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

Linux

5.1 Platform Release 1

symantec™

Veritas Storage Foundation™ for Oracle RAC Administrator's Guide

Product version: 5.1 PR1

Document version: 5.1PR1.0

Legal Notice

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

Documentation feedback

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:

sfha_docs@symantec.com

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

Contents

Section 1

SF Oracle RAC concepts and administration

Chapter 1

Overview of Veritas Storage Foundation for Oracle RAC

This chapter includes the following topics:

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 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 and raw 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.

- Support for sharing all types of files, in addition to Oracle database files, across nodes.

How SF Oracle RAC works (high-level perspective)

Real Application Clusters (RAC) is a parallel database environment that takes advantage of the processing power of multiple computers. The Oracle database is the physical data stored in tablespaces on disk, while 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 to write changed data to disk.

In traditional environments, only one instance accesses a database at a specific time. SF Oracle RAC enables all nodes to concurrently run Oracle instances and execute transactions against the same database. This software 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.

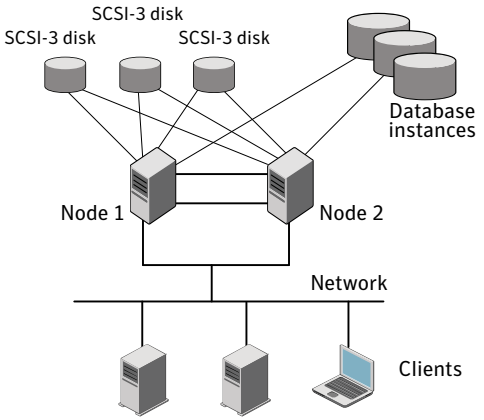
At a conceptual level, SF Oracle RAC is a cluster that manages applications (instances), networking, and storage components using resources contained in service groups. SF Oracle RAC clusters have the following properties:

- 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 below displays the basic layout and individual components required for a SF Oracle RAC installation. This basic layout includes the following components:

- Nodes that form an application cluster and are connected to both the coordinator disks and databases
- Databases for storage and backup
- SCSI-3 Coordinator disks used for I/O fencing

Figure 1-1 SF Oracle RAC basic layout and components

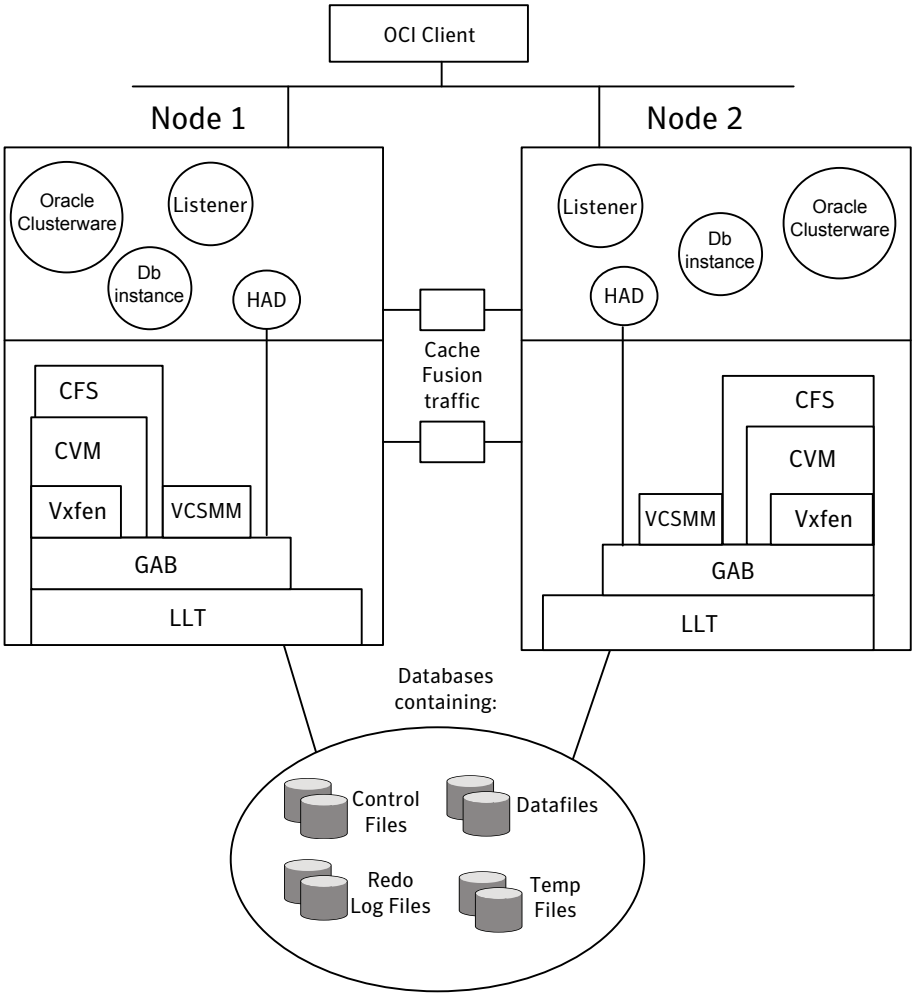


SF Oracle RAC adds the following technologies to a failover cluster environment, which are engineered specifically to improve performance, availability, and manageability of Oracle RAC environments:

- 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 Clusterware 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 failover database because another Oracle instance is already up and running. The recovery process involves applying outstanding redo log entries from the failed node.

Component products and processes of SF Oracle RAC

To understand how SF Oracle RAC manages database instances running in parallel on multiple nodes, review the architecture and communication mechanisms that provide the infrastructure for 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 25.
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 27.
Cluster Server (VCS)	Uses technology from Veritas Cluster Server to manage Oracle RAC databases and infrastructure components. See "Veritas Cluster Server" on page 30.
Database Accelerator	Provides the interface with the Oracle Disk Manager (ODM) API. See "Oracle Disk Manager" on page 29.
RAC Extensions	Manages cluster membership and communications between cluster nodes. See "RAC extensions" on page 33.

Communication infrastructure

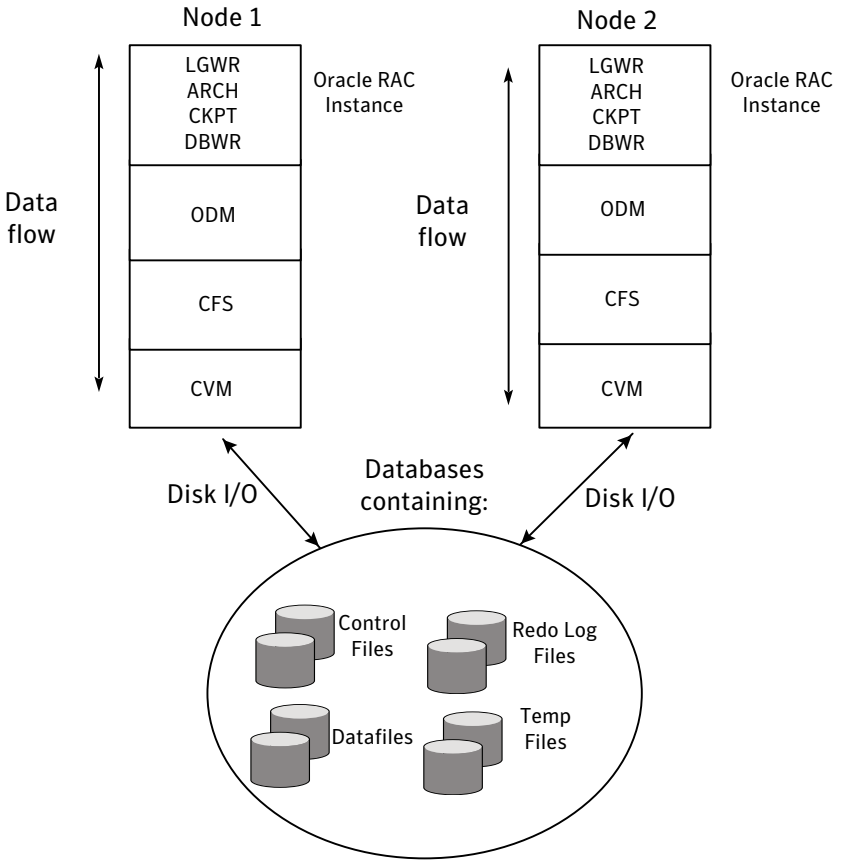
To understand the communication infrastructure, review the data flow and communication requirements.

