource (*private nics*).
Compare the nodes configured for CRS with the nodes listed by llt commands.	`Mismatch between CRS nodes` *`crs nodes`* `and LLT nodes` *`llt nodes`*`.`	Checks whether the host names configured during the Oracle Clusterware installation match the list of host names displayed on running the LLT command:`lltstat -nvv`.

CVM, CFS, and ODM health checks

This section lists the health checks performed for CVM, CFS, and ODM, the messages displayed for each check, and a brief description of the check.

Note: Warning messages indicate issues in the components or the general health of the cluster.

For recommendations on resolving the issues, see the troubleshooting chapter in this document.

Table A-5 lists the health checks performed for CVM, CFS, and ODM modules.

Table A-5 List of health checks for CVM, CFS, and ODM

List of health checks	Message	Description
Verify CVM status	`CVM is not running`	Checks whether CVM is running in the cluster.
Verify CFS status	`CFS is not running`	Checks whether CFS is running in the cluster.
Verify ODM status	`ODM is not running`	Checks whether ODM is running in the cluster.

Appendix B

Error messages

This appendix includes the following topics:

- About error messages
- VxVM error messages
- VXFEN driver error messages

About error messages

Error messages can be generated by the following software modules:

- Veritas Volume Manager (VxVM)
- Veritas Fencing (VXFEN) driver

VxVM error messages

Table B-1 contains VxVM error messages that are related to I/O fencing.

Table B-1 VxVM error messages for I/O fencing

Message	Explanation
vold_pgr_register(disk_path): failed to open the vxfen device. Please make sure that the vxfen driver is installed and configured.	The vxfen driver is not configured. Follow the instructions to set up these disks and start I/O fencing. You can then clear the faulted resources and bring the service groups online.
vold_pgr_register(disk_path): Probably incompatible vxfen driver.	Incompatible versions of VxVM and the vxfen driver are installed on the system. Install the proper version of SF Oracle RAC.

VXFEN driver error messages

Table B-2 contains VXFEN driver error messages. In addition to VXFEN driver error messages, informational messages can also be displayed.

See "VXFEN driver informational message" on page 270.

See "Node ejection informational messages" on page 271.

Table B-2 VXFEN driver error messages

Message	Explanation
Unable to register with coordinator disk with serial number: xxxx	This message appears when the vxfen driver is unable to register with one of the coordinator disks. The serial number of the coordinator disk that failed is displayed.
Unable to register with a majority of the coordinator disks. Dropping out of cluster.	This message appears when the vxfen driver is unable to register with a majority of the coordinator disks. The problems with the coordinator disks must be cleared before fencing can be enabled. This message is preceded with the message "VXFEN: Unable to register with coordinator disk with serial number xxxx."
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!	This message appears when there is a preexisting split-brain in the cluster. In this case, the configuration of vxfen driver fails. Clear the split-brain using the instructions given in the chapter on Troubleshooting SF Oracle RAC before configuring vxfen driver.
Unable to join running cluster since cluster is currently fencing a node out of the cluster	This message appears while configuring the vxfen driver, if there is a fencing race going on in the cluster. The vxfen driver can be configured by retrying after some time (after the cluster completes the fencing).

VXFEN driver informational message

The following informational message appears when a node is ejected from the cluster to prevent data corruption when a split-brain occurs.

```
VXFEN CRITICAL V-11-1-20 Local cluster node ejected from cluster
to prevent potential data corruption
```

Node ejection informational messages

Informational messages may appear on the console of one of the cluster nodes when a node is ejected from a disk or LUN.

These informational messages can be ignored.

Glossary

Agent	A process that starts, stops, and monitors all configured resources of a type, and reports their status to VCS.
Authentication Broker	The Veritas Security Services component that serves, one level beneath the root broker, as an intermediate registration authority and a certification authority. The authentication broker can authenticate clients, such as users or services, and grant them a certificate that will become part of the Veritas credential. An authentication broker cannot, however, authenticate other brokers. That task must be performed by the root broker.
Cluster	A cluster is one or more computers that are linked together for the purpose of multiprocessing and high availability. The term is used synonymously with VCS cluster, meaning one or more computers that are part of the same GAB membership.
CVM (cluster volume manager)	The cluster functionality of Veritas Volume Manager.
Disaster Recovery	Administrators with clusters in physically disparate areas can set the policy for migrating applications from one location to another if clusters in one geographic area become unavailable due to an unforeseen event. Disaster recovery requires heartbeating and replication.
disk array	A collection of disks logically arranged into an object. Arrays tend to provide benefits such as redundancy or improved performance.
DMP (Dynamic Multi-Pathing)	A feature designed to provide greater reliability and performance by using path failover and load balancing for multiported disk arrays connected to host systems through multiple paths. DMP detects the various paths to a disk using a mechanism that is specific to each supported array type. DMP can also differentiate between different enclosures of a supported array type that are connected to the same host system.
DST (Dynamic Storage Tiering)	A feature with which administrators of multi-volume VxFS file systems can manage the placement of files on individual volumes in a volume set by defining placement policies that control both initial file location and the circumstances under which existing files are relocated. These placement policies cause the files to which they apply to be created and extended on specific subsets of a file system's volume set, known as placement classes. The files are relocated to volumes in other placement

