Veritas™ Cluster Server Agent for EMC MirrorView Configuration Guide

ESX

5.1 MP2

July 2008

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

Agent version: 5.1.MP2

Document Version: 5.1.MP2.0

Legal Notice

This Symantec product may contain third party software for which Symantec is required to provide attribution to the third party ("Third Party Programs"). Some of the Third Party Programs are available under open source or free software licenses. The License Agreement accompanying the Software does not alter any rights or obligations you may have under those open source or free software licenses. Please see the Third Party Legal Notice Appendix to this Documentation or TPIP ReadMe File accompanying this Symantec product for more information on the Third Party Programs.

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

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

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

Symantec Corporation
20330 Stevens Creek Blvd.
Cupertino, CA 95014

http://www.symantec.com

Printed in the United States of America.

10 9 8 7 6 5 4 3 2 1

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 maintenance 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 Web-based support that provides rapid response and up-to-the-minute information
- Upgrade assurance that delivers automatic software upgrade protection
- Global support that is available 24 hours a day, 7 days a week
- Advanced features, including Account Management Services

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

www.symantec.com/techsupp/

Contacting Technical Support

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

www.symantec.com/techsupp/

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/techsupp/

Customer service

Customer service information is available at the following URL:

www.symantec.com/techsupp/

Customer Service is available to assist with 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 maintenance 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

Maintenance agreement resources

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

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

Additional enterprise services

Symantec offers a comprehensive set of services that allow you to maximize your investment in Symantec products and to develop your knowledge, expertise, and global insight, which enable you to manage your business risks proactively.

Enterprise services that are available include the following:

Symantec Early Warning Solutions

These solutions provide early warning of cyber attacks, comprehensive threat analysis, and countermeasures to prevent attacks before they occur.

Managed Security Services

These services remove the burden of managing and monitoring security devices and events, ensuring rapid response to real threats.

Consulting Services

Symantec Consulting Services provide on-site technical expertise from Symantec and its trusted partners. Symantec Consulting Services offer a variety of prepackaged and customizable options that include assessment, design, implementation, monitoring, and management capabilities. Each is focused on establishing and maintaining the integrity and availability of your IT resources.

Educational Services

Educational Services provide a full array of technical training, security education, security certification, and awareness communication programs.

To access more information about Enterprise services, please visit our Web site at the following URL:

www.symantec.com

Select your country or language from the site index.

Introducing the VCS agent for EMC MirrorView

This chapter includes the following topics:

About the agent for EMC MirrorView

The Veritas agent for EMC MirrorView provides support for application failover and recovery. The agent provides this support in environments that use MirrorView to replicate data between CLARiiON arrays.

The agent monitors and manages the state of replicated CLARiiON arrays that are attached to VCS nodes. The agent ensures that the system that has the MirrorView resource online also has safe and exclusive access to the configured arrays.

You can use the agent in replicated data clusters and in global clusters that run VCS.

The agent supports configuring EMC MirrorView in the synchronous or asynchronous modes. In asynchronous mode, you can replicate either individual

LUNs or replicate consistency groups. MirrorView can also replicate LUNs or metaLUNs. In synchronous mode, you cannot replicate consistency groups.

What's new in this release

The Veritas Cluster Server agent for EMC MirrorViewincludes the following new or enhanced features:

- The Veritas agent for EMC Mirrorview supports replicated data clusters in this release.

Supported software and hardware

The EMC MirrorView agent supports Veritas Cluster Server 5.1 MP2 for ESX.

Note: The Veritas agent for EMC Mirrorview supports Navisphere Secure CLI, naviseccli. The agent no longer supports the Java based CLI, navcli.jar.

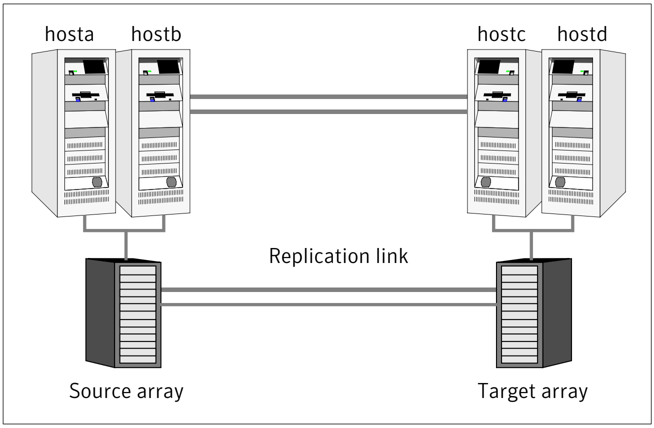
To determine the supported versions of NaviCLI and FLARE code that are on CLARiiON arrays, consult the EMC hardware compatibility list.

Note: VMotion (i.e. the live migration of the virtual machine from the source ESX host to the target) is not supported across replicated system zones.

Typical EMC MirrorView setup in a VCS cluster

Figure 1-1 displays a typical cluster setup in a MirrorView environment.

Figure 1-1 Typical clustering setup for the agent



Clustering in a MirrorView environment typically consists of the following hardware infrastructure:

- The source array (array1) has one or more hosts. The hosts have a direct connection to a CLARiiON array. The array contains the mirror that is the primary image and the direct connection uses either SCSI or Fibre Channel.
- The target array (array2) consists of one or more hosts. These hosts have a direct connection to another CLARiiON array. The array contains the mirror that is the secondary image and the connection uses either SCSI or Fibre Channel.

  The secondary image LUNs pairs with the mirrored LUNs in the source array. The target hosts and the array must be at a significant distance from the source side to survive a source-side disaster.
- Network heartbeating between the two data centers to determine their health; this network heartbeating could be LLT or TCP/IP.