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

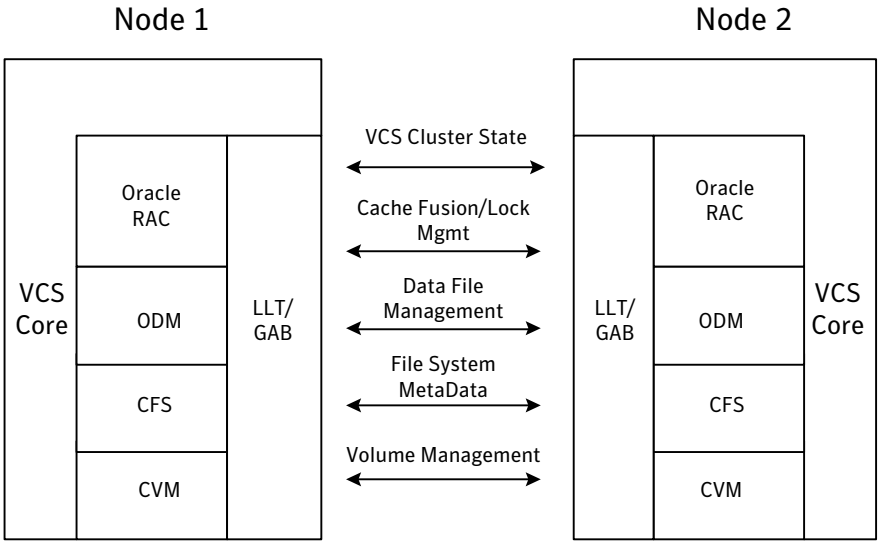


Communication requirements

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.

Low Latency Transport

LLT provides fast, kernel-to-kernel communications and monitors network connections. LLT functions as a high performance replacement for the IP stack and runs directly on top of the Data Link Protocol Interface (DLPI) layer. The use of LLT rather than IP removes latency and overhead associated with the IP stack.

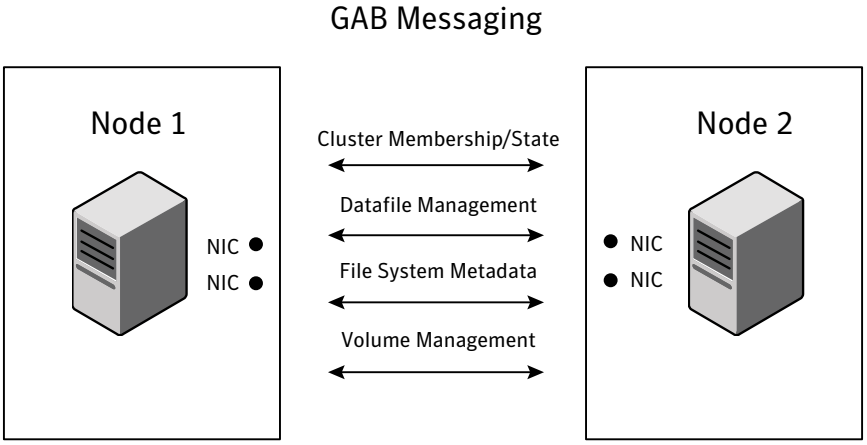
The major functions of LLT are traffic distribution and heartbeats:

Group membership services/Atomic Broadcast

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

Figure 1-5 shows the cluster communication using GAB messaging.

Figure 1-5 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.

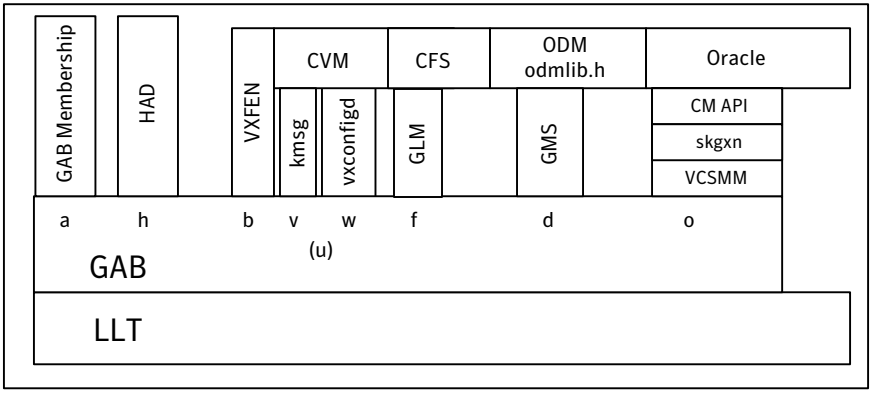
- Cluster communications
  GAB provides reliable cluster communication between SF Oracle RAC modules. GAB provides guaranteed delivery of point-to-point messages and broadcast messages to all nodes. Point-to-point messaging involves sending and acknowledging the message. Atomic-broadcast messaging ensures 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 becomes unregistered.

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 1-6 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.

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.

See Figure 1-6 on page 24.

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:

See Figure 1-6 on page 24.

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.

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.

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

See Figure 1-6 on page 24.

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 other ODM instances to support the file management features of Oracle Managed Files (OMF). OMF enables DBAs to set init.ora parameters for db data file, control file, and log file names and for those structures to be named automatically. OMF allows for the automatic deletion of physical data files when DBAs remove tablespaces.

For an illustration of the ODM and port d, see Figure 1-6.

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 agents.

VCS communication

SF Oracle RAC 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.

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

Review the following information on Oracle Clusterware, the Oracle Cluster Registry, application resources, and the voting disk.

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

Oracle Clusterware

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

Oracle Clusterware 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.
- Shared storage accessible by each node.

Oracle Cluster Registry

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

This is roughly analogous to the main.cf file and in-memory configuration in VCS. However, only one process performs I/O to the OCR file on disk.

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 control
- Rules that define dependencies between the Oracle Clusterware resources
- The current state of the cluster

The OCR data exists on a shared raw volume or a cluster file system that is accessible to each node. Use CVM mirrored volumes to protect OCR data from failures. Oracle Clusterware 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. This way, if you encounter configuration problems and are unable to restart Oracle Clusterware, you can restore the original contents.

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

Application Resources

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

Unlike VCS, Oracle Clusterware 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 Node Applications

Oracle Clusterware 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 installation.

Voting Disk

The voting disk is a heartbeat mechanism used by Oracle Clusterware to maintain cluster node membership. Voting disk data exists on a shared raw volume or a cluster file system that is accessible to each node.

The ocssd processes of Oracle Clusterware 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 determines the cluster is in a split-brain condition and panics the node.

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.

For an illustration of the connection between VCSMM, the skgxn library, and port o, see Figure 1-6.

Oracle and cache fusion traffic

Private IP address types 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.

