WHITE PAPER: MIGRATION GUIDE

Symantec.
Confidence in a connected world.

MIGRATING RED HAT CLUSTER TO VERITAS CLUSTER SERVER

Anthony Herr, Regional Product Manager
Storage and Availability Management Group
January 2011

Content

MIGRATING RED HAT CLUSTER TO VERITAS CLUSTER SERVER 1

Executive Summary 4

Third-party legal notices 4

Licensing and registration 5

Technical support 5

Introduction 5

Audience 5

Technology Overview 6

Veritas Cluster Server (VCS) 6

Veritas Cluster Server Architecture Introduction 6

Veritas Cluster Server clustering concepts 6

Cluster Communication 9

Term Comparison - RHC and VCS 10

Command Comparison - RHC and VCS 12

Veritas Cluster Server cluster heartbeats 12

Migration Procedure 12

Planning Phase 12

Application downtime required 12

Identify applications and resources under Red Hat Cluster control 13

Veritas Cluster Server Hardware Prerequisites 13

Preparing for Veritas Cluster Server Installation 14

Implementation Phase 15

Shutdown the RHC cluster 15

Uninstall RHC software 15

Veritas Cluster Server Cluster Configuration 15

Veritas Cluster Server installation 15

Veritas Cluster Server Heartbeats 15

Veritas Cluster Server Cluster Creation 16

Verify the Veritas Cluster Server Cluster 16

Veritas Cluster Server validation and testing 16

Red Hat Cluster for Linux to VCS Flowchart 17

Methods of Controlling the Veritas Cluster Server Cluster 17

Summary of VCS management capabilities 18

VCS Command Line Interface (CLI): 18
VCS Java Console: 18
VCS Cluster Simulator: 18
Veritas Operations Manager (VOM): 18
Appendix reference information 19
Migration Planning – VCS Cluster Information 19
Step-by-step migration with sample applications – RHC -> VCS 24
Migration Steps: 24
RHC Configuration Files Examples 25
Veritas Cluster Server Configuration Files Examples 26
Red Hat Cluster and Veritas Cluster Server Configuration Files Migration Example 27
Reference Documentation 27
VCS Command Line quick reference 28
Start VCS 28
Stop VCS 28
Change VCS Configuration Online 28
Get Current Cluster Status 28
Agent Operations 28
Add and Delete Users 28
System Operations 28
Resource Types 28
Resource Operations 29
Service Group Operations 29
VCS Procedures 29
VCS Directory Structure 29
Determine the Status of the Cluster 30
To Failover the ServiceGroup from One system to another 30
To Freeze/Unfreeze the ServiceGroup 30
The scripts that start VCS on boot 30
To clear a faulted resource 30
Hastart/Hastop options 30
Modifying the Cluster Config 31
Adding a new filesystem to the cluster 31
reboot/init 6/shutdown commands DO failover applications 31

Add a user to the GUI 31

Agent Scripts 32

Executive Summary

This white paper, illustrates a process to migrate a Red Hat Cluster (RHC) to Veritas Cluster Server (VCS). An introduction to the architecture of VCS is described including sections comparing RHC and VCS which contrast cluster terminology and describe architecture differences. A step-by-step process describes how to use configuration information from an existing RHC cluster to quickly migrate to a VCS cluster with similar functionality.

Licensing and registration
Veritas Cluster Server is a licensed product. See the **Veritas Cluster Server Installation Guide** for license installation instructions.

Technical support
For technical assistance, visit:
http://www.symantec.com/enterprise/support/assistance_care.jsp.
Select phone or email support. Use the Knowledge Base search feature to access resources such as TechNotes, product alerts, software downloads, hardware compatibility lists, and our customer email notification service.

Introduction

This document is intended to provide information to assist with the migration of a cluster from Red Hat Cluster (RHC) to Veritas Cluster Server (VCS) for Linux. Many customers are migrating from RHC to VCS. With this document it is our intention to illustrate the migration path to Veritas Cluster Server as an alternative to Red Hat Cluster. Please review product documentation before installing VCS.

Audience
This document is targeted for technical users of Red Hat Cluster who wish to migrate to Veritas Cluster Server on Linux. It is assumed that the reader has a general understanding of Red Hat Cluster, the Red Hat Linux Operating System and Veritas Cluster Server. For more information on Red Hat Cluster, see Red Hat's website and for Veritas Cluster Server see **http://www.symantec.com/business/cluster-server**.

Technology Overview

Veritas Cluster Server (VCS)

Veritas Cluster Server from Symantec connects multiple, independent systems into a management framework for increased availability. Each system, or node, runs its own operating system and cooperates at the software level to form a cluster. These systems can be either a physical or virtual server. VCS links commodity hardware with intelligent software to provide application failover and control. When a node or a monitored application fails, other nodes can take predefined actions to take over and bring up services elsewhere in the cluster.

Veritas Cluster Server is the industry's leading clustering solution for reducing business critical applications' planned and unplanned downtime. VCS can detect faults in an application and all its dependent components, including the associated database, operating system, network, and storage resources. When a failure is detected, Cluster Server gracefully shuts down the application, restarts it on an available server, connects it to the appropriate storage device, and resumes normal operations.

Veritas Cluster Server is supported on Red Hat Enterprise Linux, SUSE, AIX, HP-UX, Solaris and Windows. For supported storage, OS versions, and recommended patch levels please see the **Hardware Compatibility List** or the Symantec Operational Readiness Toolkit at **http://sort.symantec.com** which can assist with installation/upgrade checking utilities.

Veritas Cluster Server Architecture Introduction

This introduction is an overview of the basic concepts within Veritas Cluster Server. It is intended to provide enough information that would allow users to determine the requirements to migrate a Red Hat Cluster for Linux cluster to Veritas Cluster Server.

Veritas Cluster Server clustering concepts

Cluster

A single VCS cluster consists of multiple servers or systems, either physical or virtual, connected in various combinations to shared storage devices and network connections. VCS monitors and controls applications running in the cluster, and can restart applications in response to a variety of hardware or software faults.

A cluster is defined as all systems that share a common cluster configuration and utilize a common interconnecting network. The VCS cluster interconnect consists of redundant physical Ethernet connections, generally over two or more dedicated private networks. The communications layer carries heartbeats between systems within the cluster, as well as membership and state change information. This will be described in the cluster communications section below.

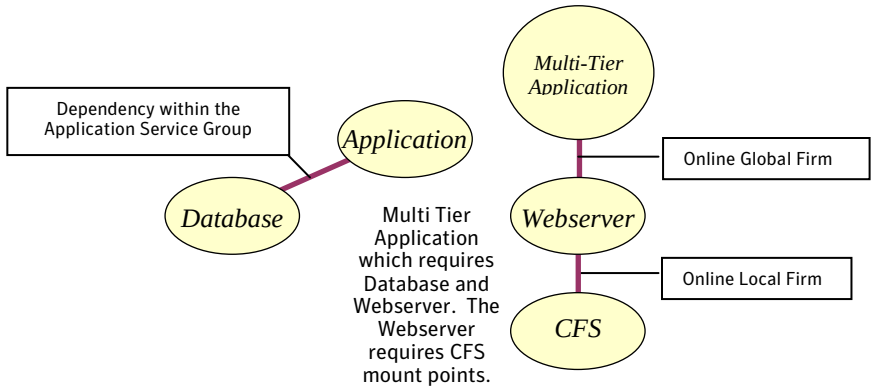
Applications can be configured to run on specific nodes in the cluster based on priority, application dependencies, or workload policies. Storage is configured to provide access to shared application data for the systems that are hosting the application. In that respect, the actual storage connectivity will determine where applications can be run: Nodes sharing access to storage are “eligible” to run an application. Shared storage is not a requirement for Veritas Cluster Server.

