IBM PowerHA SystemMirror for AIX

Standard Edition

Version 7.2

Troubleshooting PowerHA SystemMirror

IBM

Note

Before using this information and the product it supports, read the information in "Notices" on page 89.

This edition applies to IBM® PowerHA® SystemMirror® 7.2 Standard Edition for AIX® and to all subsequent releases and modifications until otherwise indicated in new editions.

Contents

About this document

This document introduces troubleshooting the PowerHA SystemMirror for AIX software. This information is also available on the documentation CD that is shipped with the operating system.

Highlighting

The following highlighting conventions are used in this document:

Bold	Identifies commands, subroutines, keywords, files, structures, directories, and other items whose names are predefined by the system. Also identifies graphical objects such as buttons, labels, and icons that the user selects.
Italics	Identifies parameters whose actual names or values are to be supplied by the user.
`Monospace`	Identifies examples of specific data values, examples of text similar to what you might see displayed, examples of portions of program code similar to what you might write as a programmer, messages from the system, or information you should actually type.

Case-sensitivity in AIX

Everything in the AIX operating system is case-sensitive, which means that it distinguishes between uppercase and lowercase letters. For example, you can use the **ls** command to list files. If you type LS, the system responds that the command `is not found`. Likewise, **FILEA**, **FiLea**, and **filea** are three distinct file names, even if they reside in the same directory. To avoid causing undesirable actions to be performed, always ensure that you use the correct case.

ISO 9000

ISO 9000 registered quality systems were used in the development and manufacturing of this product.

Related information

- The PowerHA SystemMirror Version 7.2 for AIX PDF documents are available in the PowerHA SystemMirror 7.2 PDFs topic.
- The PowerHA SystemMirror Version 7.2 for AIX release notes are available in the PowerHA SystemMirror 7.2 release notes topic.

Troubleshooting PowerHA SystemMirror

Use this information to troubleshoot the PowerHA SystemMirror software for the AIX operating system.
Related information
Administering PowerHA SystemMirror
Planning PowerHA SystemMirror
Installing PowerHA SystemMirror

What's new in Troubleshooting PowerHA SystemMirror

Read about new or significantly changed information for the Troubleshooting PowerHA SystemMirror topic collection.

How to see what's new or changed

To help you see where technical changes have been made, the information center uses:

- The $>\!|$ image to mark where new or changed information begins.
- The $|\!<$ image to mark where new or changed information ends.

December 2023

Updated the "Miscellaneous issues" on page 79 topic that describes solution for Getting arithmetic syntax error with locale setting.

June 2022

Updated the "Miscellaneous issues" on page 79 topic that describes solution for potential non-categorized PowerHA SystemMirror issue.

December 2021

Added a new "PowerHA SystemMirror LV encryption issue" on page 78 topic that describes solution for potential PowerHA SystemMirror logical volume (LV) encryption issue.

December 2018

Added information about cluster events that an administrator can initiate in the "Events initiated by the administrator" on page 16.

January 2018

Added information about Automatic Repository Disk Replacement (ARR) in the ha_trgd_repository_disk.dita topic.

Troubleshooting PowerHA SystemMirror clusters

The following sections present the recommended troubleshooting strategy for a PowerHA SystemMirror cluster. It describes the problem determination tools available from the PowerHA SystemMirror main SMIT menu. This guide also includes information on tuning the cluster for best performance, which can help you avoid some common problems.

Typically, a functioning PowerHA SystemMirror cluster requires minimal intervention. If a problem does occur, diagnostic and recovery skills are essential. Therefore, troubleshooting requires that you identify

the problem quickly and apply your understanding of the PowerHA SystemMirror software to restore the cluster to full operation.

In general, troubleshooting a PowerHA SystemMirror cluster involves:

- Becoming aware that a problem exists
- Determining the source of the problem
- Correcting the problem.

Note: These topics present the default locations of log files. If you redirected any logs, check the appropriate location.

Related concepts

Using cluster log files

These topics explain how to use the PowerHA SystemMirror cluster log files to troubleshoot the cluster. Included also are some sections on managing parameters for some of the logs.

Related reference

System components

These topics guide you through the steps to investigate system components, identify problems that you may encounter as you use PowerHA SystemMirror, and offer possible solutions.

Related information

Solving common problems

Becoming aware of the problem

When a problem occurs within a PowerHA SystemMirror cluster, you will be made aware of it through either an event notification alert, or through monitoring the errpt or hacmp.out files.

There are other ways you can be notified of a cluster problem, through mail notification, or pager notification and text messaging:

- *Mail Notification*. Although PowerHA SystemMirror standard components do not send mail to the system administrator when a problem occurs, you can create a mail notification method as a pre- or post-event to run before or after an event script executes. In a PowerHA SystemMirror cluster environment, mail notification is effective and highly recommended.

- *Remote Notification*. You can also define a notification method - numeric or alphanumeric page, or an text messaging notification to any address including a cell phone - through the SMIT interface to issue a customized response to a cluster event.

  - *Pager Notification*. You can send messages to a pager number on a given event. You can send textual information to pagers that support text display (alphanumeric page), and numerical messages to pagers that only display numbers.

  - *Text Messaging*. You can send cell phone text messages using a standard data modem and telephone land line through the standard Telocator Alphanumeric Protocol (TAP). Your provider must support this service.

    You can also issue a text message using a Falcom-compatible GSM modem to transmit SMS (Short Message Service) text-message notifications wirelessly. SMS messaging requires an account with an SMS service provider. GSM modems take TAP modem protocol as input through a RS232 line or USB line, and send the message wirelessly to the providers' cell phone tower. The provider forwards the message to the addressed cell phone. Each provider has a Short Message Service Center (SMSC).

For each person, define remote notification methods that contain all the events and nodes so you can switch the notification methods as a unit when responders change.

Note: Manually distribute each message file to each node. PowerHA SystemMirror does not automatically distribute the file to other nodes during synchronization unless the File Collections utility is set up specifically to do so.

Messages displayed on system console

The PowerHA SystemMirror system generates descriptive messages when the scripts it executes (in response to cluster events) start, stop, or encounter error conditions. In addition, the daemons that make up a PowerHA SystemMirror cluster generate messages when they start, stop, encounter error conditions, or change state. The PowerHA SystemMirror system writes these messages to the system console and to one or more cluster log files. Errors may also be logged to associated system files, such as the **errpt** file.

Related concepts

Using cluster log files
These topics explain how to use the PowerHA SystemMirror cluster log files to troubleshoot the cluster. Included also are some sections on managing parameters for some of the logs.

Related information

Planning PowerHA SystemMirror
Verifying and synchronizing a PowerHA SystemMirror cluster

Determining a problem source

Once you have determined that there is a problem, you need to find the source of the problem.

If a problem with PowerHA SystemMirror has been detected, perform the following actions for initial problem analysis:

1. Collect a PowerHA SystemMirror snapshot with the **snap -e** command. This should be done as soon as possible after the problem has been detected because the collected log files contain a time window of error.
2. Establish the state of the cluster and resource groups using the /usr/es/sbin/cluster/clstat, and /usr/es/sbin/cluster/utilities/clRGinfo commands.
3. If an event error occurred, inspect the /var/hacmp/log/hacmp.out file to locate the error. If an AIX command failed, proactively collect further debug data for the corresponding AIX component, using the snap command. The most commonly requested flag for further problem determination for PowerHA SystemMirror is **snap -egGtL**.
4. Consult the /var/hacmp/log/clverify.log, and /var/hacmp/log/autoverify.log files for the result of the most recent cluster verification. Run cluster verification.
5. If a C-SPOC command failed, consult the /var/hacmp/log/cspoc.log.long file.
6. Verify network connectivity between nodes.
7. Inspect the error log (errpt -a) to establish if errors have been logged in the time window of failure.

Stopping the cluster manager

To fix some cluster problems, you must stop the Cluster Manager on the failed node and have a surviving node take over its shared resources.

You can also stop the cluster manager process after stopping cluster services with the "*unmanage resource groups*" option. This option leaves the resources active but not monitored on the node. You can then begin the troubleshooting procedure.

If all else fails, stop the PowerHA SystemMirror cluster services on all cluster nodes. Then, manually start the application that the PowerHA SystemMirror cluster event scripts were attempting to start and run the application without the PowerHA SystemMirror software. This may require varying on volume groups, mounting file systems, and enabling IP addresses. With the PowerHA SystemMirror cluster services stopped on all cluster nodes, correct the conditions that caused the initial problem.

Using the AIX data collection utility

Use the AIX **snap** command to collect data from a PowerHA SystemMirror cluster.

The **-e** flag collects data that aids IBM support in troubleshooting a problem with PowerHA SystemMirror and its interaction with other components. In particular, the **-e** flag collects all log files of PowerHA SystemMirror utilities, ODMs maintained by PowerHA SystemMirror, some AIX ODMs, and AIX configuration data most commonly required (such as LVM, TCP/IP and installp information). The **snap -e** command runs /usr/sbin/rsct/bin/ctsnap, which collects data of the Group Services.

The PowerHA SystemMirror snapshot should be collected as soon as possible after a problem has been encountered with PowerHA SystemMirror, to ensure that the data pertaining to the time window of error are contained in the log files.

The **snap -e** command relies on the Cluster Communication Daemon subsystem (clcomd), to collect data. If this subsystem is affected by an error, the **snap -e** command might fail. In this case, collect the following data on all cluster nodes:

- tar archive of directory /var/hacmp
- tar archives of directories /etc/es/objrepos and /usr/es/sbin/cluster/etc/objrepos/active
- snap -cfgGLt

Using PowerHA SystemMirror diagnostic utilities

Both PowerHA SystemMirror and AIX supply many diagnostic tools.

The key PowerHA SystemMirror diagnostic tools (in addition to the cluster logs and messages) include:

- **clRGinfo** provides information about resource groups and for troubleshooting purposes.
- **clstat** reports the status of key cluster components - the cluster itself, the nodes in the cluster, the network interfaces connected to the nodes, the service labels, and the resource groups on each node.
- **cldisp** utility displays resource groups and their startup, fallover, and fallback policies.
- SMIT **Problem Determination Tools**, for information see the section Problem determination tools.

Using the cluster snapshot utility to check cluster configuration

You can still specify in SMIT that the logs be collected if you want them. Skipping the logs collection reduces the size of the snapshot and reduces the running time of the snapshot utility.

Working with SMIT Problem Determination Tools

The SMIT **Problem Determination Tools** menu includes the options offered by cluster snapshot utility, to help you diagnose and solve problems.

Related concepts
Problem determination tools
You can use the SMIT interface to help you troubleshoot problems with PowerHA SystemMirror.

Related information
Monitoring a PowerHA SystemMirror cluster
Saving and restoring cluster configurations

Verifying expected behavior

When the highly available applications are up and running, verify that users can access the applications.

If the applications are not up and running, you might need to look elsewhere to identify problems affecting your cluster. This document describe ways in which you should be able to locate potential problems.

Problem determination tools

You can use the SMIT interface to help you troubleshoot problems with PowerHA SystemMirror.

You can use the following tools to troubleshoot PowerHA SystemMirror. To access the following tools, enter `smit sysmirror` from the command line and select **Problem Determination Tools**.

PowerHA SystemMirror Verification
You can use this tool to verify that the configuration on all nodes is synchronized, set up a custom verification method, or set up automatic cluster verification.

View Current State
You can use this tool to display the state of the nodes, communication interfaces, resource groups, and the local event summary for the last five events.

PowerHA SystemMirror Log Viewing and Management
You can use this tool to view a list of utilities related to the log files.

Recover from PowerHA SystemMirror Script Failure
You can use this tool to recover from a script failure.

Restore PowerHA SystemMirror Configuration Database from Active Configuration
You can use this tool to automatically save any of your changes in the configuration database as a snapshot with the path `/usr/es/sbin/cluster/snapshots/UserModifiedDB`. You must save these changes before restoring the Configuration Database with the values actively being used by the Cluster Manager.

Release Locks Set By Dynamic Reconfiguration
You can use this tool to release the locks used during dynamic reconfiguration. When configuration changes are made in an active cluster, there is a multiple step process of distributing the changes to all the nodes before they are committed to the active configuration. During this process, software "locks" are put in place at different phases in order to synchronize the update process. If a failure occurs at any time during this update, the locks must not be removed. If this process occurs, you must remove the locks before any more changes can be made.

Cluster Test Tool
You can use this tool to test the recovery procedures for a new cluster before it becomes part of your production environment. You can also use this tool to test configuration changes to an existing cluster, when the cluster is not in service.

PowerHA SystemMirror Trace Facility
You can use this tool to trace PowerHA SystemMirror daemons.

PowerHA SystemMirror Error Notification
You can use this tool to create error notifications.

AIX Tracing for Cluster Resources
You can use this tool to collect AIX trace data for cluster resources when an event script is run.

Compare Active and Default Configurations
You can use this tool to compare and identify any changes in the default configuration before incorporating the changes into the active configurations.

Replace the Primary Repository Disk
You can use this tool to replace the disk that is used for the cluster repository.

Open a SMIT Session on a Node
You can use this tool to open a SMIT session on a remote node from within SMIT.

Related information
Dynamic reconfiguration issues and synchronization
Verifying and synchronizing a PowerHA SystemMirror cluster
Types of error notification

PowerHA SystemMirror verification

Select this option from the **Problem Determination Tools** menu to verify that the configuration on all nodes is synchronized, set up a custom verification method, or set up automatic cluster verification.

Table 1. Problem Determination Tools fields	
Field	**Description**
Verify PowerHA SystemMirror Configuration	Select this option to verify cluster topology resources.
Configure Custom Verification Method	Use this option to add, show and remove custom verification methods.
Automatic Cluster Configuration Monitoring	Select this option to automatically verify the cluster every twenty-four hours and report results throughout the cluster.

Verify PowerHA SystemMirror configuration

You can verify cluster topology resources and custom-defined verification methods.

To verify a PowerHA SystemMirror configuration, complete the following steps:

1. From the command line, enter `smit sysmirror`.
2. In SMIT, select **Problem Determination Tools** > **PowerHA SystemMirror Verification** > **Verifying PowerHA SystemMirror Configuration**, and press Enter.
3. Enter field values as follows:

Table 2. Verify PowerHA SystemMirrorConfiguration fields	
Field	**Value**
PowerHA SystemMirror Verification Method	By default, **Pre-Installed** will run all verification methods shipped with PowerHA SystemMirror. You can select this field to run all **Pre-Installed** programs or select **none** to specify a previously defined custom verification method.
Custom Defined Verification Method	Enter the name of a custom defined verification method. Press F4 for a list of previously defined verification methods. By default, when no methods are selected, and **none** is selected in the **Base PowerHA SystemMirror Verification Method** field, verify and synchronize will not check the base verification methods, and will generate an error message. The order in which verification methods are listed determines the sequence in which the methods run. This sequence remains the same for subsequent verifications until different methods are selected. Selecting **All** verifies all custom-defined methods.
Error Count	By default, **Verify PowerHA SystemMirror Configuration** will continue to run after encountering an error in order to generate a full list of errors. To cancel the program after a specific number of errors, type the number in this field.
Log File to store output	Enter the name of an output file in which to store verification output. By default, verification output is also stored in the **/var/hacmp/clverify/clverify.log** file.

Table 2. Verify PowerHA SystemMirrorConfiguration fields (continued)	
Field	**Value**
Verify Changes Only?	Select **no** to run all verification checks that apply to the current cluster configuration. Select **yes** to run only the checks related to parts of the PowerHA SystemMirror configuration that have changed. The **yes** mode has no effect on an inactive cluster. **Note:** The **yes** option only relates to cluster Configuration Databases. If you have made changes to the AIX configuration on your cluster nodes, you should select **no**. Only select **yes** if you have made no changes to the AIX configuration.
Logging	Selecting **on** displays all output to the console that normally goes to the **/var/hacmp/clverify/ clverify.log**. The default is **off**.

Automatic monitoring and verification of cluster configuration

The **cluster verification** utility runs on one user-selectable PowerHA SystemMirror cluster node once every 24 hours.

By default, the first node in alphabetical order runs the verification at midnight. During verification, any errors that might cause problems at some point in the future are displayed. You can change the defaults, by selecting a node and time that suit your configuration.

If the selected node is unavailable (powered off), verification does not run the automatic monitoring. When cluster verification completes on the selected cluster node, this node notifies the other cluster nodes with the following verification information:

- Name of the node where verification was run
- Date and time of the last verification
- Results of the verification.

This information is stored on every available cluster node in the PowerHA SystemMirror log file **/var/ hacmp/log/clutils.log.** If the selected node became unavailable or could not complete cluster verification, you can detect this by the lack of a report in the /**var/hacmp/log/clutils.log** file.

In case cluster verification completes and detects some configuration errors, you are notified about the following potential problems:

- The exit status of cluster verification is communicated across the cluster along with the information about cluster verification process completion.
- Broadcast messages are sent across the cluster and displayed on **stdout** . These messages inform you about detected configuration errors.
- A **cluster_notify** event runs on the cluster and is logged in **hacmp.out** (if cluster services is running).

More detailed information is available on the node that completes cluster verification in **/var/hacmp/ clverify/clverify.log**. If a failure occurs during processing, error messages and warnings clearly indicate the node and reasons for the **verification** failure.

Configuring automatic verification and monitoring of cluster configuration

You can configure the node and specify the time where cluster verification runs automatically.

Make sure the **/var** file system on the node has enough space for the **/var/hacmp/log/clutils.log** file.

To configure the node and specify the time where cluster verification runs automatically:

1. From the command line, enter smit sysmirror.
2. From the SMIT interface, select **Problem Determination Tools** > **PowerHA SystemMirror Verification** > **Automatic Cluster Configuration Monitoring**, and press Enter.
3. Enter field values as follows:

Table 3. Automatic Cluster Configuration Monitoring fields	
Field	**Value**
* Automatic cluster configuration verification	**Enabled** is the default.
Node name	Select one of the cluster nodes from the list. By default, the first node in alphabetical order will verify the cluster configuration. This node will be determined dynamically every time the automatic verification occurs.
*HOUR (00 - 23)	Midnight (00) is the default. Verification runs automatically once every 24 hours at the selected hour.

4. Verify all fields are correct, and press Enter.
5. The changes take effect when the cluster is synchronized.

Related information

Monitoring a PowerHA SystemMirror cluster

PowerHA SystemMirror log viewing and management

Select this option from the **Problem Determination Tools** menu to view a list of utilities related to the log files.

From here you can:

- View, save or delete Event summaries
- View detailed PowerHA SystemMirror log files
- Change or show PowerHA SystemMirror log file parameters
- Change or show Cluster Manager log file parameters
- Change or show a cluster log file directory
- Change all Cluster Logs directory
- Collect cluster log files for problem reporting.

Related concepts

Using cluster log files
These topics explain how to use the PowerHA SystemMirror cluster log files to troubleshoot the cluster. Included also are some sections on managing parameters for some of the logs.

Related information

Testing a PowerHA SystemMirror cluster

Recovering from PowerHA SystemMirror script failure

Select this option from the **Problem Determination Tools** menu to recover from a PowerHA SystemMirror script failure.

For example, if script failure occurs because a filesystem mount failed, you can correct the problem, mount the filesystem manually, then use this option to complete the rest of the cluster event processing.

The **Recover From PowerHA SystemMirror Script Failure** menu option sends a signal to the Cluster Manager daemon (**clstrmgrES**) on the specified node, causing it to proceed to the next step in the cluster event. If a subsequent event failure occurs, you must repeat the process of correcting the problem, then using **Recover From PowerHA SystemMirror Script Failure** option to continue to the next step. You must continue this process until the cluster state goes to "stable".

Make sure that you fix the problem that caused the script failure. You need to manually complete the remaining steps that followed the failure in the event script (see **/var/hacmp/log/hacmp.out**). Then, to

resume clustering, complete the following steps to bring the PowerHA SystemMirror event script state to EVENT COMPLETED:

1. Enter `smit hacmp`
2. In SMIT, select **Problem Determination Tools > Recover From PowerHA SystemMirror Script Failure**.
3. Select the IP label/address for the node on which you want to run the **clruncmd** command and press Enter. The system prompts you to confirm the recovery attempt. The IP label is listed in the **/etc/hosts** file and is the name assigned to the service IP address of the node on which the failure occurred.
4. Press Enter to continue. Another SMIT panel appears to confirm the success of the script recovery.

Restoring PowerHA SystemMirror configuration database from an active configuration

If cluster services are up and you make changes to the configuration, those changes have modified the default configuration directory (DCD). You may realize that the impact of those changes was not well considered and you want to undo them. Because nothing was modified in the active configuration directory (ACD), all that is needed to undo the modifications to the DCD is to restore the DCD from the ACD.

Select this option from the **Problem Determination Tools** menu to automatically save any of your changes in the Configuration Database as a snapshot with the path **/usr/es/sbin/cluster/snapshots/UserModifiedDB** before restoring the Configuration Database with the values actively being used by the Cluster Manager.

1. Enter `smit hacmp`
2. In SMIT, select **Problem Determination Tools > Restore PowerHA SystemMirror Configuration Database from Active Configuration.**

   SMIT displays:

   ```
   Are you Sure?
   ```

3. Press Enter.

   The snapshot is saved, and the active configuration is copied to the DCD. You can now view the configuration and make any further changes you desire.

Related information
Saving and restoring cluster configurations

Sample custom scripts

This section includes some scenarios where it is useful to run customer scripts, and includes some sample scripts.

Making cron jobs highly available

To help maintain the PowerHA SystemMirror environment, you need to have certain **cron** jobs execute only on the cluster node that currently holds the resources.

If a **cron** job executes in conjunction with a resource or application, it is useful to have that **cron** entry fallover along with the resource. It may also be necessary to remove that **cron** entry from the **cron** table if the node no longer possesses the related resource or application.

The following example shows one way to use a customized script to do this:

The example cluster is a two node hot standby cluster where node1 is the primary node and node2 is the backup. Node1 normally owns the shared resource group and application. The application requires that a **cron** job be executed once per day but only on the node that currently owns the resources.

To ensure that the job will run even if the shared resource group and application fall over to node2, create two files as follows:

1. Assuming that the root user is executing the **cron** job, create the file **root.resource** and another file called **root.noresource** in a directory on a non-shared file system on node1. Make these files resemble the **cron** tables that reside in the directory **/var/spool/crontabs**.

   The **root.resource** table should contain all normally executed system entries, and all entries pertaining to the shared resource or application.

   The **root.noresource** table should contain all normally executed system entries but should not contain entries pertaining to the shared resource or application.