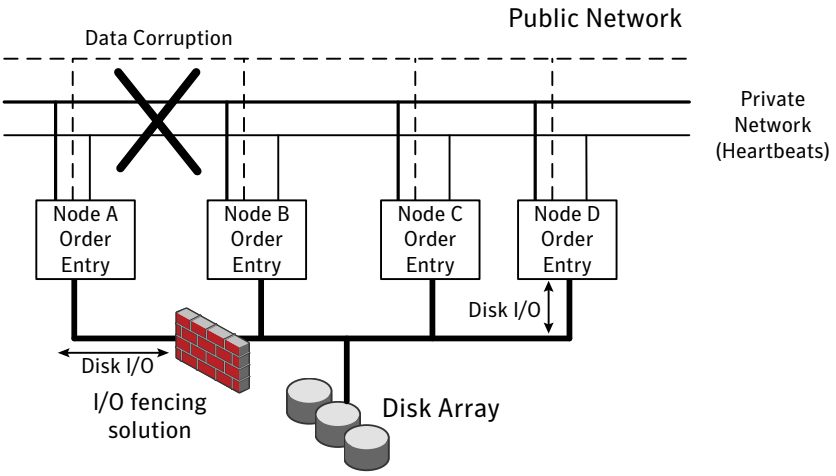
Problems arise when the mechanism that detects the failure breaks down because symptoms appear identical to those of a failed node. For example, if a system in a two-node cluster fails, the system stops sending heartbeats over the private interconnects. The remaining node then takes corrective action. 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 1-7 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 1-7 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 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.

See "Low-level communication: port relationship between GAB and processes" on page 24.

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 37.
- Coordination points—Act as a global lock during membership changes
  See "About coordination points" on page 37.

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.

The coordination points can either be disks or servers or both. Typically, a cluster must have three coordination points.

- 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 Multipathing (DMP) feature. Dynamic Multipathing (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 activeSF 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.

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 132.

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 132.
The disk array containing two of the three coordinator disks is powered off. Node B leaves the cluster and the disk array is still powered off.	Node A continues to operate as long as no nodes leave the cluster. Node A races for a majority of coordinator disks. Node A fails because only one of three coordinator disks is available. Node A panics and removes itself from the cluster.	Node B continues to operate as long as no nodes leave the cluster. Node B leaves the cluster.	Power on failed disk array and restart I/O fencing driver to enable Node A to register with all coordinator disks.

About CP server

This section discusses the CP server features.

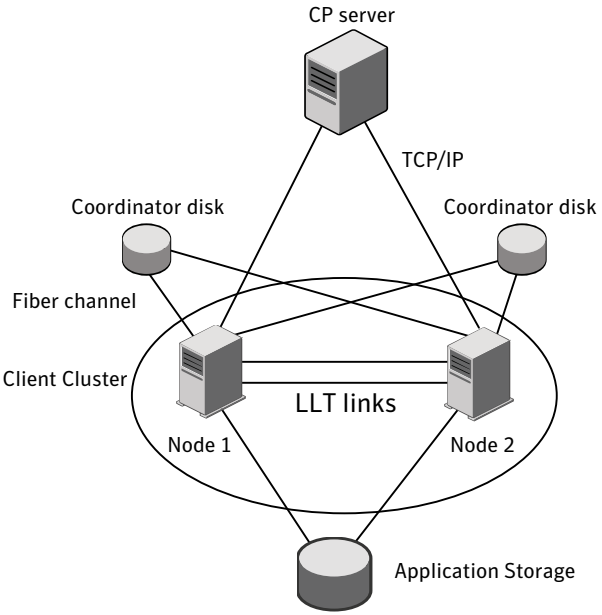
The following CP server features are described:

- 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 1-8 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 1-8 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:

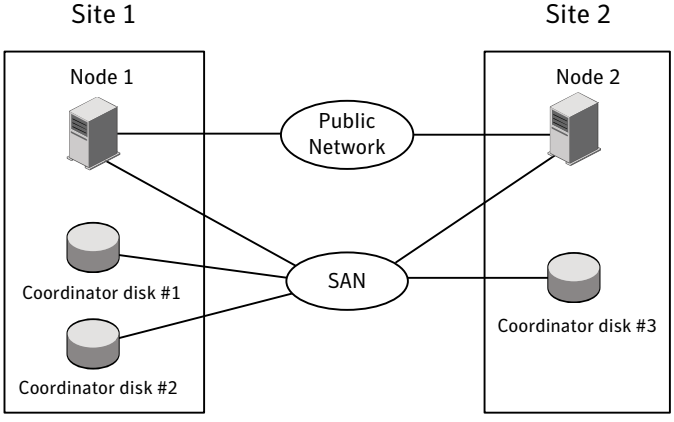
- CP server configurations are scalable, and a configuration with three CP servers can provide I/O fencing for multiple SF Oracle RAC clusters. Since a single CP server configuration can serve a large number of SF Oracle RAC clusters, the cost of multiple SF Oracle RAC cluster deployments can be significantly 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 1-9 on page 43..

  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 1-9 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, as well as the required procedures to be performed on the SF Oracle RAC cluster and CP server node or cluster.

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.

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 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 the SF Oracle RAC cluster nodes 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 107.
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 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 107.

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
Note: This procedure incurs application downtime on the SF Oracle RAC cluster.
	New CP server	Existing SF Oracle RAC cluster with fencing configured in disabled 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.
			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
Note: This procedure incurs application downtime.
	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.
			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 -all`
			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.

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 105.

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.

Recommended CP server configurations

This section discusses the following recommended CP server configurations:

- A CP server configuration where multiple SF Oracle RAC clusters use 3 CP servers as their coordination points
- A CP server configuration where multiple SF Oracle RAC clusters use a single CP server and multiple pairs of coordinator disks (2) as their coordination points

Although the recommended CP server configurations use three coordination points, three or more odd number of coordination points may be used for I/O fencing. In a configuration where multiple SF Oracle RAC clusters share a common set of CP server coordination points, the SF Oracle RAC cluster as well as the CP server use a Universally Unique Identifier (UUID) to uniquely identify a SF Oracle RAC cluster.

Figure 1-10 displays a configuration using a single CP server that is connected to multiple SF Oracle RAC clusters with each SF Oracle RAC cluster also using two coordinator disks.