Service Group

A service group is a virtual container that contains all the hardware and software resources that are required to run the managed application. Service groups allow VCS to control all the hardware and software resources of the managed application as a single unit. When a failover occurs, resources do not fail over individually— the entire service group fails over. If there is more than one service group on a system, a group may fail over without affecting the others.

Service groups can be dependent on each other. For example a finance application may be dependent on a database application. Because the managed application consists of all components that are required to provide the service, service group dependencies create more complex managed applications. When using service group dependencies, the managed

application is the entire dependency tree. The following is a graphical representation of the Service Group dependencies in a VCS cluster that controls an Application, a Database and a Webserver. The Webserver requires that CFS Mount points are online on the local VCS node before it will come online. The Application Service Group requires that the Webserver is running somewhere in the cluster before it will come online. When the Application Service Group comes online, it brings up the Database and then the Application. These are local dependencies within the Service Group.



Agents

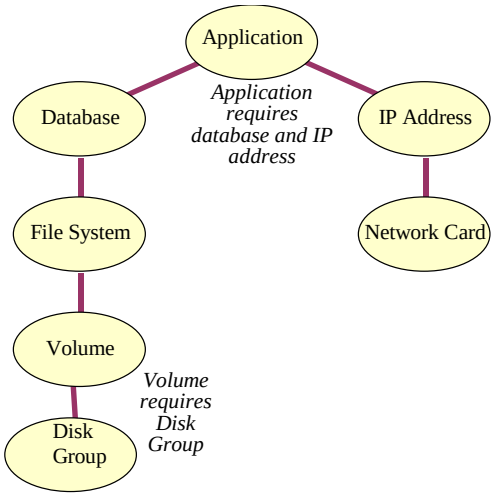
Veritas Cluster Server agents handle the start, stop, and monitoring of all resources contained within a service group. Agents receive instructions on when to start, stop, or monitor a resource from the VCS engine, and the agents then return the results of those actions to the engine. Bundled with the VCS product are a collection of agents to manage the storage and network resources required by an application.

Veritas Cluster Server also ships with agents to control all common system functions, such as file systems and network addresses. Additional agents are provided for out-of-the-box support for most enterprise applications, such as databases, application servers, and Web servers. This includes complete out-of-the-box (no customization required) support for Oracle®, DB2®, Sybase, SAP®, WebSphere, WebLogic, and many other enterprise applications. Please see **http://www.symantec.com/business/products/agents_options.jsp?pcid=pcat_business_cont&pvid=20_1**
for a complete list of applications with existing VCS Agents. All applications that can run in a High Availability environment can utilize the bundled agent that controls applications with their own start and stop scripts. Custom agents can be developed for managing applications with unique and advanced startup, shutdown and monitoring requirements. For more information on agents please see the **Veritas Cluster Server Installation Guide**.

Resources

Resources are hardware or software entities that make up the application. Types of resources include disk groups and file systems, network interface cards (NIC), IP addresses, and applications. A resource within Veritas Cluster Server is a specific instance of a service controlled by an agent. VCS may control the import of several disk groups and each one is an individual resource.

Each resource has its startup and shutdown order dictated by resource dependencies. This allows for multiple resources to be ordered based on OS or application requirements. For example, a file system resource would need the disk group resource it is contained within to be imported before the file system could be mounted when the service group is starting up. When the Veritas Operations Manager (VOM) or VCS Java GUI is used, a graphical representation of Resource dependencies can be displayed. The following is a graphical example of a service group dependency tree.



Configuration files

Veritas Cluster Server has two primary configuration files located in /etc/VRTSvcs/conf/config. These two files are main.cf, which is the primary configuration file and types.cf, which is used to define how bundled agents behave. If additional agents are installed and configured, they will have a specific types.cf file specific to their application. For example, if the Oracle agent is in use then a line at the top of the main.cf would include the OracleTypes.cf file to define how the agent is configured:

```
# cat /etc/VRTSvcs/conf/config/main.cf
include "types.cf"
include "OracleTypes.cf"
```

VCS keeps the main.cf and all types.cf files in sync on all nodes in the cluster. The cluster configuration is stored in the previously mentioned files. When the cluster is started those files are validated and read into the HAD process, which will be further discussed in the cluster communication section, on the local cluster node. If the main.cf file is changed while the cluster is online, no changes are introduced in the running cluster. There are two methods to modify the configuration within a running cluster:

1) Run CLI commands to modify the cluster configuration
2) Use the GUI to run commands to modify the cluster configuration

After the cluster is modified and the configuration is closed the changes are written to the main.cf and types.cf files on all nodes in the cluster to ensure all configs files stay in sync.

Cluster File System

GFS may exist in your current environment to enable filesystem access from each cluster node. This feature is not available by default with Veritas Cluster Server, though VCS is tightly integrated in the Storage Foundation Cluster File System product. Veritas Storage Foundation Cluster File System enables concurrent file access from multiple servers to provide a flexible, high-performance and highly available platform for shared data in a SAN environment. Veritas Storage Foundation Cluster File System is built for commercial transactional workloads and provides a single file system schema that is cache coherent. With Veritas Storage Foundation Cluster File System, cluster-wide volume and file system configuration allows for simplified management; and extending clusters is simplified as new servers adopt cluster-wide configurations. If GFS currently is in use then you should install and configure SF CFS, which includes VCS.

Cluster Communication

Veritas Cluster Server uses a cluster interconnect for network communications between cluster systems. Each system runs as an independent unit and shares information at the cluster level. On each system the VCS High Availability Daemon (HAD), which is the decision maker for the cluster, maintains a view of the cluster configuration. This daemon operates as a replicated state machine, which means all systems in the cluster have a synchronized state of the cluster configuration. This is accomplished by the following:

- All systems run an identical copy of HAD.
- HAD on each system maintains the state of its own resources, and sends all cluster information about the local system to all other machines in the cluster.
- HAD on each system receives information from the other cluster systems to update its own view of the cluster.
- Each system follows the same code path for actions on the cluster.

HAD communicates over a high-performance, low-latency replacement for the IP stack consisting of two components, Group Membership Services/Atomic Broadcast (GAB) and Low Latency Transport (LLT). These two components operate in a manner similar to the TCP and IP protocols in that they connect nodes and communicate information between them. In order to make these protocols as efficient as possible, a few layers in the TCP/IP stack have been removed. Because of this GAB and LLT heartbeat traffic is not routable though it can be configured using UDP. The following sections go into more detail on the specific protocols.

Low Latency Transport (LLT)

The Low Latency Transport protocol has two major functions.