2. Copy the files to the other node so that both nodes have a copy of the two files.

3. On both systems, run the following command at system startup:

   ```
   crontab root.noresource
   ```

   This will ensure that the **cron** table for root has only the "no resource" entries at system startup.

4. You can use either of two methods to activate the *root.resource* **cron** table. The first method is the simpler of the two.

   • Run **crontab root.resource** as the last line of the application start script. In the application stop script, the first line should then be **crontab root.noresource**. By executing these commands in the application start and stop scripts, you are ensured that they will activate and deactivate on the proper node at the proper time.

   • Run the **crontab** commands as a post_event to node_up_complete and node_down_complete.

     – Upon node_up_complete on the primary node, run **crontab root.resources**.

     – On node_down_complete run **crontab root.noresources**.

   • The takeover node must also use the event handlers to execute the correct **cron** table. Logic must be written into the node_down_complete event to determine if a takeover has occurred and to run the **crontab root.resources** command. On a reintegration, a pre-event to node_up must determine if the primary node is coming back into the cluster and then run a **crontab root.noresource** command.

Making print queues highly available

In the event of a fallover, the currently queued print jobs can be saved and moved over to the surviving node.

The print spooling system consists of two directories: /**var/spool/qdaemon** and **/var/spool/lpd/qdir**. One directory contains files containing the data (content) of each job. The other contains the files consisting of information pertaining to the print job itself. When jobs are queued, there are files in each of the two directories. In the event of a fallover, these directories do not normally fallover and therefore the print jobs are lost.

The solution for this problem is to define two file systems on a shared volume group. You might call these file systems **/prtjobs** and **/prtdata**. When PowerHA SystemMirror starts, these file systems are mounted over **/var/spool/lpd/qdir** and **/var/spool/qdaemon**.

Write a script to perform this operation as a post event to node_up. The script should do the following:

1. Stop the print queues

2. Stop the print queue daemon

3. Mount **/prtjobs** over **/var/spool/lpd/qdir**

4. Mount **/prtdata** over **/var/spool/qdaemon**

5. Restart the print queue daemon

6. Restart the print queues.

   In the event of a fallover, the surviving node will need to do the following:

7. Stop the print queues

8. Stop the print queue daemon

9. Move the contents of **/prtjobs** into **/var/spool/lpd/qdir**

10. Move the contents of **/prtdata** into /**var/spool/qdaemon**

11. Restart the print queue daemon

12. Restart the print queues.

13. To do this, write a script called as a post-event to **node_down_complete** on the takeover. The script needs to determine if the **node_down** is from the primary node.

Using cluster log files

These topics explain how to use the PowerHA SystemMirror cluster log files to troubleshoot the cluster. Included also are some sections on managing parameters for some of the logs.

Viewing PowerHA SystemMirror cluster log files

Your first approach to diagnosing a problem affecting your cluster should be to examine the cluster log files for messages output by the PowerHA SystemMirror subsystems. These messages provide valuable information for understanding the current state of the cluster. The following sections describe the types of messages output by the PowerHA SystemMirror software and the log files into which the system writes these messages.

For most troubleshooting, the **/var/hacmp/log/hacmp.out** file will be the most helpful log file. Resource group handling has been enhanced in recent releases and the **hacmp.out** file has been expanded to capture more information on the activity and location of resource groups after cluster events. For instance, the **hacmp.out** file captures details of resource group parallel processing that other logs (such as the cluster history log) cannot report. The event summaries included in this log make it easier to see quickly what events have occurred recently in the cluster.

Reviewing cluster message log files

The PowerHA SystemMirror software writes the messages it generates to the system console and to several log files. Each log file contains a different subset of messages generated by the PowerHA SystemMirror software. When viewed as a group, the log files provide a detailed view of all cluster activity.

The following list describes the log files into which the PowerHA SystemMirror software writes messages and the types of cluster messages they contain. The list also provides recommendations for using the different log files. Note that the default log directories are listed here; you have the option of redirecting some log files to a chosen directory. If you have redirected any logs, check the appropriate location.

Table 4. Cluster message log files

Log file name	Description
system error log	Contains time-stamped, formatted messages from all AIX subsystems, including scripts and daemons. For information about viewing this log file and interpreting the messages it contains, see the section Understanding the system error log. **Recommended Use:** Because the **system error log** contains time-stamped messages from many other system components, it is a good place to correlate cluster events with system events.
/tmp/clconvert.log	Contains a record of the conversion progress when upgrading to a recent PowerHA SystemMirror release. The installation process runs the **cl_convert** utility and creates the **/tmp/clconvert.log** file. **Recommended Use:** View the **clconvert.log** to gauge conversion success when running **cl_convert** from the command line.
/var/ha/log/grpglsm	Contains time-stamped messages in ASCII format. These track the execution of internal activities of the RSCT Group Services Globalized Switch Membership daemon. IBM support personnel use this information for troubleshooting. The file gets trimmed regularly. Therefore, save it promptly if there is a chance you may need it.
/var/hacmp/adm/ cluster.log	Contains time-stamped, formatted messages generated by PowerHA SystemMirror scripts and daemons. **Recommended Use:** Because this log file provides a high-level view of current cluster status, check this file first when diagnosing a cluster problem.

Table 4. Cluster message log files (continued)	
Log file name	**Description**
/var/hacmp/adm/ history/ cluster.mmddyyyy	Contains time-stamped, formatted messages generated by PowerHA SystemMirror scripts. The system creates a cluster history file every day, identifying each file by its file name extension, where *mm* indicates the month, *dd* indicates the day, and *yyyy* the year. For information about viewing this log file and interpreting its messages, see the section Understanding the cluster history log file.
	Recommended Use: Use the cluster history log files to get an extended view of cluster behavior over time.
	Note that this log is not a good tool for tracking resource groups processed in parallel. In parallel processing, certain steps formerly run as separate events are now processed differently and these steps will not be evident in the cluster history log. Use the **hacmp.out** file to track parallel processing activity.
/var/log/clcomd/ clcomddiag.log	Contains time-stamped, formatted, diagnostic messages generated by **clcomd**.
	Recommended Use: Information in this file is for IBM Support personnel.
/var/hacmp/log/ autoverify.log	Contains any warnings or errors that occurred during the automatic cluster verification run.
/var/hacmp/log/ clavan.log	Contains the state transitions of applications managed by PowerHA SystemMirror. For example, when each application managed by PowerHA SystemMirror is started or stopped and when the node stops on which an application is running.
	Each node has its own instance of the file. Each record in the **clavan.log** file consists of a single line. Each line contains a fixed portion and a variable portion:
	Recommended Use: By collecting the records in the **clavan.log** file from every node in the cluster, a utility program can determine how long each application has been up, as well as compute other statistics describing application availability time.
/var/hacmp/log/ clinfo.log /var/hacmp/log/ clinfo.log.n, n=1,..,7	The **clinfo.log** file records the output generated by the event scripts as they run. This information supplements and expands upon the information in the **/var/hacmp/log/hacmp.out** file.
	You can install Client Information (Clinfo) services on both client and server systems - client systems (cluster.es.client) will not have any HACMP ODMs (for example HACMPlogs) or utilities (for example clcycle); therefore, the Clinfo logging will not take advantage of cycling or redirection.
	The default debug level is 0 or "off". You can enable logging using command line flags. Use the clinfo -l flag to change the log file name.
/var/hacmp/log/ clstrmgr.debug /var/hacmp/log/ clstrmgr.debug.n, n=1,..,7	Contains time-stamped, formatted messages generated by the **clstrmgrES** subsystem. The default messages are verbose and are typically adequate for troubleshooting most problems, however IBM support may direct you to enable additional debugging.
	Recommended Use: Information in this file is for IBM Support personnel.
/var/hacmp/log/ clstrmgr.debug.long /var/hacmp/log/ clstrmgr.debug.long.n , n=1,..,7	Contains high-level logging of cluster manager activity, in particular its interaction with other components of PowerHA SystemMirror and with RSCT, which event is currently being run, and information about resource groups (for example, their state and actions to be performed, such as acquiring or releasing them during an event.
	Recommended Use: Information in this file is for IBM Support personnel.
/var/hacmp/log/ cspoc.log	Contains time-stamped, formatted messages generated by PowerHA SystemMirror C-SPOC commands. The **cspoc.log** file resides on the node that invokes the C-SPOC command.
	Recommended Use: Use the C-SPOC log file when tracing a C-SPOC command's execution on cluster nodes.
/var/hacmp/log/ cspoc.log.long	Contains a high-level of logging for the C-SPOC utility - commands and utilities that have been invoked by C-SPOC on specified nodes and their return status.
/var/hacmp/log/ cspoc.log.remote	Contains logging of the execution of C-SPOC commands on remote nodes with ksh option xtrace enabled (set -x).
/var/hacmp/log/ hacmp.out /var/ hacmp/log/ hacmp.out.n n=1,..,7	Contains time-stamped, formatted messages generated by PowerHA SystemMirror scripts on the current day.
	In verbose mode (recommended), this log file contains a line-by-line record of every command executed by scripts, including the values of all arguments to each command. An event summary of each high-level event is included at the end of each event's details. For information about viewing this log and interpreting its messages, see the section Understanding the hacmp.out log file.
	Recommended Use: Because the information in this log file supplements and expands upon the information in the **/var/hacmp/adm/cluster.log** file, it is the primary source of information when investigating a problem.

Log file name	Description
Table 4. Cluster message log files (continued)	
Log file name	**Description**
/var/hacmp/log/ oraclesa.log	Contains information about any Oracle specific errors that occur when using this Smart Assist and is used by the Oracle Smart Assist.
/var/hacmp/log/sa.log	Contains information about any general errors that occur when using the Smart Assists and is used by the Smart Assist infrastructure.
/var/log/clcomd/ clcomddiag.log	Contains time-stamped, formatted messages generated by Cluster Communications daemon (**clcomd**) activity. The log shows information about incoming and outgoing connections, both successful and unsuccessful. Also displays a warning if the file permissions for **/usr/es/sbin/cluster/etc/rhosts** are not set correctly - users on the system should not be able to write to the file. **Recommended Use**: Use this file to troubleshoot communication problems of PowerHA SystemMirror utilities.
/var/hacmp/log/ clconfigassist.log	Contains debugging information for the Two-Node Cluster Configuration Assistant. The Assistant stores up to ten copies of the numbered log files to assist with troubleshooting activities.
/var/hacmp/clverify/ clverify.log	The **clverify.log** file contains the verbose messages output by the cluster verification utility. The messages indicate the node(s), devices, command, etc. in which any verification error occurred.
/var/hacmp/log/ clutils.log	Contains information about the date, time, results, and which node performed an automatic cluster configuration verification. It also contains information for the file collection utility, the two-node cluster configuration assistant, and the cluster test tool.
/var/hacmp/log/ cl_testtool.log	Includes excerpts from the **hacmp.out** file. The Cluster Test Tool saves up to three log files and numbers them so that you can compare the results of different cluster tests. The tool also rotates the files with the oldest file being overwritten

Related reference

Understanding the cluster.log file

The **/var/hacmp/adm/cluster.log** file is a standard text file. When checking this file, first find the most recent error message associated with your problem. Then read back through the log file to the first message relating to that problem. Many error messages cascade from an initial error that usually indicates the problem source.

Understanding the cluster history log file

The **cluster history log** file is a standard text file with the system-assigned name **/usr/es/sbin/cluster/ history/cluster.mmddyyyy**, where _mm_ indicates the month, _dd_ indicates the day in the month and _yyyy_ indicates the year.

Understanding the hacmp.out log file

The **/var/hacmp/log/hacmp.out** file is a standard text file. The system cycles **hacmp.out** log file seven times. Each copy is identified by a number appended to the file name. The most recent log file is named **/var/hacmp/log/hacmp.out**; the oldest version of the file is named **/var/hacmp/log/ hacmp.out.7**.

Related information

Upgrading a PowerHA SystemMirror cluster

Verifying and synchronizing a PowerHA SystemMirror cluster

Understanding the cluster.log file

The **/var/hacmp/adm/cluster.log** file is a standard text file. When checking this file, first find the most recent error message associated with your problem. Then read back through the log file to the first message relating to that problem. Many error messages cascade from an initial error that usually indicates the problem source.

Format of messages in the cluster.log file

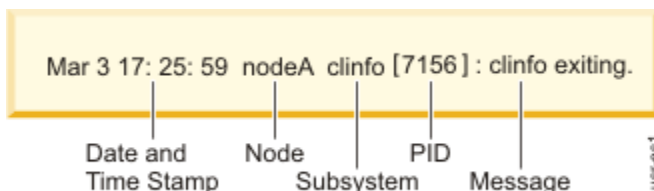The entries in the **/var/hacmp/adm/cluster.log** file use the following format:

Figure 1. Format for entries

Each entry has the following information:

Table 5. cluster.log file	
Entry	**Description**
Date and Time stamp	The day and time on which the event occurred.
Node	The node on which the event occurred.
Subsystem	The PowerHA SystemMirror subsystem that generated the event. The subsystems are identified by the following abbreviations: • **clstrmgrES** - the Cluster Manager daemon • **clinfoES** - the Cluster Information Program daemon
PID	The process ID of the daemon generating the message (not included for messages output by scripts).
Message	The message text.

The entry in the previous example indicates that the Cluster Information program (**clinfoES**) stopped running on the node named *nodeA* at 5:25 P.M. on March 3.

Because the **/var/hacmp/adm/cluster.log** file is a standard ASCII text file, you can view it using standard AIX file commands, such as the **more** or **tail** commands. However, you can also use the SMIT interface. The following sections describe each of the options.

Viewing the cluster.log file using SMIT

To view the **/var/hacmp/adm/cluster.log** file using SMIT:

1. Enter smit hacmp.
2. In SMIT, select **Problem Determination Tools > PowerHA SystemMirror Log Viewing and Management** > **View Detailed PowerHA SystemMirror Log Files** and press Enter.
3. Select **Scan the PowerHA SystemMirror for AIX System Log** and press Enter. This option references the **/var/hacmp/adm/cluster.log** file.

   Note: You can select to either *scan* the contents of the **cluster.log** file as it exists, or you can *watch* an active log file as new events are appended to it in real time. Typically, you scan the file to try to find a problem that has already occurred; you watch the file as you test a solution to a problem to determine the results.

Understanding the hacmp.out log file
The **/var/hacmp/log/hacmp.out** file is a standard text file. The system cycles **hacmp.out** log file seven times. Each copy is identified by a number appended to the file name. The most recent log file is named **/var/hacmp/log/hacmp.out**; the oldest version of the file is named **/var/hacmp/log/hacmp.out.7**.

Given the recent changes in the way resource groups are handled and prioritized in fallover circumstances, the **hacmp.out** file contains event summaries that are useful in tracking the activities and location of your resource groups.

You can customize the wait period before a warning message appears. Since this affects how often the **config_too_long** message is posted to the log, the **config_too_long** console message may not be evident in every case where a problem exists. When a cluster event runs longer than expected, a warning message is added to **hacmp.out**. This can occur if there is an event script failure or if a system command hangs or is just running slowly.

When checking the **hacmp.out** file, search for EVENT FAILED messages. These messages indicate that a failure has occurred. Then, starting from the failure message, read back through the log file to determine exactly what went wrong. The **hacmp.out** log file provides the most important source of information when investigating a problem.

Event preambles
When a cluster event processes resource groups with dependencies or replicated resources, an event preamble is included in the **hacmp.out** file.

This preamble shows the sequence of events in which the cluster manager plans to attempt and bring the resource groups online on the correct nodes and sites. It also considersthe individual group dependencies and site configuration.

Note: The preamble represents the sequence of events the cluster manager enques during the planning stage of the event. When an individual event fails, or the cluster manager recalculates the plan for any reason, a new preamble is generated. Not all events in the original preamble are necessarily run.

Example

```
PowerHA SystemMirror Event Preamble

-----------------------------------------------------------------

Node Down Completion Event has been enqueued.
-----------------------------------------------------------------
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
PowerHA SystemMirror Event Preamble
Action:  Resource:
-----------------------------------------------------------------
Enqueued rg_move acquire event for resource group rg3.

Enqueued rg_move release event for resource group rg3.

Enqueued rg_move secondary acquire event for resource group 'rg1'.
Node Up Completion Event has been enqueued.
-----------------------------------------------------------------
```

Event summaries
Event summaries that appear at the end of each event's details make it easier to check the **hacmp.out** file for errors. The event summaries contain pointers back to the corresponding event, which allow you to easily locate the output for any event.

See the section Non-verbose and verbose output of the hacmp.out log file for an example of the output.

You can also view a compilation of only the event summary sections pulled from current and past **hacmp.out** files. The option for this display is found on the **Problem Determination Tools > PowerHA SystemMirror Log Viewing and Management > View/Save/Remove Event Summaries > View Event Summaries** SMIT panel. For more detail, see the section View compiled hacmp.out event summaries.

Related reference
Viewing compiled hacmp.out event summaries
In the **hacmp.out** file, event summaries appear after those events that are initiated by the Cluster Manager. For example, **node_up** and **node_up_complete** and related subevents such as **node_up_local** and **node_up_remote_complete**.

Non-verbose and verbose output of the hacmp.out log file

You can select either verbose or non-verbose output.

Events initiated by the administrator
PowerHA includes several APIs which allow the cluster administrator to initiate cluster events.

These APIs include smit, clmgr and other command line utilities which perform various operations like starting and stopping cluster services or initiating a configuration change.

When these APIs are used in an active cluster, the administrative operations are logged in *hacmp.out* with the *admin_op* event. This event captures the details of the operation, like starting or stopping cluster services, along with any error information associated with the operation.

For example, if the administrator attempts to start cluster services when cluster services are already active, the logging for the *admin_op* event will explain that the operation cannot be performed because cluster services are already active.

Once the administrator operation has been verified and logged, the cluster manager will initiate the cluster events associated with the operation.

For example, if the administrator starts cluster services, the *admin_op* event will log the details of the operation,which will be followed by the logging of the corresponding *node_up* event.

hacmp.out in HTML format
You can view the **hacmp.out** log file in HTML format by setting formatting options on the **Problem Determination Tools > PowerHA SystemMirror Log Viewing and Management > Change/Show PowerHA SystemMirror Log File Parameters** SMIT panel.

For instructions see the section Setting the level and format of information recorded in the hacmp.out file.

Related tasks
Setting the level and format of information recorded in the hacmp.out file
You can set the level of information recorded in the **/var/hacmp/log/hacmp.out** file:

Resource group acquisition failures and volume group failures in hacmp.out
Reported resource group acquisition failures (failures indicated by a non-zero exit code returned by a command) are tracked in **hacmp.out**.

This information includes:

- The start and stop times for the event
- Which resource groups were affected (acquired or released) as a result of the event
- In the case of a failed event, an indication of which resource action failed.

  You can track the path the Cluster Manager takes as it tries to keep resources available.

  In addition, the automatically configured AIX Error Notification method that runs in the case of a volume group failure writes the following information in the **hacmp.out** log file:
- AIX error label and ID for which the method was launched
- The name of the affected resource group
- The node's name on which the error occurred.

Messages for resource group recovery upon node_up
The **hacmp.out** file, event summaries, and **clstat** include information and messages about resource groups in the ERROR state that attempted to get online on a joining node, or on a node that is starting up.

Similarly, you can trace the cases in which the acquisition of such a resource group has failed, and PowerHA SystemMirror launched an **rg_move** event to move the resource group to another node in the nodelist. If, as a result of consecutive **rg_move** events through the nodes, a non-concurrent resource group still failed to get acquired, PowerHA SystemMirror adds a message to the **hacmp.out** file.

Interface events reported for networks
When you add a network interface on a network, the actual event that runs in this case is called **join_interface**. This is reflected in the **hacmp.out** file.

Similarly, when a network interface failure occurs, the actual event that is run in is called **fail_interface**. This is also reflected in the **hacmp.out** file. Remember that the event that is being run in this case simply indicates that a network interface on the given network has failed.

Resource group processing messages in the hacmp.out file
The **hacmp.out** file allows you to fully track how resource groups have been processed in PowerHA SystemMirror.

This topic provides a brief description, for detailed information and examples of event summaries with *job types*, see the section Tracking resource group parallel and serial processing in the hacmp.out file.

For each resource group that has been processed by PowerHA SystemMirror, the software sends the following information to the **hacmp.out** file:

- Resource group name
- Script name
- Name of the command that is being executed.

The general pattern of the output is:

```
resource_group_name:script_name [line number] command line
```

In cases where an event script does not process a specific resource group, for instance, in the beginning of a **node_up** event, a resource group's name cannot be obtained. In this case, the resource group's name part of the tag is blank.

For example, the **hacmp.out** file may contain either of the following lines:

```
cas2:node_up_local[199] set_resource_status ACQUIRING
:node_up[233] cl_ssa_fence up stan
```

In addition, references to the individual resources in the event summaries in the **hacmp.out** file contain reference tags to the associated resource groups. For instance:

```
Mon.Sep.10.14:54:49.EDT 2003.cl _swap_IP_address.192.168.1.1.cas2.ref
```

Related reference
Tracking resource group processing in the hacmp.out file
Output to the **hacmp.out** file allows you to isolate details related to a specific resource group and its resources. Based on the content of the **hacmp.out** event summaries, you can determine whether or not the resource groups are being processed in the expected order.

Config_too_long message in the hacmp.out file
For each cluster event that does not complete within the specified event duration time, **config_too_long** messages are logged in the **hacmp.out** file.

The messages are then sent to the console according to the following pattern:

- The first five **config_too_long** messages appear in the **hacmp.out** file at 30-second intervals
- The next set of five messages appears at an interval that is double the previous interval until the interval reaches one hour
- These messages are logged every hour until the event completes or is terminated on that node.

You can customize the waiting period before a **config_too_long** message is sent.

Related information
Planning for cluster events

Non-verbose and verbose output of the hacmp.out log file
You can select either verbose or non-verbose output.

Non-verbose output

In non-verbose mode, the **hacmp.out** log contains the start, completion, and error notification messages output by all PowerHA SystemMirror scripts. Each entry contains the following information:

Table 6. hacmp.out log file

Entry	Description
Date and Time Stamp	The day and time on which the event occurred.
Message	Text that describes the cluster activity.
Return Status	Messages that report failures include the status returned from the script. This information is not included for scripts that complete successfully.
Event Description	The specific action attempted or completed on a node, file system, or volume group.

Verbose output

In verbose mode, the **hacmp.out** file also includes the values of arguments and flag settings passed to the scripts and commands.