Figure 1-10 Single CP server connecting to multiple SF Oracle RAC clusters

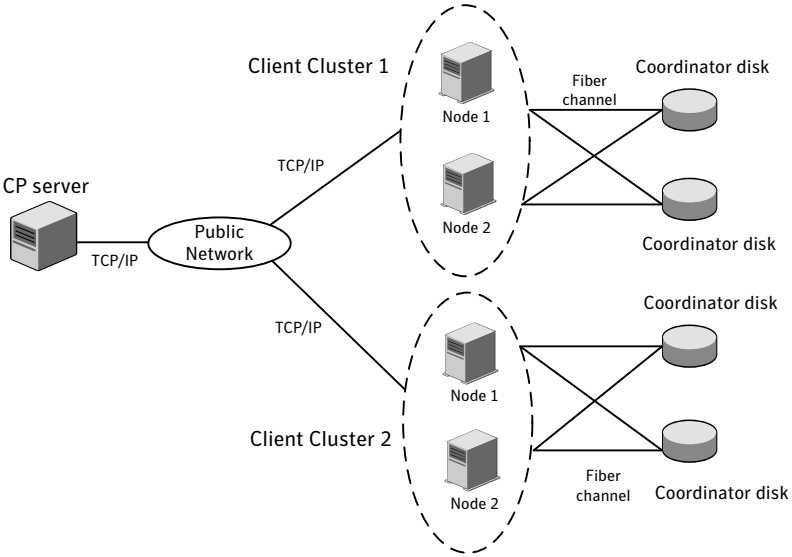
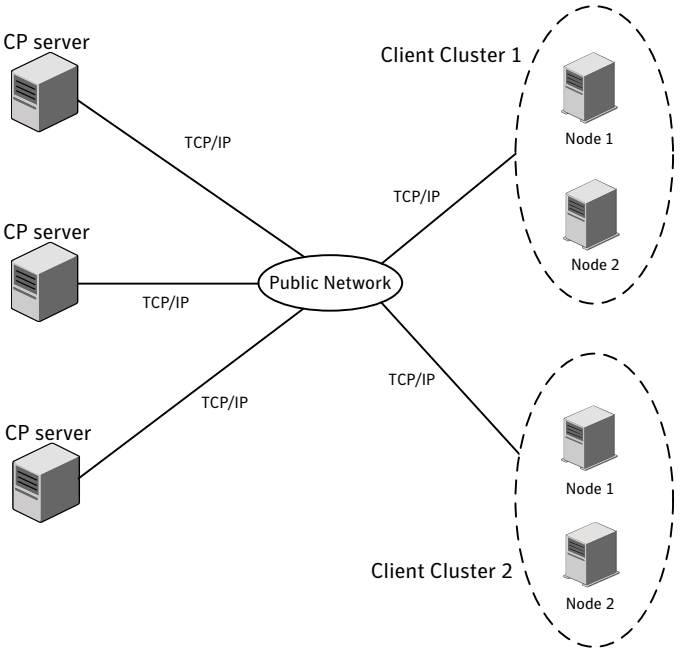


Figure 1-11 displays a configuration using 3 CP servers that are connected to multiple SF Oracle RAC clusters.

Figure 1-11 Three CP servers connecting to multiple 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.

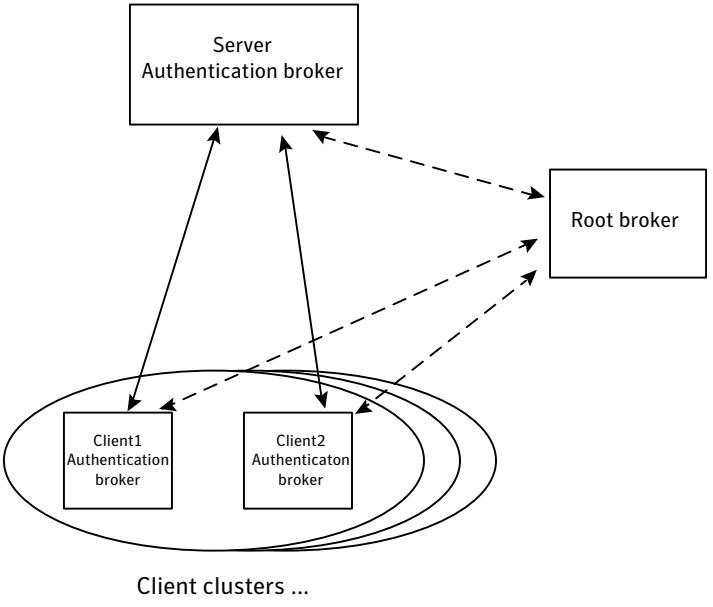
The CP server node or cluster needs to confirm the authenticity of the SF Oracle RAC cluster nodes that communicate with it as a coordination point and only accept requests from known SF Oracle RAC cluster nodes. Requests from unknown clients are rejected as non-authenticated. Similarly, the fencing 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 1-12 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 1-12 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.

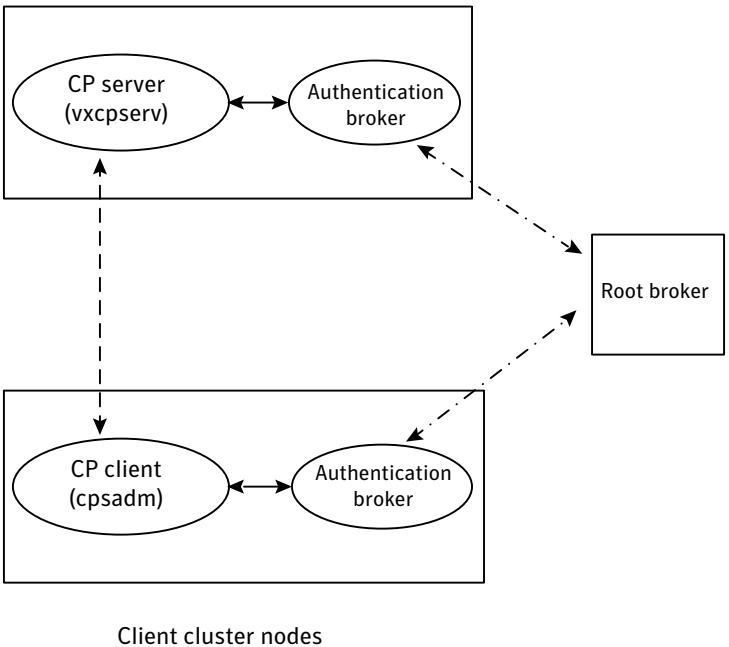
How secure communication between the CP servers and SF Oracle RAC clusters work

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 1-13 displays a schematic of the end-to-end communication flow with security enabled on CP server and SF Oracle RAC clusters.

Figure 1-13 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:

- CP server and authentication broker configured on it, and
- 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**.

Chapter 2

Administering SF Oracle RAC and its components

This chapter includes the following topics:

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 58.
- Starting or stopping SF Oracle RAC on each node
  See "Starting or stopping SF Oracle RAC on each node" on page 59.
- Applying Oracle patches
  See "Applying Oracle patches" on page 63.
- Adding LLT links to increase capacity
  See "Adding LLT links to increase capacity" on page 64.

- Removing LLT links
  See "Removing LLT links" on page 66.
- Adding aggregated links
  See "Adding aggregated links" on page 67.
- Adding storage to an SF Oracle RAC cluster
  See "Adding storage to an SF Oracle RAC cluster" on page 67.
- Recovering from storage failure
  See "Recovering from storage failure" on page 68.
- Enhancing the performance of SF Oracle RAC clusters
  See "Enhancing the performance of SF Oracle RAC clusters" on page 68.
- Creating snapshots for offhost processing
  See "Creating snapshots for offhost processing" on page 69.
- Verifying the ODM port
  See "Verifying the ODM port" on page 69.
- Verifying the nodes in a cluster
  See "Verifying the nodes in a cluster" on page 70.

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:\
/usr/lib/vxvm/bin:/opt/VRTSvcs/bin:\
/opt/VRTS/bin:/opt/VRTSvcs/rac/bin:\
/opt/VRTSob/bin:\
$PATH; export PATH
```

Starting or stopping SF Oracle RAC on each node

Use one of the following options to start or stop SF Oracle RAC on each node in the cluster.

To start SF Oracle RAC using the SF Oracle RAC installer	See "Starting SF Oracle RAC using the SF Oracle RAC installer" on page 59.
To stop SF Oracle RAC using the SF Oracle RAC installer	See "Stopping SF Oracle RAC using the SF Oracle RAC installer" on page 59.
To start SF Oracle RAC manually	See "Starting SF Oracle RAC manually on each node" on page 60.
To stop SF Oracle RAC manually	See "Stopping SF Oracle RAC manually on each node" on page 61.

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
```

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 -local
```

3. Stop SF Oracle RAC:

```
# /opt/VRTS/install/installsfrac -stop 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 VCSMM:

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

6. Start ODM:

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

7 Start VCS, CVM, and CFS:

```
# hastart
```

8 Verify that all GAB ports are up and running:

```
# gabconfig -a
GAB Port Memberships
===============================================================
Port a gen ada401 membership 01
Port b gen ada40d membership 01
Port d gen ada409 membership 01
Port f gen ada41c membership 01
Port h gen ada40f membership 01
Port o gen ada406 membership 01
Port v gen ada416 membership 01
Port w gen ada418 membership 01
```

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 | 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', '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 VCSMM:

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

8 Stop fencing:

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

9 Stop GAB:

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

10 Stop LLT:

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

Note: The command `svcadm disable` is persistent across reboots. See the `svcadm`(1M) manual page for more information.