Traffic distribution

LLT provides the communications backbone for GAB. LLT distributes (load balances) inter-system communication across all configured network links. This distribution ensures all cluster communications are evenly distributed across all network links for performance and fault resilience. If a link fails, traffic is redirected to the remaining links. A maximum of eight network links are supported.

Heartbeat

LLT is responsible for sending and receiving heartbeat traffic over each configured network link. LLT heartbeat is an Ethernet broadcast packet. This broadcast heartbeat method allows a single packet to notify all other cluster members the sender is functional, as well as provide necessary address information for the receiver to send unicast traffic back to the sender. The heartbeat is the only broadcast traffic generated by VCS. Each system sends 2 heartbeat packets per second per interface. All other cluster communications, including all status and configuration traffic is point to point unicast. This heartbeat is used by the Group Membership Services to determine cluster membership.

Group Membership Services/Atomic Broadcast (GAB)

The Group Membership Services/Atomic Broadcast protocol (GAB) has two major functions.

Cluster membership

GAB maintains cluster membership by receiving input on the status of the heartbeat from each system via LLT, as described below. When a system no longer receives heartbeats from a cluster peer, LLT passes the heartbeat loss to GAB. GAB marks the peer as DOWN and excludes it from the cluster. In most configurations, membership arbitration is used to prevent network partitions.

Cluster communications

GAB's second function is reliable cluster communications. GAB provides guaranteed delivery of messages to all cluster systems. The Atomic Broadcast functionality is used by HAD to ensure that all systems within the cluster receive all configuration change messages, or are rolled back to the previous state, much like a database atomic commit. While the communications function in GAB is known as Atomic Broadcast, no actual network broadcast traffic is generated. An Atomic Broadcast message is a series of point to point unicast messages from the sending system to each receiving system, with a corresponding acknowledgement from each receiving system.

About data protection

Membership arbitration by itself is inadequate for complete data protection because it assumes that all systems will either participate in the arbitration or are already down. Rare situations can arise which must also be protected against. Some examples are:

- A system hang causes the kernel to stop processing for a period of time.
- The system resources were so busy that the heartbeat signal was not sent. A break and resume function is supported by the hardware and executed. Dropping the system to a system controller level with a break command can result in the heartbeat signal timeout.

In these types of situations, the systems are not actually down, and may return to the cluster after cluster membership has been recalculated. This could result in data corruption as a system could potentially write to disk before it determines it should no longer be in the cluster.

Combining membership arbitration with data protection of the shared storage eliminates all of the above possibilities for data corruption.

Data protection fences off or removes access to the shared data storage from any system that is not a current and verified member of the cluster. Access is blocked by the use of SCSI-3 persistent reservations.

SCSI-3 Persistent Reservation

SCSI-3 Persistent Reservation (SCSI-3 PR) supports device access from multiple systems, or from multiple paths from a single system. At the same time it blocks access to the device from other systems, or other paths.

Veritas Cluster Server logic determines when to online a service group on a particular system. If the service group contains a disk group, the disk group is imported as part of the service group being brought online. When using SCSI-3 PR, importing the disk group puts registration and reservation on the data disks. Only the system that has imported the storage with SCSI-3 reservation can write to the shared storage. This prevents a system that did not participate in membership arbitration from corrupting the shared storage.

SCSI-3 PR ensures persistent reservations across SCSI bus resets. Membership arbitration combined with data protection is termed I/O Fencing. Coordination Point Server (CPS), Introduced in VCS version 5.1, can be used instead of a physical disk for use with I/O Fencing. CPS takes the place of a single disk. Multiple CPS servers could be used to replace all SCSI-3 PR disks within a cluster. The primary use case for Coordination Point Servers is within a distributed computing environment as the communication occurs over IP.

Note: Use of SCSI 3 PR protects against all elements in the IT environment that might be trying to write illegally to storage, not only VCS related elements.

Non-SCSI-3 Fencing

Due to some customers inability to implement SCSI-3 PR based on architecture issues or array incompatibility, VCS also offers an option to use a Non-SCSI-3 Fencing mechanism. This option uses network based CPS to act as a cluster arbitrators. As with the traditional SCSI-3 PR implementation of fencing, a race condition occurs when cross cluster communication is interrupted. With Non-SCSI-3 Fencing, there are no SCSI keys written to the shared storage, so it uses cluster arbitration and delay to provide a best effort method for preventing a split-brain condition within the cluster.

Term Comparison - RHC and VCS

Term	Red Hat Cluster	Veritas Cluster Server
Cluster	Cluster	Cluster
Cluster Member	Cluster Node	System
Framework to used to online, offline and monitor applications controlled by the cluster	Toolkit	Agent
A grouping of application services together	Services	Service Group
A grouping of application services that run on all nodes in the cluster at the same time	Services	Parallel Service Group

Heartbeat technologies	Ethernet	Ethernet
Cluster Split-Brain Protection	Power Fencing, SCSI-3 PR, Fibre Channel Switch Fencing and GNDB Fencing	SCSI-3 PGR/Coordination Point Server
Number of supported nodes	16	64
Cluster File System Support	GFS/GFS2	SFCFS

Command Comparison - RHC and VCS

Command Purpose	Red Hat Cluster	Veritas Cluster Server
Cluster startup	service cman start service clvmd start service gfs start service rgmanager start	hastart
Cluster shutdown	service rgmanager stop service gfs stop service clvmd stop service cman stop	hastop
Bring Online an application package/group	clusvcadm -e <group>	hagrp –online <SG> -sys <System>
Bring Offline an application package/group	clusvcadm -s <group>	hagrp –offline <SG> -sys <System>
Display the cluster status	clustat	hastatus -sum
Additional Terms	<group> = Service Name	SG = Service group System = Cluster Node

Veritas Cluster Server cluster heartbeats

Because VCS communicates using LLT and GAB protocols, it does not use IP communication in the default configuration. This requires that the connections between nodes not be routed and that each heartbeat NIC use a different VLAN. At least 2 NICs are required per cluster for heartbeats. Configurations requiring IP communication (e.g. stretched clusters utilizing WAN links) can alternatively use "LLT over UDP" (see appendix section of the **Veritas Cluster Server Install Guide**).

Migration Procedure

Planning Phase

In order to ensure a successful transition from RHC to VCS, several items need to be considered. To begin with the cluster heartbeats and data protection strategies need to be mapped out to determine if the current RHC heartbeats can be used for VCS. After the cluster communication is documented then each service under Red Hat Cluster control needs to be considered. If a VCS Agent is available for the resource to be managed, then the appropriate attributes need to be identified to properly control that resource. Each Agent has different attributes used to control resources. For example, an IP resource would require attributes such as identifying the NIC card to be used and the NetMask used to configure the IP as well as any specific options used when the IP is configured.

Planning is required to ensure an optimal implementation. The VCS configuration can be generated prior to the migration to make certain that all Single Points of Failure (SPOF) are identified and all RHC services are migrated.

The Planning phase of this document is intended to present a methodology to be used to properly prepare the user to migrate from RHC to VCS. Included is a sample migration which will show the steps taken during this process. Please use appropriate care when planning your migration.