  See "About cluster heartbeats" on page 21.
- In a replicated data cluster environment, all hosts are part of the same cluster. You must connect them with the dual and dedicated networks that support LLT.

  In a global cluster environment, you must attach all hosts in a cluster to the same CLARiiON array.

EMC MirrorView agent functions

The VCS agent for EMC MirrorView monitors and manages the state of replicated CLARiiON LUNs attached to VCS nodes. Agent functions bring resources online, take them offline, and perform different monitoring actions.

The agent performs the following functions:

online — Creates a lock file on the local host. This lock indicates that the resource is online and makes the mirrors available for the application to use. The agent performs specific actions depending on the state of the mirrors.

See "About the MirrorView agent's online function" on page 12.

offline — Removes the lock file on the local host.

monitor — Verifies that the lock file exists. If the lock file exists, the monitor entry point reports the status of the resource as online. If the lock file does not exist, the monitor entry point reports the status of the resource as offline.

open — Removes the lock file from the host where the function is called. This action prevents potential concurrency violation if the service group fails over to another node.

Note that the agent does not remove the lock file if the agent was started after a `hastop -force` command.

clean — Removes the lock file.

info — The info function gives the information about the mirrors (in case of synchronous mode of replication). It also gives information about the mirrors/groups in case of asynchronous mode of replication. It uses the `-sync listsyncprogress` and `-async -list` or `-async listgroups` commands to get this information.

resynch — Performs a resynchronization action.

About the MirrorView agent's online function

The agent's online operation performs specific actions depending on the state of the mirrors.

Synchronous state

If the state of all local mirrors is MIRRORED, the agent creates a lock file on the local host. This lock indicates that the resource is online and makes the mirrors available for the application to use.

If one or more mirrors are not in the MIRRORED state, the agent runs a NaviCLI command. With this command, the agent brings them into the MIRRORED state, which enables the application to use them.

- For secondary images in the synchronized state, the agent runs the `mirror -sync -promoteimage` command to promote the remote mirror. This command also converts the current primary to secondary.
- For secondary images in the CONSISTENT state, the agent waits to check if the image has transitioned to the SYNCHRONIZED state.
- If the images have transitioned to the SYNCHRONIZED state, the agent then runs the `mirror -sync -promoteimage` command to promote the remote mirror. This command also converts the current primary to secondary.
- If the image has not transitioned to the SYNCHRONIZED state, the agent checks if the remote array is accessible. If the remote array is accessible, then this condition indicates link failure—the image would be in a fractured condition.

In case of fracture:

- If the SplitTakeover attribute is set to 1, the agent forcibly promotes the secondary image.
- If the SplitTakeover attribute is set to 0, the agent does not try to promote the secondary image forcibly, and becomes the administrator’s decision.

Asynchronous state

You can configure the online function for either consistency groups or mirrors in asynchronous mode.

Consistency groups

If the state of the group is SYNCHRONIZED, the agent creates a lock file on the local host to indicate that the resource is online. This lock makes the LUNs available for the application to use.

If one or more mirrors are not in the MIRRORED state, the agent checks to see if the remote array is accessible.

- If the remote array is not accessible, then the agent checks the value of the attribute SplitTakeover before proceeding with any further actions.

- If the SplitTakeover attribute is set to 1, the agent forcibly promotes the secondary image.
- If the SplitTakeover attribute is set to 0, the agent does not try to promote the secondary image forcibly, and becomes the administrator's decision.
- If the remote array is accessible, then the agent runs the `mirror -async -promotegroup` command to promote the remote group.
- In case of a successful promotegroup operation, the operation also converts the current primary to secondary.
- If the promotegroup operation is not successful, then the agent initiates a synchronization.
  The agent periodically checks if the group is SYNCHRONIZED. After a successful synchronization, the agent promotes the group using the `mirror -async -promotegroup` command. If the synchronization is not successful, the agent times out.

Mirrors

If the state of all local mirrors is MIRRORED, the agent creates a lock file on the local host. The lock indicates that the resource is online and makes the mirrors available for the application to use.

If one or more mirrors are not in the MIRRORED state, the agent checks to see if the remote array is accessible.

- If the remote array is not accessible, then the agent checks the value of the attribute SplitTakeover before proceeding with any further actions.
- If the SplitTakeover attribute is set to 1, the agent forcibly promotes the secondary image.
- If the SplitTakeover attribute is set to 0, the agent does not try to promote the secondary image forcibly, and becomes the administrator's decision.
- If the remote array is accessible, then the agent runs the `mirror -async -promoteimage` command to promote the remote mirrors.
- A successful promoteimage operation converts the current primary to secondary.
  If the promoteimage operation is not successful, then the agent initiates a synchronization.
  The agent periodically checks if the group is SYNCHRONIZED. After a successful synchronization, the agent promotes the secondary mirror using the `mirror -async -promoteimage` command. If the synchronization is not successful, the agent times out.

Additional considerations in an ESX environment

The agent performs the following actions before coming online:

- Changes the following ESX server settings:
  LVM.DisallowSnapshotLun=0.
  LVM.EnableResignature=0.
  Refer to the VMware documentation for more information on these settings.
- Rescans all Host Bus Adapters (HBA).

Chapter 2

Configuring the agent for EMC MirrorView

This chapter includes the following topics:

Configuration concepts for the EMC MirrorView agent

Review the resource type definition and the attribute definitions for the agent.

Resource type definition for the MirrorView agent

The resource type definition defines the agent in VCS.

```
type MirrorView (
    static keylist SupportedActions = { resync }
    static int MonitorInterval = 300
    static int NumThreads = 1
    static int OfflineMonitorInterval = 0
    static int RestartLimit = 1
    static str ArgList[] = { NaviCliHome, LocalArraySPNames,
    RemoteArraySPNames, Mode, GrpName, MirNames, SplitTakeover }
    str NaviCliHome = "/opt/Navisphere/bin"
    str LocalArraySPNames[]
    str RemoteArraySPNames[]
    str Mode
    str GrpName
str MirNames[]
int SplitTakeover
temp str VCSResLock
)
```

Attribute definitions for the MirrorView agent

Review the description of the agent attributes.