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

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

2 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
```

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
```

Adding LLT links to increase capacity

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.

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

Refer to the `lltconfig` manual page for more information.

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 /etc/lltab.

To add a new LLT link

1 Log in to one of the nodes in the cluster as the root user and run the following command:

```
# lltconfig -d device -t device_tag
```

Where:

-t *devtag* is the device tag used to identify the link

-d *device* is the network device path

See the `lltconfig`(1M) for more options and detailed information.

For example:

```
# lltconfig -t eth2 -d eth2
```

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 or MultiPrivNIC.

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"
)
```

Removing LLT links

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

To remove LLT links

1. Log in to one of the nodes in the cluster as the root user and run the following command:

   ```
   # lltconfig -u device_tag
   ```

   Where:

   -u *devtag* is the device tag used to identify the link you want to remove

   See the `lltconfig`(1M) for more options and detailed information.

   For example:

   ```
   # lltconfig -u eth3
   ```

2. 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
   ```

Adding aggregated links

You can add aggregated interfaces to address bandwidth limitations or failover issues.

To add aggregated links

1. Log in as the root user on each node in the cluster.
2. Run the following command to add aggregated links to LLT:

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

3. Using vi or any text editor, open the file `/etc/lltab` and append the following entry in the file to make the change persistent across reboot operations:

```
link tag device_name systemid_range link_type sap mtu_size
```

Where:

tag is a link name to identify the link

device_name is the name of the bonded interface.

It is followed by a colon (:) and an integer which specifies the unit or physical point of attachment (PPA). The PPA defines which Ethernet adapter to use.

systemid_range represents the range of systems for which the command is valid. If the command is valid for all systems, specify a dash (-).

linktype indicates the type of link. Specify ether as the link type.

SAP specifies the service access point (SAP) to bind on the network links. SAP defines the point at which Ethernet services are accessible to LLT. The default value for SAP is 0xcafe.

MTU specifies the maximum transmission unit to use for sending packets on network links.

For example:

```
link link1 bond0- ether - -
```

Adding storage to an SF Oracle RAC cluster

Use the `vxassist` 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 `vxassist` (1M) manual page for information on various options.

To add storage to an SF Oracle RAC cluster

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

```
# vxresize [-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:

```
# vxresize -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`

For more information, see the `vxresize`(1M) manual page.

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

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

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.
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 70.

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
```

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 devices and drivers" on page 71.
- Configuring VCS to start Oracle with a specified Pfile
  See "Configuring VCS to start Oracle with a specified Pfile" on page 72.
- Verifying VCS configuration
  See "Verifying VCS configuration" on page 72.
- Starting and stopping VCS
  See "Starting and stopping VCS" on page 72.

If you encounter issues while administering VCS, refer to the troubleshooting section for assistance.

Viewing available Veritas devices and drivers

To view the available Veritas devices:

```
# lsdev -C -c vxdrv

vxdmp     Available  Veritas VxDMP Device Driver
vxglm0    Available  N/A
vxgms0    Available  N/A
vxio      Available  Veritas VxIO Device Driver
vxportal0 Available  VERITAS VxPORTAL Device Driver
vxqio0    Defined    VERITAS VxQIO Device Driver
vxspec    Available  Veritas VxSPEC Device Driver
```

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
```

```
457f000    4ed20 /usr/lib/drivers/gab
```

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

```
# lsmod |grep vx

55de000     2000 /etc/vx/kernel/dmpjbod
55dc000     2000 /etc/vx/kernel/dmpap
55da000     2000 /etc/vx/kernel/dmpaa
55b5000    21000 /usr/lib/drivers/vxodm.ext_61
55b2000     3000 /usr/lib/drivers/vxspec
4de0000   7ce000 /usr/lib/drivers/vxio
4d7a000    62000 /usr/lib/drivers/vxdmp
4cee000    3d000 /usr/lib/drivers/vxgms.ext_61
4c7e000    4d000 /usr/lib/drivers/vxfen
4bf3000     e000 /usr/lib/drivers/vxqio.ext_61
4bef000     3000 /usr/lib/drivers/vxportal.ext_61
4969000   232000 /usr/lib/drivers/vxfs.ext_61
48fa000    3e000 /usr/lib/drivers/vxglm.ext
```

Configuring VCS to start Oracle with a specified Pfile

If you want to configure VCS such that Oracle starts with a specified Pfile, modify the main.cf file for the Oracle group as follows:

```
Oracle ora1 (
                Sid @galaxy = vrts1
                Sid @nebula = vrts2
                Owner = oracle
                Home = "/app/oracle/orahome"
                StartUpOpt = SRVCTLSTART
                ShutDownOpt = SRVCTLSTOP
                pfile="/app/oracle/orahome/dbs/initprod1.ora"
                )
```

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.

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 73.
- About vxfentsthdw utility
  See "About the vxfentsthdw utility" on page 74.
- About vxfenadm utility
  See "About the vxfenadm utility" on page 82.
- About vxfenclearpre utility
  See "About the vxfenclearpre utility" on page 87.
- About vxfenswap utility
  See "About the vxfenswap utility" on page 88.

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

See "Troubleshooting I/O fencing" on page 131.

About administering I/O fencing

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

vxfentsthdw	Tests hardware for I/O fencing See "About the vxfentsthdw utility" on page 74.
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 82.
vxfenclearpre	Removes SCSI-3 registrations and reservations from disks See "About the vxfenclearpre utility" on page 87.
vxfenswap	Replaces coordinator disks without stopping I/O fencing See "About the vxfenswap utility" on page 88.
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 86.
- 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 2-2 describes the methods that the utility provides to test storage devices.

Table 2-2 vxfentsthdw options

vxfentsthdw option	Description	When to use
-n	Utility uses rsh for communication.	Use when rsh is used for communication.
-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 78.

Table 2-2 vxfentsthdw options *(continued)*

vxfentsthdw option	Description	When to use
-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.	
-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 77.
-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 79.
-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 81.

Table 2-2 vxfentsthdw options *(continued)*

vxfentsthdw option	Description	When to use
-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 81.

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 77.

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 secondary 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.

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/sdy

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 87.

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:

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.

Reboot the server to proceed with normal cluster startup...
#
```

7 Restart all nodes in the cluster.

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.

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

See the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide* for details on the coordinator disk 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 89.
  See "Replacing the coordinator diskgroup in a cluster that is online" on page 91.
  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 135.
- 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 94.
- 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 95.

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. 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 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 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.

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

10 Review the message that the utility displays and confirm that you want to commit the new set of coordinator disks. Else skip to step 11.

```
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.

11 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 77.

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
```

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
...
```

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 2-3.

For descriptions of specific types of `cpsadm` command actions, see Table 2-4.

Table 2-3 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 2-4).
-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 2-3 cpsadm command parameters *(continued)*

Parameter	Name	Description
-o	victim node host name	This parameter specifies the victim node's host name.

