Application Note: Clustering configurations supported for VCS with vSphere

Application Note: Setting up Veritas Cluster Server with VMware vSphere

Documentation version: 1.0

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/

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/

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

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

- Product release level

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

Licensing and registration

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

www.symantec.com/business/support/

Customer service

Customer service information is available at the following URL:

www.symantec.com/business/support/

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

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

Support agreement resources

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

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

Contents

Chapter 1

Veritas Cluster Server setup for VMware vSphere

This chapter includes the following topics:

- VCS cluster on a single ESX host
- VCS cluster across ESX hosts
- VCS cluster between physical and virtual machines
- Disaster recovery setups with clustering configurations
- Supported vSphere versions and guest operating systems
- Feature compatibility matrix for various shared storage configurations

VCS cluster on a single ESX host

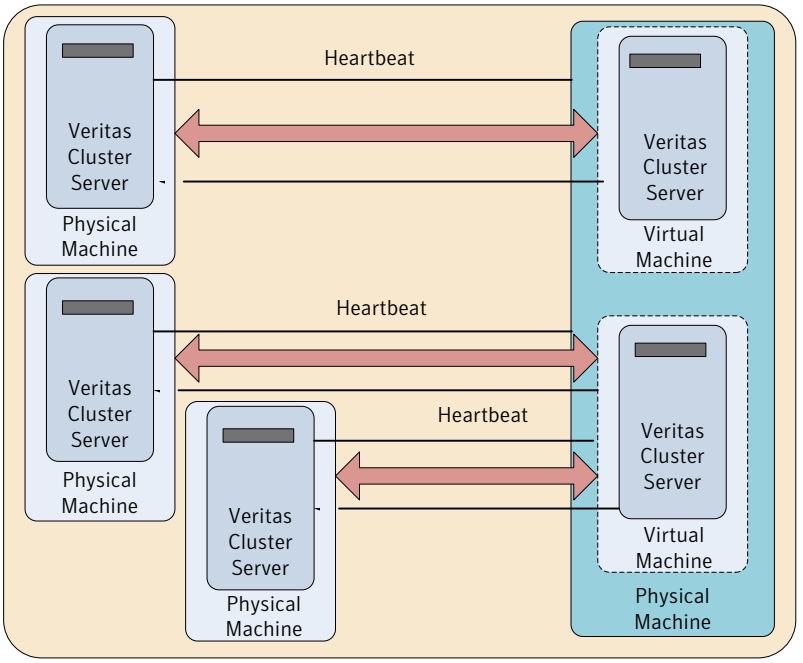
In this configuration, a VCS cluster runs on the same ESX host connected to a local or remote storage. This configuration protects against failures at application and operating system level but cannot protect against hardware failure. The VCS cluster shares a private network for the private heartbeat.

VCS cluster across ESX hosts

This configuration protects against both software and hardware failures. In this configuration, the VCS nodes are distributed across two or more physical ESX hosts. The shared storage can be on SAN or NFS, and SAN can be either iSCSI SAN or FC SAN. The virtual machine on each ESX host runs the clustering software. The virtual machines on each ESX host share a private network with each other for the private heartbeat.

VCS cluster between physical and virtual machines

In this configuration, the failover node is a VM. You can set up a standby VCS cluster node on a virtual machine for one or more VCS cluster nodes on a physical system. Therefore, in case of a hardware failure on any one or all of the physical systems, the standby VCS cluster on the virtual machine can take over the operations of the physical systems and vice versa. The shared storage in this configuration must be based on RDM, with either physical or virtual SCSI bus sharing.



Disaster recovery setups with clustering configurations

Some of the clustering configurations can be used as a disaster recovery (DR) setup. One of the nodes must be configured as Primary site and the other as DR site. Both sites must be in different geographical locations to isolate them from suffering from the same environmental disaster. Thus, the clustering configurations can provide seamless data access even in natural disasters such as floods, hurricanes, earthquakes, and so on.

VCS cluster across ESX hosts for DR

In this configuration, the failover node is a VM. You can set up a standby VCS cluster node on a virtual machine for one or more VCS cluster nodes on one or more physical systems. Therefore, in case of a hardware failure on any one or all of the physical systems, the standby VCS cluster on the virtual machine can take over the operations of the physical systems and vice versa.