Verbose output example with event summary

Some events (those initiated by the Cluster Manager) are followed by event summaries, as shown in these excerpts:

```
....
Mar 25 15:20:30 EVENT COMPLETED: network_up alcuin tmssanet_alcuin_bede

PowerHA SystemMirror Event Summary
Event: network_up alcuin tmssanet_alcuin_bede
Start time: Tue Mar 25 15:20:30 2003

End time: Tue Mar 25 15:20:30 2003

Action:  Resource:Script Name:
----------------------------------------------------------------------
No resources changed as a result of this event
----------------------------------------------------------------------
```

Event summary for the settling time

CustomRG has a settling time configured. A less priority node joins the cluster:

```
Mar 25 15:20:30 EVENT COMPLETED: node_up alcuin

PowerHA SystemMirror Event Summary
Event: node_up alcuin
Start time: Tue Mar 25 15:20:30 2003

End time: Tue Mar 25 15:20:30 2003

Action:  Resource:  Script Name:
----------------------------------------------------------------

No action taken on resource group 'CustomRG'.
The Resource Group 'CustomRG' has been configured
to use 20 Seconds Settling Time. This group will be
processed when the timer expires.
----------------------------------------------------------------------
```

Event summary for the fallback timer

CustomRG has a daily fallback timer configured to fall back on 22 hrs 10 minutes. The resource group is on a less priority node (bede). Therefore, the timer is ticking; the higher priority node (alcuin) joins the cluster:

```
The message on bede
...

Mar 25 15:20:30 EVENT COMPLETED: node_up alcuin

PowerHA SystemMirror Event Summary
Event: node_up alcuin
Start time: Tue Mar 25 15:20:30 2003

End time: Tue Mar 25 15:20:30 2003

Action:  Resource:  Script Name:
----------------------------------------------------------------
No action taken on resource group 'CustomRG'.
The Resource Group 'CustomRG' has been configured
to fallback on Mon Mar 25 22:10:00 2003
-----------------------------------------------------------------------
The message on alcuin ...
Mar 25 15:20:30 EVENT COMPLETED: node_up alcuin

PowerHA SystemMirror Event Summary
Event: node_up alcuin
Start time: Tue Mar 25 15:20:30 2003

End time: Tue Mar 25 15:20:30 2003

Action:  Resource:  Script Name:
--------------------------------------------------------------
The Resource Group 'CustomRG' has been configured
to fallback using daily1 Timer Policy
-----------------------------------------------------------------------
```

View the hacmp.out file using SMIT
You can view the **/var/hacmp/log/hacmp.out** file using SMIT.

To view the **/var/hacmp/log/hacmp.out** file using SMIT:

1. Enter `smit hacmp`
2. In SMIT, select **Problem Determination Tools > PowerHA SystemMirror Log Viewing and Management** > **View Detailed PowerHA SystemMirror Log Files** and press Enter.
3. On the **View Detailed PowerHA SystemMirror Log Files** menu, you can select to either *scan* the contents of the**/var/hacmp/log/hacmp.out** file or *watch* as new events are appended to the log file. Typically, you will scan the file to try to find a problem that has already occurred and then watch the file as you test a solution to the problem. In the menu, the **/var/hacmp/log/hacmp.out** file is referred to as the PowerHA SystemMirror Script Log File.
4. Select **Scan the PowerHA SystemMirror Script Log File** and press Enter.
5. Select a script log file and press Enter.

Setting the level and format of information recorded in the hacmp.out file
You can set the level of information recorded in the **/var/hacmp/log/hacmp.out** file:

Note: These preferences take place as soon as you set them.

To set the level of information recorded in the **/var/hacmp/log/hacmp.out** file:

1. Enter `smit hacmp`
2. In SMIT, select **Problem Determination Tools > PowerHA SystemMirror Log Viewing and Management > Change/Show PowerHA SystemMirror Log File Parameters**.

   SMIT prompts you to specify the name of the cluster node you want to modify. Runtime parameters are configured on a per-node basis.

3. Type the node name and press Enter.

   SMIT displays the **PowerHA SystemMirror Log File Parameters** panel.
4. To obtain verbose output, set the value of the **Debug Level** field to **high.**
5. To change the **hacmp.out** display format, select **Formatting options for hacmp.out**. Select a node and set the formatting to **HTML (Low)**, **HTML (High)**, **Default (None)**, or **Standard**.

   Note: If you set your formatting options for **hacmp.out** to **Default (None)**, then no event summaries will be generated. For information about event summaries, see the section Viewing compiled hacmp.out event summaries.
6. To change the level of debug information, set the value of **Cluster Manager debug level** field to either **standard** or **high**.

Related reference

Viewing compiled hacmp.out event summaries
In the **hacmp.out** file, event summaries appear after those events that are initiated by the Cluster Manager. For example, **node_up** and **node_up_complete** and related subevents such as **node_up_local** and **node_up_remote_complete**.

Viewing compiled hacmp.out event summaries

In the **hacmp.out** file, event summaries appear after those events that are initiated by the Cluster Manager. For example, **node_up** and **node_up_complete** and related subevents such as **node_up_local** and **node_up_remote_complete**.

Note that event summaries do not appear for all events; for example, when you move a resource group through SMIT.

The **View Event Summaries** option displays a compilation of all event summaries written to a node's **hacmp.out** file. This utility can gather and display this information even if you have redirected the **hacmp.out** file to a new location. You can also save the event summaries to a file of your choice instead of viewing them via SMIT.

Note: Event summaries pulled from the **hacmp.out** file are stored in the **/usr/es/sbin/cluster/ cl_event_summary.txt** file. This file continues to accumulate as **hacmp.out** cycles, and is not automatically truncated or replaced. Consequently, it can grow too large and crowd your **/usr** directory. You should clear event summaries periodically, using the **Remove Event Summary History** option in SMIT.

This feature is node-specific. Therefore, you cannot access one node's event summary information from another node in the cluster. Run the **View Event Summaries** option on each node for which you want to gather and display event summaries.

The event summaries display is a good way to get a quick overview of what has happened in the cluster lately. If the event summaries reveal a problem event, you will probably want to examine the source **hacmp.out** file to see full details of what happened.

Note: If you have set your formatting options for **hacmp.out** to **Default (None)**, then no event summaries will be generated. The **View Event Summaries** command will yield no results.

How event summary view information is gathered
The **Problem Determination Tools > PowerHA SystemMirror Log Viewing and Management -> View/ Save/Remove PowerHA SystemMirror Event Summaries -> View Event Summaries** option gathers information from the **hacmp.out** log file, not directly from PowerHA SystemMirror while it is running. Consequently, you can access event summary information even when PowerHA SystemMirror is not running. The summary display is updated once per day with the current day's event summaries.

In addition, the display shows the resource group location and state information. This information reflects output from the **clRGinfo** command.

Note that **clRGinfo** displays resource group information more quickly when the cluster is running. If the cluster is not running, wait a few minutes and the resource group information will eventually appear.

Viewing event summaries
You can view a compiled list of event summaries on a node using SMIT.

To view a compiled list of event summaries on a node:

1. Enter `smit hacmp`
2. In SMIT, select **View Event Summaries** and press Enter. SMIT displays a list of event summaries generated on the node. SMIT will notify you if no event summaries were found.

Saving event summaries to a specified file
You can store the compiled list of a node's event summaries to a file using SMIT.

To store the compiled list of a node's event summaries to a file:

1. Enter `smit hacmp`
2. In SMIT, select **View/Save/Remove PowerHA SystemMirror Event Summaries**.
3. Select **Save Event Summaries to a file**.
4. Enter the path/file name where you wish to store the event summaries.

Depending on the format you select (for example .txt or .html), you can then move this file to be able to view it in a text editor or browser.

Understanding the system error log

The PowerHA SystemMirror software logs messages to the system error log whenever a daemon generates a state message.

The PowerHA SystemMirror messages in the system error log follow the same format used by other AIX subsystems. You can view the messages in the system error log in short or long format.

In short format, also called summary format, each message in the **system error log** occupies a single line. The description of the fields in the short format of the **system error log**:

Table 7. system error log	
Field	**Description**
Error_ID	A unique error identifier.
Time stamp	The day and time on which the event occurred.
T	Error type: **permanent (P), unresolved (U), or temporary (T)**.
CL	Error class: **hardware (H), software (S), or informational (O)**.
Resource_name	A text string that identifies the AIX resource or subsystem that generated the message. PowerHA SystemMirror messages are identified by the name of their daemon.
Error_description	A text string that briefly describes the error.

In long format, a page of formatted information is displayed for each error.

Unlike the PowerHA SystemMirror log files, the **system error log** is not a text file.

The AIX **errpt** command generates an error report from entries in the system error log. For information on using this command see the **errpt** man page.

To view the AIX **system error log**, you must use the AIX SMIT:

1. Enter `smit`
2. In SMIT, select **Problem Determination Tools** > **PowerHA SystemMirror Log Viewing and Management > View Detailed PowerHA SystemMirror Log Files > Scan the PowerHA SystemMirror for AIX System Log** and press Enter.

   SMIT displays the error log.

Understanding the cluster history log file

The **cluster history log** file is a standard text file with the system-assigned name **/usr/es/sbin/cluster/ history/cluster.mmddyyyy**, where *mm* indicates the month, *dd* indicates the day in the month and *yyyy* indicates the year.

You should decide how many of these log files you want to retain and purge the excess copies on a regular basis to conserve disk storage space. You may also decide to include the **cluster history log** file in your regular system backup procedures.

The description of the fields in the **cluster history log** file messages:

Table 8. cluster history log file	
Field	**Description**
Date and Time stamp	The date and time at which the event occurred.
Message	Text of the message.
Description	Name of the event script.

Note: This log reports specific events. Note that when resource groups are processed in parallel, certain steps previously run as separate events are now processed differently, and therefore do not show up as events in the cluster history log file. You should use the **hacmp.out** file, which contains greater detail on resource group activity and location, to track parallel processing activity.

Because the **cluster history log** file is a standard text file, you can view its contents using standard AIX file commands, such as **cat**, **more**, and **tail**. You cannot view this log file using SMIT.

Collecting cluster log files for problem reporting

If you encounter a problem with PowerHA SystemMirror and report it to IBM support, you may be asked to collect log files pertaining to the problem. In PowerHA SystemMirror, the **Collect PowerHA SystemMirror Log Files for Problem Reporting** SMIT panel aids in this process.

> ⚠️ **CAUTION:** Use this panel only if requested by the IBM support personnel. If you use this utility without direction from IBM support, be careful to fully understand the actions and the potential consequences.

To collect cluster log files for problem reporting:

1. Enter `smit hacmp`
2. In SMIT, select **Problem Determination Tools > PowerHA SystemMirror Log Viewing and Management > Collect Log Files for Problem Reporting**.
3. Type or select values in entry fields:

Table 9. Collect Log Files for Problem Reporting fields	
Field	**Value**
Log Destination Directory	Enter a directory name where cluster logs will be collected. The default is **/tmp**.
Collection Pass Number	Select a value in this field. The default is **2** (collect). Select **1** to calculate the amount of space needed. Select **2** to collect the actual data.
Nodes to Collect Data from	Enter or select nodes from which the data will be collected. Separate node names with a comma. The default is **All** nodes.
Debug	The default is **No**. Use this option if IBM Support requests to turn on debugging.
Collect RSCT Log Files	The default is **Yes**. Skip collection of RSCT data.

Managing cluster log files

PowerHA SystemMirror automatically manages **cluster log** files. The individual logs are limited to a maximum size and are removed after they reach a certain age, or are overwritten by newer versions.

In general, PowerHA SystemMirror defaults to the following rules for all log files.

Table 10. General rules for log files	
Item	**Rule**
Maximum size	Log files that are over 1 MB in size are cycled.
Maximum number of outdated logs	No more than 7 prior versions of the file are preserved.
Maximum age	Log files older than one day are cycled.

If you want to customize the values that are specified in the general rules, you can override them by specifying different values in the **/etc/environment** file on each cluster node.

To override the default values, add the following entries:

Table 11. Override values	
Item	**Description**
CLCYCLE_MAX_SIZE=*<size in bytes>*	Add this entry to limit the maximum size of any saved log file.
CLCYCLE_MAX_LOGS=*<number of old files to save>*	Add this entry to change the number of old log files that are preserved by the **clcycle** command.
CLCYCLE_MAX_DAYS=*<cycle log files older than this number of days>*	Add this entry to change the age at which log files are to be cycled.
CLCYCLE_CLUSTER_LOG= *<FALSE| TRUE>*	Add this entry to ensure that the **cluster.log** file is not managed by PowerHA SystemMirror by default. Instead, PowerHA SystemMirror adds entries to the **syslog.conf** file, which causes the **syslog subsystem** to manage the size, age, and backup copies of the **cluster.log** file. **Note:** If you want PowerHA SystemMirror to manage the **cluster.log** file, specify **CLCYCLE_CLUSTER_LOG=TRUE** in the **/etc/environmment** file.

Tracking resource group processing in the hacmp.out file

Output to the **hacmp.out** file allows you to isolate details related to a specific resource group and its resources. Based on the content of the **hacmp.out** event summaries, you can determine whether or not the resource groups are being processed in the expected order.

Parallel processing order reflected in event summaries

Several features are listed in the **hacmp.out** file and in the event summaries that might help you follow the flow of parallel resource group processing.

- Each line in the **hacmp.out** file flow includes the name of the resource group to which it applies.
- The event summary information includes details about all resource types.
- Each line in the event summary indicates the related resource group.

The following example shows an event summary for resource groups named `cascrg1` and `cascrg2` that are processed in parallel:

```
PowerHA SystemMirror Event Summary
```

```
Event: node_ up electron

Start time: Wed May 8 11: 06: 30 2002
End time: Wed May 8 11: 07: 49 2002

Action:  Resource:  Script Name: -------------------------------------------------------------

Acquiring resource group: cascrg1 process_ resources
Search on: Wed. May. 8. 11: 06: 33. EDT. 2002. process_ resources. cascrg1. ref
Acquiring resource group: cascrg2 process_ resources
Search on: Wed. May. 8. 11: 06: 34. EDT. 2002. process_ resources. cascrg2. ref
Acquiring resource: 192. 168. 41. 30 cl_ swap_ IP_ address
Search on: Wed. May. 8. 11: 06: 36. EDT. 2002. cl_ swap_ IP_ address. 192. 168. 41. 30
Acquiring resource: hdisk1 cl_ disk_ available
Search on: Wed. May. 8. 11: 06: 40. EDT. 2002. cl_ disk_ available.
hdisk1. cascrg1
Acquiring resource: hdisk2 cl_ disk_ available
Search on: Wed. May. 8. 11: 06: 40. EDT. 2002. cl_ disk_ available.
hdisk2. cascrg2 Resource online: hdisk1 cl_ disk_ available
Search on: Wed. May. 8. 11: 06: 42. EDT. 2002. cl_ disk_ available.
hdisk1. cascrg1 Resource online: hdisk2 cl_ disk_ available
Search on: Wed. May. 8. 11: 06: 43. EDT. 2002. cl_ disk_ available.
hdisk2. cascrg2
```

As shown here, all processed resource groups are listed first, followed by the individual resources that are being processed.

Job types: Parallel resource group processing

When resource group dependencies or sites are configured in the cluster, check the event preamble which lists the plan of action the Cluster Manager. This plan describes the process the resource groups follow for the prescribed events.

Execution of individual events is traced in the **hacmp.out** file. If there is a problem with an event, or it did not produce the expected results, certain patterns, and keywords are presented in the **hacmp.out**. This file is used to try to identify the source of the problem.

The following information is provided for users who are interested in understanding the low-level details of cluster event processing. It is not intended as a reference for use in primary problem determination.

If you have a problem with PowerHA SystemMirror Enterprise Edition follow your local problem reporting and support procedures as a primary response.

The cluster manager uses an approach described as "parallel processing" for planning cluster events. Parallel processing combines several different recovery steps in a single event in order to maximize the efficiency and speed of event processing. With parallel processing, the **process_resources** event script is used as a main event for processing different resources based on resource types. The **process_resources** event uses a keyword "JOB_TYPE" to identify the resources currently being processed.

Job types are listed in the **hacmp.out** log file. This list assists you to identify the sequence of events that take place during acquisition or release of different types of resources. Depending on the cluster resource groups configuration, other specific job types that take place during parallel processing of resource groups.

- There is one job type for each resource type, which includes, but is not
  limited to the following: DISKS, FILESYSTEMS, TAKEOVER_LABELS, TAPE_RESOURCES, AIX_FAST_CONNECTIONS, APPLICATIONS, COMMUNICATION_LINKS, USERDEF_RESOURCES, CONCURRENT_VOLUME_GROUPS, EXPORT_FILESYSTEMS, MOUNT_FILESYSTEMS and REMOUNT_FILESYSTEMS.
- There are also a number of job types that are used to help capitalize on the benefits of parallel processing: SETPRKEY, TELINIT, SYNC_VGS, LOGREDO, NFS_STOP, and UPDATESTATD. Now the related operations are run only one time per event, rather than with each resource group. This change is one of the primary areas of benefit from parallel resource group processing, especially for small clusters.

JOB_TYPE=ONLINE
In the complete phase of an acquisition event, after all resources for all resource groups have been successfully acquired, the **ONLINE** job type is run. This job ensures that all successfully acquired resource

groups are set to the online state. The **RESOURCE_GROUPS** variable contains the list of all groups that were acquired.

```
:process_resources[1476] clRGPA
:clRGPA[48] [[ high = high ]]
:clRGPA[48] version= 1. 16
:clRGPA[50] usingVer= clrgpa
:clRGPA[55] clrgpa
:clRGPA[56] exit 0
:process_resources[1476] eval JOB_TYPE= ONLINE RESOURCE_GROUPS="
cascrg1 cascrg2 conc_ rg1"

:process_resources[1476] JOB_TYPE= ONLINE RESOURCE_GROUPS=
cascrg1 cascrg2 conc_rg1 :process_resources[1478] RC= 0
:process_resources[1479] set +a
:process_resources[1481] [ 0 -ne 0 ]
:process_resources[1700] set_resource_group_state UP
```

JOB_TYPE= OFFLINE
In the complete phase of a release event, after all resources for all resource groups have been successfully released, the **OFFLINE** job type is run. This job ensures that all successfully released resource groups are set to the offline state. The **RESOURCE_GROUPS** variable contains the list of all groups that were released.

```
conc_rg1 :process_resources[1476] clRGPA
conc_rg1 :clRGPA[48] [[ high = high ]]
conc_rg1 :clRGPA[48] version= 1. 16
conc_rg1 :clRGPA[50] usingVer= clrgpa
conc_rg1 :clRGPA[55] clrgpa
conc_rg1 :clRGPA[56] exit 0
conc_rg1 :process_resources[1476] eval JOB_TYPE= OFFLINE RESOURCE_GROUPS=" cascrg2 conc_ rg1"

conc_ rg1:process_resources[1476] JOB_TYPE= OFFLINE RESOURCE_GROUPS= cascrg2 conc_rg1

conc_ rg1 :process_resources[1478] RC= 0
conc_rg1 :process_resources[1479] set +a
conc_rg1 :process_resources[1481] [ 0 -ne 0 ]
conc_rg1 :process_resources[1704] set_resource_group_state DOWN
```

JOB_TYPE=ERROR
If an error occurred during the acquisition or release of any resource, the **ERROR** job type is run. The variable **RESOURCE_GROUPS** contains the list of all groups where acquisition or release failed during the current event. These resource groups are moved into the error state. When this job is run during an acquisition event, PowerHA SystemMirror uses the Recovery from Resource Group Acquisition Failure feature and launches an **rg_move** event for each resource group in the error state.

```
conc_rg1: process_resources[1476] clRGPA
conc_rg1: clRGPA[50] usingVer= clrgpa
conc_rg1: clRGPA[55] clrgpa
conc_rg1: clRGPA[56] exit 0
conc_rg1: process_resources[1476] eval JOB_ TYPE= ERROR RESOURCE_GROUPS=" cascrg1"

conc_rg1: process_resources[1476] JOB_TYPE= ERROR RESOURCE_GROUPS= cascrg1
conc_rg1: process_resources[1478] RC= 0
conc_rg1: process_resources[1479] set +a
conc_rg1: process_resources[1481] [ 0 -ne 0 ]
conc_rg1: process_resources[1712] set_resource_group_state ERROR
```

Related information
Resource group behavior during cluster events

JOB_TYPE=NONE
After all processing is complete for the current **process_resources** script, the final job type of **NONE** is used to indicate that processing is complete and the script can return. When exiting after receiving this job, the **process_resources** script always returns 0 for success.

```
conc_rg1: process_resources[1476] clRGPA
conc_rg1: clRGPA[48] [[ high = high ]]
conc_rg1: clRGPA[48] version= 1.16
conc_rg1: clRGPA[50] usingVer= clrgpa
```

```
conc_rg1: clRGPA[55] clrgpa
conc_rg1: clRGPA[56] exit 0
conc_rg1: process_resources[1476] eval JOB_TYPE= NONE
conc_rg1: process_resources[1476] JOB_TYPE= NONE
conc_rg1: process_resources[1478] RC= 0
conc_rg1: process_resources[1479] set +a
conc_rg1: process_resources[1481] [ 0 -ne 0 ]
conc_rg1: process_resources[1721] break
conc_rg1: process_resources[1731] exit 0
```

JOB_TYPE=ACQUIRE

The **ACQUIRE** job type occurs at the beginning of any resource group acquisition event. Search **hacmp. out** for **JOB_ TYPE= ACQUIRE** and view the value of the **RESOURCE_ GROUPS** variable to see a list of which resource groups are being acquired in parallel during the event.

```
:process_resources[1476] clRGPA
:clRGPA[48] [[ high = high ]]
:clRGPA[48] version= 1. 16
:clRGPA[50] usingVer= clrgpa
:clRGPA[55] clrgpa
:clRGPA[56] exit 0
:process_resources[1476] eval JOB_TYPE= ACQUIRE RESOURCE_GROUPS=" cascrg1 cascrg2"
:process_resources[1476] JOB_TYPE= ACQUIRE RESOURCE_GROUPS= cascrg1 cascrg2
:process_resources[1478] RC= 0
:process_resources[1479] set +a
:process_resources[1481] [ 0 -ne 0 ]
:process_resources[1687] set_resource_group_state ACQUIRING
```