Table 2-4 describes the specific `cpsadm` command action types. This command's action data (-a) may contain some of the above `cpsadm` command parameters.

Table 2-4 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 2-4 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 2-5 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 2-5 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_user
```

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
```

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 does not support migration from server-based fencing (vxfen_mode=customized) to disk-based fencing (vxfen_mode=scsi3) and vice-versa in an online cluster.

However, in a cluster that is offline you can migrate from disk-based fencing to server-based fencing and vice-versa:

- Disk-based to server-based fencing:
  Perform the tasks on the CP server and on the SF Oracle RAC cluster nodes as described in the “Enable fencing in a client cluster with a new CP server” scenario.
- Server-based to disk-based fencing:
  Perform the tasks on the SF Oracle RAC cluster nodes as described in the “Enable fencing in a client cluster with a new CP server” scenario.

See “Deployment and migration scenarios for CP server” on page 44.

You can cancel the coordination point replacement operation at any time using the `vxfenswap -a cancel` command.

See “About the vxfenswap utility” on page 88.

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.

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.

If security is not configured, please refer to:

See "About secure communication between the SF Oracle RAC cluster and CP server" on page 50.

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.

7 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\
-f cps_operator -g vx

User _HA_VCS_galaxy@HA_SERVICESgalaxy.symantec.com successfully added

# cpsadm -s mycps.symantecexample.com -a add_user -e\
_HA_VCS_nebula6@HA_SERVICES@nebula.symantec.com\
-f cps_operator -g vx

User _HA_VCS_nebula@HA_SERVICES@nebula.symantec.com successfully added
```

8 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.
```

9 Modify /etc/vxfenmode file on each SF Oracle RAC cluster node to set security=1.

10 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 113.
- Using cfsmount to mount CFS file systems
  See "Using cfsmount to mount CFS file systems" on page 113.
- Resizing CFS file systems
  See "Resizing CFS file systems" on page 113.
- Verifying the status of CFS file systems
  See "Verifying the status of CFS file systems" on page 114.
- Verifying CFS port
  See "Verifying CFS port" on page 114.

If you encounter issues while administering CFS, refer to the troubleshooting section for assistance.

See "Troubleshooting CFS" on page 153.

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 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 171.
- Establishing CVM cluster membership manually
  See “Establishing CVM cluster membership manually” on page 115.

- Importing a shared disk group manually
  See "Importing a shared disk group manually" on page 116.
- Deporting a shared disk group manually
  See "Deporting a shared disk group manually " on page 116.
- Starting shared volumes manually
  See "Starting shared volumes manually" on page 171.
- Evaluating the state of CVM ports
  See "Evaluating the state of CVM ports" on page 116.
- 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 116.
- Verifying CVM membership state
  See "Verifying CVM membership state" on page 117.
- Verifying the state of CVM shared disk groups
  See "Verifying the state of CVM shared disk groups" on page 117.
- Verifying the activation mode
  See "Verifying the activation mode" on page 118.

If you encounter issues while administering CVM, refer to the troubleshooting section for assistance.

See "Troubleshooting CVM" on page 149.

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
```

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 and port 'w' for vxconfigd communication between the cluster nodes. The following command displays the state of CVM ports:

```
# gabconfig -a | egrep "Port [vw]"
```

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 diskgroupname |grep activation

local-activation: shared-write
```

If "shared-write" activation is not set, run the following command:

```
# vxdg -g diskgroupname set activation=sw
```

Administering Oracle

This section provides instructions for the following Oracle administration tasks:

- Creating a database
  See “Creating a database” on page 119.
- Increasing swap space for Oracle
  See “Increasing swap space for Oracle” on page 119.
- Stopping Oracle Clusterware
  See “Stopping Oracle Clusterware” on page 119.
- Determining Oracle Clusterware object status
  See “Determining Oracle Clusterware object status” on page 120.
- Configuring virtual IP addresses for Oracle Clusterware
  See “Configuring virtual IP addresses for Oracle Clusterware” on page 121.
- Configuring Oracle group to start and stop Oracle database instances
  See “Configuring Oracle group to start and stop Oracle database instances” on page 121.
- Configuring listeners
  See “Configuring listeners” on page 121.
- Starting or stopping Oracle listener
  See “Starting or stopping Oracle listener” on page 121.
- Starting and stopping Oracle service groups
  See “Starting and stopping Oracle service groups” on page 122.

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

See "Troubleshooting Oracle" on page 155.

Creating a database

To create a database, run the dbca command as follows:

```
$ export DISPLAY=display.ip

$ dbca
```

For more information, consult the Oracle documentation.

Increasing swap space for Oracle

The minimum swap space requirement for Oracle RAC 11g is 8 GB. The operating system requirement for minimum swap space is two times the size of RAM.

Between the minimum requirements of Oracle RAC and the operating system, make sure that you meet the minimum requirement that is higher. For example, if the operating system requirement for minimum swap space computes to 5 GB on your Oracle RAC 11g systems, make sure that you meet the minimum swap space requirement of Oracle RAC, that is 8 GB.

To increase swap space for Oracle

1 Check the amount of free swap space available.

```
# swap -l

device                maj,min        total       free
/dev/hd6               10,  2       512MB       505MB
```

2 Increase the swap space as required.

```
# chps -s '40' hd6
```

Stopping Oracle Clusterware

If you need to manually stop Oracle Clusterware outside of VCS control, run the following command:

```
# $GRID_HOME/bin/crsctl stop crs
Stopping resources.
Successfully stopped CRS resources
Stopping CSSD.
```

```
Shutting down CSS daemon.
Shutdown request successfully issued.
```

Determining Oracle Clusterware object status

To determine the status of Oracle Clusterware objects:

```
$GRID_HOME/bin/crsctl stat res -t
--------------------------------------------------------------------
NAME           TARGET  STATE     SERVER    STATE_DETAILS
--------------------------------------------------------------------
Local Resources
--------------------------------------------------------------------
ora.LISTENER.lsnr
               ONLINE  ONLINE    galaxy
               ONLINE  ONLINE    nebula
ora.asm
               OFFLINE OFFLINE   galaxy
               OFFLINE OFFLINE   nebula

.
.
.
--------------------------------------------------------------------
Cluster Resources
--------------------------------------------------------------------
ora.LISTENER_SCAN1.lsnr
      1        ONLINE  ONLINE       nebula
ora.LISTENER_SCAN2.lsnr
      1        ONLINE  ONLINE       galaxy
ora.LISTENER_SCAN3.lsnr
      1        ONLINE  ONLINE       galaxy
ora.dtrac.db
      1        ONLINE  ONLINE       galaxy                 Open
      2        ONLINE  ONLINE       nebula                 Open
.
.
.
```

The Oracle Clusterware objects 'gsd', 'ons', 'vip', and 'lsnr' are the nodes applications for the nodes galaxy and nebula.

Configuring virtual IP addresses for Oracle Clusterware

To configure virtual IP addresses for Oracle Clusterware:

```
# export DISPLAY=display.ip
```

```
# $GRID_HOME/bin/vipca
```

where display.ip is the display IP address.

For more information, consult the Oracle documentation.

Configuring Oracle group to start and stop Oracle database instances

Symantec recommends that your Oracle group within VCS be configured to start and stop Oracle database instances.

The following sample main.cf extract shows how to configure VCS to start your Oracle database instances:

```
Oracle Ora_1 (
                Critical = 0
                Sid @galaxy = PROD11
                Sid @nebula = PROD12
                Owner = oracle
                Home = "/oracle/orahome"
                StartUpOpt = SRVCTLSTART
                ShutDownOpt = SRVCTLSTOP
                OnlineTimeout = 900
                )
```

Configuring listeners

To configure listeners:

```
$ export DISPLAY=display.ip
```

```
$ netca
```

For more information, consult Oracle documentation.