Required attributes

You must assign values to the following required attributes:

NaviCliHome

NaviCLI installation directory

`"/opt/Navisphere/bin"`

Type and dimension: string-scalar

LocalArraySPNames

The list of storage processors within the array to which the local hosts are connected. Can be names or IP addresses.

Type and dimension: string-vector

RemoteArraySPNames

The list of storage processors within the array to which the remote hosts are connected. Can be names or IP addresses.

Type and dimension: string-vector

Mode

The replication mode, which is either: `sync` or `async`.

Type and dimension: string-scalar

GrpName

The name of the consistency group to which the mirrors belong. This function applies only if the mode is `async`.

Type and dimension: string-scalar

MirNames

This function specifies the mirrors with only one replication mode: `sync` or `async`.

Type and dimension: string-vector

SplitTakeover

This integer indicates whether VCS should forcefully promote a secondary to a primary.

In case of a link-failure between the two arrays, the state of the mirror remains consistent or out-of-sync. Under such circumstances, if the application has to failover—due to disaster or user-driven action—mirrors are not in a SYNCHRONIZED state.

If the value of the SplitTakeOver attribute is 1:

- The agent fails over when it discovers link failures
- The agent determines that mirrors are out of sync

If the value of the attribute is 0, agent does not fail over and the administrator must to determine what to do.

Type and dimension: integer-scalar

Internal attribute

Do not modify internal attributes. The MirrorView agent currently supports the following internal attribute:

VCSResLock

This agent uses this attribute to guarantee serialized management in case of a parallel application. Do not modify this value.

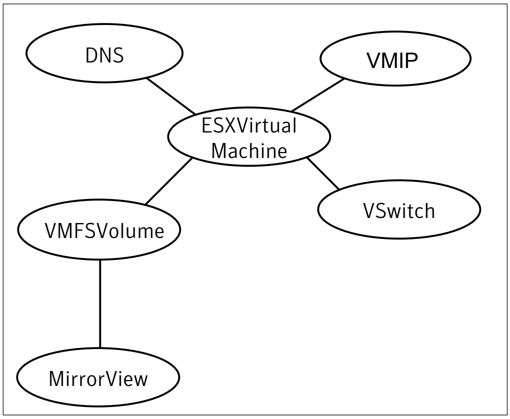
Type and dimension: temporary-string

Sample configuration for the MirrorView agent

Figure 2-1 shows a VCS service group that has a resource of type MirrorView

The VMFSVolumeresource depends on the MirrorView resource.

Figure 2-1 Dependency tree



You can configure a resource of type MirrorView in the `main.cf` file. In this example, the resource is configured for asynchronous mode and consistency groups.

```
MirrorView mir (
    NaviCliHome = "/opt/Navisphere/bin"
    LocalArraySPNames @sys1= = { "Local_SP1_Name", "Local_SP1_IP" }
    LocalArraySPNames @sys2 = { "Local_SP2_Name", "Local_SP2_IP" }
    RemoteArraySPNames @sys1 = { "Local_SP2_IP", "Remote_SP2_Name" }
    RemoteArraySPNames @sys2= { "Local_SP1_IP", "Remote_SP1_Name" }
    Mode = async
    GrpName = consistency_grp1
    SplitTakeover = 0
    )
```

If you want to configure the resource in synchronous mode and specify the individual mirror names, configure the MirNames attribute, instead of the GrpNames attribute, as follows:

```
    Mode = sync
    MirNames = { "sync_mir1", "sync_mir2" }
    GrpName = ""
```

If you want to configure the resource in asynchronous mode and specify the individual mirror names, configure the MirNames attribute, instead of the GrpNames attribute, as follows:

```
Mode = async
MirNames = { "async_mir1", "async_mir2" }
GrpName = ""
```

Before you configure the agent for MirrorView

Before you configure the agent, review the following information:

- Verify that the clustering infrastructure is in place.
  If you plan to configure the agent in a global cluster, make sure the global service group for the application is configured.
  For more information, see the *Veritas Cluster Server User's Guide*.
- Review the configuration concepts, which describe the agent's type definition and attributes.
  See "Configuration concepts for the EMC MirrorView agent" on page 17.
- Verify that you have installed the agent on all systems in the cluster.
- Verify the hardware setup for the agent.
  See "Typical EMC MirrorView setup in a VCS cluster" on page 10.
- Make sure that the cluster has an effective heartbeat mechanism in place.
  See "About cluster heartbeats" on page 21.
- Set up an effective heartbeat mechanism to prevent split-brain.
  See "About preventing split-brain" on page 23.
- Set up system zones in replicated data clusters.
  See "About configuring system zones in replicated data clusters" on page 22.

About cluster heartbeats

In a replicated data cluster, ensure robust heartbeating by using dual, dedicated networks over which the Low Latency Transport (LLT) runs. Additionally, you can configure a low-priority heartbeat across public networks.

In a global cluster, VCS sends ICMP pings over the public network between the two sites for network heartbeating. To minimize the risk of split-brain, VCS sends ICMP pings to highly available IP addresses. VCS global clusters also notify the administrators when the sites cannot communicate.

Heartbeat loss may occur due to the failure of all hosts in the primary cluster. In such a scenario, a failover may be required even if the array is alive. In any case, a host-only crash and a complete site failure must be distinguished. In a host-only crash, only the ICMP heartbeat signals a failure by an SNMP trap. No cluster failure notification occurs because a surviving heartbeat exists. This trap is the only notification to fail over an application.