Application downtime required

It should be noted that the migration process will require application downtime. This is because when the Red Hat Cluster is taken offline the applications must be brought offline. In comparison to Red Hat Cluster, VCS does not need to bring applications offline to stop the cluster. In VCS, the cluster can be taken offline while the applications the cluster maintains will continue to be online. This allows for cluster modification to occur without impacting the state of a running application. VCS also has a concept called freezing a service group. This puts the service group in a maintenance mode which will continue to monitor an application but will not take action if an application is brought offline without cluster knowledge. Please see the Veritas Cluster Server User Guide for additional information.

Identify applications and resources under Red Hat Cluster control

Identify all resources currently being controlled by the RHC cluster. These resources are everything from the NIC and failover IP address, to the Disk Group and File Systems, as well as the applications. To properly identify resources for migration, attention is required to understand the available agents using VCS. The following is a list of Agents available for VCS based on agent categories:

Application	apache_agent	sapwebas_agent	powercentersvcmgr_agent
	tuxedo_agent	oracleapps_agent	weblogic_agent
	oracleas_agent	websphere_agent	saplivecache_agent
	webspheremq_agent	sapnw_agent	
Database	db2_agent	sapmaxdb_agent	informix_agent
	sybase_agent	oracle_agent	
Replication	dataguard_agent	ntap_agent	db2hadr_agent
	srdf_agent	htc_agent	srdfstar_agent
	metro_mirror_agent	mirrorview_agent	svccopyservices_agent
Storage*	DiskGroup	DiskReservation	Volume
	Mount	LVMlogicalvolume	LVMvolumegroup
	SANVolume		
Network*	IP	NIC	IPMultiNIC
	MultiNICA	DNS	
File Share*	NFS	NFSRestart	Share
	SambaServer	NetBIOS	SambaShare
Service*	Application	Process	ProcessOnOnly
Infrastructure*	NotifierMngr	VRTSWebApp	Proxy
	Phantom	RemoteGroup	
Testing*	ElifNone	FileNone	FileOnOff
	FileOnOnly		
*Agents that are bundled with the Product. All other agents are bundled in a free Agent Pack to allow for updates on a continuous basis			

To determine how to properly implement and the capabilities of each agent please see the **Veritas Cluster Server Bundled Agents Reference Guide** and the **Veritas Cluster Server Agent Pack**.

Veritas Cluster Server Hardware Prerequisites

The primary hardware requirement for Veritas Cluster Server is related to cluster communication over heartbeats. VCS requires a minimum of 2 NICs to be used for heartbeats. VCS, as with any installed application, has disk space requirements. The current requirements can be found in the **VCS Installation Guide for Linux** or through the Symantec Operations Readiness Toolkit website – **http://sort.symantec.com**

There may be additional hardware requirements to protect the applications and avoid Single Points of Failure (SPOF). When architecting the environment, it is essential to validate that all required resources are examined to guard against error conditions. If SCSI-3 is planned to be included in the environment then the shared storage array will need to have that feature enabled and disks will need to be assigned to the coordinator disk group. To validate the implementation availability of this feature, please see the **Veritas Cluster Server Installation guide** for more information.Veritas Cluster Server Linux OS Prerequisites

With Veritas Cluster Server 5.1 SP1 the following OS versions are supported:

- Red Hat Enterprise Linux 5 (RHEL 5) with Update 3 (2.6.18-128.el5 kernel) or later on AMD Opteron or Intel Xeon EM64T (x86_64)

- SUSE Linux Enterprise Server 10 (SLES 10) with SP2 (2.6.16.60-0.21 kernel) or SP3 on AMD Opteron or Intel Xeon EM64T (x86_64)
- SUSE Linux Enterprise Server 11 (SLES 11) (2.6.27.19-5-default kernel) or SUSE Linux Enterprise Server 11 (SLES 11) with SP1 on AMD Opteron or Intel Xeon EM64T (x86_64)
- Oracle Enterprise Linux (OEL 5) with Update 3 or later
- VCS does not support Xen kernels with any distribution.

Note: 64-bit operating systems are only supported.

Please see the **VCS Release notes** for the latest details.

Preparing for Veritas Cluster Server Installation

There are several steps that need to be performed as pre-installation tasks. These include establishing the heartbeat connections, validating shared storage is in place, deciding if SCSI-3 PGR for I/O Fencing will be implemented and obtaining a license key depending on the version of VCS to be installed(permanent, temporary or keyless) to be used during installation. For all pre-installation tasks please see the **Veritas Cluster Server Installation Guide for Linux**. Appropriate documentation for Storage Foundation Cluster File System can also be found here.

Implementation Phase

Shutdown the RHC cluster

To ensure an easy backout plan, be sure to backup your RHC cluster configuration file (**/etc/cluster/cluster.conf).** Before stopping the cluster, ensure that all services are offline by running the following command:
clustat

For each package that is still running, issue the command for them to shutdown:
clusvcadm –s <group>

The following command sequence is then used to stop and disable the Red Hat Cluster daemons. It's required to run the command sequence on every single node in the cluster.

service rgmanager stop
service gfs stop
service clvmd stop
service cman stop

chkconfig --level 2345 rgmanager off
chkconfig --level 2345 gfs off
chkconfig --level 2345 clvmd off
chkconfig --level 2345 cman off

Uninstall RHC software

To uninstall Red Hat Cluster, use the yum command to erase on all the Red Hat Cluster rpms that are installed. The uninstall process can be done at a later date to allow for a migration backout plan. VCS and RHC can be installed on the same box as long as when VCS controls the applications Red Hat Cluster daemons are disabled, the startup of Red Hat Cluster processes are disabled and only VCS is controlling the application resources. Here is an example of the uninstallation command for the Red Hat Cluster rpm:

yum erase cman

NOTE: Depending on your configuration, more components may have to be uninstalled.

Veritas Cluster Server Cluster Configuration

If I/O Fencing is to be utilized within the VCS cluster then the disks to be used need to be validated, initialized, setup in a disk group and made ready to be included within the configuration. As a note, SCSI-3 PR using I/O Fencing requires VxVM. Non-SCSI-3 Fencing (NSF) and Coordination Point Servers (CPS) can be used for cluster arbitration. For full instructions on how to setup and validate if SCSI-3 can be used for VCS in the environment or how to implement NSF and CPS within your cluster, please see the **VCS Installation Guide for Linux** for further information. VCS can be configured using LVM as the volume manager as well as VxVM for application data

Veritas Cluster Server installation

Veritas Cluster Server is installed via the Veritas Common Product Installer or installvcs script. For details for usage, please reference the VCS installation procedures as outlined in the Veritas **Cluster Server Installation Guide for Linux**.

Veritas Cluster Server Heartbeats

Veritas Cluster Server Heartbeats will be established during the binary installation process. The installer script asks which NIC will be used for heartbeats. The NICs can be different on each node in the cluster but it is preferred to have the configurations be as similar as possible. VCS Heartbeats need to be on separate networks or VLAN to add redundancy and reduce the possibility of a single LAN causing all Heartbeat links to go down at once.

Veritas Cluster Server Cluster Creation

Veritas Cluster Server can be modified using 3 different methods: Java GUI or connection to the Veritas Operations Server (VOS), VCS CLI commands to modify an already running cluster or editing the cluster configuration file (main.cf) when the cluster is offline. During VCS installation a configuration file is created. It contains the systems that were designated during the installation process. It may also contain services to send out SNMP/SMTP alerts if they were configured during installation.