JOB_TYPE=RELEASE

The **RELEASE** job type occurs at the beginning of any resource group release event. Search **hacmp. out** for **JOB_ TYPE= RELEASE** and view the value of the **RESOURCE_ GROUPS** variable to see a list of which resource groups are being released in parallel during the event.

```
:process_resources[1476] clRGPA
:clRGPA[48] [[ high = high ]]
:clRGPA[48] version= 1. 16
:clRGPA[50] usingVer= clrgpa
:clRGPA[55] clrgpa
:clRGPA[56] exit 0
:process_resources[1476] eval JOB_ TYPE= RELEASE RESOURCE_ GROUPS=" cascrg1 cascrg2"
:process_resources[1476] JOB_ TYPE= RELEASE RESOURCE_ GROUPS= cascrg1 cascrg2
:process_resources[1478] RC= 0
:process_resources[1479] set +a
:process_resources[1481] [ 0 -ne 0 ]
:process_resources[1691] set_ resource_ group_ state RELEASING
```

JOB_TYPE= SSA_FENCE

The **SSA_FENCE** job type is used to handle fencing and unfencing of SSA disks. The variable **ACTION** indicates what should be done to the disks listed in the **HDISKS** variable. All resources groups (both parallel and serial) use this method for disk fencing.

```
:process_resources[1476] clRGPA FENCE
:clRGPA[48] [[ high = high ]]
:clRGPA[55] clrgpa FENCE
:clRGPA[56] exit 0
:process_resources[1476] eval JOB_TYPE= SSA_ FENCE ACTION= ACQUIRE
HDISKS=" hdisk6" RESOURCE_GROUPS=" conc_ rg1 " HOSTS=" electron"
:process_ resources[1476] JOB_TYPE= SSA_FENCE ACTION= ACQUIRE
HDISKS= hdisk6 RESOURCE_GROUPS= conc_rg1 HOSTS=electron
:process_ resources[1478] RC= 0
:process_ resources[1479] set +a
:process_ resources[1481] [ 0 -ne 0 ]
:process_ resources[1675] export GROUPNAME= conc_ rg1 conc_ rg1
:process_ resources[1676] process_ ssa_ fence ACQUIRE
```

Note: Notice that disk fencing uses the **process_resources** script, and, therefore, when disk fencing occurs, it may mislead you to assume that resource processing is taking place, when, in fact, only disk fencing is taking place. If disk fencing is enabled, you will see in the **hacmp.out** file that the disk fencing operation occurs *before* any resource group processing. Although the **process_ resources** script handles SSA disk fencing, the resource groups are processed serially. **cl_ ssa_ fence** is called once for each

resource group that requires disk fencing. The **hacmp.out** content indicates which resource group is being processed.

```
conc_ rg1: process_resources[8] export GROUPNAME
conc_ rg1: process_resources[10] get_ list_ head hdisk6
conc_ rg1: process_resources[10] read LIST_OF_HDISKS_ FOR_ RG
conc_ rg1: process_resources[11] read HDISKS
conc_ rg1: process_resources[11] get_ list_ tail hdisk6
conc_ rg1: process_resources[13] get_ list_ head electron
conc_ rg1: process_resources[13] read HOST_ FOR_ RG
conc_ rg1: process_resources[14] get_ list_ tail electron
conc_ rg1: process_resources[14] read HOSTS
conc_ rg1: process_resources[18] cl_ ssa_fence ACQUIRE electron hdisk6
conc_ rg1: cl_ssa_fence[43] version= 1. 9. 1. 2
conc_ rg1: cl_ssa_fence[44]
conc_ rg1: cl_ssa_fence[44]
conc_ rg1: cl_ssa_fence[46] STATUS= 0
conc_ rg1: cl_ssa_fence[48] (( 3 < 3
conc_ rg1: cl_ssa_fence[56] OPERATION= ACQUIRE
```

JOB_TYPE=SERVICE_LABELS
The **SERVICE_LABELS** job type handles the acquisition or release of service labels. The variable **ACTION** indicates what should be done to the service IP labels listed in the **IP_LABELS** variable.

```
conc_ rg1: process_ resources[ 1476] clRGPA
conc_ rg1: clRGPA[ 55] clrgpa
conc_ rg1: clRGPA[ 56] exit 0
conc_ rg1: process_ resources[ 1476] eval JOB_ TYPE= SERVICE_ LABELS
ACTION= ACQUIRE IP_ LABELS=" elect_ svc0: shared_ svc1, shared_ svc2"
RESOURCE_ GROUPS=" cascrg1 rotrg1" COMMUNICATION_ LINKS=": commlink1"
conc_ rg1: process_ resources[1476] JOB_ TYPE= SERVICE_ LABELS
ACTION= ACQUIRE IP_ LABELS= elect_ svc0: shared_ svc1, shared_ svc2
RESOURCE_ GROUPS= cascrg1 rotrg1 COMMUNICATION_ LINKS=: commlink1
conc_ rg1: process_ resources[1478] RC= 0
conc_ rg1: process_ resources[1479] set +a
conc_ rg1: process_ resources[1481] [ 0 -ne 0 ]
conc_ rg1: process_ resources[ 1492] export GROUPNAME= cascrg1
```

This job type launches an **acquire_service_addr** event. Within the event, each individual service label is acquired. The content of the **hacmp.out** file indicates which resource group is being processed. Within each resource group, the event flow is the same as it is under serial processing.

```
cascrg1: acquire_service_addr[ 251] export GROUPNAME
cascrg1: acquire_service_addr[251] [[ true = true ]]
cascrg1: acquire_service_addr[254] read SERVICELABELS
cascrg1: acquire_service_addr[254] get_ list_ head electron_ svc0
cascrg1: acquire_service_addr[255] get_ list_ tail electron_ svc0
cascrg1: acquire_service_addr[255] read IP_ LABELS
cascrg1: acquire_service_addr[257] get_ list_ head
cascrg1: acquire_service_addr[257] read SNA_ CONNECTIONS
cascrg1: acquire_service_addr[258] export SNA_ CONNECTIONS
cascrg1: acquire_service_addr[259] get_ list_ tail
cascrg1: acquire_service_addr[259] read _SNA_ CONNECTIONS
cascrg1: acquire_service_addr[270] clgetif -a electron_ svc0
```

JOB_TYPE=VGS
The **VGS** job type handles the acquisition or release of volume groups. The variable **ACTION** indicates what should be done to the volume groups being processed, and the names of the volume groups are listed in the **VOLUME_GROUPS** and **CONCURRENT_VOLUME_GROUPS** variables.

```
conc_rg1 :process_resources[1476] clRGPA
conc_rg1 :clRGPA[55] clrgpa
conc_rg1 :clRGPA[56] exit 0

conc_rg1 :process_resources[1476] eval JOB_TYPE= VGS ACTION= ACQUIRE
CONCURRENT_VOLUME_GROUP=" con_vg6" VOLUME_GROUPS=""
casc_vg1: casc_vg2" RESOURCE_GROUPS=" cascrg1 cascrg2 "
EXPORT_FILESYSTEM=""

conc_rg1 :process_resources[1476] JOB_TYPE= VGS
ACTION= ACQUIRE CONCURRENT_VOLUME_GROUP= con_vg6 VOLUME_GROUPS= casc_vg1: casc_ vg2
RESOURCE_GROUPS= cascrg1 cascrg2 EXPORT_FILESYSTEM=""
```

```
conc_rg1 :process_resources[1478] RC= 0
conc_rg1 :process_resources[1481] [ 0 -ne 0 ]
conc_rg1 :process_resources[1529]
export GROUPNAME= cascrg1 cascrg2
```

This job type runs the **cl_activate_vgs** event utility script, which acquires each individual volume group. The content of the **hacmp.out** file indicates which resource group is being processed, and within each resource group, the script flow is the same as it is under serial processing.

```
cascrg1 cascrg2 :cl_activate_vgs[256] 1> /usr/ es/ sbin/ cluster/ etc/ lsvg. out. 21266
2> /tmp/ lsvg. err

cascrg1: cl_activate_vgs[260] export GROUPNAME
cascrg1: cl_activate_vgs[262] get_ list_head
casc_vg1: casc_vg2
cascrg1: cl_activate_vgs[ 62] read_LIST_OF_VOLUME_GROUPS_FOR_RG
cascrg1: cl_activate_vgs[263] get_list_tail casc_vg1: casc_vg2
cascrg1: cl_activate_vgs[263] read VOLUME_GROUPS
cascrg1: cl_activate_vgs[265] LIST_OF_VOLUME_GROUPS_ FOR_ RG=
cascrg1: cl_activate_vgs[ 270] fgrep -s -x casc_ vg1 /usr/ es/ sbin/
cluster/ etc/ lsvg. out. 21266
cascrg1: cl_activate_vgs[275] LIST_OF_VOLUME_GROUPS_FOR_RG= casc_vg1
cascrg1: cl_activate_vgs[275] [[ casc_ vg1 = ]]
```

Disk fencing
Disk fencing uses the **process_resources** script with the JOB_TYPE=SSA_FENCE.

Processing in clusters with dependent resource groups or sites
Resource groups in clusters that are configured with dependent groups or sites, that are handled with dynamic event phasing.

These events process one or more resource groups at a time. Multiple nonconcurrent resource groups can be processed within one **rg_move** event.

Related information
Applications and PowerHA SystemMirror

Sample event with siblings output to hacmp.out
This topic contains a sample event with siblings output to **hacmp.out**.

```
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
Mar 28 09:40:42 EVENT START: rg_move a2 1ACQUIRE
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
:process_resources[1952] eval JOB_TYPE=ACQUIRE RESOURCE_GROUPS="rg3"
SIBLING_GROUPS="rg1 rg3" SIBLING_NODES_BY_GROUP="b2 : b2"
SIBLING_ACQUIRING_GROUPS="" SIBLING_ACQUIRING_NODES_BY_GROUP=""
PRINCIPAL_ACTION="ACQUIRE" AUXILLIARY_ACTION="NONE"
:process_resources[1952] JOB_TYPE=ACQUIRE RESOURCE_GROUPS=rg3
SIBLING_GROUPS=rg1 rg3 SIBLING_NODES_BY_GROUP=b2 : b2
SIBLING_ACQUIRING_GROUPS= SIBLING_ACQUIRING_NODES_BY_GROUP=
PRINCIPAL_ACTION=ACQUIRE AUXILLIARY_ACTION=NONE
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
:rg_move_complete[157] eval FORCEDOWN_GROUPS="" RESOURCE_GROUPS=""
HOMELESS_GROUPS="" ERRSTATE_GROUPS="" PRINCIPAL_ACTIONS=""
ASSOCIATE_ACTIONS="" AUXILLIARY_ACTIONS="" SIBLING_GROUPS="rg1 rg3"
SIBLING_NODES_BY_GROUP="b2 : b2" SIBLING_ACQUIRING_GROUPS="" SIBLING
_ACQUIRING_NODES_BY_GROUP="" SIBLING_RELEASING_GROUPS=""
SIBLING_RELEASING_NODES_BY_GROUP=""
:rg_move_complete[157] FORCEDOWN_GROUPS= RESOURCE_GROUPS=
HOMELESS_GROUPS= ERRSTATE_GROUPS= PRINCIPAL_ACTIONS= ASSOCIATE_ACTIONS=
AUXILLIARY_ACTIONS= SIBLING_GROUPS=rg1 rg3 SIBLING_NODES_BY_GROUP=b2 :
b2 SIBLING_ACQUIRING_GROUPS= SIBLING_ACQUIRING_NODES_BY_GROUP
= SIBLING_RELEASING_GROUPS= SIBLING_RELEASING_NODES_BY_GROUP=
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
:process_resources[1952] eval JOB_TYPE=SYNC_VGS ACTION=ACQUIRE
VOLUME_GROUPS="vg3,vg3sm" RESOURCE_GROUPS="rg3 "
:process_resources[1952] JOB_TYPE=SYNC_VGS
ACTION=ACQUIRE_VOLUME_GROUPS=vg3,vg3sm RESOURCE_GROUPS=rg3
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
rg3:process_resources[1952] eval JOB_TYPE=ONLINE RESOURCE_GROUPS="rg3"
rg3:process_resources[1952] JOB_TYPE=ONLINE RESOURCE_GROUPS=rg3
rg3:process_resources[1954] RC=0
rg3:process_resources[1955] set +a
```

```
rg3:process_resources[1957] [ 0 -ne 0 ]
rg3:process_resources[2207] set_resource_group_state UP
rg3:process_resources[3] STAT=0
rg3:process_resources[6] export GROUPNAME
rg3:process_resources[7] [ UP != DOWN ]
rg3:process_resources[9] [ REAL = EMUL ]
rg3:process_resources[14] clchdaemons -d clstrmgr_scripts -t resource_locator -n a1 -o rg3 -v UP
rg3:process_resources[15] [ 0 -ne 0 ]
rg3:process_resources[26] [ UP = ACQUIRING ]
rg3:process_resources[31] [ UP = RELEASING ]
rg3:process_resources[36] [ UP = UP ]
rg3:process_resources[38] cl_RMupdate rg_up rg3 process_resources
Reference string: Sun.Mar.27.18:02:09.EST.2005.process_resources.rg3.ref
rg3:process_resources[39] continue
rg3:process_resources[80] return 0
rg3:process_resources[1947] true
rg3:process_resources[1949] set -a
rg3:process_resources[1952] clRGPA
rg3:clRGPA[33] [[ high = high ]]
rg3:clRGPA[33] version=1.16
rg3:clRGPA[35] usingVer=clrgpa
rg3:clRGPA[40] clrgpa
rg3:clRGPA[41] exit 0
rg3:process_resources[1952] eval JOB_TYPE=NONE
rg3:process_resources[1952] JOB_TYPE=NONE
rg3:process_resources[1954] RC=0
rg3:process_resources[1955] set +a
rg3:process_resources[1957] [ 0 -ne 0 ]
rg3:process_resources[2256] break
rg3:process_resources[2267] [[ FALSE = TRUE ]]
rg3:process_resources[2273] exit 0
:rg_move_complete[346] STATUS=0
:rg_move_complete[348] exit 0
Mar 27 18:02:10 EVENT COMPLETED: rg_move_complete a1 2 0
```

Managing a node's PowerHA SystemMirror log file parameters

Each cluster node supports two log file parameters.

These allow you to:

- Set the level of debug information output by the PowerHA SystemMirror scripts. By default, PowerHA SystemMirror sets the debug information parameter to high, which produces detailed output from script execution.
- Set the output format for the **hacmp.out** log file.

To change the log file parameters for a node:

1. Enter `smit hacmp`
2. In SMIT, select **Problem Determination Tools > PowerHA SystemMirror Log Viewing and Management > Change/Show PowerHA SystemMirror Log File Parameters** and press Enter.
3. Select a node from the list.
4. Enter field values as follows:

Table 12. Change/Show PowerHA SystemMirror Log File Parameters fields	
Field	**Value**
Debug Level	Cluster event scripts have two levels of logging. The **low** level only logs events and errors encountered while the script executes. The **high** (default) level logs all commands performed by the script and is strongly recommended. The **high** level provides the level of script tracing needed to resolve many cluster problems.
Formatting options for hacmp.out	**Select one of these: Default (None)** (no special format), **Standard** (include search strings), **HTML (Low)** (limited HTML formatting), or **HTML (High)** (full HTML format).

5. Press Enter to add the values into the PowerHA SystemMirror for AIX Configuration Database.

6. Return to the main PowerHA SystemMirror emenu. Select **Extended Configuration > Extended Verification and Synchronization**.

   The software checks whether cluster services are running on any cluster node. If so, there will be no option to skip verification.
7. Select the options you want to use for verification and Press Enter to synchronize the cluster configuration and node environment across the cluster.

Related information

Verifying and synchronizing a PowerHA SystemMirror cluster

Logging for clcomd

Logging for the **clcomd** daemon to **clcomd.log** and **clcomddiag.log** is turned on by default.

The information in **clcomd.log** provides information about all connections to and from the daemon, including information for the initial connections established during discovery. Because **clcomddiag.log** contains diagnostic information for the daemon, you usually do not use this file in troubleshooting situations.

The following example shows the type of output generated in the **clcomd.log** file. The second and third entries are generated during the discovery process.

```
Wed May 7 12:43:13 2003: Daemon was successfully started
Wed May 7 12:44:10 2003: Trying to establish connection to node
 temporarynode0000001439363040
Wed May 7 12:44:10 2003: Trying to establish connection to node
 temporarynode0000002020023310
Wed May 7 12:44:10 2003: Connection to node temporarynode0000002020023310, success, 192.0.24.4->
Wed May 7 12:44:10 2003: CONNECTION: ACCEPTED: test2: 192.0.24.4->192.0.24.4
Wed May 7 12:44:10 2003: WARNING: /usr/es/sbin/cluster/etc/rhosts permissions
 must be -rw-------
Wed May 7 12:44:10 2003: Connection to node temporarynode0000001439363040: closed
Wed May 7 12:44:10 2003: Connection to node temporarynode0000002020023310: closed
Wed May 7 12:44:10 2003: CONNECTION: CLOSED: test2: 192.0.24.4->192.0.24.4
Wed May 7 12:44:11 2003: Trying to establish connection to node test1
Wed May 7 12:44:11 2003: Connection to node test1, success, 192.0.24.4->192.0.24.5
Wed May 7 12:44:11 2003: Trying to establish connection to node test3.
```

You can view the content of the **clcomd.log** or **clcomddiag.log** file by using the AIX **vi** or **more** commands.

You can turn off logging to **clcomddiag.log** temporarily (until the next reboot, or until you enable logging for this component again) by using the AIX **tracesoff** command. To permanently stop logging to **clcomddiag.log**, start the daemon from SRC without the **-d** flag by using the following command:

```
chssys -s clcomd -a ""
```

Redirecting PowerHA SystemMirror cluster log files

During normal operation, PowerHA SystemMirror produces several output log files that you can use to monitor and debug your systems. You can store a cluster log in a location other than its default directory if you so choose. If you do this, keep in mind that the minimum disk space for most cluster logs is 2MB. 14MB is recommended for **hacmp.out**.

Note: Logs should be redirected to local file systems and not to shared or NFS file systems. Having logs on those file systems may cause problems if the file system needs to unmount during a fallover event. Redirecting logs to NFS file systems may also prevent cluster services from starting during node reintegration.

The log file redirection function does the following:

- Checks the location of the target directory to determine whether it is part of a local or remote file system.
- Performs a check to determine whether the target directory is managed by PowerHA SystemMirror. If it is, any attempt to redirect a log file will fail.

- Checks to ensure that the target directory is specified using an absolute path (such as "/mylogdir") as opposed to a relative path (such as "mylogdir").

These checks decrease the possibility that the chosen file system may become unexpectedly unavailable.

Note: The target directory must have read-write access.

System components

These topics guide you through the steps to investigate system components, identify problems that you may encounter as you use PowerHA SystemMirror, and offer possible solutions.

If no error messages are displayed on the console and if examining the log files proves fruitless, you next investigate each component of your PowerHA SystemMirror environment and eliminate it as the cause of the problem.

Investigating system components

Both PowerHA SystemMirror and AIX provide utilities you can use to determine the state of a PowerHA SystemMirror cluster and the resources within that cluster. Using these commands, you can gather information about volume groups or networks.

Your knowledge of the PowerHA SystemMirror system is essential. You must know the characteristics of a normal cluster beforehand and be on the lookout for deviations from the norm as you examine the cluster components. Often, the surviving cluster nodes can provide an example of the correct setting for a system parameter or for other cluster configuration information.

You should review the PowerHA SystemMirror cluster components that you can check and describes some useful utilities. If examining the cluster log files does not reveal the source of a problem, investigate each system component to move through the layers. You should investigate the components in the following order:

1. Application layer
2. PowerHA SystemMirror layer
3. Logical Volume Manager layer
4. TCP/IP layer
5. AIX layer
6. Physical network layer
7. Physical disk layer
8. System hardware layer

You should also know what to look for when examining each layer and know the tools you should use to examine the layers.

Checking highly available applications

As a first step to finding problems affecting a cluster, check each highly available application running on the cluster. Examine any application-specific log files and perform any troubleshooting procedures recommended in the application's documentation.

In addition, check the following:

- Do some simple tests; for example, for a database application try to add and delete a record.
- Use the **ps** command to check that the necessary processes are running, or to verify that the processes were stopped properly.
- Check the resources that the application expects to be present to ensure that they are available, the file systems and volume groups for example.

Checking the PowerHA SystemMirror layer

If checking the application layer does not reveal the source of a problem, check the PowerHA SystemMirror layer.

The two main areas to investigate are:

- PowerHA SystemMirror components and required files
- Cluster topology and configuration.

Note: These steps assume that you have checked the log files and that they do not point to the problem.

Checking PowerHA SystemMirror components

A PowerHA SystemMirror cluster is made up of several required files and daemons. The following sections describe what to check for in the PowerHA SystemMirror layer.

Checking PowerHA SystemMirror required files

Make sure that the PowerHA SystemMirror files required for your cluster are in the proper place, have the proper permissions (readable and executable), and are not zero length. The PowerHA SystemMirror files and the AIX files modified by the PowerHA SystemMirror software are listed in the README file that accompanies the product.

Checking cluster services and processes

Check the status of the following PowerHA SystemMirror daemons:

- The Cluster Manager (**clstrmgrES**) daemon
- The Cluster Communications (**clcomdES**) daemon
- The Cluster Information Program (**clinfoES**) daemon.

When these components are not responding normally, determine if the daemons are active on a cluster node. Use either the options on the SMIT **System Management (C-SPOC)Cluster ServicesShow Cluster Services** panel or the **lssrc** command.

For example, to check on the status of all daemons under the control of the SRC, enter:

```
lssrc -a | grep active
 syslogdras 290990active
 sendmail  mail270484active
 portmapportmap286868active
 inetd  tcpip  295106active
 snmpd  tcpip  303260active
 dpid2  tcpip  299162active
 hostmibd  tcpip  282812active
 aixmibdtcpip  278670active
 biodnfs 192646active
 rpc.statd nfs 254122 active
 rpc.lockd nfs 274584active
 qdaemonspooler196720active
 writesrv  spooler250020active
 ctrmc  rsct98392 active
 clcomdES  clcomdES  204920active
 IBM.CSMAgentRMrsct_rm90268 active
 IBM.ServiceRM rsct_rm229510active
 IBM.ERRM  rsct_rm188602active
 IBM.AuditRMrsct_rm151722active
 topsvcstopsvcs602292active
 grpsvcsgrpsvcs569376active
 emsvcs emsvcs 561188active
 emaixosemsvcs 557102active
 clstrmgrEScluster544802active
 gsclvmd565356active
 IBM.HostRMrsct_rm442380active
```

To check on the status of all cluster daemons under the control of the SRC, enter: `lssrc -g cluster`

Note: When you use the **-g** flag with the **lssrc** command, the status information does not include the status of subsystems if they are inactive. If you need this information, use the **-a** flag instead. For more information on the **lssrc** command, see the man page.