About configuring system zones in replicated data clusters

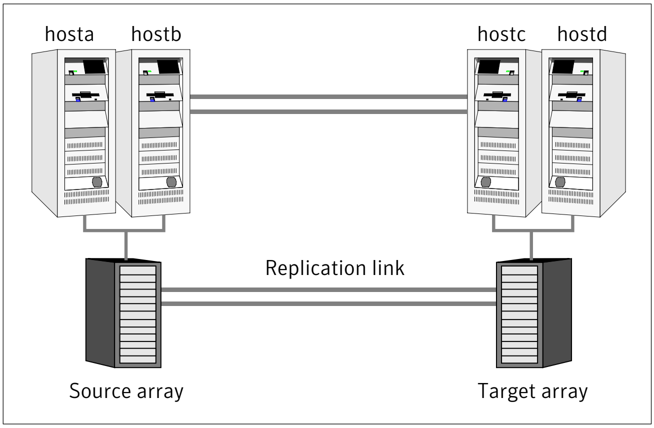
In a replicated data cluster, you can prevent unnecessary MirrorView failover or failback by creating system zones. VCS attempts to fail over applications within the same system zone before failing them over across system zones.

Configure the hosts that are attached to an array as part of the same system zone to avoid unnecessary failover.

Figure 2-2 depicts a sample configuration where hosta and hostb are in one system zone, and hostc and hostd are in another system zone.

Use the SystemZones attribute to create these zones.

Figure 2-2 Example system zone configuration



Global clusters do not require system zones because failover occurs on a remote cluster if all local targets have been exhausted.

As long as a secondary image is available, MirrorView sends the writes to the secondary image immediately in synchronous mode. It does so periodically in asynchronous mode.

If the period is too long, you can perform synchronization using the resync action entry point. The supported resync action is defined in the MirrorView resource type.

About preventing split-brain

Split-brain occurs when all heartbeat links between the primary and secondary hosts are cut. In this situation, each side mistakenly assumes that the other side is down. You can minimize the effects of split-brain by ensuring that the cluster heartbeat links pass through a similar physical infrastructure as the replication links. When you ensure that both pass through the same infrastructure, if one breaks, so does the other.

Sometimes you cannot place the heartbeats alongside the replication links. In this situation, a possibility exists that the cluster heartbeats are disabled, but the replication link is not. A failover transitions the original source to target and vice-versa. In this case, the application faults because its underlying volumes become write-disabled, causing the service group to fault. VCS tries to fail it over to another host, causing the same consequence in the reverse direction. This phenomenon continues until the group comes online on the final node. You can avoid this situation by setting up your infrastructure such that loss of heartbeat links also mean the loss of replication links.

Configuring the agent for MirrorView

You can adapt most clustered applications to a disaster recovery environment by:

- Converting their LUNs to CLARiiON LUNs
- Synchronizing the mirrors
- Adding the EMC MirrorView agent to the service group

After configuration, the application service group must follow the dependency diagram.

See “Sample configuration for the MirrorView agent” on page 19.

Configuring the agent manually in a global cluster

Configuring the agent manually in a global cluster involves the following tasks:

To configure the agent in a global cluster

1. Start Cluster Manager and log on to the cluster.
2. If the agent resource type (MirrorView) is not added to your configuration, add it. From the Cluster Explorer **File** menu, choose **Import Types** and select:

   /etc/VRTSvcs/conf/MirrorViewTypes.cf.
3. Click **Import**.
4. Save the configuration.

5 Add a resource of type MirrorView at the bottom of the service group.

6 Configure the attributes of the MirrorView resource.

7 If the service group is not configured as a global group, configure the service group using the Global Group Configuration Wizard.

See the *Veritas Cluster Server User's Guide* for more information.

8 Change the ClusterFailOverPolicy from the default, if necessary. Symantec recommends keeping the default, which is Manual, to minimize the chance of failing over on a split-brain.

9 Repeat step 5 through step 8 for each service group in each cluster that uses replicated data.

Configuring the agent manually in a replicated data cluster

Configuring the agent manually in a replicated data cluster involves the following tasks:

To configure the agent in a replicated data cluster

1 Start Cluster Manager and log on to the cluster.

2 If the agent resource type (MirrorView) is not added to your configuration, add it. From the Cluster Explorer **File** menu, choose **Import Types** and select:

```
/etc/VRTSvcs/conf/MirrorViewTypes.cf.
```

3 Click **Import**.

4 Save the configuration.

5 In each service group that uses replicated data, add a resource of type MirrorView at the top of the service group.

6 Configure the attributes of the MirrorView resource. Note that some attributes must be localized to reflect values for the hosts that are attached to different arrays.

7 Set the SystemZones attribute for the service group to reflect which hosts are attached to the same array.

3

Managing and testing clustering support for EMC MirrorView

This chapter includes the following topics:

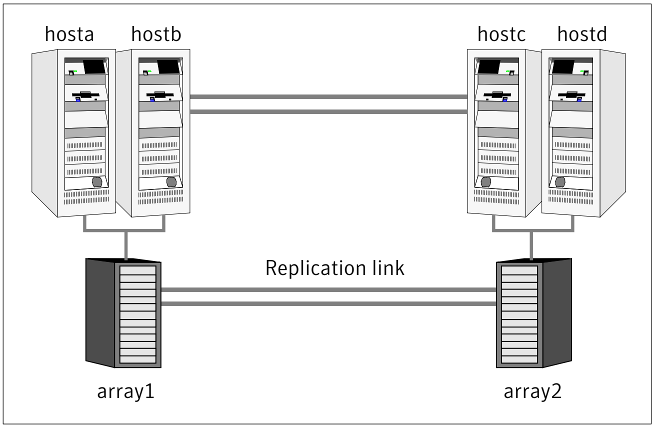
Typical test setup for the EMC MirrorView agent

A typical test environment includes the following characteristics:

- Two hosts (hosta and hostb) are attached to the source CLARiiONarray.
- Two hosts (hostc and hostd) are attached to the target CLARiiON array.
- The application runs on hosta and devices in the local array are read-write enabled in the SYNCHRONIZED state.
- A replicated data cluster has two dedicated heartbeat links.
  A global cluster has one network heartbeat.