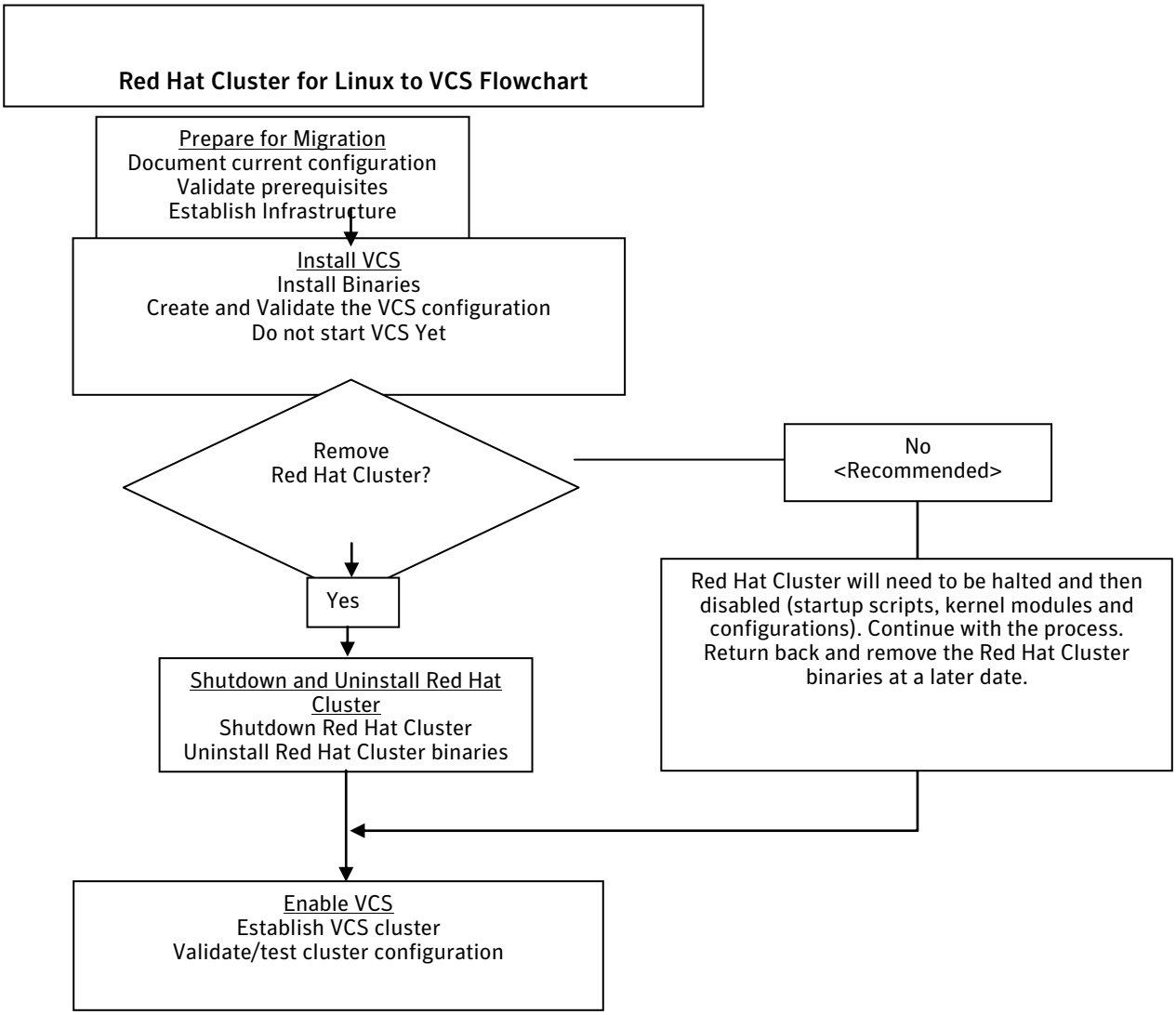
It is at this point that the information from the Red Hat Cluster (RHC) needs to be migrated into the VCS cluster. Each service in the RHC, which is necessary for the application to function, will need to be established within VCS. These services can be implemented using any of the three methods to modify the VCS cluster. Examples of this are provided in a later section of this document.

Verify the Veritas Cluster Server Cluster

When all of the services migrated from Red Hat Cluster are now configured within VCS, several additional steps should be taken to ensure the ability to properly administer the cluster. These steps include adding VCS users with appropriate privileges and determining which method will be used to control the cluster.

Veritas Cluster Server validation and testing

A plan needs to be established to validate the cluster functionality. VCS has local HA FireDrill capabilities that can be used to determine if the cluster was setup properly. In addition to using the FireDrill function, cluster testing should be performed to confirm that the configuration acts as expected.


Red Hat Cluster for Linux to VCS Flowchart
Prepare for Migration
Document current configuration
Validate prerequisites
Establish Infrastructure
Install VCS
Install Binaries
Create and Validate the VCS configuration
Do not start VCS Yet
Remove
Red Hat Cluster?
No
<Recommended>
Red Hat Cluster will need to be halted and then disabled (startup scripts, kernel modules and configurations). Continue with the process. Return back and remove the Red Hat Cluster binaries at a later date.
Yes
Shutdown and Uninstall Red Hat Cluster
Shutdown Red Hat Cluster
Uninstall Red Hat Cluster binaries
Enable VCS
Establish VCS cluster
Validate/test cluster configuration

Methods of Controlling the Veritas Cluster Server Cluster

There are several methods of managing the Veritas Cluster Server Clusters. Historically VCS administrators have used the command line to control the cluster, maintain the configuration, and monitor the status of the cluster. This option of management continues to be available. There are two major disadvantages of the CLI, knowledge of the commands to be used and their lack of graphical representation. Since VCS was introduced it has included a GUI that could be installed to manage the cluster in a graphical mode. The Java console enables the cluster to be managed by users from their local PCs.

Along with the command line for VCS and the Java Console, Symantec has developed management utilities that enable features beyond just controlling the cluster. VCS Management Console allows for the management of several clusters at once. It supports advanced capabilities in reporting and configuration checking utilities. Storage Foundation Manager is an additional console that includes the ability to maintain Storage Foundation as well as VCS with limited functionality in one application. The final Console in the Symantec management strategy is the Veritas Operations Manager, which is the combination of Storage Foundation Manager and VCS Management Console. VOM will also include capabilities not found within SFM or VCS MC such as the ability to determine appropriate patches, run reports on VCS trends and detect barriers to successful failover, both for Global clusters as well as local clusters.

With all of the management utilities available to maintain the VCS cluster, they are included with Veritas Cluster Server.

Summary of VCS management capabilities

VCS Command Line Interface (CLI):

Single cluster management UI
Commands are consistent across Operating Systems and VCS versions
Every node in the cluster can be used to run commands against the cluster
No additional packages are required for use

VCS Java Console:

Single cluster management UI
Will no longer be packaged with SFHA/VCS when version 5.1 is GA and will be available as a download from **http://go.symantec.com/vcsmc**
It is recommended to use VOM as future features may not be available with the Java GUI.