Starting or stopping Oracle listener

If you have issues with Oracle listener, you can stop and restart the listener as oracle user.

To start Oracle listener:

```
$ lsnrctl start
```

To stop Oracle listener:

```
$ lsnrctl stop
```

To check the status of Oracle listener:

```
$ lsnrctl status
```

For more information, consult the Oracle documentation.

Starting and stopping Oracle service groups

To start the Oracle service group "oracle_grp" on nodes "galaxy" and "nebula":

```
# hagrp -online oracle_grp -sys galaxy
```

```
# hagrp -online oracle_grp -sys nebula
```

To stop the Oracle service group "oracle_grp" on nodes "galaxy" and "nebula":

```
# hagrp -offline oracle_grp -sys galaxy
```

```
# hagrp -offline oracle_grp -sys nebula
```

Section 2

Performance and troubleshooting

Chapter 3

Troubleshooting SF Oracle RAC

This chapter includes the following topics:

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:

```
# 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:

```
# 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	*$ORACLE_BASE*\ /oraInventory/logs/\ installActions*date_time*.log	Contains errors that occurred during Oracle RAC installation. It clarifies the nature of the error and when it occurred during the installation. **Note:** Verify if there are any installation errors logged in this file, since they may prove to be critical errors. If there are any installation problems, send this file to Tech Support for debugging the issue.
Oracle alert log	For Oracle RAC 11g: *$ORACLE_BASE*/diag/rdbms/*db_name*/\ *instance_name*/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 128.
Agent log files 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*.

Table 3-1 List of log files *(continued)*

Log file	Location	Description

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/adm/streames`	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:

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' and validate using the command
  'vxkeyless set NONE'
```

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 set NONE
  ```

- 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 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 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 132.

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 -g 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 135. 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]:
 [Character Format]: VFBEAD00
 [Node Format]: Cluster ID: 48813 Node ID: 0 Node Name: unknown
```

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 87.

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 87.

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 95.

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 88.

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
  ```

- If the coordinator attribute value is set to on, you must turn off this attribute for the coordinator disk group.

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

- Destroy the disk group.

  ```
  # vxdg destroy vxfencoorddg
  ```

7 Add the new disk to the node, initialize it as a VxVM disk, and add it to the vxfencoorddg disk group.

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

8 Test the recreated disk group for SCSI-3 persistent reservations compliance.

  See "Testing the coordinator disk group using vxfentsthdw -c option" on page 77.

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
```

The vxfenswap utility faults when echo or cat is used in .bashrc file

The vxfenswap utility faults when you use echo or cat to print messages in the .bashrc file for the nodes.

To recover the vxfenswap utility fault

- Verify whether the rcp or scp functions properly.

  If the vxfenswap operation is unsuccessful, use the `vxfenswap -cancel` command if required to roll back any changes that the utility made.

  See "About the vxfenswap utility" on page 88.

Troubleshooting on the CP server

All the 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 single node VCS or SFHA cluster hosting 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.

CP server service group issues

If you cannot bring up the CPSSG service group after the CP server configuration, verify that the CPSSG service group and its resources are valid and properly configured in the VCS configuration.

Check the VCS engine log to see if any of the CPSSG service group resources are FAULTED. The engine log is located in the following directory:

/var/VRTSvcs/log/engine_[ABC].log

The resources that are configured under the CPSSG service groups are displayed in the following figures:

- CPSSG group and dependency figure for CP server hosted on a single node VCS cluster:
- CPSSG group and dependency figure for CP server hosted on an SFHA cluster:

Note: For information about general VCS troubleshooting procedures, refer to the Veritas™ Cluster Server User's Guide, Version 5.1.

Testing the connectivity of the CP server

The connectivity of the CP server can be tested using the `cpsadm` command. The following `cpsadm` command tests 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.

Issuing the command on the SF Oracle RAC cluster nodes requires the environment variables CPS_USERNAME and CPS_DOMAINTYPE to be set.

Troubleshooting server-based I/O fencing on the SF Oracle RAC cluster

The file /var/VRTSvcs/log/vxfen/vxfend_[ABC].log contains logs and text files that may be useful in understanding and/or troubleshooting fencing-related issues on a SF Oracle RAC cluster node.

Issues during server-based fencing start up on SF Oracle RAC cluster node

The following issues may occur during fencing start up on the SF Oracle RAC cluster node:

- `cpsadm` command on the SF Oracle RAC cluster gives connection error
- Authentication failure
- Authorization failure
- Preexisting split-brain

cpsadm command on the SF Oracle RAC cluster node 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 correct CP server virtual IP/virtual hostname and port number are being used by the SF Oracle RAC cluster nodes.
  Check the /etc/vxfenmode file.
- Ensure that the running CP server is using the same virtual IP/virtual hostname and port number.

Authentication failure

If secure communication has been configured between the CP server and the SF Oracle RAC cluster nodes, authentication failure can occur due to the following causes:

- Symantec Product Authentication Services 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 50.

- The hostname of the SF Oracle RAC cluster nodes is not the same hostname used when configuring the Symantec Product Authentication Service.
  The hostname of the SF Oracle RAC cluster nodes must be set to the hostname used when configuring the Symantec Product Authentication Service. The fully qualified hostname registered with the Symantec Product Authentication Service can be viewed 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 nodes.

Authorization failure

Authorization failure occurs when the CP server's SF Oracle RAC cluster nodes or users are not added in the CP server configuration. Therefore, fencing on the SF Oracle RAC 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 SF Oracle RAC cluster node and user in the CP server configuration and restart fencing. Refer to the following section:

Preexisting split-brain

To illustrate preexisting split-brain, assume 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. The situation is 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 using the `cpsadm` command.

The following `cpsadm` command can be used 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.

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.
- The locales of the SF Oracle RAC cluster nodes are different.
  The locales of the SF Oracle RAC cluster nodes must be the same on all SF Oracle RAC cluster nodes where I/O fencing is configured, and the `vxfen_mechanism` parameter in the /etc/vxfenmode file is set to "`cps`".
  Under these conditions, ensure that each SF Oracle RAC cluster node has the same locale installed.
- There is no network connectivity from one or more SF Oracle RAC cluster nodes to the CP server(s).
- The cluster or 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

The following procedure can be use to troubleshoot a mixed I/O fencing configuration (configuration using both coordinator disks and CP server for I/O fencing). This procedure involves using 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.

Troubleshooting mixed I/O fencing configuration (coordinator disks and CP server)

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.

Enter the `vxfenadm -s` command on each of the SF Oracle RAC cluster nodes.

```
# vxfenadm -s /dev/vx/rdmp/3pardata0_190
```

```
Device Name: /dev/vx/rdmp/3pardata0_190
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
```

```
# vxfenadm -s /dev/vx/rdmp/3pardata0_191
```

```
Device Name: /dev/vx/rdmp/3pardata0_191
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 CPS server, run the `cpsadm list nodes` command to review a list of nodes in the cluster.