Figure 3-1 depicts a typical test environment.

Figure 3-1 Typical test setup



Testing service group migration

Verify the service group can migrate to different hosts in the cluster and across clusters.

To perform the service group migration test

1. In the Service Groups tab of the Cluster Explorer configuration tree, right-click the service group.

   Migrate the service group to a host that is attached to the same array.

2. Click **Switch To**, and click the system that is attached to the same array (hostb) from the menu.

   The service group comes online on hostb and local image remains in the MIRRORED state.

3. In the Service Groups tab of the Cluster Explorer configuration tree, right-click the service group.

   Migrate the service group to a host that is attached to a different array.

4 Click **Switch To**, and click the system that is attached to the another array (hostc) from the menu.

The service group comes online on hostc and the role of the images there transition to primary.

Accumulate dirty tracks on the new source-side and update them back on the target:

```
hares -action mirrorview_res_name resync -sys hostc
```

The variable *mirrorview_res_name* represents the name of the MirrorView resource.

5 In the Service Groups tab of the Cluster Explorer configuration tree, right-click the service group.

After the devices transition to a source SYNCHRONIZED state, migrate the service group back to its original host.

6 Click **Switch To**, and click the system on which the group was initially online (hosta).

The group comes online on hosta. The devices return to the RW/SYNCINPROG state at the array that is attached to hosta and hostb, and then eventually transition to the SYNCHRONIZED state.

Testing host failure

In this scenario, the host where the application runs is lost. Eventually all the hosts in the system zone or cluster are lost.

To perform the host failure test

1 Halt or shut down the host where the application runs (hosta).

The service group fails over to hostb and devices are in the SYNCHRONIZING state.

2 Halt or shut down hostb.

In a replicated data cluster, the group fails over to hostc or hostd depending on the FailOverPolicy in the cluster.

In a global cluster, a cluster down alert appears and gives you the opportunity to fail over the service group manually.

In both environments, the role of the devices changes from secondary to primary and starts on the target host.

3 Reboot the two hosts that were shut down.

4 Switch the service group to its original host when VCS starts.

Do the following:

- In the **Service Groups** tab of the Cluster Explorer configuration tree, right-click the service group.
- Click **Switch To**, and click the system on which the service group was initially online (hosta).
  The service group comes online on hosta and devices transition to the SYNCHRONIZING state and then to the SYNCHRONIZED state.

Performing a disaster test

Test how robust your cluster is in case of a disaster.

To perform a disaster test

1 Shut down all hosts on the source side and shut down the source array.

If you can not shut down the source array, change the value of the RemoteArraySPNames in the target side to non-existent names and IP addresses. This action mimics a disaster scenario from the target's point of view.

2 In a replicated data cluster, the service group fails over to hostc or hostd in the following conditions:

- All devices were originally in the SYNCHRONIZED state.
- No synchronization was in progress at the time of disaster.

3 In a global cluster, the administrator is notified of the failure. The administrator can then initiate the failover.

Performing the failback test

You can set up your cluster for a failback test.

The failback test verifies the application can fail back to its original host after a failover to a remote site.

To perform the failback test for asynchronous mode with Consistency groups

1 Remove all the mirrors form the consistency group on the old primary.

2 Destroy the consistency group on the old primary.

3 Forcefully destroy the remote mirrors on the old primary.

4 Remove the LUNs from the storage group on the old primary.

5 Remove the mirrors from the consistency group on the new primary.

6 Add secondary images to each of the remote mirrors on the new primary.

7 Add the mirrors into the consistency group on the new primary.

Between step 5 and step 7, the LUNs become vulnerable to data corruption. For example, if one of the LUNs has sustained hardware damage and failed.

During this window, the mirrors are not a part of the consistency group. The writes to other mirrors that were a part of the consistency group are not stopped. This situation could result in data corruption.

8 Add the LUNs, where the secondary image resides, into the appropriate storage group on the old primary.

To perform the failback test for synchronous and asynchronous mode with Individual mirrors

1 Forcefully destroy the remote mirrors on the old primary.

2 Remove the LUNs from the storage group on the old primary.

3 Add secondary images to each of the remote mirrors on the new primary.

4 Add the LUNs, where the secondary image resides, into the appropriate storage group on the old primary.

In either of the modes, the original contents of the old primary are lost.

Failure scenarios for EMC MirrorView

Review the failure scenarios and agent behavior in response to failure.

Site disaster

In a total site failure, all hosts and the array are completely disabled, either temporarily or permanently.

In a global cluster, VCS detects site failure by the loss of all configured heartbeats.

A total disaster renders the devices on the surviving array in the FRACTURED state. If the SplitTakeover attribute is set to its default value of 1, the online entry point runs the ‘promote’ operation. If the attribute is set to 0, no takeover occurs and the online entry point times out and faults.

The online entry point detects whether any synchronization was in progress when the source array was lost. Since the target devices are inconsistent until the synchronization completes, the agent does not write-enable the devices, but it times out and faults. You must restore consistent data from a snapshot or tape backup.

All host or all application failure

Even if both arrays are operational, the service group fails over in the following conditions:

- All hosts on the source CLARiiON side are disabled.
- The application cannot start successfully on any source host.