	classes when they meet specified naming, timing, access rate, and storage capacity-related conditions. See also Veritas File System (VxFS)
Failover	A failover occurs when a service group faults and is migrated to another system.
GAB (Group Atomic Broadcast)	A communication mechanism of the VCS engine that manages cluster membership, monitors heartbeat communication, and distributes information throughout the cluster.
HA (high availability)	The concept of configuring the product to be highly available against system failure on a clustered network using Veritas Cluster Server (VCS).
HAD (High Availability Daemon)	The core VCS process that runs on each system. The HAD process maintains and communicates information about the resources running on the local system and receives information about resources running on other systems in the cluster.
IP address	An identifier for a computer or other device on a TCP/IP network, written as four eight-bit numbers separated by periods. Messages and other data are routed on the network according to their destination IP addresses. See also virtual IP address
Jeopardy	A node is in jeopardy when it is missing one of the two required heartbeat connections. When a node is running with one heartbeat only (in jeopardy), VCS does not restart the applications on a new node. This action of disabling failover is a safety mechanism that prevents data corruption.
latency	For file systems, this typically refers to the amount of time it takes a given file system operation to return to the user.
LLT (Low Latency Transport)	A communication mechanism of the VCS engine that provides kernel-to-kernel communications and monitors network communications.
logical volume	A simple volume that resides on an extended partition on a basic disk and is limited to the space within the extended partitions. A logical volume can be formatted and assigned a drive letter, and it can be subdivided into logical drives. See also LUN
LUN	A LUN, or logical unit, can either correspond to a single physical disk, or to a collection of disks that are exported as a single logical entity, or virtual disk, by a device driver or by an intelligent disk array's hardware. VxVM and other software modules may be capable of automatically discovering the special characteristics of LUNs, or you can use disk tags to define new storage attributes. Disk tags are administered by using the `vxdisk` command or the graphical user interface.
main.cf	The file in which the cluster configuration is stored.

mirroring	A form of storage redundancy in which two or more identical copies of data are maintained on separate volumes. (Each duplicate copy is known as a mirror.) Also RAID Level 1.
Node	The physical host or system on which applications and service groups reside. When systems are linked by VCS, they become nodes in a cluster.
resources	Individual components that work together to provide application services to the public network. A resource may be a physical component such as a disk group or network interface card, a software component such as a database server or a Web server, or a configuration component such as an IP address or mounted file system.
Resource Dependency	A dependency between resources is indicated by the keyword "requires" between two resource names. This indicates the second resource (the child) must be online before the first resource (the parent) can be brought online. Conversely, the parent must be offline before the child can be taken offline. Also, faults of the children are propagated to the parent.
Resource Types	Each resource in a cluster is identified by a unique name and classified according to its type. VCS includes a set of pre-defined resource types for storage, networking, and application services.
root broker	The first authentication broker, which has a self-signed certificate. The root broker has a single private domain that holds only the names of brokers that shall be considered valid.
SAN (storage area network)	A networking paradigm that provides easily reconfigurable connectivity between any subset of computers, disk storage and interconnecting hardware such as switches, hubs and bridges.
Service Group	A service group is a collection of resources working together to provide application services to clients. It typically includes multiple resources, hardware- and software-based, working together to provide a single service.
Service Group Dependency	A service group dependency provides a mechanism by which two service groups can be linked by a dependency rule, similar to the way resources are linked.
Shared Storage	Storage devices that are connected to and used by two or more systems.
shared volume	A volume that belongs to a shared disk group and is open on more than one node at the same time.
SFCFS (Storage Foundation Cluster File System)	
SNMP Notification	Simple Network Management Protocol (SNMP) developed to manage nodes on an IP network.