The command syntax is as follows:

```
# 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.

For example:

```
# /opt/VRTS/bin/cpsadm -s 10.140.94.101 -a list_nodes

ClusName        UUID                                    Hostname(Node ID) Registered
gl-rh2  {25aeb8c6-1dd2-11b2-95b5-a82227078d73}  node_101(0)         0
gl-rh2  {25aeb8c6-1dd2-11b2-95b5-a82227078d73}  node_102(1)         0
cpstest {a0cf10e8-1dd1-11b2-87dc-080020c8fa36}  node_220(0)         0
cpstest {a0cf10e8-1dd1-11b2-87dc-080020c8fa36}  node_240(1)         0
ictwo   {f766448a-1dd1-11b2-be46-5d1da09d0bb6}  node_330(0)         0
ictwo   {f766448a-1dd1-11b2-be46-5d1da09d0bb6}  sassette(1)         0
fencing {e5288862-1dd1-11b2-bc59-0021281194de}  CDC-SFLAB-CD-01(0)  0
fencing {e5288862-1dd1-11b2-bc59-0021281194de}  CDC-SFLAB-CD-02(1)  0
gl-su2  {8f0a63f4-1dd2-11b2-8258-d1bcc1356043}  gl-win03(0)         0
gl-su2  {8f0a63f4-1dd2-11b2-8258-d1bcc1356043}  gl-win04(1)         0
gl-su1  {2d2d172e-1dd2-11b2-bc31-045b4f6a9562}  gl-win01(0)         0
gl-su1  {2d2d172e-1dd2-11b2-bc31-045b4f6a9562}  gl-win02(1)         0
gl-ax4  {c17cf9fa-1dd1-11b2-a6f5-6dbd1c4b5676}  gl-ax06(0)          0
gl-ax4  {c17cf9fa-1dd1-11b2-a6f5-6dbd1c4b5676}  gl-ax07(1)          0
gl-ss2  {da2be862-1dd1-11b2-9fb9-0003bac43ced}  galaxy(0)         1
gl-ss2  {da2be862-1dd1-11b2-9fb9-0003bac43ced}  nebula(1)         1
```

5 Review the CP server list membership.

On the CP server, run the following command to review the list membership. The command syntax is as follows:

```
# 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.

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 87.

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 87.

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.

For additional examples:

- RHEL 4.0 Example:
  `/etc/modprobe.conf` includes:

```
options scsi_mod dev_flags="HITACHI:OPEN-3:0x240" options
scsi_mod max_luns=512
```

- SLES9 Example:
  `/boot/efi/efi/SuSE/elilo.conf` includes:

```
append = "selinux=0 splash=silent elevator=cfq
scsi_mod.default_dev_flags=0x240"
```

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.

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.

Network interfaces change their names after reboot

On SUSE systems, network interfaces change their names after reboot even with `HOTPLUG_PCI_QUEUE_NIC_EVENTS=yes` and `MANDATORY_DEVICES="..."` set.

Workaround: Use `PERSISTENT_NAME= ethX` where X is the interface number for all interfaces.

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.

Oracle log files

The following Oracle log files are helpful for resolving issues with Oracle components:

- Oracle log file
- CRS core dump file
- Oracle css log file

- OCSSD core dump file

Oracle log file

The Oracle log file contains the logs pertaining to the CRS resources such as the virtual IP, Listener, and database instances. It indicates some configuration errors or Oracle problems, since CRS does not directly interact with any of the Symantec components.

To check the Oracle log file, access the following locations.

- For Oracle 10g Release 1, access:

```
$CRS_HOME/crs/log
```

- For Oracle 10g Release 2, access:

```
$CRS_HOME/log/hostname/crsd
```

  where *hostname* is the string returned by the `hostname` command.

CRS core dump file

The CRS core dumps for the `crsd.bin` daemon are written here. Use this file for further debugging.

To check for crs core dumps access:

```
$CRS_HOME/crs/init
```

Oracle css log file

The css logs indicate actions such as reconfigurations, missed checkins, connects, and disconnects from the client CSS listener. If there are any membership issues, they will show up here. If there are any communication issues over the private networks, they are logged here. The `ocssd` process interacts with `vcsmm` for cluster membership.

To check the Oracle css log file, access the following locations.

- For Oracle 10g Release 1, access:

```
$CRS_HOME/css/log
```

- For Oracle 10g Release 2, access:

```
$CRS_HOME/log/hostname/cssd
```

  where *hostname* is the string returned by the `hostname` command.

OCSSD core dump file

Core dumps from the `ocssd` and the `pid` for the `css` daemon whose death is treated as fatal are located here. If there are any abnormal restarts for `css` the core files, they are found here.

To check for ocssd core dumps, access the following locations:

```
$CRS_HOME/css/init
```

Oracle Notes

Review the following Oracle notes, when dealing with the following specific Oracle issues:

259301.1	CRS and 10g Real Application Clusters
280589.1	CRS Installation Does Not Succeed if One or More Cluster Nodes Present are Not to be Configured for CRS.
265769.1	10g RAC: Troubleshooting CRS Reboots
279793.1	How to Restore a Lost Vote Disk in 10g
239998.1	10g RAC: How to Clean Up After a Failed CRS Install Two items missing in this Oracle note are: ■ Remove the /etc/oracle/ocr.loc file. This file contains the location for the Cluster registry. If this file is not removed then during the next installation the installer will not query for the OCR location and will pick it from this file. ■ If there was a previous 9i Oracle installation, then remove the following file: /var/opt/oracle/srvConfig.loc. If this file is present the installer will pick up the Vote disk location from this file and may create the error "the Vote disk should be placed on a shared file system" even before specifying the Vote disk location.
272332.1	CRS 10g Diagnostic Collection Guide

Oracle user must be able to read /etc/llttab File

Check the permissions of the file `/etc/llttab`. Oracle must be allowed to read it.

Relinking of VCSMM library fails after upgrading from version 4.1 MP2

After you upgrade from SF Oracle RAC 4.1 MP2, the relinking process may fail with the following message:

```
/app/crshome/lib/libskgxn2.so is not a VCSMM library
on one or more node(s) of your cluster. It should be a symbolic link to
/opt/ORCLcluster/lib/libskgxn2.so file, which must be a VCSMM library file.
```

The process fails because the Veritas `skgxn` library is copied directly to the Oracle Clusterware home directory (`/app/crshome/lib`) instead of linking the library in the Oracle Clusterware home directory to the library `/opt/ORCLcluster/lib/libskgxn2.so`.

To resolve the issue, create a symbolic link for the library from the Oracle Clusterware home directory to the library `/opt/ORCLcluster/lib/libskgxn2.so` as follows:

```
# ln -s /opt/ORCLcluster/lib/libskgxn2.so \
/app/crshome/lib/libskgxn2.so
```

After relinking the VCSMM library, relink the ODM library as described in the *Veritas Storage Foundation for Oracle RAC Installation and Configuration Guide*.

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
```

root.sh hangs after Oracle binaries installation

This may occur when using OCR on a raw volume if Oracle patch Number 4045013 is not applied.

To prevent root.sh from hanging

1. Install Oracle Database Binaries.
2. Apply Oracle patch Number 4045013, available on metalink.oracle.com.

   In patch search criteria, specify 4045013 as patch number and Linux Opteron in Platform/Architecture.
3. Execute `root.sh`.

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 $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
  ```

OCR and Vote disk related issues

Verify that the permissions are set appropriately as given in the Oracle installation guide.

See "Oracle Clusterware fails after restart" on page 160.

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 3-2 lists the warning messages displayed during the health check and corresponding recommendations for resolving the issues.

Table 3-2 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 3-2 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 9i 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 9i 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) package 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

The following 8 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)

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 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 35.

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 82.

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 -g 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 74.

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 105. **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

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:

- 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.

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

Error messages

This appendix includes the following topics:

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 A-1 contains VxVM error messages that are related to I/O fencing.

Table A-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 A-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 178.

See "Node ejection informational messages" on page 179.

Table A-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 Multipathing)	A feature of Veritas Volume Manager 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 SF Manager 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

K

L

M

O

R

S

T

V