In replicated data cluster environments, the failover can be automatic, whereas in global cluster environments failover requires user confirmation by default.

In both replicated data cluster and global cluster environments, multiple service groups can fail over in parallel.

Replication link failure

Before the MirrorView takes any action, it waits for the synchronization to complete in the following situations:

- The two arrays are healthy and the link that failed is restored.
- A failover is initiated while synchronization is in progress.

After the synchronization completes, the MirrorView runs the promote operation.

If the agent times out before the synchronization completes, the resource faults.

If the SplitTakeover attribute is set to 0, the agent does not attempt a promote operation, but it times out and faults. If you write-enable the devices manually, the agent can come online after it is cleared.

Rescanning Host Bus Adapters (HBAs) on VCS nodes

While executing rescan HBA operations on VCS nodes, use the following guidelines to make sure that the LVM settings conform to VCS requirements:

- Note the current values of the following LVM settings:
  DisallowSnapshotLun
  EnableResignature
- While rescanning snapshot LUNs for new datastores, set EnableResignature=1.
- If rescanning with DisallowSnapshotLun=0 and EnableResignature=0, make sure that snapshot LUNs that do not contain already resignatured datastores are not presented to the server.

4

Setting up a fire drill

This chapter includes the following topics:

About fire drills

A fire drill procedure verifies the fault-readiness of a disaster recovery configuration. This procedure is done without stopping the application at the primary site and disrupting user access.

A fire drill is performed at the secondary site using a special service group for fire drills. The fire drill service group is identical to the application service group, but uses a fire drill resource in place of the replication agent resource. The fire drill service group uses a copy of the data that is used by the application service group.

In clusters employing EMC MirrorView, the MirrorViewSnap resource manages the replication relationship during a fire drill.

Bringing the fire drill service group online demonstrates the ability of the application service group to come online at the remote site when a failover occurs.

The MirrorViewSnap agent supports fire drills for storage devices that are managed using Veritas Volume Manager 5.0 MP1 RP2, which is a component of Veritas Storage Foundation.

Note: The agent does not support fire drills for asynchronous configurations.

Note: In an ESX environment, you can also perform a fire drill locally to generate a last known good copy of your application data. If the replicated data produced by the fire drill tests successfully, it is the last known good copy.

Considerations for using MirrorView and SnapView together

MirrorView is an EMC software application that maintains a copy or image of a logical unit (LUN) at a separate location. This copy or image is a provision for disaster recovery. You can use this image in the event that a disaster disables the production image. The production image is called the primary image; the copy image is called the secondary image.

SnapView is a storage-system-based software application that enables you to create a copy of a LUN by using either clones or snapshots. A clone is an actual copy of a LUN and takes time to create, depending on the size of the source LUN. A clone is a full copy. A snapshot is a virtual point-in-time copy of a LUN and takes only seconds to create. A snapshot uses a copy-on-write principle.

Fire drills use SnapView with MirrorView. The VCS MirrorView fire drill agent, MirrorViewSnap, does not support clones because of the following considerations:

- If a LUN is a MirrorView primary or secondary image, you cannot create a clone group for that image. If a LUN is a member of a clone group as the source or clone, it cannot serve as a MirrorView primary or secondary image.
- If the MirrorView Synchronous option is installed, you can create a snapshot of the primary or secondary image. However, Symantec recommends that you take a snapshot of a secondary image only if the state of the image is either SYNCHRONIZED or CONSISTENT. If the image is in the SYNCHRONIZING state or in the OUT-OF-SYNC state, the snapshot data is not useful.
- If the MirrorView Asynchronous option is installed, you can create a snapshot of the primary or secondary image. However, Symantec recommends that you take a snapshot of a secondary image only if the last update started has

completed successfully. If the update did not complete successfully because the image is fractured or the update is still in progress, the snapshot data is not useful.

The VCS MirrorView fire drill agent, MirrorViewSnap, does not support clones because of these considerations.

About the MirrorViewSnap agent

The MirrorViewSnap agent uses SnapView to take a snapshot of a source LUN and then makes the snapshot available to the secondary server. The MirrorViewSnap resource is configured in the fire drill service group.

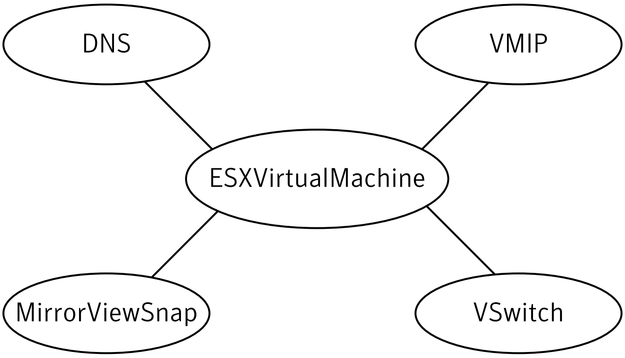
Agent fire drill functionality

The MirrorViewSnap agent makes a local copy of the LUNs in the mirrors that are specified as part of the fire drill configuration. The agent resides at the bottom of the dependency tree because a snapshot of data must be made available before the application can use it.

Note: The MirrorViewSnap agent creates SnapView snapshots, not SnapView clones.

Refer to Figure 4-1 for the MirrorViewSnap agent dependency tree.

Figure 4-1 MVSnap agent dependency tree



A snapshot is a virtual LUN. When a secondary server views a snapshot, it is viewing a point-in-time copy of a source LUN. You determine the point in time when you start a SnapView session. The session keeps track of the source LUN's data at a particular point in time. During a session, the production server can still write to the source LUN and modify data.