To view additional information on the status of a daemon run the **clcheck_server** command. The **clcheck_server** command makes additional checks and retries beyond what is done by **lssrc** command. For more information, see the **clcheck_server** man page.

To determine whether the Cluster Manager is running, or if processes started by the Cluster Manager are currently running on a node, use the **ps** command.

For example, to determine whether the **clstrmgrES** daemon is running, enter:

```
ps -ef | grep clstrmgrES
root  18363  3346  3 11:02:05 -  10:20 /usr/es/sbin/cluster/clstrmgrES
root  19028  19559 2 16:20:04 pts/10  0:00 grep clstrmgrES
```

See the **ps** man page for more information about using this command.

Checking for cluster configuration problems

For a PowerHA SystemMirror cluster to function properly, all the nodes in the cluster must agree on the cluster topology, network configuration, and ownership and takeover of PowerHA SystemMirror resources. This information is stored in the Configuration Database on each cluster node.

To begin checking for configuration problems, ask yourself if you (or others) have made any recent changes that may have disrupted the system. Have components been added or deleted? Has new software been loaded on the machine? Have new PTFs or application updates been performed? Has a system backup been restored? Then run verification to ensure that the properPowerHA SystemMirror-specific modifications to AIX software are in place and that the cluster configuration is valid.

The cluster verification utility checks many aspects of a cluster configuration and reports any inconsistencies. Using this utility, you can perform the following tasks:

• Verify that all cluster nodes contain the same cluster topology information
• Check that all network interface cards are properly configured, and that shared disks are accessible to all nodes that can own them
• Check for agreement among all nodes on the ownership of defined resources, such as file systems, log files, volume groups, disks, and application controllers
• Check for invalid characters in cluster names, node names, network names, network interface names and resource group names
• Verify takeover information.

The verification utility will also print out diagnostic information about the following:

• Custom snapshot methods
• Custom verification methods
• Custom pre or post events
• Cluster log file redirection.

From the main PowerHA SystemMirror SMIT panel, select **Problem Determination Tools > PowerHA SystemMirror Verification > Verify PowerHA SystemMirror Configuration**. If you find a configuration problem, correct it, then resynchronize the cluster.

Note: Some errors require that you make changes on each cluster node. For example, a missing application start script or a volume group with `autovaryon=TRUE` requires a correction on each affected node. Some of these issues can be taken care of by using PowerHA SystemMirror File Collections.

Run the **/usr/es/sbin/cluster/utilities/cltopinfo** command to see a complete listing of cluster topology. In addition to running the PowerHA SystemMirror verification process, check for recent modifications to the node configuration files.

The command `ls -lt /etc` lists all the files in the **/etc** directory and shows the most recently modified files that are important to configuring AIX, such as:

- **etc/inett.conf**
- **etc/hosts**
- **etc/services**

It is also very important to check the resource group configuration for any errors that may not be flagged by the verification process. For example, make sure the file systems required by the application controllers are included in the resource group with the application.

Check that the nodes in each resource group are the ones intended, and that the nodes are listed in the proper order. To view the cluster resource configuration information from the main PowerHA SystemMirror SMIT panel, select **Extended Configuration > Extended Resource Configuration > PowerHA SystemMirror Extended Resource Group Configuration > Show All Resources by Node or Resource Group**.

You can also run the **/usr/es/sbin/cluster/utilities/clRGinfo** command to see the resource group information.

Note: If cluster configuration problems arise after running the cluster verification utility, do not run C-SPOC commands in this environment as they may fail to execute on cluster nodes.

Related information
Verifying and synchronizing a PowerHA SystemMirror cluster

Checking a cluster snapshot file

The PowerHA SystemMirror cluster snapshot facility (**/usr/es/sbin/cluster/utilities/clsnapshots**) allows you to save in a file, a record all the data that defines a particular cluster configuration. It also allows you to create your own custom snapshot methods, to save additional information important to your configuration. You can use this snapshot for troubleshooting cluster problems.

The default directory path for storage and retrieval of a snapshot is **/usr/es/sbin/cluster/snapshots**.

Note that you cannot use the cluster snapshot facility in a cluster that is running different versions of PowerHA SystemMirror concurrently.

Related information
Saving and restoring cluster configurations

Information saved in a cluster snapshot
The primary information saved in a cluster snapshot is the data stored in the PowerHA SystemMirror Configuration Database classes (such as HACMPcluster, HACMPnode, and HACMPnetwork). This is the information used to recreate the cluster configuration when a cluster snapshot is applied.

The cluster snapshot does not save any user customized scripts, applications, or other configuration parameters that are not for PowerHA SystemMirror. For example, the name of an application controller and the location of its start and stop scripts are stored in the PowerHA SystemMirror server Configuration Database object class. However, the scripts themselves as well as any applications they may call are not saved.

The cluster snapshot does not save any device data or configuration-specific data that is outside the scope of PowerHA SystemMirror. For instance, the facility saves the names of shared file systems and volume groups; however, other details, such as NFS options or LVM mirroring configuration are not saved.

If you moved resource groups using the Resource Group Management utility **clRGmove**, once you apply a snapshot, the resource groups return to behaviors specified by their default nodelists. To investigate a cluster after a snapshot has been applied, run **clRGinfo** to view the locations and states of resource groups.

In addition to this Configuration Database data, a cluster snapshot also includes output generated by various PowerHA SystemMirror and standard AIX commands and utilities. This data includes the current

state of the cluster, node, network, and network interfaces as viewed by each cluster node, as well as the state of any running PowerHA SystemMirror daemons.

The cluster snapshot includes output from the following commands:

- cllscf
- df
- lsfs
- netstat
- cllsnw
- exportfs
- lslpp
- no
- cllsif
- ifconfig
- lslv
- clchsyncd
- clshowres
- ls
- lsvg
- cltopinfo

Skipping the logs collection reduces the size of the snapshot and speeds up running the snapshot utility.

You can use SMIT to collect cluster log files for problem reporting. This option is available under the **Problem Determination Tools** > **PowerHA SystemMirror Log Viewing and Management > Collect Cluster log files for Problem Reporting** SMIT menu. It is recommended to use this option only if requested by the IBM support personnel.

Note that you can also use the AIX **snap -e** command to collect PowerHA SystemMirror cluster data, including the **hacmp.out** and **clstrmgr.debug** log files.

Related information

Saving and restoring cluster configurations

Cluster snapshot files
The cluster snapshot facility stores the data it saves in two separate files, the Configuration Database data file and the Cluster State Information File, each displaying information in three sections.

Configuration Database Data File (.odm)
This file contains all the data stored in the PowerHA SystemMirror Configuration Database object classes for the cluster.

This file is given a user-defined basename with the **.odm** file extension. Because the Configuration Database information must be largely the same on every cluster node, the cluster snapshot saves the values from only one node. The cluster snapshot Configuration Database data file is an ASCII text file divided into three delimited sections:

Section	Description
Table 13. Database Data file (.odm) sections	
Section	Description
Version section	This section identifies the version of the cluster snapshot. The characters <VER identify the start of this section; the characters </VER identify the end of this section. The cluster snapshot software sets the version number.

Table 13. Database Data file (.odm) sections (continued)	
Section	**Description**
Description section	This section contains user-defined text that describes the cluster snapshot. You can specify up to 255 characters of descriptive text. The characters <DSC identify the start of this section; the characters </DSC identify the end of this section.
ODM data section	This section contains the PowerHA SystemMirror Configuration Database object classes in generic AIX ODM stanza format. The characters <ODM identify the start of this section; the characters </ODM identify the end of this section.

The following is an excerpt from a sample cluster snapshot Configuration Database data file showing some of the ODM stanzas that are saved:

```
<VER
1.0
</VER

<DSC
My Cluster Snapshot
</DSC

<ODM

PowerHA SystemMirror cluster:
 id = 1106245917
 name = "HA52_TestCluster"
 nodename = "mynode"
 sec_level = "Standard"
 sec_level_msg = ""
 sec_encryption = ""
 sec_persistent = ""
 last_node_ids = ""
 highest_node_id = 0
 last_network_ids = ""
 highest_network_id = 0
 last_site_ides = ""
 highest_site_id = 0
 handle = 1
 cluster_version = 7
 reserved1 = 0
 reserved2 = 0
 wlm_subdir = ""
 settling_time = o
 rg_distribution_policy = "node"
 noautoverification = 0
 clvernodename = ""
 clverhour = 0

PowerHA SystemMirror node:
 name = "mynode"
 object = "VERBOSE_LOGGING"
 value = "high"
.
.
</ODM
```

Cluster State Information File (.info)
This file contains the output from standard AIX and PowerHA SystemMirror system management commands.

This file is given the same user-defined basename with the **.info** file extension. If you defined custom snapshot methods, the output from them is appended to this file. The Cluster State Information file contains three sections:

Table 14. Cluster State information file (.info)	
Section	**Description**
Version section	This section identifies the version of the cluster snapshot. The characters <VER identify the start of this section; the characters </VER identify the end of this section. The cluster snapshot software sets this section.
Description section	This section contains user-defined text that describes the cluster snapshot. You can specify up to 255 characters of descriptive text. The characters <DSC identify the start of this section; the characters </DSC identify the end of this section.
Command output section	This section contains the output generated by AIX and PowerHA SystemMirror ODM commands. This section lists the commands executed and their associated output. This section is not delimited in any way.

Checking the logical volume manager

When troubleshooting a PowerHA SystemMirror cluster, you need to check the LVM entities for volume groups, physical and logical volumes, and file systems.

Checking volume group definitions

Check to make sure that all shared volume groups in the cluster are active on the correct node. If a volume group is not active, vary it on using the appropriate command for your configuration.

In the SMIT panel **Initialization and Standard Configuration > Configure PowerHA SystemMirror Resource Groups > Change/Show Resources for a Resource Group (standard)** , all volume groups listed in the **Volume Groups** field for a resource group should be varied on the node(s) that have the resource group online.

Using the lsvg command to check volume groups

To check for inconsistencies among volume group definitions on cluster nodes, use the **lsvg** command to display information about the volume groups defined on each node in the cluster:

```
lsvg
```

The system returns volume group information similar to the following:

```
rootvg
```

```
datavg
```

To list only the active (varied on) volume groups in the system, use the **lsvg -o** command as follows:

```
lsvg -o
```

The system returns volume group information similar to the following:

```
rootvg
```

To list all logical volumes in the volume group, and to check the volume group status and attributes, use the **lsvg -l** command and specify the volume group name as shown in the following example:

```
lsvg -l rootvg
```

Note: The volume group must be varied on to use the **lsvg-l** command.

You can also use PowerHA SystemMirror SMIT to check for inconsistencies: **System Management (C-SPOC) > PowerHA SystemMirror Logical Volume Management > Shared Volume Groups** option to display information about shared volume groups in your cluster.

Checking the varyon state of a volume group

You may check the status of the volume group by issuing the **lsvg <** *vgname* **>** command.

Depending on your configuration, the **lsvg** command returns the following options:

`vg state` could be `active` (if it is active varyon), or `passive only` (if it is passive varyon).

`vg mode` could be `concurrent` or `enhanced concurrent`.

Here is an example of **lsvg** output:

```
# lsvg myvg

VOLUME GROUP:  Volume_Group_01  VG IDENTIFIER: 0002231b00004c00000000f2801blcc3
VG STATE:  active PP SIZE: 16 megabyte(s)
VG PERMISSION:read/write TOTAL PPs: 1084 (17344 megabytes)
MAX LVs:256FREE PPs:977 (15632 megabytes)
LVs:4USED PPs:107 (1712 megabytes)
OPEN LVs:  0QUORUM:2
TOTAL PVs:2VG DESCRIPTORS:3
STALE PVs:0  STALE PPs  0
ACTIVE PVs:  2AUTO ON:no
MAX PPs per PV1016 MAX PVs: 32
LTG size:  128 kilobyte (s)  AUTO SYNC:no
HOT SPARE:no
```

Using the C-SPOC utility to check shared volume groups

To check for inconsistencies among volume group definitions on cluster nodes in a two-node C-SPOC environment:

1. Enter `smitty hacmp`
2. In SMIT, select **System Management (C-SPOC) > PowerHA SystemMirror Logical Volume Management > Shared Volume Groups > List All Shared Volume Groups** and press Enter to accept the default (**no**).

A list of all shared volume groups in the C-SPOC environment appears. This list also contains enhanced concurrent volume groups included as resources in non-concurrent resource groups.

You can also use the C-SPOC **cl_lsvg** command from the command line to display this information.

Checking physical volumes

To check for discrepancies in the physical volumes defined on each node, obtain a list of all physical volumes known to the systems and compare this list against the list of disks specified in the **Disks** field of the **Command Status** panel. Access the **Command Status** panel through the SMIT **Extended Configuration > Extended Resource Configuration > PowerHA SystemMirror Extended Resource Group Configuration > Show All Resources by Node or Resource Group** panel.

To obtain a list of all the physical volumes known to a node and to find out the volume groups to which they belong, use the **lspv** command. If you do not specify the name of a volume group as an argument, the **lspv** command displays every known physical volume in the system. For example:

```
lspv
hdisk00000914312e971arootvg
hdisk100000132a78e213rootvg
hdisk200000902a78e21adatavg
hdisk300000321358e354datavg
```

The first column of the display shows the logical name of the disk. The second column lists the physical volume identifier of the disk. The third column lists the volume group (if any) to which it belongs.

Note that on each cluster node, AIX can assign different names (hdisk numbers) to the same physical volume. To tell which names correspond to the same physical volume, compare the physical volume identifiers listed on each node.

If you specify the logical device name of a physical volume (hdisk*x*) as an argument to the **lspv** command, it displays information about the physical volume, including whether it is active (varied on). For example:

```
lspv hdisk2
PHYSICAL VOLUME:hdisk2 VOLUME GROUP:abalonevg
PV IDENTIFIER:  0000301919439ba5 VG IDENTIFIER: 00003019460f63c7
PV STATE:active VG STATE:active/complete
STALE PARTITIONS:  0  ALLOCATABLE:yes
PP SIZE: 4 megabyte(s)LOGICAL VOLUMES:2
TOTAL PPs:  203 (812 megabytes) VG DESCRIPTORS: 2
FREE PPs:192 (768 megabytes)
USED PPs:11 (44 megabytes)
FREE DISTRIBUTION: 41..30..40..40..41
USED DISTRIBUTION:00..11..00..00..00
```

If a physical volume is inactive (not varied on, as indicated by question marks in the **PV STATE** field), use the appropriate command for your configuration to vary on the volume group containing the physical volume. Before doing so, however, you may want to check the system error report to determine whether a disk problem exists. Enter the following command to check the system error report:

```
errpt -a|more
```

You can also use the **lsdev** command to check the availability or status of all physical volumes known to the system.

Checking logical volumes

To check the state of logical volumes defined on the physical volumes, use the **lspv -l** command and specify the logical name of the disk to be checked.

As shown in the following example, you can use this command to determine the names of the logical volumes defined on a physical volume:

```
lspv -l hdisk2
LV NAMELPs PPs DISTRIBUTIONMOUNT POINT
lv02 50  50  25..00..00..00..25/usr
lv04 44  44  06..00..00..32..06/clusterfs
```

Use the **lslv** *logicalvolume* command to display information about the state (opened or closed) of a specific logical volume, as indicated in the **LV STATE** field. For example:

```
lslv nodeAlv

LOGICAL VOLUME: nodeAlv   VOLUME GROUP:nodeAvg
LV IDENTIFIER:  00003019460f63c7.1PERMISSION:  read/write
VG STATE:active/complete  LV STATE:opened/syncd
TYPE: jfs  WRITE VERIFY:off
MAX LPs: 128  PP SIZE: 4 megabyte(s)
COPIES:  1  SCHED POLICY:parallel
LPs:  10PPs:  10
STALE PPs:0  BB POLICY:relocatable
INTER-POLICY:minimum  RELOCATABLE: yes
INTRA-POLICY:middleUPPER BOUND: 32
MOUNT POINT: /nodeAfsLABEL:/nodeAfs
MIRROR WRITE CONSISTENCY: on
EACH LP COPY ON A SEPARATE PV ?: yes
```

If a logical volume state is inactive (or closed, as indicated in the **LV STATE** field), use the appropriate command for your configuration to vary on the volume group containing the logical volume.

Using the C-SPOC utility to check shared logical volumes

To check the state of shared logical volumes on cluster nodes:

In SMIT select **System Management (C-SPOC) > PowerHA SystemMirror Logical Volume Management > Shared Logical Volumes > List All Shared Logical Volumes by Volume Group** . A list of all shared logical volumes appears.

You can also use the C-SPOC **cl_lslv** command from the command line to display this information.

Checking file systems

Check to see if the necessary file systems are mounted and where they are mounted. Compare this information against the PowerHA SystemMirror definitions for any differences. Check the permissions of the file systems and the amount of space available on a file system.

Use the following commands to obtain this information about file systems:

- The **mount** command
- The **df** command
- The **lsfs** command.

Use the **cl_lsfs** command to list file system information when running the C-SPOC utility.

Obtaining a list of file systems

Use the **mount** command to list all the file systems, both JFS and NFS, currently mounted on a system and their mount points.

For example:

```
mount

node mountedmounted over vfs date  options
--------------------------------------------------------------------------
/dev/hd4 /  jfs Oct 06 09:48  rw,log=/dev/hd8
/dev/hd2 /usr  jfs Oct 06 09:48  rw,log=/dev/hd8
/dev/hd9var /var  jfs Oct 06 09:48  rw,log=/dev/hd8
/dev/hd3 /tmp  jfs Oct 06 09:49  rw,log=/dev/hd8
/dev/hd1 /home jfs Oct 06 09:50  rw,log=/dev/hd8
pearl /home/home nfs Oct 07 09:59  rw,soft,bg,intr
jade  /usr/local  /usr/localnfs Oct 07 09:59  rw,soft,bg,intr
```

Determine whether and where the file system is mounted, then compare this information against the PowerHA SystemMirror definitions to note any differences.

Checking available file system space

To see the space available on a file system, use the **df** command.

For example:

```
df

File System Total KB free %usediused %iused  Mounted on
/dev/hd4  12288 530856%  896 21%/
/dev/hd2 4136962676893%19179 18%/usr
/dev/hd9var8192 373654%  115  5%/var
/dev/hd38192 7576 7%72  3%/tmp
/dev/hd14096 3932 4%17  1%/home
/dev/crab1lv8192 7904 3%17  0%/crab1fs
/dev/crab3lv  1228811744 4%16  0%/crab3fs
/dev/crab4lv  1638415156 7%17  0%/crab4fs
/dev/crablv4096 325220%17  1%/crabfs
```

Check the **%used** column for file systems that are using more than 90% of their available space. Then check the **free** column to determine the exact amount of free space remaining.

Checking mount points, permissions, and file system information

Use the **lsfs** command to display information about mount points, permissions, file system size and so on.

For example:

```
lsfs

Name Nodename Mount PtVFS  Size  Options Auto
/dev/hd4 --/jfs  24576 --  yes
/dev/hd1 --/homejfs  8192 --  yes
/dev/hd2 --/usrjfs 827392  --  yes
/dev/hd9var--/varjfs  16384--  yes
/dev/hd3 -- /tmp  jfs  16384--  yes
/dev/hd7 --/mntjfs  -- --  no
/dev/hd5 --/blvjfs  -- --  no
/dev/crab1lv --/crab1fsjfs  16384rw  no
/dev/crab3lv --/crab3fsjfs  24576rw  no
/dev/crab4lv --/crab4fsjfs  32768rw  no
/dev/crablv-- /crabfs  jfs  8192  rw  no
```

Important: For file systems to be NFS exported, be sure to verify that logical volume names for these file systems are consistent throughout the cluster.

Using the C-SPOC utility to check shared file systems
Check to see whether the necessary shared file systems are mounted and where they are mounted on cluster nodes in a two-node C-SPOC environment.

In SMIT select **System Management (C-SPOC) > PowerHA SystemMirror Logical Volume Management > Shared Filesystems** . Select from either **Journaled Filesystems > List All Shared Filesystems** or **Enhanced Journaled Filesystems > List All Shared Filesystems** to display a list of shared file systems.

You can also use the C-SPOC **cl_lsfs** command from the command line to display this information.

Checking the automount attribute of file systems
At boot time, AIX attempts to check all the file systems listed in **/etc/filesystems** with the **check=true** attribute by running the **fsck** command.

If AIX cannot check a file system, it reports the following error:

```
Filesystem helper: 0506-519 Device open failed
```

For file systems controlled by PowerHA SystemMirror, this error message typically does not indicate a problem. The file system check fails because the volume group on which the file system is defined is not varied on at boot time.

To avoid generating this message, edit the **/etc/filesystems** file to ensure that the stanzas for the shared file systems do not include the **check=true** attribute.

Checking the TCP/IP subsystem

You can investigate the TCP/IP subsystem using AIX commands.

These commands include the following:

- Use the **netstat** command to make sure that the network interfaces are initialized and that a communication path exists between the local node and the target node.
- Use the **ping** command to check the point-to-point connectivity between nodes.
- Use the **ifconfig** command on all network interfaces to detect bad IP addresses, incorrect subnet masks, and improper broadcast addresses.
- Scan the **/var/hacmp/log/hacmp.out** file to confirm that the **/etc/rc.net** script has run successfully. Look for a zero exit status.
- If IP address takeover is enabled, confirm that the **/etc/rc.net** script has run and that the service interface is on its service address and not on its base (boot) address.
- Use the **lssrc -g tcpip** command to make sure that the **inetd** daemon is running.
- Use the **lssrc -g portmap** command to make sure that the **portmapper** daemon is running.
- Use the **arp** command to make sure that the cluster nodes are not using the same IP or hardware address.

- Use the **netstat** command to:
  - Show the status of the network interfaces defined for a node.
  - Determine whether a route from the local node to the target node is defined.

    The **netstat -in** command displays a list of all initialized interfaces for the node, along with the network to which that interface connects and its IP address. You can use this command to determine whether the service and boot interfaces are on separate subnets. The subnets are displayed in the **Network** column.

```
netstat -in

Name Mtu NetworkAddress  IpktsIerrs  OpktsOerrsColl
lo0 1536 <Link> 18406   0 18406   00
lo0 1536 127 127.0.0.118406   0 18406   00
en1 1500 <Link> 11116260 58643   00
en1 1500 100.100.86.100.100.86.136   11116260 58643   00
en0 1500 <Link> 943656 0 52208   00
en0 1500 100.100.83.100.100.83.136   943656 0 52208   00
tr1 1492 <Link> 18790 165600
tr1 1492 100.100.84.100.100.84.136   18790 165600
```

Look at the first, third, and fourth columns of the output. The **Name** column lists all the interfaces defined and available on this node. Note that an asterisk preceding a name indicates the interface is down (not ready for use). The **Network** column identifies the network to which the interface is connected (its subnet). The **Address** column identifies the IP address assigned to the node.