VCS Cluster Simulator:

Veritas Cluster Server Simulator helps administrators simulate high availability environments from their laptops
It enables the ability to test multiple application failover scenarios without impacting production
Creating cluster configurations simplify installations as the configuration is available to test before installation.
The **download location** for the cluster simulator also contains a flash demo on the product

Veritas Operations Manager (VOM):

Multi-Cluster Management and Reporting tool
Supports stretch clusters and global clusters including site-to-site migration and DR
Includes proactive checks with Firedrill scheduling
Management Server installs on Win, Linux and Solaris
Can be downloaded from **http://go.symantec.com/vom**
More on SCORE: **http://score.corp.symantec.com/products/237**

Appendix reference information

Migration Planning – VCS Cluster Information

This section is provided as a sample cluster configuration form. Information gathered from the Red Hat Cluster can be used to configure Veritas Cluster Server. Fill in as much data as possible in the below forms to ease the VCS cluster configuration. For more information on how to implement VCS please see the earlier portions of this document or the **Veritas Cluster Server for Linux Installation guide**.

LVM Volume Group Information
==========

Volume Group Name: _____

Logical Volume Name: _____ Mount Point: _____

Logical Volume Name: _____ Mount Point: _____

Logical Volume Name: _____ Mount Point: _____

Volume Group Name: _____

Logical Volume Name: _____ Mount Point: _____

Logical Volume Name: _____ Mount Point: _____

Logical Volume Name: _____ Mount Point: _____

Volume Group Name: _____

Logical Volume Name: _____ Mount Point: _____

Logical Volume Name: _____ Mount Point: _____

Logical Volume Name: _____ Mount Point: _____

VxVM Disk Group

==========

==========

Disk Group Name: _____

Volume Names: _____ Mount Point: _____

Volume Names: _____ Mount Point: _____

Volume Names: _____ Mount Point: _____

Disk Group Name: _____

Volume Names: _____ Mount Point: _____

Volume Names: _____ Mount Point: _____

Volume Names: _____ Mount Point: _____

Disk Group Name: _____

Volume Names: _____ Mount Point: _____

Volume Names: _____ Mount Point: _____

Volume Names: _____ Mount Point: _____

VCS Cluster Information

==========

==========

Cluster Name: _____

Cluster Nodes: _____

Cluster Number: _____

CVM/Oracle RAC Information

==========

==========

Oracle RAC Version: _____ Node Names running RAC: _____

CVM Disk Group(s): _____

CVM Volume(s): _____

Heartbeat Information

==========

==========

Heartbeat NICs (2 minimum): _____ _____ _____ _____ _____

SCSI-3 I/O Fencing Information (SCSI-3 is Optional and Non-SCSI-3 Fencing can also be used)

==========

==========

SCSI-3 Disk Group: _____

SCSI-3 Disks: _____ _____ _____

______________ ______________ ______________ ______________

Have these Devices been tested with the vxfentsthdw command?(Yes/No) ______

CPS (IP address of Server): ______________ ______________ ______________

GUI User Security (User Admin is created by default)

==
==========

User name: ______________________

Access: __________ Cluster Admin/Cluster Operator/Group Admin/Group Operator/Guest

Service group: _____________ required if Group Admin or Group Operator

Service group Information

==
==========

Service group Name: ________________ Service group type (Parallel or Failover): ___________

System List: ____

Auto Start List: ____

Resources:

Resource Name: ____________ Type of Resource/Agent to be used: _______________

Resource Attributes (Different for each type of resource): __________________________

____

Is the Resource Critical? _____________

Resource Name: ____________ Type of Resource/Agent to be used: _______________

Resource Attributes (Different for each type of resource): __________________________

____

Is the Resource Critical? _____________

Resource Name: ____________ Type of Resource/Agent to be used: _______________

Resource Attributes (Different for each type of resource): __________________________

____

Is the Resource Critical? _____________

Resource Name: ____________ Type of Resource/Agent to be used: _______________

Resource Attributes (Different for each type of resource): __________________________

____

Is the Resource Critical? _____________

Resource Name: ____________ Type of Resource/Agent to be used: _______________

Resource Attributes (Different for each type of resource): __________________________

____

Is the Resource Critical? _____________

Dependencies (What is the order of startup)

Resource named: ________________ requires that ________________ resource is online first

Resource named: _____ requires that _____ resource is online first

Resource named: _____ requires that _____ resource is online first

Resource named: _____ requires that _____ resource is online first

Resource named: _____ requires that _____ resource is online first

Service group Dependencies (What is the dependency between service groups?)

Service group named: _____ requires that _____ service group is _____ (online or offline) _____ (local or global) _____ (firm – if this mandatory)

Network Information (goes in the appropriate service groups):

==========

Virtual IP _____ subnet mask_____ associated NIC: _____

Network Hosts: _____ a List of IPs Used to tests if a NIC is online by ping

Notification Information:

==========

SMTP Server: _____

SMTP Recipients: _____ Notification Level: _____-Information/Warning/Error

SMTP Recipients: _____ Notification Level: _____-Information/Warning/Error

SMTP Recipients: _____ Notification Level: _____-Information/Warning/Error

SMTP Recipients: _____ Notification Level: _____-Information/Warning/Error

SNMP Server: _____

Attribute Information

==========

Each level within the cluster has default values. These attributes can be modified to enable the preferred behavior. The following is a sample of attributes that can be modified. For a full listing please see the **Veritas Cluster Server Administrators Guide**.

Agent Attributes:

An Agent is the binaries that control an application or process. This control of an application is the startup, shutdown, monitor and clean procedures. Each Agent has specific attributes necessary to control and monitor the application/process defined. When there is a specific instance of an application, for example a NIC card, then that is a resource. There are additional attributes that are used with the agent to control how it functions. The following are a couple of default variables that can be modified to control how the cluster behaves on a per Agent basis:

MonitorInterval (How often is a resource monitored?) ____ Default 60 (seconds)

OfflineMonitorInterval (Same as MonitorInterval but on the Offline node) ____ Default 300 (seconds)

RestartLimit (The number of times a resource can restart before failing) ___ Default 0

OnlineRetryLimit (The limit in attempting to bring a resource online during startup) ___ Default 0

IMF(Should this resource type be monitored using IMF?) ____ Default No

Resource Attributes:
Each Resource has the attributes to control an application using an Agent. For example a Mount Resource requires information on the specific File System to be managed. Beyond the specific information passed to the Agent to manage the Resource there are default values that change the behavior of service group. Here is an example of an attributes that can be modified for each Resource:

Critical (This specifies if the resource goes offline unexpectedly will it cause the service group to failover)

Step-by-step migration with sample applications – RHC -> VCS

Migration Steps:

1. Perform pre-planning steps to gather existing configuration information and application information to migrate to Veritas Cluster Server
    - This includes ensuring that VCS installation binaries and a license key is are available unless keyless licensing is to be used with VCS 5.1SP1
    - There are additional pre-planning steps needed to utilize certain features within the product such as the Veritas Operations Manager and authentication broker. For additional information please see the **Veritas Cluster Server on Linux installation guide**
    - To install all nodes within a cluster at one time, trusted SSH communication needs to be in place before VCS is installed
2. Validate Heartbeat network communication
    - Ensure that NICs can communicate only to their corresponding pair
      For example, NIC1 on Box1 can only ping NIC1 on Box2 and cannot ping NIC2 on Box2
3. Bring the Red Hat Cluster down on all nodes and disable the cluster from startup
    - Run the command sequence specified in the Implementation Phase section of this document.
    - Backup all Red Hat Cluster configuration files
      Use #chkconfig to disable all services specified in the Implementation Phase section of this document.
4. Install the VCS software
    - With the CD in place run installer or go into the cluster_server directory and run installvcs
    - Continue through the installation menus with information regarding the cluster setup in the pre-planning steps. You have a choice when using the installer to just install the binaries (RPMs) or to install the binaries and configure the cluster. If the pre-planning phase has been completed, the install and configure option should be selected. The info required for use with this installation and configuration method includes:
        i. License Key (unless Keyless Licensing is to be used with VCS version 5.1SP1)
        ii. Cluster name and number to be used
        iii. Heartbeat NICs
        iv. Will the Symantec Product Authentication Service be used? If not then the configuration of VCS Users (Username, User Access, Password)
        v. Establishing communication with a Veritas Operations Manager if one is to be used
        vi. The setup of SNMP and SMTP notification if these will be used
        vii. The setup and configuration of I/O Fencing (SCSI-3, Non-SCSI-3, CPS)
5. At this point the cluster has been established and a base configuration was created. Our next step is to configure the Services under RHC within VCS. As a note, this step can be done prior to VCS binaries being installed to reduce downtime
    - Take each Red Hat Cluster Service and port it to VCS
    - Examples of this can be seen in the Appendix that shows the output of the RHC configuration files and the VCS Configuration files
    - Depending on your environment you can edit the configuration files manually or when the cluster is use the CLI, Java GUI or the Veritas Operations Manager to configure the cluster.
6. With any cluster software installation is to validate it is configured correctly.
7. In our example we moved the startup scripts out of place in Step #3 rather than uninstalling the Red Hat Cluster binaries. When the migration is complete and tested Red Hat Cluster needs to be uninstalled. Put the original files back in place and uninstall the Red Hat Cluster packages.

RHC Configuration Files Examples

The following files were used in setting up a generic configuration within RHC used to cluster an Apache. We have a two node cluster (node01 and node02) that manages a single package (apache-pkg).

/etc/cluster/cluster.conf (for brevity all comments as well as executable lines of code are not included):

```
<?xml version="1.0"?>
<cluster alias="apachecluster" config_version="32" name="apachecluster">
    <fence_daemon clean_start="0" post_fail_delay="0" post_join_delay="3"/>
    <clusternodes>
        <clusternode name="node02.solutions.com" nodeid="2" votes="1">
            <fence/>
        </clusternode>
        <clusternode name="node01.solutions.com" nodeid="1" votes="1">
            <fence/>
        </clusternode>
    </clusternodes>
    <cman expected_votes="1" two_node="1"/>
    <fencedevices/>
    <rm>
        <failoverdomains>
            <failoverdomain name="productiondomain" nofailback="0" ordered="0"
restricted="0">
                <failoverdomainnode name="node02.solutions.com" priority="1"/>
                <failoverdomainnode name="node01.solutions.com" priority="1"/>
            </failoverdomain>
        </failoverdomains>
        <resources>
            <script file="/etc/rc.d/init.d/httpd" name="httpd"/>
            <lvm lv_name="apachevol" name="apachegroup_LVM" vg_name="apachegroup"/>
            <ip address="10.14.192.168" monitor_link="1"/>
            <fs device="/dev/apachegroup/apachevol" force_fsck="0" force_unmount="0"
fsid="12493" fstype="ext2" mountpoint="/docroot" name="docroot_Mount" self_fence="0"/>
        </resources>
        <service autostart="1" exclusive="1" name="apachegrp">
            <lvm ref="apachegroup_LVM">
                <fs ref="docroot_Mount">
                    <ip ref="10.14.192.168">
                        <script ref="httpd"/>
                    </ip>
                </fs>
            </lvm>
        </service>
    </rm>
    <totem consensus="4800" join="60" token="10000"
token_retransmits_before_loss_const="20"/>
</cluster>
```

Veritas Cluster Server Configuration Files Examples

The following are the files used in the configuration of Veritas Cluster Server for the same Apache Service within VCS it is called a service group. As a note, the files are collected from the redhat1 system. Files specific to the system, like the /etc/VRTSvcs/conf/sysname file reflect this.

/etc/llttab:

```
set-node /etc/VRTSvcs/conf/sysname
set-cluster 200
link eth1 eth1 - ether - -
link eth2 eth2 - ether - -
```

/etc/VRTSvcs/conf/sysname:

redhat1

/etc/llthosts:

0 redhat1
1 redhat2

/etc/VRTSvcs/conf/config/main.cf:

```
include "types.cf"

cluster redhatcluster (
    UserNames = { admin = HopHojOlpKppNxpJom }
    Administrators = { admin }
    )

system redhat1 (
    )

system redhat2 (
    )

group apachegrp (
        SystemList = { redhat1 = 0, redhat2 = 1 }
        AutoStart = 0
        AutoStartList = { redhat1 }
        )

        Apache apacheserver (
                httpdDir = "/usr/sbin"
                HostName = apache-1
                User = root
                SecondLevelMonitor = 1
                ConfigFile = "/apache1/conf/httpd.conf"
                )

        DiskGroup apachedg (
                DiskGroup = apachedg
                )

        IP apacheip (
                Device = eth0
                Address = "192.168.1.2"
```

```
        NetMask = "255.255.255.0"
        )

Mount apachemnt (
        MountPoint = "/apache1"
        BlockDevice = "/dev/vx/dsk/apachedg/apachevol"
        FSType = vxfs
        MountOpt = rw
        FsckOpt = "-y"
        )

NIC apachenic (
        Device = eth0
        )

Volume apachevol (
        DiskGroup = apachedg
        Volume = apachevol
        )

apacheip requires apachenic
apachemnt requires apachevol
apacheserver requires apacheip
apacheserver requires apachemnt
apachevol requires apachedg
```

Red Hat Cluster and Veritas Cluster Server Configuration Files Migration Example

	Red Hat Cluster configuration **/etc/cluster/cluster.conf**	Veritas Cluster Server configuration **/etc/VRTSvcs/conf/config/main.cf**
This section of the document will be provided after the analysis of Red Hat Cluster Version 6 is complete.		