Agent prerequisites

Before initiating a fire drill, you must complete the initial setup for the MirrorViewSnap agent. After the initial setup is completed, VCS can:

- Take a snapshot
- Make the snapshot read-writable
- Start the application, which is already running on the production cluster, to verify the production data

To set up the MirrorViewSnap agent

1. Establish the MirrorView relationship.
2. Reserve sufficient unallocated LUNs in the reserved LUN pool.
3. Install and enable the SnapView licence.
4. Install the Navisphere CLI.
5. Before running a fire drill on a cluster at the secondary site , ensure that no VCS global service groups are online on the secondary site. Do this by ensuring that any virtual machines that are configured for disaster recovery through VCS are not in the powered-on state.

MirrorViewSnap agent functions

TheMirrorViewSnap agent performs the following functions:

online

Destroys any existing snapshot, takes a new snapshot of the LUNs in the mirror/consistency group, and then make it available for use.

The online function performs the following steps:

- Destroys any existing snapshot. (If a snapshot already exists, it was taken as a part of a previous online function for the current MirrorViewSnap type of resource.) If the agent is performing an online function for the first time or if an existing snapshot was manually destroyed, you receive error messages in the log file that pertain to the inability to destroy the snapshot. You may safely ignore these messages.
- Retrieves the state of the mirror/consistency group. If the mirror/consistency group is not in either SYNCHRONIZED/CONSISTENT state, the fire drill cannot continue. The agent logs an error and exits.
- Creates the MirrorViewSnap object.
- Takes the snapshot. In case of any error, the agent rolls back the changes for all the other mirrors.
- Creates a lock file and exits.

offline

Removes the lock file created by the online function.

monitor

Verifies the existence of the lock file to make sure the resource is online.

If the lock file exists, Monitor reports the state of the MirrorViewSnap resource as ONLINE. Otherwise, it reports the state as OFFLINE.

clean

Removes the lock file created by the online function.

open

Checks for the existence of the lock file.

If the lock file exists, open waits until at least one of its parent resources is probed:

- If any resources that depend on the MirrorViewSnap are online, the agent calls open in response to being restarted after it was killed or after HAD was killed. In either case, the agent does not remove the lock file.
- If any resources that depend on the MirrorViewSnap are offline, the agent is being restarted in response to HAD being started. Then, open removes the lock file and the agent reports OFFLINE.

Processing the snapshot

Processing the snapshot requires the following actions;

- Detects the new datastore (created by the snapshot) and identifies the new virtual machine

- Registers the new virtual machine with the ESX host using the new snapshot datastore that was created.
- Populates the CfgFile attribute of the ESXVirtualMachine resource (parent resource) in the fire drill service group. The display name of the new virtual machine is the value of the vmname attribute, suffixed by _snapshot. The suffix clearly identifies the virtual machine as the snapshot virtual machine.
- Generates new MAC addresses for each vmnic and patches the configuration files of the new virtual machine with the new MAC addresses.
- Disconnects the new virtual machine from the public switches by setting the ethernet[n].networkName attribute in the vmx file to None. This action enables you to configure IP address and other parameters in the guest OS of the virtual machine after it boots up.

Resource type definition for the MirrorViewSnap agent

Following are the resource type definitions for the MirrorViewSnapSnap agent:

```
type MirrorViewSnap (
    static keylist SupportedActions = { resync }
    static int MonitorInterval = 600
    static int NumThreads = 1
    static int OnlineTimeout = 600
    static int MonitorTimeout = 600
    static int ActionTimeout = 300
    static int RestartLimit = 1
    static str ArgList[] = { StorageGrpName, TargetResName,
    NaviCliHome, LocalArraySPNames }
    str StorageGrpName
    str TargetResName
    str NaviCliHome = "/opt/Navisphere/bin"
    str LocalArraySPNames[]
    temp str Responsibility
)
```

MirrorViewSnap agent attributes

Configure the following attributes to customize the behavior of the MirrorViewSnap agent:

TargetResName

Name of the resource managing the LUNs to be snapshot.

Set this attribute to the name of the MirrorView resource if the data being snapshot is replicated.

- Set the attribute to the name of the ESXVirtualMachine resource if the data is not replicated.

Type-dimension: string-scalar

StorageGrpName

Name of the storage group that contains the snapshot. The host that runs the fire drill must be a part of this storage group. Otherwise, the fire drill fails.

Type-dimension: string-scalar

Responsibility

Do not modify. For internal use only.

Used by the agent to keep track of resynchonizing snapshots.

Type-Dimension: temporary string

Sample configuration for a fire drill service group

The sample configuration of a fire drill service group is identical to an application service group with a hardware replication resource. However, in a fire drill service group, the MirrorViewSnap resource replaces the MirrorView resource as shown in the main.cf file.

The fire drill groups for running the fire drill on both a secondary site (normal fire drill) and locally (to generate a last known good copy) are denoted by the following:

```
group test_mirrorview_fd
```

```
group test_mirrorview_fd_local
```

The vmname attribute is not a mandatory attribute for the ESXVirtualMachine resource. However, to work correctly, both a local fire drill and a remote fire drill require that you configure the vmname attribute. The sample main.cf file provides an example.

Note: In the MirrorViewSnap resource type definition, ensure that the MonitorInterval attribute is set to 600 and that the ActionTimeout attribute is set to 300.

See “Resource type definition for the MirrorViewSnap agent” on page 36..