The **netstat -rn** command indicates whether a route to the target node is defined. To see all the defined routes, enter:

```
netstat -rn
```

Information similar to that shown in the following example is displayed:

```
Routing tables
DestinationGateway   Flags   Refcnt UseInterface
Netmasks:
(root node)
(0)0
(0)0 ff00 0
(0)0 ffff 0
(0)0 ffff ff80 0
(0)0 70 204 1 0
(root node)Route Tree for Protocol Family 2:
(root node)
127 127.0.0.1U 3   1436   lo0
127.0.0.1 127.0.0.1UH0456   lo0
100.100.83.128100.100.83.136   U 6 18243   en0
100.100.84.128100.100.84.136   U 1   1718   tr1
100.100.85.128100.100.85.136   U 2   1721   tr0
100.100.86.128100.100.86.136   U 8 21648   en1
100.100.100.128   100.100.100.136 U 039   en0
(root node)Route Tree for Protocol Family 6:
(root node)
(root node)
```

To test for a specific route to a network (for example 100.100.83), enter:

```
netstat -nr | grep '100\.100\.83'

100.100.83.128100.100.83.136   U 6 18243   en0
```

The same test, run on a system that does not have this route in its routing table, returns no response. If the service and boot interfaces are separated by a bridge, router, or hub and you experience problems communicating with network devices, the devices may not be set to handle two network segments as one physical network. Try testing the devices independent of the configuration, or contact your system administrator for assistance.

Note that if you have only one interface active on a network, the Cluster Manager will not generate a failure event for that interface.

See the **netstat** man page for more information on using this command.

Related information

Network interface events

Checking point-to-point connectivity

The **ping** command tests the point-to-point connectivity between two nodes in a cluster. Use the **ping** command to determine whether the target node is attached to the network and whether the network connections between the nodes are reliable.

Be sure to test all TCP/IP interfaces configured on the nodes (service and boot).

For example, to test the connection from a local node to a remote node named *nodeA* enter:

```
/etc/ping nodeA

PING testcluster.nodeA.com: (100.100.81.141): 56 data bytes
64 bytes from 100.100.81.141: icmp_seq=0 ttl=255 time=2 ms
64 bytes from 100.100.81.141: icmp_seq=1 ttl=255 time=1 ms
64 bytes from 100.100.81.141: icmp_seq=2 ttl=255 time=2 ms
64 bytes from 100.100.81.141: icmp_seq=3 ttl=255 time=2 ms
```

Type `Control-C` to end the display of packets. The following statistics appear:

```
----testcluster.nodeA.com PING Statistics----
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 1/1/2 ms
```

The **ping** command sends packets to the specified node, requesting a response. If a correct response arrives, **ping** prints a message similar to the output shown earlier indicating no lost packets. This indicates a valid connection between the nodes.

If the **ping** command hangs, it indicates that there is no valid path between the node issuing the **ping** command and the node you are trying to reach. It could also indicate that required TCP/IP daemons are not running. Check the physical connection between the two nodes. Use the **ifconfig** and **netstat** commands to check the configuration. A "bad value" message indicates problems with the IP addresses or subnet definitions.

Note that if "DUP!" appears at the end of the **ping** response, it means the **ping** command has received multiple responses for the same address. This response typically occurs when network interfaces have been misconfigured, or when a cluster event fails during IP address takeover. Check the configuration of all interfaces on the subnet to verify that there is only one interface per address. For more information, see the **ping** man page.

In addition, you can assign a *persistent node IP label* to a cluster network on a node. When for administrative purposes you wish to reach a specific node in the cluster using the **ping** or **telnet** commands without worrying whether an service IP label you are using belongs to any of the resource groups present on that node, it is convenient to use a persistent node IP label defined on that node.

Related information

Planning PowerHA SystemMirror

Configuring PowerHA SystemMirror cluster topology and resources (extended)

Checking the IP address and netmask

Use the **ifconfig** command to confirm that the IP address and netmask are correct. Invoke **ifconfig** with the name of the network interface that you want to examine.

For example, to check the first Ethernet interface, enter:

```
ifconfig en0

en0: flags=2000063<UP,BROADCAST,NOTRAILERS,RUNNING,NOECHO>
   inet 100.100.83.136 netmask 0xffffff00 broadcast 100.100.83.255
```

```
    inet6 fe80::214:5eff:fe4d:6045/64
    tcp_sendspace 131072 tcp_recvspace 65536 rfc1323 0
```

If the specified interface does not exist, **ifconfig** replies:

```
No such device
```

The **ifconfig** command displays multiple lines of output. The first line shows the interface's name and characteristics. Check for these characteristics:

Field	Value
Table 15. ifconfig command output	
UP	The interface is ready for use. If the interface is down, use the **ifconfig** command to initialize it. For example: `  ifconfig en0 up` If the interface does not come up, replace the interface cable and try again. If it still fails, use the **diag** command to check the device.
RUNNING	The interface is working. If the interface is not running, the driver for this interface may not be properly installed, or the interface is not properly configured. Review all the steps necessary to install this interface, looking for errors or missed steps. **Note:** The **ifconfig** command displays only the configuration and not the functional state of the adapter. It is possible that the configured state might be *UP* but some other problem is preventing communications.

The remaining output from the **ifconfig** command includes information for each address configured on the interface. Check these fields to make sure the network interface is properly configured.

See the **ifconfig** man page for more information.

Using the arp command

Use the **arp** command to view what is currently held to be the IP addresses associated with nodes listed in a host's arp cache. For example:

```
arp -a

flounder (100.50.81.133) at 8:0:4c:0:12:34 [ethernet]
cod (100.50.81.195) at 8:0:5a:7a:2c:85 [ethernet]
pollock (100.50.81.147) at 10:0:5a:5c:36:b9 [ethernet]
```

This output shows what the host node currently believes to be the IP and MAC addresses for nodes flounder, cod, seahorse and pollock. (If IP address takeover occurs without Hardware Address Takeover, the MAC address associated with the IP address in the host's arp cache may become outdated. You can correct this situation by refreshing the host's arp cache.)

See the **arp** man page for more information.

Checking the AIX operating system

To view hardware and software errors that may affect the cluster, use the **errpt** command.

Be on the lookout for disk and network error messages, especially permanent ones, which indicate real failures. See the **errpt** man page for more information.

Checking physical networks

You should check your physical networks and connections.

Checkpoints for investigating physical connections include:

- Check the serial line between each pair of nodes.
- If you are using Ethernet:
  - Use the **diag** command to verify that the network interface card and cables good.
  - Ethernet adapters for the IBM System p can be used either with the transceiver that is on the card or with an external transceiver. There is a jumper on the NIC to specify which you are using. Verify that your jumper is set correctly.
  - Make sure that hub lights are on for every connected cable.

Related information
Planning cluster network connectivity

Checking disks and disk adapters

Use the **diag** command to verify that the adapter card is functioning properly. If problems arise, be sure to check the jumpers, cables, and terminators along the SCSI bus.

For SCSI disks, including IBM SCSI disks and arrays, make sure that each array controller, adapter, and physical disk on the SCSI bus has a unique SCSI ID. Each SCSI ID on the bus must be an integer value from 0 through 15, although some SCSI adapters may have limitations on the SCSI ID that can be set. See the device documentation for information about any device-specific limitations. A common configuration is to set the SCSI ID of the adapters on the nodes to be higher than the SCSI IDs of the shared devices. Devices with higher IDs take precedence in SCSI bus contention.

For example, if the standard SCSI adapters use IDs 5 and 6, assign values from 0 through 4 to the other devices on the bus. You may want to set the SCSI IDs of the adapters to 5 and 6 to avoid a possible conflict when booting one of the systems in service mode from a **mksysb** tape of other boot devices, since this will always use an ID of 7 as the default.

If the SCSI adapters use IDs of 14 and 15, assign values from 3 through 13 to the other devices on the bus.

You can check the SCSI IDs of adapters and disks using either the **lsattr** or **lsdev** command. For example, to determine the SCSI ID of the adapter *scsi1* (SCSI-3), use the following **lsattr** command and specify the logical name of the adapter as an argument:

```
lsattr -E -l scsi1 | grep id
```

Do not use wildcard characters or full pathnames on the command line for the device name designation.

Important: If you restore a backup of your cluster configuration onto an existing system, be sure to recheck or reset the SCSI IDs to avoid possible SCSI ID conflicts on the shared bus. Restoring a system backup causes adapter SCSI IDs to be reset to the default SCSI ID of 7.

If you note a SCSI ID conflict, see the *Planning Guide* for information about setting the SCSI IDs on disks and disk adapters.

To determine the SCSI ID of a disk, enter:

```
lsdev -Cc disk -H
```

Related information
Planning PowerHA SystemMirror

Recovering from PCI hot plug NIC failure

If an unrecoverable error causes a PCI hot-replacement process to fail, you may be abandoned in a state where your NIC is unconfigured and still in maintenance mode. The PCI slot holding the card and/or the new card may be damaged at this point. User intervention is required to get the node back in fully working order.

For more information, refer to your hardware manuals or search for information about devices on IBM's website.

Checking the cluster communications daemon

In some cases, if you change or remove IP addresses in the AIX adapter configuration, and this takes place after the cluster has been synchronized, the Cluster Communications daemon cannot validate these addresses against the /etc/cluster/rhosts file or against the entries in the PowerHA SystemMirror Configuration Database. When this problem occurs, you may see one or more errors from PowerHA SystemMirror while working with the configuration or during verification and synchronization.

Or, you may obtain an error during the cluster synchronization.

In this case, you must update the information that is saved in the **/etc/cluster/rhosts** file on all cluster nodes, and refresh the **clcomd** command to make it aware of the changes. When you synchronize and verify the cluster again, the **clcomd** command starts using IP addresses added to the PowerHA SystemMirror Configuration Database.

To refresh the Cluster Communications daemon, use:

```
refresh -s clcomd
```

Also, configure the **/etc/cluster/rhosts** file to contain all the addresses currently used by PowerHA SystemMirror for inter-node communication, and then copy this file to all cluster nodes. The **/etc/cluster/rhosts** file can contain IPv4 and IPv6 addresses.

Related concepts
Cluster communications issues
This section describes solutions for potential cluster communication issues.

Checking system hardware

Check the power supplies and LED displays to see if any error codes are displayed. Run the AIX **diag** command to test the system unit.

Without an argument, **diag** runs as a menu-driven program. You can also run **diag** on a specific piece of hardware. For example:

```
diag -d hdisk0 -c

Starting diagnostics.
Ending diagnostics.
```

This output indicates that hdisk0 is okay.

PowerHA SystemMirror installation issues

These topics describe some potential installation issues.

Cannot find file system at boot time

This topic discusses what happens when AIX cannot find a file system at boot time.

Problem

At boot time, AIX tries to check, by running the **fsck** command, all the file systems listed in **/etc/filesystems** with the **check=true** attribute. If it cannot check a file system. AIX reports an error. The system displays the following:

```
  +------------------------------------------------------------+
Filesystem Helper: 0506-519 Device open failed
  +------------------------------------------------------------+
```

Solution

For file systems controlled by PowerHA SystemMirror, this error typically does not indicate a problem. The file system check failed because the volume group on which the file system is defined is not varied on at boot-time. To prevent the generation of this message, edit the **/etc/filesystems** file to ensure that the stanzas for the shared file systems do not include the **check=true** attribute.

cl_convert does not run due to failed installation

This topic discusses what happens when **cl_convert** does not run due to a failed installation.

Problem

When you install PowerHA SystemMirror, **cl_convert** is run automatically. The software checks for an existing PowerHA SystemMirror configuration and attempts to convert that configuration to the format used by the version of the software bring installed. However, if installation fails, **cl_convert** will fail to run as a result. Therefore, conversion from the Configuration Database of a previous PowerHA SystemMirror version to the Configuration Database of the current version will also fail.

Solution

Run **cl_convert** from the command line. To gauge conversion success, refer to the **clconvert.log** file, which logs conversion progress.

Root user privilege is required to run **cl_convert**.

⚠️ **CAUTION:** Before converting be sure that your ODMDIR environment variable is set to **/etc/es/objrepos**.

For information on **cl_convert** flags, refer to the **cl_convert** man page.

Configuration files could not be merged during installation

This topic discusses configuration file problems during installation.

Problem

During the installation of PowerHA SystemMirror client software, the following message appears:

```
  +------------------------------------------------------------+
Post-installation Processing...
  +------------------------------------------------------------+
Some configuration files could not be automatically merged into
the system during the installation. The previous versions of these files
```

```
have been saved in a configuration directory as listed here. Compare
the saved files and the newly installed files to determine if you need
to recover configuration data. Consult product documentation
to determine how to merge the data.
Configuration files, which were saved in /usr/lpp/save.config:
/usr/es/sbin/cluster/utilities/clexit.rc
```

Solution

As part of the PowerHA SystemMirror installation process, copies of PowerHA SystemMirror files that could potentially contain site-specific modifications are saved in the **/usr/lpp/save.config** directory before they are overwritten. As the message states, you must merge site-specific configuration information into the newly installed files.

Troubleshooting PowerHA SystemMirror and Tivoli System Automation for Multiplatform

You cannot install Tivoli® System Automation for Multiplatform on the same node that already has PowerHA SystemMirror Version 7.1.0, or later, installed.

PowerHA SystemMirror 7.1.0, or later, is built upon the Cluster Aware AIX (CAA) capabilities. Tivoli System Automation for Multiplatform is built upon Reliable Scalable Cluster Technology (RSCT) capabilities. Therefore, you cannot use PowerHA SystemMirror and Tivoli System Automation for Multiplatform on the same node because they are built upon different clustering capabilities.

Solving common problems

The following topics describe some common problems and recommendations.

PowerHA SystemMirror startup issues

This section describes solutions for potential PowerHA SystemMirror startup issues.

- ODMPATH environment variable not set correctly
- clinfo daemon exits after starting
- Node powers off after cluster services are started
- configchk command returns an unknown host message
- Cluster Manager does not respond during reconfiguration
- clcomd and clstrmgrES daemons fail to start on newly installed AIX nodes
- Pre-event or post-event does not exist on a node after upgrading the node
- Node fails during configuration with 869 LED display
- Node cannot rejoin cluster after being dynamically removed
- Resource group migration is not persistent after cluster startup
- Cluster does not startup after upgrade to PowerHA SystemMirror
- Cluster verification errors when nodes have different fileset levels

ODMPATH environment variable not set correctly

Problem
    Queried object not found.

Solution

    PowerHA SystemMirror has a dependency on the location of certain Object Data Model (ODM) repositories to store configuration data. The ODMPATH environment variable allows ODM commands and subroutines to query locations other than the default location if the queried object is not present

in the default location. You can set this variable, but it must include the default location, `/etc/objrepos`, or the integrity of configuration information might be lost.

clinfo daemon exits after starting

Problem

The "smux-connect" error occurs after you start the **clinfoES** daemon with the **-a** option. Another process is using port 162 to receive traps.

Solution

Check to see whether another process, such as the **trapgend** smux subagent of NetView® for AIX or the System Monitor for AIX **sysmond** daemon, is using port 162. If port 162 is used, restart the **clinfoES** daemon without the **-a** option and configure Tivoli NetView for z/OS for AIX to receive the SNMP traps. This error does not occur if the **clinfoES** daemon is started by using the **startsrc** command.

Node powers off after cluster services are started

Problem

The node is powered off or the node does not respond after cluster services are started. The `errpt` report shows an operator message that is logged by the **clexit.rc** script, which issues a **halt -q** command to the system.

Solution

Use the cluster verification utility to uncover discrepancies in cluster configuration information on all cluster nodes.

Correct any configuration errors uncovered by the cluster verification utility. Make the necessary changes by using the PowerHA SystemMirror Configuration SMIT panels. After you correct the problem, select the **Verify and Synchronize PowerHA SystemMirror Configuration** option to synchronize the cluster configuration across all nodes. Then, select the **Start Cluster Services** option from the **System Management (C-SPOC)** > **PowerHA SystemMirror Manage Services** SMIT panel to start the Cluster Manager.

The problem is resolved after the configuration passes verification and is synchronized across the cluster. If the problem persists, report the problem to IBM support.

For more information about the **snap -e** command, see the section Using the AIX data collection utility.

configchk command returns an unknown host message

Problem

The `/etc/hosts` file on each cluster node does not contain the IP labels of other nodes in the cluster. For example, in a four-node cluster, Node A, Node B, and Node C's `/etc/hosts` files do not contain the IP labels of the other cluster nodes.

If this situation occurs, the **configchk** command returns the following message to the console:

```
"your hostname not known," "Cannot access node x."
```

This message indicates that the `/etc/hosts` file on Node x does not contain an entry for your node.

Solution

Before you start the PowerHA SystemMirror software, ensure that the `/etc/hosts` file on each node includes the service and boot IP labels of each cluster node.

Cluster Manager does not respond during reconfiguration

Problem

The Cluster Manager does not respond during reconfiguration and display a message that is similar to the following sample:

```
The cluster has been in reconfiguration too long;Something may be wrong.
```

```
An event script has failed.
```

Solution

Determine why the script failed by examining the `/var/hacmp/log/hacmp.out` file to see what process exited with a nonzero status. The error messages in the `/var/hacmp/adm/cluster.log` file might also be helpful. Fix the problem identified in the log file. Then, run the **clruncmd** command either at the command line, or by using the SMIT **Problem Determination Tools** > **PowerHA SystemMirror Script Failure** panel. The **clruncmd** command signals the Cluster Manager to resume cluster processing.

clcomd and clstrmgrES daemons fail to start on newly installed AIX nodes

Problem
On newly installed AIX nodes, **clcomd** and **clstrmgrES** daemons fail to start.

Solution

Manually indicate to the system console (for the AIX installation assistant) that the AIX installation is finished.

This problem usually occurs on newly installed AIX nodes. During the first restart, AIX runs the installation assistant from `/etc/inittab` and does not proceed with other entries in this file. AIX installation assistant waits for your input on system console. AIX runs the installation assistant on every subsequent boot until you indicate that installation is complete. After the installation is complete, the system starts the cluster communications daemon **clcomd** and the Cluster Manager daemon **clstrmgrES**.

Pre-event or post-event does not exist on a node after upgrading the node

Problem
The cluster verification utility indicates that a pre- or post-event does not exist on a node after you upgrade to a new version of the PowerHA SystemMirror software.

Solution

Ensure that a script by the defined name exists and is executable on all cluster nodes.

Each node must contain a script that is associated with the defined pre-event or post-event. While the contents of the script do not have to be the same on each node, the name of the script must be consistent across the cluster. If no action is required on a particular node, a **no-op** script with the same event-script name must be placed on nodes on which processing does not occur.

Node fails during configuration with 869 LED display

Problem
The system does not respond and the code 869 is displayed.

Solution

A number of situations can cause this display to occur. Make sure all devices connected to the SCSI bus have unique SCSI IDs to avoid SCSI ID conflicts. In particular, check that the adapters and devices on each cluster node that is connected to the SCSI bus have a different SCSI ID. By default,

AIX assigns an ID of 7 to a SCSI adapter when it configures the adapter. For more information about checking and setting SCSI IDs, see the *Planning Guide*.

Node cannot rejoin cluster after being dynamically removed

Problem

Node cannot rejoin cluster after the node is dynamically removed.

Solution

When you remove a node from the cluster, the cluster definition remains in the node's configuration database. If you start cluster services on the removed node, the node reads this cluster configuration data and attempts to rejoin the cluster from which it is removed. The other nodes no longer recognize this node as a member of the cluster and refuse to allow the node to join. The node that is requesting to join the cluster has the same cluster name as the existing cluster. The node can cause the cluster to become unstable or the existing nodes to fail.

To ensure that a removed node cannot be restarted with outdated configuration database information, complete the following procedure to remove the cluster definition from the node:

1. Stop cluster services on the node to be removed by using the following command:

   ```
   clstop -R
   ```

   Important: You must stop cluster services on the node before the node is removed from the cluster.

   The **-R** flag removes the PowerHA SystemMirror entry in the **/etc/inittab** file, preventing cluster services from being automatically started when the node is rebooted.

2. Remove the PowerHA SystemMirror entry from the **rc.net** file by using the following command:

   ```
   clchipat false
   ```

3. Remove the cluster definition from the node's configuration database by using the following command:

   ```
   clrmclstr
   ```

You can also perform this task from the SMIT panel by selecting **Extended Configuration > Extended Topology Configuration > Configure a PowerHA SystemMirror Cluster > Remove a PowerHA SystemMirror Cluster**.

Resource group migration is not persistent after cluster startup

Problem

You requested resource group migration operation by using the Resource Group Migration Utility, in which **Persists across Cluster Reboot** option is set to `true` or the resource group migration operation is performed by using the **clRGmove** command. After you stop and restart cluster services, this policy is not followed on one of the nodes in the cluster.

Solution

This problem occurs when you specified persistent resource group migration, and a node was down and inaccessible. In this scenario, the node did not obtain information about the persistent resource group migration, and after the cluster services are restarted, the node is the first to join the cluster. Thus, the resource group migration is not persistent. To restore the persistent migration setting, you must again specify it in the **Extended Resource Configuration** > **PowerHA SystemMirror Resource Group Configuration** SMIT panel.

Cluster does not startup after upgrade to PowerHA SystemMirror

Problem

The ODM entry for group hacmp is removed on SP nodes. This problem manifests itself as the inability to start the cluster or **clcomd** errors.

Solution

To further improve security, the PowerHA SystemMirror configuration database has the following enhancements:

- **Ownership**: All PowerHA SystemMirror ODM files are owned by user root and group hacmp. In addition, all PowerHA SystemMirror binary files that are intended for use by non-root users are also owned by user root and group hacmp.
- **Permissions**: All PowerHA SystemMirror ODM files, except for the hacmpdisksubsystem file with 600 permissions, are set with 640 permissions (readable by user root and group hacmp, writable by user root). All PowerHA SystemMirror binary files that are intended for use by non-root users are installed with 2555 permissions (readable and executable by all users, with the **setgid** bit turned on so that the program runs as group hacmp).

During the installation, PowerHA SystemMirror creates the group hacmp on all nodes if it does not exist. By default, the "champ" group has permission to read the PowerHA SystemMirror ODMs, but does not have any other special authority. For security reasons, the authority of "champ" group must not be expanded.