VCS cluster between physical and virtual machines for DR

In this configuration for DR, either the physical or the virtual machine can be configured as the Primary site and the other as the DR site. Thus, in case of a disaster, the Primary site fails over to the DR site to restore accessibility and uninterrupted operations.

Supported vSphere versions and guest operating systems

The following guest operating systems are supported:

Table 1-1 displays the supported VCS versions on different guest OS for the specified vSphere versions

	For vSphere 4.0			For vSphere4.1		
	Linux (all flavors supported by VCS)	Windows (all flavors supported by VCS)	Solaris (x86)	Linux (all flavors supported by VCS)	Windows (all flavors supported by VCS)	Solaris (x86)
Supported VCS versions	5.0	5.1SP1	5.0	5.0	5.1SP1	5.0
	5.1	5.1SP1AP1	5.1	5.1	5.1SP1AP1	5.1

Feature compatibility matrix for various shared storage configurations

Table 1-2 indicates which vSphere features are compatible with differen storage configurations

VM storage configurations	Compatible vSphere features			
	VMotion	DRS	VMware HA [*1]	VMware Snapshots
vmdk on VMFS	N	N	N[*2]	N
Physical and Virtual RDM with FC SAN	N	N	Y	N
Physical and Virtual RDM with iSCSI SAN	N	N	Y	N
vmdk on NFS	N	N	N	N
iSCSI inside guest	Y	Y	Y	Y[*3]
NFS inside guest (using Mount agent)	Y	Y	Y	Y[*3]
SnapDrive (only windows)	Y	Y	Y	Y[*3]

*1: Refer to appendix A, since certain settings are needed for the VMware HA to work well with VCS-in-guest.

*2: In the event of a hardware failure, all the VMs may failover to another host, but this method requires VMs to be on the same host.

*3: Taking snapshots is possible if VCS and all applications are fully stopped. Before reverting to the snapshot, shutdown VCS and all applications and then revert to the snapshot. Ensure that no software updates or configuration changes related to VCS have been applied post the snapshot.

Appendix A

Coexistence of VMware HA with VCS in guest

This appendix includes the following topics:

- Motivation
- Settings
- Recommendations
- Availability levels

Motivation

VMware HA is a VMware solution for high-availability of virtual machines to protect against host failures, hardware failures, OS crashes, etc.

VCS-in-guest in VMware environments provides high-availability of applications inside guest by providing protection from host failures, hardware failures, OS crashes and also application failures at software layer (for example, in cases of application hang, file-level corruptions at OS level that cannot be resolved with reboot).

Since there is a cost involved in maintaining standby VMs, you may choose to protect only specific applications using VCS-in-guest and protect the remaining applications using VMware HA. By using VMware HA in conjunction with VCS-in-guest, when a host fails, standy VCS-nodes running on that host are automatically restarted by VMware HA on a new host without the need for user-intervention, potentially eliminating the need to maintain multiple standbys.

See “Availability levels” on page 16.

Settings

Since VMware HA and VCS-in-guest work with different timers, there are certain settings that you must enforce to make sure they work well together.

The following are the settings that you need to ensure the two HA solutions work together.

- 'VMware HA' *das.failuredetectiontime* time must be around 10 times the LLT *peerinact* time. To set the LLT *peerinact* time, refer to the VCS documentation.
  This setting is to avoid the VCS-in-guest cluster seeing nodes from restarting and rejoining as a result of VMware HA actions, before VCS can detect the failure, and to compensate for timer drifts between VCS-in-guest and VMware HA.
  For example, for 8 secs for LLT peerinact, *das.failuredetectiontime* = 80 secs.
  Since this setting adversely affects VMware HA failover time, the following alternative can be used.
  Add a delay of 10 * *peerinact* on LLT startup script of each VM.
  - Edit the LLT startup script in unix environment.
    Example:

```
/etc/init.d/llt on linux.
#
# start (load kernel module and configure) LLT.
# if either (or both) already done, do not do anything.
#
start()
{
sleep 60
# Check to see if this script is allowed to start llt
if ["$LLT_START"!=1]; then
exit 2
fi
```

  - Follow these steps in Windows environment
    - Step 1: Change the VCS services to "manual" start by running the following commands on command line:

```
sc config llt start= demand
sc config gab start= demand
sc config VCSComm start= demand
sc config had start= demand
```