Reference Documentation

For additional information on Veritas Cluster Server for Linux see our document repository located at: **http://sfdoccentral.symantec.com/index.html**

VCS Command Line quick reference

Start VCS

hastart (-force) (-stale)

Stop VCS

```
# hastop -local [-force | -evacuate]
# hastop -sys system_name [-force | -evacuate]
# hastop -all [-force]
```

```
-local stops HAD on the system where you
 type the command.
-sys stops had on the system you specify.
-all stops had on all systems in the
 cluster.
```

Change VCS Configuration Online

```
haconf –makerw
```
...make changes...
```
haconf –dump –makrero
```

Get Current Cluster Status

hastatus -summary

Agent Operations

Stop and start agents manually.

haagent -start *agent_name* -sys *system_name*
haagent -stop *agent_name* -sys *system_name*

Add and Delete Users

Add a user with read/write access to the VCS configuration.	# hauser -add *user_name* Enter a password when prompted.
Add a user with read-only access.	# hauser -add VCSGuest Press Return when prompted for a password.
Modify a user.	# hauser -modify *user_name* Enter a new password when prompted.
Delete a user.	# hauser -delete *user_name*
Display a user. If *user_name* is not specified, all users are displayed.	# hauser -display *[user_name*]

System Operations

List systems in the cluster.	# hasys -list
Get detailed information about each system.	# hasys -display *[system_name*]
Add a system. Increase the system count in the GAB startup script.	# hasys -add *system_name*
Delete a system.	# hasys -delete *system_name*

Resource Types

List resource types.	# hatype -list
Get detailed information about a resource type.	# hatype -display *[type_name*]
List all resources of a particular type.	# hatype -resources *type_name*
Add a resource type.	# hatype -add *resource_type*
Set the value of static attributes.	# hatype -modify ...
Delete a resource type.	# hatype -delete *resource_type*

Resource Operations

List all resources	# hares -list
List a resource's dependencies.	# hares -dep *[resource_name*]
Get detailed information about a resource.	# hares -display *[resource_name*]
Add a resource.	# hares -add *resource_name resource_type service_group*
Modify the attributes of the new resource.	# hares -modify *resource_name attribute_name value*
Delete a resource, type.	# hares -delete *resource_name*
Online a resource, type.	# hares -online *resource_name* -sys *system_name*
Offline a resource, type.	# hares -offline *resource_name* -sys *system_name*
Cause a resource's agent to immediately monitor the resource on a particular system.	# hares -probe *resource_name* -sys *system_name*
Clear a faulted resource.	# hares -clear *resource_name* [-sys *system_name*]
Make a resource's attribute value local.	# hares –local *resource_name attribute_name value*
Make a resource's attribute value global.	# hares –global *resource_name attribute_name value*
Specify a dependency between two resources.	# hares -link *parent_res child_res*
Remove the dependency relationship between two resources:	# hares -unlink *parent_res child_res*

Service Group Operations

List all service groups.	# hagrp -list
List a service group's resources.	# hagrp -resources *[service_group*]
List a service group's dependencies.	# hagrp -dep *[service_group*]
Get detailed information about a service group.	# hagrp -display *[service_group*]
Start a service group and bring its resources online.	# hagrp -online *service_group* –sys *system_name*
Stop a service group and take its resources offline.	# hagrp -offline *service_group* -sys *system_name*
Switch a service group from one system to another. *(failover groups only)*	# hagrp -switch *service_group* -to *to_system*
Freeze a service group (disable onlining and offlining).	# hagrp -freeze *service_group* [-persistent]
Thaw a service group (reenable onlining and offlining).	# hagrp -unfreeze *service_group* [-persistent]
Enable a service group.	# hagrp -enable *service_group* [-sys *system_name]*
Disable a service group.	# hagrp -disable *service_group* [-sys *system_name]*
Enable all the resources in a service group.	# hagrp -enableresources *service_group*
Disable all the resources in a service group.	# hagrp -disableresources *service_group*
Specify the dependency relationship between two service groups.	# hagrp -link *parent_group child_group relationship*
Remove the dependency between two service groups.	# hagrp -unlink *parent_group child_group*

VCS Procedures

VCS Directory Structure

Binaries /opt/VRTSvcs/bin
Configuration /etc/VRTSvcs/conf/config
Logs /var/VRTSvcs/log

Determine the Status of the Cluster

hastatus –sum
hastatus
or check out the /var/VRTSvcs/log/engine.log_A

To Failover the ServiceGroup from One system to another

hagrp –switch <SG> –to <SYSTEM>

To Freeze/Unfreeze the ServiceGroup

hagrp –freeze <SG>
hagrp –unfreeze <SG>

The scripts that start VCS on boot

/etc/rc2.d/S70llt
/etc/rc2.d/S92gab
/etc/rc3.d/S99vcs

To clear a faulted resource

First determine the reason for the fault from the log files and messages files
Second run the command:
hares –clear <RESOURCE>

Hastart/Hastop options

Hastart has to be started from each box if the cluster goes down.
If you reboot the cluster (vcs) will be started upon boot.
Hastop has two primary options (-local or –all).
When stopping the cluster you have to consider if you want just the local system within the cluster or if the entire cluster need to have VCS stopped.
The “hastop –all –force” command will stop VCS on all nodes in the cluster but will not stop the resources. This allows for VCS to be shutdown without affecting the applications that VCS is configured to manage.

Modifying the Cluster Config
There are three ways to modify the cluster:
1) Take all systems offline and edit the main.cf configuration file. Run “hacf –verify .”
2) Edit the cluster from the GUI while the system is up.
3) Run commands to modify the cluster while it is up.

Adding a new filesystem to the cluster
1) Create the volume from Volume Manager
2) Freeze the ServiceGroup you will be working on/modifying
3) Click to open the Cluster Configuration file
4) On the GUI click on add a resource
We will add a Mount Resource for each mounted filesystem
5) For the Mount Resource you will need the Block Device, Mount Point, and the FS Type
The Last step is to add dependencies
6) To add dependencies select the mount resource and then click on the volume resource
7) Next add all other dependencies (Mnt -> DG, if the Mount needs another mount, etc.)
8) Finally dump the cluster config to propagate the config to all other boxes
9) Then close the Cluster Config
10) When the cluster boots up and all mount points are added and are up unfreeze the ServiceGroup

reboot/init 6/shutdown commands DO failover applications

The application will come offline and the system will be rebooted. The rebooting system is executing the K10vcs rc script which contains:

$HASTOP –sysoffline

This translates to: hastop -local -evacuate -noautodisable

The “evacuate” option initiates the ServiceGroup failover. When the system comes online the ServiceGroup should be located on a different system in the cluster.

Add a user to the GUI

The cluster needs to be open to writing first, so run the command:
haconf –makerw
Next add a user with the command:
hauser –add <user>
The system will prompt you for a password.
If none is entered then the user has read-only permissions
If the added user needs more than guest permissions run the command:
haclus -modify Administrators/Operators -add <username>
hagrp -modify <grpname> Administrators/Operators -add <username>
When finished close the cluster config by running the command:
haconf –dump –makero
This command will dump the config out to all systems connected to the cluster currently,
And then close the config.

Agent Scripts
The agents rely on scripts to bring the resources online/offline/monitor.
The scripts are located in /opt/VRTSvcs/bin or /opt/VRTSagents/ha/bin directory.
Each Agent has its own directory and the online/offline/monitor/clean files
The custom agent written is located in the directory of that agent type

About Symantec

Symantec is a global leader in providing security, storage and systems management solutions to help businesses and consumers secure and manage their information. Headquartered in Mountain View, Calif., Symantec has operations in 40 countries. More information is available at www.symantec.com.

For specific country offices and contact numbers, please visit our Web site. For product information in the U.S., call toll-free 1 (800) 745 6054.

Symantec Corporation
World Headquarters
350 Ellis Street
Mountain View, CA 94043 USA
+1 (408) 517 8000
1 (800) 721 3934
www.symantec.com

02/08