If you use programs that access the PowerHA SystemMirror ODMs directly, you might need to rewrite them if they are intended to be run by non-root users:

- All access to the ODM data by non-root users must be provided by using PowerHA SystemMirror utilities.
- If you are using the PSSP File Collections facility to maintain the consistency of /etc/group, the new group hacmp that is created on individual cluster nodes during installation might be lost when the next file synchronization occurs.

Cluster verification errors when nodes have different fileset levels

Problem

When clusters have nodes at different fileset levels such as cluster.es.server.diag, the **clverify** command can become unresponsive or dump the core.

Solution

Generally, PowerHA SystemMirror nodes have the same fileset level. However, clusters might have nodes different fileset levels when you perform a node-by-node rolling PTF upgrade. These types of errors prevent successful cluster startup.

In this scenario, when you start the cluster, ignore verification errors by selecting **System Management (C-SPOC)** > **Manage PowerHA SystenMirror Services** > **Start Cluster Services** in SMIT. In the Start Cluster Services panel, set **Ignore verification errors?** to true and start the cluster.

Note: Ensure that the nodes are at the same fileset levels to avoid this issue. Ignoring verification errors must be avoided.

Related concepts

Disk and file system issues
This section describes solutions for potential disk and file system issues.

PowerHA SystemMirror takeover issues
This section describes solutions for potential takeover issues.

Network and switch issues

This section describes solutions for potential network and switch issues.

Cluster communications issues
This section describes solutions for potential cluster communication issues.

Client issues
This section describes solutions for potential PowerHA SystemMirror client issues.

SNMP-based status commands issues
This section describes solutions for problems that can cause SNMP-based status commands such as `clstat`, `cldump`, and `cldisp` to fail.

Miscellaneous issues
This section describes solutions for potential non-categorized PowerHA SystemMirror issues.

Disk and file system issues

This section describes solutions for potential disk and file system issues.

- AIX volume group commands fail during a varyon operation on a shared volume group
- varyonvg command fails on a volume group
- cl_nfskill command fails when performing a forced unmount operation
- cl_scdiskreset command fails
- fsck command fails at boot time
- System cannot mount specified file systems
- Cluster disk replacement process fails
- File system change not recognized by lazy update
- clam_nfsv4 application monitor fails
- Troubleshooting LVM split-site mirroring
- Troubleshooting repository disks
- Troubleshooting disk fencing

AIX volume group commands fail during a varyon operation on a shared volume group

Problem

The `/var/hacmp/log/hacmp.out` file shows that the **cl_nfskill** command fails when you perform a forced unmount of an Network File System (NFS) mounted file system. NFS provides certain levels of locking a file system that resists forced unmounting by the **cl_nfskill** command.

Solution

When you configure a shared volume group, set the **Activate volume group AUTOMATICALLY at system restart?** field to no on the SMIT **System Management (C-SPOC)** > **PowerHA SystemMirror Logical Volume Management** > **Shared Volume Groups** > **Create a Shared Volume Group** panel. After you import the shared volume group on the other cluster nodes, use the following command to ensure that the volume group on each node is not set to `autovaryon` at boot:

```
chvg -an vgname
```

varyonvg command fails on a volume group

This section discusses different problems that are indicated by a **varyonvg** command failing on a volume group.

Problem 1

The PowerHA SystemMirror software (the `/var/hacmp/log/hacmp.out` file) indicates that the **varyonvg** command failed when trying to vary on a volume group.

Solution

Ensure that the volume group is not set to `autovaryon` on any node and that the volume group (unless it is in concurrent access mode) is not already varied on by another node.

The **lsvg -o** command can be used to determine whether the shared volume group is active. Enter:`lsvg volume_group_name` on the node with the volume group activated, and check the **AUTO ON** field to determine whether the volume group is automatically set to be on. If the **AUTO ON** field is set to yes, you can modify the value by using the `chvg -an volume_group_name` command.

Problem 2

The volume group information on disk differs from the information that is present in the Device Configuration Database.

Solution

Correct the Device Configuration Database on the nodes that have incorrect information:

1. Use the **smit exportvg** fast path to export the volume group information. The volume group information is removed from the Device Configuration Database.
2. Use the **smit importvg** fast path to import the volume group. A new Device Configuration Data Base entry is created directly from the information on disk. After you import the volume group, ensure that the volume group is not set to **autovaryon** at the next system restart.
3. In SMIT, select **Problem Determination Tools** > **Recover From PowerHA System Mirror Script Failure**. The **clruncmd** command is run and the Cluster Manager resumes cluster processing.

Problem 3

The PowerHA SystemMirror software indicates that the **varyonvg** command failed because the volume group might not be found.

Solution

The volume group is not defined to the system. If a new volume group is created and exported, or if an mksysb system backup is restored, you must import the volume group. Follow the steps described in Problem 2.2 to verify that the correct volume group name is being referenced.

Problem 4

The PowerHA SystemMirror software indicates that the **varyonvg** command failed because the logical volume is incomplete.

Solution

The **varyonvg** command fails because the forced vary on attribute is configured for the volume group in SMIT, and when the forced vary on operation is performed, PowerHA SystemMirror dis not find a complete copy of the specified logical volume for the volume group. Also, it is possible that you requested a forced vary on operation but did not specify the **super strict** allocation policy for the mirrored logical volumes. In this case, the vary on operation might not be successful.

cl_nfskill command fails when performing a forced unmount operation

Problem

The **/var/hacmp/log/hacmp.out** file shows that the **cl_nfskill** command fails on a forced unmount operation of an NFS-mounted file system. NFS provides certain levels of locking a file system that resists forced unmount operation by the **cl_nfskill** command.

Solution

Make a copy of the `/etc/locks` file in a separate directory before you run the **cl_nfskill** command. Delete the original `/etc/locks` file and run the **cl_nfskill** command. After the command succeeds, re-create the `/etc/locks` file by using the saved copy.

cl_scdiskreset command fails

Problem

The **cl_scdiskreset** command logs error messages to the `/var/hacmp/log/hacmp.out` file. To break the reserve held by one system on a SCSI device, the PowerHA SystemMirror disk utilities issue the **cl_scdiskreset** command. The **cl_scdiskreset** command might fail if back-level hardware exists on the SCSI bus (adapters, cables, or devices) or if a SCSI ID conflict exists on the bus.

Solution

See the appropriate sections in the Using cluster log files topic to check the SCSI adapters, cables, and devices. Ensure that you have the latest adapters and cables. The SCSI IDs for each SCSI device must be different.

fsck command fails at boot time

Problem

At boot time, AIX runs the **fsck** command to check all the file systems listed in **/etc/filesystems** with the **check=true** attribute. If it cannot check a file system, AIX displays an error.

Solution

For file systems controlled by PowerHA SystemMirror, this message typically does not indicate a problem. The file system check fails because the volume group that is defining the file system is not varied on. The boot procedure does not automatically vary on PowerHA SystemMirror-controlled volume groups.

To prevent this message, make sure that all the file systems under PowerHA SystemMirror control do not have the **check=true** attribute in their **/etc/filesystems** stanzas. To delete this attribute or change it to **check=false**, edit the **/etc/filesystems** file.

System cannot mount specified file systems

Problem

The **/etc/filesystems** file is not updated to reflect changes to log names for a logical volume. If you change the name of a logical volume after the file systems is created for that logical volume, the **/etc/filesystems** entry for the log does not get updated. When you mount the file systems, the PowerHA SystemMirror software tries to get the required information about the logical volume name from the old log name. Because this information is not updated, the file systems cannot be mounted.

Solution

Update the **/etc/filesystems** file after you modify logical volume names.

Cluster disk replacement operation fails

Problem

The disk replacement operation failed to complete due to a **node_down** event.

Solution

After the node is online, export the volume group, and then import the volume group before you start PowerHA SystemMirror on this node.

File system change not recognized by lazy update

Problem

If you change the name of a file system, or remove a file system and then perform a lazy update, lazy update does not run the **imfs -lx** command before the **imfs** command is run. This action might lead to a failure during fallover or prevent a successful restart of the PowerHA SystemMirror cluster services.

Solution

Use the C-SPOC utility to change or remove file systems to ensure that the **imfs -lx** command is run before the **imfs** command and that the changes are updated on all nodes in the cluster.

Error Reporting provides detailed information about inconsistency in volume group state across the cluster. If this happens, take manual corrective action. If the file system changes are not updated on all nodes, update the nodes manually with this information.

clam_nfsv4 application monitor fails

Problem

The clam_nfsv4 application monitor takes more than 60 seconds to complete. The monitor is not responding and is stopped. Therefore, a fallover occurs on the Network File System (NFS) node. This fallover usually occurs if the system that hosts the application monitor is experiencing high-performance workloads.

Solution

You must reduce the system workloads to correct this problem. You can also apply APAR IV08873 to your system, which reduces the amount of time it takes to run the clam_nfsv4 application monitor script.

Troubleshooting LVM split-site mirroring

Problem
PowerHA SystemMirror and LVM do not have information about the physical location for disks, other than the information that was specified when the mirror pools were defined.

Solution

Review the following information to identify possible solutions for problems with LVM split-site mirroring:

- Verify that the assignment of disks to mirror pools by entering `smitty cl_mirrorpool_mgt` from the command line, and select **Show Mirror Pools for a Volume Group**.
- Verify that the mirroring for individual file systems and logical volumes is correct by entering `smitty cl_lv` from the command line, and selecting **Show Characteristics of a Logical Volume**.
- Verify that your volume groups are super strict by entering `smitty cl_vgsc` from the command line, and select **Change/Show characteristics of a Volume Group**.
- Examine the AIX error log for problems associated with the disks in the volume group if resynchronization fails. You can manually resynchronize the volume group by entering `smitty cl_syncvg` from the command line, and select **Synchronize LVM Mirrors by Volume Group**.

Troubleshooting repository disks

Problem
If any node in the cluster encounters errors with the repository disk or a failure while accessing the disk, the cluster enters a limited or restricted mode of operation. In this mode of operation most topology-related operations are not allowed, and any node that is restarted cannot rejoin the cluster.

Solution

When the repository disk fails, you are notified of the disk failure. PowerHA SystemMirror continues to notify you of the repository disk failure until it is resolved. To determine what the problem is with the repository disk, you can view the following log files:

- `hacmp.out`
- AIX error log (using the **errpt** command)

Example: hacmp.out log

Following is an example of an error message in the `hacmp.out` log file when a repository disk fails: `ERROR: rep_disk_notify : Tue Jan 10 13:38:22 CST 2012 : Node "r6r4m32"(0x54628FEA1D0611E183EE001A64B90DF0) on Cluster r6r4m31_32_33_34 has lost access to repository disk hdisk75.`

Example: AIX error log

When a node loses access to the repository disk, an entry is made in the AIX error log of each node that has a problem. Following is an example of an error message in the error log file when a repository disk fails.

Note: To view the AIX error log, you must use the **errpt** command.

```
LABEL:          OPMSG
IDENTIFIER:     AA8AB241

Date/Time:      Tue Jan 10 13:38:22 CST 2012
Sequence Number: 21581
Machine Id:     00CDB2C14C00
Node Id:        r6r4m32
Class:          O
Type:           TEMP
WPAR:           Global
Resource Name:  clevmgrd

Description
OPERATOR NOTIFICATION

User Causes
ERRLOGGER COMMAND

        Recommended Actions
        REVIEW DETAILED DATA

Detail Data
MESSAGE FROM ERRLOGGER COMMAND
Error: Node 0x54628FEA1D0611E183EE001A64B90DF0 has lost access to repository disk hdisk75.
```

Replacing a failed or lost repository disk

If a repository disk fails, the repository disk must be recovered on a different disk to restore all cluster operations. The circumstances for your cluster environment and the type of the repository disk failure determine the possible methods for recovering the repository disk.

Automatic Repository Disk Replacement (ARR)

PowerHA SystemMirror Version 7.2.0, or later, uses the ARR capability of CAA (in AIX Version 7.2, or later, or in AIX Version 7.1 with Technology Level 4, or later), to handle repository disk failures. ARR automatically replaces a failed repository disk with a backup repository disk. The ARR function is available only if you configure a backup repository disk by using PowerHA SystemMirror. For more information about ARR, see the Repository disk failure topic.

You must clean up the failed repository disk because the ARR does not clean the disk as it is not accessible. To clean up the failed repository disk, use the following command:

```
CAA_FORCE_ENABLED=true rmcluster -r <disk name>
```

The following are two possible scenarios where a repository disk fails and the possible methods for restoring the repository disk on a new storage disk.

Repository disk fails but the cluster is still operational

In this scenario, the repository disk access is lost on one or more nodes in the cluster. When this failure occurs, Cluster Aware AIX (CAA) continues to operate in restricted mode by using repository disk information that is cached in memory. If CAA remains active on a single node in the cluster, the information from the previous repository disk information can be used to rebuild a new repository disk.

To rebuild the repository disk after a failure, complete the following steps from any node where CAA is still active:

1. Verify that CAA is active on the node by using the **lscluster -c** command and then the **lscluster -m** command.

2. Replace the repository disk by completing the steps in the Replacing a repository disk with SMIT topic. PowerHA SystemMirror recognizes the problem and interacts with CAA to rebuild the repository disk on the new storage disk.

   Note: This step updates the repository information that is stored in the PowerHA SystemMirror configuration data.

   You do not need to perform Step 1 and Step 2, if the ARR function is available.

3. Synchronize thePowerHA SystemMirror cluster configuration information by selecting **Cluster Nodes and Networks** > **Verify and Synchronize Cluster Configuration** from the SMIT interface.

Repository disk fails and the nodes in the cluster are restarted

In this rare scenario, a series of critical failures occurs that result in a worst case scenario where access to the repository disk is lost and all nodes in the cluster were rebooted. Thus, none of the nodes in the cluster remained online during the failure and you cannot rebuild the repository disk from the AIX operating systems memory. When the nodes are brought back online, they cannot start CAA because a repository disk is not present in the cluster. To fix this problem, it is ideal to bring back the repository disk and allow the cluster self-heal. If that is not possible, you must rebuild the repository disk on a new storage disk and use it to start the CAA cluster.

To rebuild the repository disk and start cluster services, complete the following steps:

1. On a node in the cluster, rebuild the repository by completing the steps in the Replacing a repository disk with SMIT topic. PowerHA SystemMirror recognizes the problem and interacts with CAA to rebuild the repository disk on the new storage disk.

   Note: This step updates the repository information that is stored in the PowerHA SystemMirror configuration data and rebuilds the repository disk from the CAA cluster cache file.

   If the ARR function is available, you do not need to perform Step 1, and the disk is replaced automatically.

   After the repository disk is replaced, run the verify and synchronization operations. If some of the nodes are down, the verify and synchronization operations might fail with errors. To run the verify and synchronization operations successfully, enter the following command:

   ```
   #/usr/es/sbin/cluster/utilities/cldare -f -dr
   ```

   You can ignore the cl_rsh errors if any.

2. Start cluster services on the node that hosts the repository disk by completing the steps in the Starting cluster services topic.

3. All other nodes in the cluster continue to attempt to access the original repository disk. You must configure these nodes to use the new repository disk and start CAA cluster services. Verify that the CAA cluster is not active on any of these nodes by using the **lscluster -m**

command. If the CAA cluster is not active or the local node is in the DOWN state, enter the following commands to remove the old repository disk information:

```
export CAA_FORCE_ENABLED=true
clusterconf -fu
```

4. To have other nodes join the CAA cluster, use the following command on the active node with the newly created repository disk:

```
clusterconf -p
```

For AIX Version 7.1 with Technology Level 4, or later, you do not need to perform Step 3 and Step 4. After you complete Step 2, all nodes that were rebooted must wait for about 10 minutes to use the new repository disk.

5. Verify that CAA is active by first using the **lscluster -c** command and then the **lscluster -m** command.

6. Synchronize thePowerHA SystemMirror cluster configuration information about the newly created repository disk to all other nodes by selecting **Cluster Nodes and Networks** > **Verify and Synchronize Cluster Configuration** from the SMIT interface.

7. Start PowerHA SystemMirror cluster services on all nodes (besides the first node where the repository disk was created) by selecting **System Management (C-SPOC)** > **PowerHA SystemMirror Services** > **Start Cluster Services** from the SMIT interface.

Snapshot migration and repository disk

The snapshot migration process for an online cluster requires that the cluster information in the snapshot matches the online cluster information. This requirement also applies to repository disks. If you change a repository disk configuration, you must update the snapshot to reflect these changes and then complete the snapshot migration process.

Troubleshooting disk fencing

Disk fencing feature is only available for the quarantine policies in PowerHA SystemMirror.

Problem 1
Disk fencing is no longer needed for your environment. You can disable disk fencing and release the reservation for a disk or a volume group.

Solution
To disable disk fencing and release the reservation for a disk or a volume group, complete the following steps:

1. Stop the cluster services on all cluster nodes.

2. From the command line, enter **smit sysmirror**.

3. From the SMIT interface, select **Custom Cluster Configuration** > **Cluster Nodes and Networks** > **Initial Cluster Setup (Custom)** > **Configure Cluster Split and Merge Policy** > **Quarantine Policy** > **Disk Fencing**, and press Enter.

4. Specify **No** for the **Disk Fencing** field, and press Enter to save your changes.

5. Verify and synchronize the cluster.

6. Start the cluster services on all cluster nodes.

Problem 2
A resource group goes into an error state in an active cluster. The resource group is put into an error state because a node fails to register and put a reserve on a single volume group in the resource group.

Solution
To fix this problem with the resource group, complete the following steps:

1. From the command line, enter **smit sysmirror.**

2. From the SMIT interface, select **Problem Determination Tools** > **Recover Resource Group from SCSI Persistent Reserve Error**, and press Enter.

3. Select the resource that is in an error state, and press Enter.

4. From the SMIT interface, select **System Management (C-SPOC)** > **Resource Group and Applications** > **Bring a Resource Group Online**, and press Enter.

5. Select the resource group that you want to bring back online, and press Enter.

Note: If the problem persists, contact IBM support.

Problem 3

If the quarantine policy is **Disk Fencing**, the PowerHA SystemMirror sets up the SCSI Persistent Reserve state for all shared disks when it is started. PowerHA SystemMirror also sets up the Persistent Reserve keys for all paths to the devices. If later, new or changed paths are added to the device, the Persistent Reserve keys are not set up for those paths.

Solution

To update the SCSI Persistent Reserves for the disk paths that are new or changed, complete the following steps from the command line:

- To release the existing disk paths reservations, run one of the following commands:

```
clmgr modify physical_volume <disk> SCSIPR_ACTION=clear
clmgr modify volume_group <vg> SCSIPR_ACTION=clear
```

where *disk* is the name of the disk that is the part of the volume group and *vg* is the name of the volume group.

- To restore the reservations, stop the cluster services by using the **unmanage** option and restart the cluster services. To restore the reservation on each cluster node, you must stop and start the cluster services on each cluster node by using the following commands:

```
clmgr stop node MANAGE=unmanage
clmgr start node
```

Related concepts

PowerHA SystemMirror startup issues
This section describes solutions for potential PowerHA SystemMirror startup issues.

PowerHA SystemMirror takeover issues
This section describes solutions for potential takeover issues.

Network and switch issues
This section describes solutions for potential network and switch issues.

Cluster communications issues
This section describes solutions for potential cluster communication issues.

Client issues
This section describes solutions for potential PowerHA SystemMirror client issues.

SNMP-based status commands issues
This section describes solutions for problems that can cause SNMP-based status commands such as `clstat`, `cldump`, and `cldisp` to fail.

Miscellaneous issues
This section describes solutions for potential non-categorized PowerHA SystemMirror issues.

Network and switch issues

This section describes solutions for potential network and switch issues.

- Unexpected network interface failure in switched networks
- Verification multicast communication
- Persisting IPv6 addresses during system restart

- [Cluster nodes cannot communicate](#)
- [Distributed SMIT causes unpredictable results](#)
- [Recovering from PCI hot plug NIC failure](#)
- [IP label for PowerHA SystemMirror disconnected from AIX interface](#)
- [Packets lost during data transmission](#)
- [Troubleshooting multicast](#)
- [Troubleshooting unicast](#)
- [Troubleshooting virtual local area networks](#)

Unexpected network interface failure in switched networks

Problem

Unexpected network interface failures can occur in PowerHA SystemMirror configurations by using switched networks if the networks and the switches are incorrectly configured.

Solution

Take care to configure your switches and networks correctly.

Verification multicast communication

Problem

By default, PowerHA SystemMirror uses unicast communications for heartbeat. For cluster communication, you can optionally select to configure a multicast address or have CAA automatically select the multicast address if your network is configured to support multicast communication. If you use multicast communication, do not create a cluster until you verify that multicast packets can be sent successfully across all nodes that are part of the cluster.

Solution

To test end-to-end multicast communication for all nodes used to create the cluster on your network, run the **mping** command to send and receive packets between nodes.

If you are running PowerHA SystemMirror Version 7.1.1, or later, you cannot create a cluster if the **mping** command fails. If the **mping** command fails, your network is not set up correctly for multicast communication. If so, review the documentation for your switches and routers to enable multicast communication.

You can run the **mping** command with a specific multicast address; otherwise, the command uses a default multicast address. You must use the multicast addresses that are used for creating the cluster as input for the **mping** command.

Note: The **mping** command uses the interface that has the default route. To use the **mping** command for testing multicast communication on a different interface that does not have the default route, you must temporarily add a static route with the required interface to the multicast IP address.

The following example shows a success case and a failure case for the **mping** command, where Node A is the receiver and Node B is the sender.