- Step 2: Create a batch file. Put the following in a batch file named `startVCSwithDelay.bat` and store it in `cluster %VCS_HOME%\bin –`

```
@echo off
set delaytime=160
set LOGFILE=%VCS_HOME%\log\startVCSwithDelay.log
echo "-----------------------------------------" >
"%LOGFILE%"
date /T >> "%LOGFILE%"
hostname >> "%LOGFILE%"
time /T >> "%LOGFILE%
echo "Startup delay %delaytime% seconds" >> "%LOGFILE%"
ping localhost -n %delaytime% > nul
time /T >> "%LOGFILE%"
echo "Starting VCS services" >> "%LOGFILE%"
hastart >> "%LOGFILE%"
:wait_for_svcs
hasys -state >> "%LOGFILE%"
if %errorlevel% == 0 ( goto started )
echo "-------------------" >> "%LOGFILE%"
time /T >> "%LOGFILE%"
ping localhost -n 1 > nul
goto wait_for_svcs
:started
time /T >> "%LOGFILE%"
echo "Done" >> "%LOGFILE%"
```

- Step 3: Create a scheduled task with the previously created batch file. The task should trigger on system startup and it should run under “system” account.
  In Windows 2008: **All Programs -> Administrative Tools -> Task Scheduler - > Create Basic task**
  In Windows 2003: **All Programs -> Control Panel -> Scheduled Tasks- > the Scheduled Task**

- VM Monitoring must be turned off for all nodes in “VCS-in-guest” cluster.

Recommendations

1. Create anti-affinity rules between all nodes in a VCS-in-guest cluster, configured as a part of “VCS cluster across ESX Hosts”. This rule ensures that

in general two VMs belong to a single VCS-in-guest cluster do not end up on the same host. If multiple VMs of a single VCS-in-guest cluster run on the same host, failure of that host affects multiple VCS nodes.

2. Configure 'management networks' (service console networks) as the VCS-in-guest LLT private networks and also always configure two private networks. VMware HA sends its heartbeats on management networks, so if we use the same networks for VCS, the advantage is that user doesn't need to provide additional redundant networks for VCS-in-guest.
3. Configure non-SCSI3 fencing on VCS-in-guest Unix clusters. This protects against data corruption in case of network partitions.
4. For windows VCS-in-guest clusters and linux VCS-in-guest clusters without fencing, configure the host isolation response as power-off VM. This provides partial protection from potential data corruption in case of host isolation. However, to gain advantage of this setting, configure *das.failuredetectiontime* to be much less than half of the *peerinact* value, and configure the delay in LLT startup to 10 * *peerinact*.

   In this case when a host gets isolated, VMware turns off the VCS-in-guest VM that is isolated prior to the other VCS-in-guest VMs seeing this node as faulted.

   Example values: *das.failuredetectiontime* = 16 , *peerinact* = 32, LLT startup script delay = 320 seconds

---

Note: Timers in the guest and host may drift and this method is not fully guaranteed to provide protection against potential data corruption.

Availability levels

1. In certain situations , VMware HA with VCS-in-guest would yield higher availability than standalone VCS-in-guest.

   Example: Consider three ESX hosts H1-H3-H3 and two VMs N1-N2 in a VCS-in-guest cluster. N1 is running on H1 and N2 is running on H2. N1 is running the application. If H2 fails, then VMware HA restarts N2 on H3. Now if H1 fails, application would failover to N2. In absence of VMware-HA this would not be possible.
2. In certain situations , VMware HA + VCS-in-guest would yield lesser availability than standalone VCS-in-guest.

Example : Consider two ESX hosts H1-H2 and two VMs N1-N2 in a VCS-in-guest cluster. N1 is running on H1 and N2 is running on H2. N1 is running the app. If H2 fails, then VMware HA restarts N2 on H1. N1 and N2 end up on the same host.

Subsequently if H2 is back again and if H1 fails, both N1 and N2 fail and from the point of view of VCS-in-guest cluster, it would be as if the entire cluster rebooted.

Note: This would yield lesser availability than stand-alone VCS-in-guest where when H2 is back again and N2 would have started on H2, thus allowing the application to failover quickly to N2 when H1 fails.

The essence here is that ideally VMs of a VCS-in-guest cluster must distribute themselves evenly across hosts, otherwise there is a possibility of losing availability. It is difficult to enforce this through DRS because DRS triggers only periodically, which leaves a window where VMs may not be distributed evenly.