You can configure a resource of type MirrorViewSnap in the main.cf file as follows.

```
//MirrorView firedrill service group
group test_mirrorview_fd(
SystemList = { dr_sys0 = 0, dr_sys1 = 1 }
)
ESXVirtualMachine vm_res_fd (
Critical = 0
vmname = vm_fd_test
CfgFile = " "
)
MirrorViewSnap mvs_snapres_fd (
Critical = 0
StorageGrpName = fd_storage_grp_name
TargetResName = MirrorView_vm_res_fd
LocalArraySPNames = { "dr_array1.veritas.com",
"dr_array2.veritas.com" }
)
vm_res_fd requires mvs_snapres_fd
//Global MirrorView service group
group MirrorView_SG (
SystemList = { dr_sys0 = 0, dr_sys1 = 1 }ClusterList = { cluster_dr = 0, cluste
)
MirrorView MirrorView_vm_res_fd (
LocalArraySPNames = { "dr_array.veritas.com" }
RemoteArraySPNames = { "prim_array.veritas.com" }
Mode = sync
MirNames = { fd_replication_MirName}
)
//Local MirrorView firedrill service group
group test_mirrorview_fd_local (
SystemList = { dr_local_sys0 = 0, dr_local_sys1 = 1 }
)
ESXVirtualMachine vm_res_fd_local (
Enabled = 0
vmname = local_vm_fd
CfgFile = " "
)
MirrorViewSnap mvsnp_res_fd_local (
Critical = 0
StorageGrpName = fd_storage_grp_name
TargetResName = vm_res_fd
LocalArraySPNames = { "dr_local_array1.veritas.com" }
)
vm_res_fd_local requires mvsnp_res_fd_local
```

Before you configure the fire drill service group

Before you configure the fire drill service group, follow the steps below:

- Make sure the application service group is configured with a MirrorView resource.
- Make sure the infrastructure to take snapshots is properly configured between the source and target arrays.
- Make sure the MirrorView relationship is established.
- Reserve sufficient unallocated LUNs in the reserved LUN pool.
- Install and enable the SnapView license.
- Install the Navisphere CLI

Additional requirements for running a fire drill in an ESX environment

Follow these guidelines for fire drills in an ESX environment:

- Ensure that each service group contains only one virtual machine resource to be able to support fire drill.
- Symantec recommends dedicating one node in the cluster for fire drills.
- Configure fire drill service groups on this node. Do not configure replication resources on this node.
- Configure your array such that snapshot LUNs (i.e. target LUNs on which hardware snapshots are taken) are visible to this node only; the other nodes must not see snapshot LUNs. The restriction occurs because other nodes in the cluster may have LVM settings LVM.EnableResignature = 0 and LVM.DisAllowSnapshotLUN = 0 set by the replication agents, in which case VMWare recommends that no snapshot LUNs should be exposed to such ESX hosts.
- Install VMware tools to all virtual machines in the fire drill configuration. Otherwise, the missed virtual machine heartbeats cause the virtual machine resource to fault.
- Ensure that no virtual machines are configured with raw device mapping.

- Ensure that the resources configured within the fire drill service group are not marked as critical. Not marking the fire drill service group resources as critical prevents service group failover.

Configuring the fire drill service group

This section describes how to use Cluster Manager (Java Console) to create the fire drill service group.

To create the fire drill service group

1. Open the Veritas Cluster Manager (Java Console).
2. Log on to the cluster and click **OK**.
3. Click the Service Group tab in the left pane and click the **Resources** tab in the right pane.
4. Right-click the cluster in the left pane and click **Add Service Group**.
5. In the Add Service Group dialog box, provide information about the new service group
6. In Service Group name, enter a name for the fire drill service group
7. From the Available Systems box, select the node that is dedicated for fire drills.
8. Click the fire drill service group in the left pane and click the Resources tab in the right pane.
9. Add a resource of type MirrorViewSnap and configure its attributes.
10. Add a resource of type ESXVirtualMachine and configure its attributes.

    See "Sample configuration for a fire drill service group" on page 37..
11. Link resources such that the ESXVirtualMachine resource depends on the MirrorViewSnap resource.

Configuring the new virtual machine for the fire drill

To configure the new virtual machine for the fire drill

1. Bring the fire drill service group online on the node dedicated for fire drills.

   Reboot the new virtual machine. This virtual machine now comes up without any connections to any public network. You may see some errors and the application may report a faulted status; ignore these errors.

2. Assign a unique IP address and host name to the new virtual machine.

3 Take the fire drill service group offline.

4 If you want to connect the new virtual machine to the internet, connect the virtual machine to at least one public switch while it is shut down.

Running the fire drill

Before running a fire drill on the secondary site, ensure that it does not have any global service groups that have failed or switched over to the secondary site in the online state.

Additionally, ensure that you do not run a local fire drill on the secondary site, even if the secondary site has become the new primary site after a failover to the remote cluster. These restrictions occur because of ESX server behavior during datastore rescans with LVM settings: EnableResignature = 1; DisAllowSnapshotLun = 0; which may cause datastores to be resignatured. Refer to the *Release Notes* for more information.

You are now ready to run the fire drill.

To run the fire drill

1 Bring the fire drill service group online. The configured applications come up using snapshot data.

2 Ensure the validity of your application by using it to perform tasks that are appropriate for the application.

Being able to use the application indicates a successful fire drill and ensures that your replicated data is valid. The snapshot is stored in the current state until you run the next fire drill.

3 Take the fire drill service group offline.

Failing to take the fire drill service group offline could cause failures in your environment. For example, if the application service group fails over to the node hosting the fire drill service group, there would be resource conflicts, resulting in both service groups faulting.

Verifying a successful fire drill

Run the fire drill routine periodically to verify the application service group can fail over to the remote node.

To verify a successful fire drill

1. Bring the fire drill service group online on a node that does not have the application running. Verify that the fire drill service group comes online.

   This action validates your disaster recovery configuration. The production service group can fail over to the secondary site in the event of an actual failure (disaster) at the primary site.

2. If the fire drill service group does not come online, review the VCS engine log for more information.

3. Take the fire drill offline after its functioning has been validated.

   Failing to take the fire drill offline could cause failures in your environment. For example, if the application service group fails over to the node hosting the fire drill service group, there would be resource conflicts, resulting in both service groups faulting.