State	The current activity status of a resource, group or system. Resource states are given relative to both systems.
Storage Checkpoint	A facility that provides a consistent and stable view of a file system or database image and keeps track of modified data blocks since the last Storage Checkpoint.
System	The physical system on which applications and service groups reside. When a system is linked by VCS, it becomes a node in a cluster. See Node
types.cf	A file that describes standard resource types to the VCS engine; specifically, the data required to control a specific resource.
VCS (Veritas Cluster Server)	An open systems clustering solution designed to eliminate planned and unplanned downtime, simplify server consolidation, and allow the effective management of a wide range of applications in multiplatform environments.
Virtual IP Address	A unique IP address associated with the cluster. It may be brought up on any system in the cluster, along with the other resources of the service group. This address, also known as the IP alias, should not be confused with the base IP address, which is the IP address that corresponds to the host name of a system.
VxFS (Veritas File System)	A component of the Veritas Storage Foundation product suite that provides high performance and online management capabilities to facilitate the creation and maintenance of file systems. A file system is a collection of directories organized into a structure that enables you to locate and store files.
VxVM (Veritas Volume Manager)	A Symantec product installed on storage clients that enables management of physical disks as logical devices. It enhances data storage management by controlling space allocation, performance, data availability, device installation, and system monitoring of private and shared systems.
VVR (Veritas Volume Replicator)	A data replication tool designed to contribute to an effective disaster recovery plan.

Index

Symbols

/sbin/vcsmmconfig
    starting VCSMM 221

A

agents
    intelligent resource monitoring 36
    poll-based resource monitoring 36
AMF driver 36

C

Changing the CVM master 152
cluster
    Group Membership Services/Atomic Broadcast (GAB) 28
    interconnect communication channel 25
Cluster File System (CFS)
    architecture 32
    communication 33
    overview 32
Cluster master node
    changing 152
Cluster Volume Manager (CVM)
    architecture 30
    communication 31
    overview 30
commands
    format (verify disks) 216
    vxdctl enable (scan disks) 216
communication
    communication stack 24
    data flow 23
    GAB and processes port relationship 29
    Group Membership Services/Atomic Broadcast GAB 28
    interconnect communication channel 25
    requirements 24
coordination point definition 54
coordinator disks
    DMP devices 45
coordinator disks *(continued)*
    for I/O fencing 45
CP server
    deployment scenarios 55
    migration scenarios 55
CP server user privileges 133
cpsadm command 134
CVM master
    changing 152

D

data corruption
    preventing 42
data disks
    for I/O fencing 45
drivers
    tunable parameters 237

E

environment variables
    MANPATH 74
error messages
    node ejection 271
    VxVM errors related to I/O fencing 269

F

fire drills
    about 157
    disaster recovery 157
    for global clusters 157
format command 216

G

GAB
    tunable parameters 238
GAB tunable parameters
    dynamic 240
        Control Seed port 240
        Gab queue limit 240
        Halt on process death 240

GAB tunable parameters *(continued)*
dynamic *(continued)*
Halt on rejoin 240
IOFENCE timeout 240
Isolate timeout 240
Keep on killing 240
Kill_ntries 240
Missed heartbeat halt 240
Quorum flag 240
Stable timeout 240
static 238
flowctrl 238
gab_conn_wait 238
gab_kill_ntries 238
gab_kstat_size 238
isolate_time 238
logbufsize 238
msglogsize 238
numnids 238
numports 238
getcomms
troubleshooting 166
getdbac
troubleshooting script 166

H

hagetcf (troubleshooting script) 166

I

I/O fencing
communication 44
operations 44
preventing data corruption 42
testing and scenarios 48
intelligent resource monitoring
disabling 100
enabling 100
IP address
troubleshooting VIP configuration 223

K

kernel
tunable driver parameters 237

L

LLT
about 25
tunable parameters 245
log files 205

M

MANPATH environment variable 74
Master node
changing 152
messages
node ejected 271
VXFEN driver error messages 270

O

Oracle Clusterware installation
removing Oracle Clusterware if installation fails 223
Oracle Disk Manager (ODM)
overview 34
Oracle instance
definition 19
Oracle patches
applying 79

R

reservations
description 43

S

SCSI-3 PR 43
secure communication 66
security 65
server-based fencing
replacing coordination points
online cluster 143
SF Oracle RAC
about 17
architecture 19, 22
communication infrastructure 23
error messages 269
high-level functionality 19
overview of components 23
tunable parameters of kernel drivers 237
SF Oracle RAC components
Cluster Volume Manager (CVM) 30
SF Oracle RAC installation
pre-installation tasks
setting MANPATH 74
SFRAC
tunable parameters 237
Switching the CVM master 152

T

troubleshooting
    CVMVolDg 218
    error when starting Oracle instance 221
    File System Configured Incorrectly for ODM 226
    getcomms 166
        troubleshooting script 166
    getdbac 166
    hagetcf 166
    Oracle log files 222
    overview of topics 215, 219–220, 226
    restoring communication after cable disconnection 216
    running scripts for analysis 166
    scripts 166
    SCSI reservation errors during bootup 196
    shared disk group cannot be imported 216

V

VCSMM
    vcsmmconfig command 221
vxdctl command 216
VXFEN driver error messages 270
VXFEN driver informational message 270
VxVM
    error messages related to I/O fencing 269
VxVM (Volume Manager)
    errors related to I/O fencing 269