Success case:

```
Receiver

root@nodeA:/# mping -r -R -c 5
mping version 1.1
Listening on 227.1.1.1/4098:

Replying to mping from 9.3.207.195 (nodeB.aus.stglabs.ibm.com) bytes=32 seqno=0 ttl=1
Replying to mping from 9.3.207.195 (nodeB.aus.stglabs.ibm.com) bytes=32 seqno=1 ttl=1
Replying to mping from 9.3.207.195 (nodeB.aus.stglabs.ibm.com) bytes=32 seqno=2 ttl=1
Replying to mping from 9.3.207.195 (nodeB.aus.stglabs.ibm.com) bytes=32 seqno=3 ttl=1
Replying to mping from 9.3.207.195 (nodeB.aus.stglabs.ibm.com) bytes=32 seqno=4 ttl=1

Sender
```

```
root@nodeB:/# mping -R -s -c 5
mping version 1.1
mpinging 227.1.1.1/4098 with ttl=1:

32 bytes from 9.3.207.190 (nodeA.aus.stglabs.ibm.com) seqno=0 ttl=1 time=0.985 ms
32 bytes from 9.3.207.190 (nodeA.aus.stglabs.ibm.com) seqno=1 ttl=1 time=0.958 ms
32 bytes from 9.3.207.190 (nodeA.aus.stglabs.ibm.com) seqno=2 ttl=1 time=0.998 ms
32 bytes from 9.3.207.190 (nodeA.aus.stglabs.ibm.com) seqno=3 ttl=1 time=0.863 ms
32 bytes from 9.3.207.190 (nodeA.aus.stglabs.ibm.com) seqno=4 ttl=1 time=0.903 ms

--- 227.1.1.1 mping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 0.863/0.941/0.998 ms
```

Failure case:

```
Receiver

root@nodeA:/# mping -r -R -c 5 -6
mping version 1.1
Listening on ff05::7F01:0101/4098:

Replying to mping from fe80::18ae:19ff:fe72:1a15 bytes=48 seqno=0 ttl=1
Replying to mping from fe80::18ae:19ff:fe72:1a15 bytes=48 seqno=1 ttl=1
Replying to mping from fe80::18ae:19ff:fe72:1a15 bytes=48 seqno=2 ttl=1
Replying to mping from fe80::18ae:19ff:fe72:1a15 bytes=48 seqno=3 ttl=1
Replying to mping from fe80::18ae:19ff:fe72:1a15 bytes=48 seqno=4 ttl=1

Sender

root@nodeB:/# mping -R -s -c 5 -6
mping version 1.1
mpinging ff05::7F01:0101/4098 with ttl=1:


--- ff05::7F01:0101 mping statistics ---
5 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.000/0.000/0.000 ms
```

Note: To verify a result, you must check the sender side of the **mping** command only. Also, note the percentage of packet loss. To verify whether multicast is working on a network, you must perform the **mping** tests with both nodes tested as both the sender and receiver. Typically, the non-verbose output provides you the necessary information. However, if you choose to use the **-v** flag with the **mping** command, a good knowledge about the internals of the program is necessary, without which the verbose output can be misunderstood. You can also check the return code from the sender side of the **mping** command. If an error occurs, the sender side returns **255**. Upon success, it returns **0**.

Cluster Aware AIX (CAA) selects a default multicast address if you do not specify a multicast address when you create the cluster. The default multicast address is created by combining the logical OR of the value (228.0.0.0) with the low 24 bits of the IP address of the node. For example, if the IP address is 9.3.199.45, then the default multicast address would be 228.3.199.45.

The Internet Protocol version 6 (IPv6) addresses are supported by PowerHA SystemMirror Version 7.1.2, or later. When IPv6 addresses are configured in the cluster, Cluster Aware AIX (CAA) activates heartbeat for the IPv6 addresses with an IPv6 multicast address. You must verify that the IPv6 connections in your environment can communicate with multicast addresses.

To verify that IPv6 multicast communications are configured correctly in your environment, you can run the **mping** command with the **-6** option. When you run the **mping** command, it verifies the IPv6 multicast communications with the default IPv6 multicast address. To specify a specific IPv6 multicast address, run the **mping** command with the **-a** option and specify an IPv6 multicast address. You do not need to specify the **-6** option when you use the **-a** option. The **mping** command automatically determines the family of the address that is passed with the **-a** option.

Persisting IPv6 addresses during system restart

Problem

Internet Protocol version 6 (IPv6) is designed for dynamic configuration as is the AIX operating system. IPv6 addresses do not persist during a system reboot operation.

Solution

To configure IPv6 addresses after a reboot, you can manually run the **autoconf6** command. Alternatively, PowerHA SystemMirror will run the **autoconf6** command automatically before starting cluster services.

To configure the **autoconf6** command to run automatically for the AIX operating system, complete the following steps to change the /etc/rc.tcpip file:

1. Uncomment the following lines to run the **autoconf6** command:

   ```
   # Start up autoconf6 process
   start /usr/sbin/autoconf6
   ```

   Note: You can specify individual interfaces by entering the **-i** flag. For example,

   ```
   # Start up autoconf6 process
   start /usr/sbin/autoconf6 "" "-i en1"
   ```

2. Uncomment the following lines to start the ndpd daemons:

   ```
   # Start up ndpd-host daemon
   start /usr/sbin/ndpd-host "$src_running"

   # Start up the ndpd-router daemon
   start /usr/sbin/ndpd-router "$src_running"
   ```

Cluster nodes cannot communicate

Problem

If your configuration has two or more nodes that are connected by a single network, you might experience a partitioned cluster. A partitioned cluster occurs when cluster nodes cannot communicate. In normal circumstances, a service network interface failure on a node causes the Cluster Manager to recognize and handle a **swap_adapter** event, where the service IP label or IP address is replaced with another IP label or IP address. Heartbeats are exchanged by the way of shared disks. However, there is a chance that the node becomes isolated from the cluster. Although the Cluster Managers on other nodes are aware of the attempted **swap_adapter** event, they cannot communicate with the now isolated (partitioned) node because no communication path exists.

Solution

Ensure that the network is configured for no single point of failure.

Distributed SMIT causes unpredictable results

Problem

Using the AIX utility DSMIT on operations other than starting or stopping PowerHA SystemMirror cluster services, can cause unpredictable results.

Solution

DSMIT manages the operation of networked IBM System p processors. It includes the logic necessary to control execution of AIX commands on all networked nodes. To avoid a conflict with PowerHA SystemMirror, use DSMIT only to start and stop PowerHA SystemMirror cluster services.

Recovering from PCI hot plug NIC failure

Problem

If an unrecoverable error causes a PCI hot-replacement process to fail, the NIC might be abandoned in an unconfigured state and the node might be abandoned in maintenance mode. The PCI slot that is holding the NIC or the new NIC might be damaged.

Solution

User intervention is required to get the node back in fully working order.

IP label for PowerHA SystemMirror disconnected from AIX interface

Problem

When you define network interfaces to the cluster configuration by entering or selecting an IPPowerHA SystemMirror label, PowerHA SystemMirror discovers the associated AIX network interface name. PowerHA SystemMirror expects this relationship to remain unchanged. If you change the name of the AIX network interface name after you configure and synchronize the cluster, PowerHA SystemMirror does not function correctly.

Solution

If this problem occurs, you can reset the network interface name from the SMIT PowerHA SystemMirror **System Management (C-SPOC)** panel.

Packets lost during data transmission

Problem

If data is intermittently lost during transmission, it is possible that the maximum transmission unit (MTU) is set to different sizes on different nodes. For example, if Node A sends 8 K packets to Node B, which can accept 1.5 K packets, Node B assumes that the message is complete and the data might be lost.

Solution

Run the cluster verification utility to ensure that all of the network interface cards on all cluster nodes during the same network have the same setting for MTU size. If the MTU size is inconsistent across the network, an error is displayed, and you can determine which nodes to adjust.

Troubleshooting multicast

Problem

Use the `mping` command to test whether your nodes can send and receive multicast packets. If the `mping` command fails, you need to identify what the problem is in your network environment.

Solution

To troubleshoot multicast problems in your network, review the following guidelines:

- Review the documentation for the switches that are used for multicast communication.
- Disable Internet Group Management Protocol (IGMP) snooping on the switches that are used for multicast communication.

  Note: If your network infrastructure does not allow IGMP snooping to be disabled permanently, you might be able to troubleshoot problems by temporarily disabling snooping on the switches, and then adding more network components one at a time.

- Eliminate any cascaded switches between the nodes in the cluster. In other words, have only a single switch between the nodes in the cluster.

Troubleshooting unicast

Problem

By default, PowerHA SystemMirror uses unicast socket-based communications between nodes in the cluster.

If you are having problems with unicast communications, follow general network troubleshooting procedures. For example,:

- Use the **ifconfig** and **netstat** commands to verify the IP address configuration and routing.
- Use the **ping** and **traceroute** commands to verify that nodes and adapters can communicate.

If the problem cannot be identified, use the **iptrace** command to trace low-level packet activity.

Troubleshooting virtual local area networks

Problem

To troubleshoot VLAN interfaces defined to PowerHA SystemMirror and detect an interface failure, consider these interfaces as interfaces defined on single adapter networks.

In particular, list the network interfaces that belong to a VLAN in the *ping_client_list* variable in the `/usr/es/sbin/cluster/etc/clinfo.rc` script and run the **clinfo** command. Whenever a cluster event occurs, **clinfo** monitors and detects a failure of the listed network interfaces. Due to the nature of virtual local area networks, other mechanisms to detect the failure of network interfaces are not effective.

Related concepts

PowerHA SystemMirror startup issues
This section describes solutions for potential PowerHA SystemMirror startup issues.

Disk and file system issues
This section describes solutions for potential disk and file system issues.

PowerHA SystemMirror takeover issues
This section describes solutions for potential takeover issues.

Cluster communications issues
This section describes solutions for potential cluster communication issues.

Client issues
This section describes solutions for potential PowerHA SystemMirror client issues.

SNMP-based status commands issues
This section describes solutions for problems that can cause SNMP-based status commands such as `clstat`, `cldump`, and `cldisp` to fail.

Miscellaneous issues
This section describes solutions for potential non-categorized PowerHA SystemMirror issues.

Cluster communications issues

This section describes solutions for potential cluster communication issues.

- Message encryption fails
- Cluster nodes fail to communicate with each other

Message encryption fails

Problem

Encryption or decryption fails after security is enabled and the **clcomd** daemon communication fails across nodes. To verify whether your encryption or decryption fails, you can view the **clcomddiag.log** file.

Solution

Disable security by using the SMIT from the master node or any node, and then stop and start PowerHA SystemMirror communication daemon on all nodes.

Verify that the cluster node has the following file sets installed before you enable security:

- For data encryption with DES message authentication: `rsct.crypt.des`
- For data encryption standard Triple DES message authentication: `rsct.crypt.3des`
- For data encryption with Advanced Encryption Standard (AES) message authentication: `rsct.crypt.aes256.` . The **clic** version 4.7 file set must be installed.

If needed, install the file sets mentioned earlier from the AIX Expansion Pack CD-ROM.

If the files sets are installed after PowerHA SystemMirror is already running, start and stop the PowerHA SystemMirror Cluster Communications daemon for PowerHA SystemMirror to use these file sets. To restart the Cluster Communications daemon:

Example:

```
stopscr -s clcomd
startsrc -s clcomd
```

If the file sets are present, and you get an encryption error, the encryption file sets were installed, or reinstalled, after PowerHA SystemMirror was running. Restart the Cluster Communications daemon as described earlier.

Cluster nodes fail to communicate with each other

Problem

Cluster nodes are unable to communicate with each, and you have one of the following configured:

- Message authentication, or message authentication and encryption enabled
- Use of persistent IP labels for VPN tunnels.

Solution

Make sure that the network is operational, see the section Network and switch issues. Check whether the cluster has persistent IP labels. If it does, make sure that they are configured correctly and that you can ping the IP label. If you are using message authentication, or message authentication and encryption:

- Make sure that each cluster node has the same setting for message authentication mode. If the modes are different, on each node set message authentication mode to None and configure message authentication again.
- Make sure that each node has the same type of encryption key in the `/usr/es/sbin/ cluster/etc` directory. Encryption keys cannot reside in other directories.

If persistent IP labels are configured for a VPN, perform the following steps:

1. Change **User Persistent Labels** to No.
2. Synchronize cluster configuration.
3. Change **User Persistent Labels** to Yes.

Related concepts

PowerHA SystemMirror startup issues
This section describes solutions for potential PowerHA SystemMirror startup issues.

Disk and file system issues
This section describes solutions for potential disk and file system issues.

PowerHA SystemMirror takeover issues

This section describes solutions for potential takeover issues.

Network and switch issues
This section describes solutions for potential network and switch issues.

Client issues
This section describes solutions for potential PowerHA SystemMirror client issues.

SNMP-based status commands issues
This section describes solutions for problems that can cause SNMP-based status commands such as `clstat`, `cldump`, and `cldisp` to fail.

Miscellaneous issues
This section describes solutions for potential non-categorized PowerHA SystemMirror issues.

PowerHA SystemMirror takeover issues

This section describes solutions for potential takeover issues.

If you are investigating resource group movement in PowerHA SystemMirror and what to know why an **rg_move** event occurred, you must check the `/var/hacmp/log/hacmp.out` file. The way resource groups are handled and prioritized in fallover circumstances, particularly in PowerHA SystemMirror, the `hacmp.out` file and its event summaries are important in tracking the activity and resulting location of your resource groups. In addition, with parallel processing of resource groups, the `hacmp.out` file reports details that cannot be seen in the cluster history log or the **clstrmgr.debug** log file. Always check the `hacmp.out` log early on when you investigate resource group movement after takeover activity.

- varyonvg command fails during takeover
- Highly available applications fail
- PowerHA SystemMirror selective fallover is not triggered by a volume group loss of quorum error in AIX
- Group Services sends GS_DOM_MERGE_ER message
- cfgmgr command causes unwanted behavior in cluster
- Network interfaces swap fails due to an rmdev *device busy* error

varyonvg command fails during takeover

Problem

The PowerHA SystemMirror software failed to vary on a shared volume group. The volume group name is either missing or is incorrect in the PowerHA SystemMirror Configuration Database object class.

Solution

- Check the `/var/hacmp/log/hacmp.out` file to find the error related to vary on operation failure.
- List all the volume groups known to the system by using the **lsvg** command. Check whether the volume group names that are used in the PowerHA SystemMirror resource Configuration Database object class are correct. To change a volume group name in the Configuration Database, in SMIT, select **Initialization and Standard Configuration** > **PowerHA SystemMirror** > **Resource Groups** > **Change/Show Resource Groups**. Select the resource group where you want the volume group to be included. Use the `Volume Groups` or `Concurrent Volume Groups` field on the **Change/Show Resources and Attributes for a Resource Group** panel to set the volume group names. After you correct the problem, select **Problem Determination Tools** > **Recover From PowerHA SystemMirror Script Failure** panel to run the **clruncmd** command to signal the Cluster Manager to resume cluster processing.
- Run the cluster verification utility to verify cluster resources.

Highly available applications fail

Problem

When an application is manually stopped after you stop cluster services, and resource groups are placed in an UNMANAGED state, the application does not restart with reintegration of the node.

Solution

Check that the relevant application entry in the `/usr/es/sbin/cluster/server.status` file is removed before node reintegration.

The application entry in the `/usr/es/sbin/cluster/server.status` file lists all applications already running on the node, and PowerHA SystemMirror does not restart the applications with entries in the `server.status` file.

Delete the relevant application `server.status` entry before reintegration, for PowerHA SystemMirror to recognize that the highly available application is not running, and that it must be restarted on the node.

PowerHA SystemMirror selective fallover is not triggered by a volume group loss of quorum error in AIX

Problem

PowerHA SystemMirror does not selectively move the affected resource group to another cluster node when a volume group quorum loss occurs.

Solution

If quorum is lost for a volume group that belongs to a resource group on a cluster node, the system checks whether the LVM_SA_QUORCLOSE error is displayed in the node's AIX error log file. The system communicates to the Cluster Manager to selectively move the affected resource group. PowerHA SystemMirror uses this error notification method only for mirrored volume groups with quorum enabled.

If fallover does not occur, check that the LVM_SA_QUORCLOSE error appeared in the AIX error log. When the AIX error log buffer is full, new entries are discarded until buffer space becomes available and an error log entry informs you of this problem. To resolve this issue, increase the size of the AIX error log internal buffer for the device driver.

Group Services sends GS_DOM_MERGE_ER message

Problem

A Group Services merge message is displayed and the node that is receiving the message shuts itself down. You see a GS_DOM_MERGE_ER error log entry, and a message in the Group Services daemon log file:

```
"A better domain XXX has been discovered, or domain master requested to dissolve the
domain."
```

A Group Services merge message is sent when a node loses communication with the cluster and then the node tries to reestablish communication.

Solution

It might be difficult to determine the state of the missing node and its resources. To avoid a possible data divergence if the node rejoins the cluster, you must shut down the node and successfully complete the takeover of its resources.

For example, if a cluster node cannot communicate with other nodes, but continues to work through its process table, the other nodes no longer receive keepalive messages from the missing node. The missing node is considered as failed and the remaining nodes then process the necessary events to acquire the disks, IP addresses, and other resources from the missing node. This attempt to take over

resources results in the dual-attached disks that are receiving resets to release them from the missing node and to start IP address takeover scripts.

As the disks are being acquired by the takeover node or after the disks are acquired and applications are running, the missing node completes its process table or clears an application problem. The missing node attempts to resend keepalive messages and rejoin the cluster. The disks and IP address are successfully taken over, it becomes possible to have a duplicate IP address on the network. The disks might start to experience extraneous traffic on the data bus.

Because the reason for the missing node cannot be determined, you can assume that the problem might repeat later, causing extra downtime of the node and the cluster and its applications. To ensure the highest cluster availability, Group Service merge messages must be sent to any missing cluster node to identify node isolation. This action permits successful takeover of resources, and eliminates the possibility of data corruption that can occur if both the takeover node and the rejoining missing node attempt to write to the disks. Also, if two nodes exist on the network with the same IP address, transactions might be missed and applications might hang.

When you have a partitioned cluster, the nodes on each side of the partition detect this and run a **node_down** for the nodes on the opposite side of the partition. When you run the **node_down** command or after the communication is restored, if the two sides of the partition do not agree that nodes are still members of the cluster, a decision is made as to which partition must remain up. The other partition is shut down by a GA merge from nodes in the other partition or by a node that is sending a Group Service merge to itself.

In clusters that consist of more than two nodes, the decision is based the partition that has the most nodes, and that partition stays up. If equal number of nodes are there in each partition cluster, the nodes that remain up is determined by the node number, which is also generally the first in alphabetical order. The lowest node number in cluster remains.

Group Services domain merge messages indicate that a node isolation problem is handled to keep the resources as highly available as possible, giving you time to later investigate the problem and its cause. When a domain merge occurs, Group Services and the Cluster Manager exit. The `clstrmgr.debug` file contains the following error:

```
"announcementCb: GRPSVCS announcement code=n; exiting"
"CHECK FOR FAILURE OF RSCT SUBSYSTEMS (topsvcs or grpsvcs)"
```

`cfgmgr` command causes unwanted behavior in cluster

Problem

SMIT commands like **Configure Devices Added After IPL** use the `cfgmgr` command. Sometimes this command can cause unwanted behavior in a cluster. For instance, if there is a network interface swap, the `cfgmgr` command tries to swap the network interfaces again, causing the Cluster Manager to fail.

Solution

For more information about modifying `rc.net` file, see the *Installation Guide*. You can use this technique always, not just for IP address takeover. However, as it adds to the overall takeover time, it is not recommended.

Network interfaces swap fails due to an rmdev *device busy* error

Problem

Network interfaces swap fails due to a **rmdev** *device busy* error. For example, `/var/hacmp/log/hacmp.out` shows a message that is similar to the following sample:

```
Method error (/etc/methods/ucfgdevice):
0514-062 Cannot perform the requested function because the specified device is busy.
```

Solution

Check whether the following applications are run on the system. These applications might keep the device busy:

SNA

Use the following commands to see whether SNA is running:

```
lssrc -g sna
```

Use the following command to stop SNA:

```
stopsrc -g sna
```

If that does not work, use the following command:

```
stopsrc -f -s sna
```

If that does not work, use the following command:

```
/usr/bin/sna -stop sna -t forced
```

If that does not work, use the following command:

```
/usr/bin/sna -stop sna -t cancel
```

Tivoli NetView for z/OS or Netmon

Ensure that the **sysmond** daemon is started with a **-H** flag. The network interface is opened and closed each time SM/6000 goes out to read the status. The **cl_swap_HW_address** script is run successfully when you run the **rmdev** command after the **ifconfig detach** before you swap the hardware address.

Use the following command to stop all Tivoli NetView for z/OS daemons:

```
/usr/OV/bin/nv6000_smit stopdaemons
```

IPX

Use the following commands to see whether IPX is running:

```
ps -ef |grep npsd
ps -ef |grep sapd
```

Use the following command to stop IPX:

```
/usr/lpp/netware/bin/stopnps
```

NetBIOS

Use the following commands to see whether NetBIOS is running:

```
ps -ef | grep netbios
```

Use the following commands to stop NetBIOS and unload NetBIOS streams:

```
mcsadm stop; mcs0 unload
```

• Unload various streams if applicable (that is, if the file exists):

```
cd /etc
strload -uf /etc/dlpi.conf
strload -uf /etc/pse.conf
strload -uf /etc/netware.conf
strload -uf /etc/xtiso.conf
```

- Some customer applications keep a device busy. Ensure that the shared applications are stopped properly.

Related concepts

PowerHA SystemMirror startup issues
This section describes solutions for potential PowerHA SystemMirror startup issues.

Disk and file system issues
This section describes solutions for potential disk and file system issues.

Network and switch issues
This section describes solutions for potential network and switch issues.

Cluster communications issues
This section describes solutions for potential cluster communication issues.

PowerHA SystemMirror takeover issues
This section describes solutions for potential takeover issues.

Client issues
This section describes solutions for potential PowerHA SystemMirror client issues.

SNMP-based status commands issues
This section describes solutions for problems that can cause SNMP-based status commands such as `clstat`, `cldump`, and `cldisp` to fail.

Miscellaneous issues
This section describes solutions for potential non-categorized PowerHA SystemMirror issues.

Client issues

This section describes solutions for potential PowerHA SystemMirror client issues.

- Network interface swap causes client connectivity issue
- Clients cannot access applications
- Clients cannot find clusters
- clinfo daemon is not running
- clinfo daemon does not report that a node is down

Network interface swap causes client connectivity issue

Problem

The client cannot connect to the cluster. The ARP cache on the client node still contains the address of the failed node, not the fallover node.

Solution

Issue a **ping** command to the client from a cluster node to update the client's ARP cache. Be sure to include the client name as the argument to this command. The **ping** command updates a client's ARP cache even if the client is not running **clinfoES**. You might need to add a call to the ping command in your application's pre-event or post-event processing scripts to automate this update on specific clients.

Clients cannot access applications

Problem

The **SNMP process** failed.

Solution

Check the `/etc/hosts` file on the node on which **SNMP** failed to ensure that it contains IP labels or addresses of cluster nodes.

Clients cannot find clusters

Problem

The **clstat** utility that is running on a client cannot find any clusters. The **clinfoES** daemon does not manage the data structures that it created for its clients (like **clstat**) effectively because it cannot locate an SNMP process with which it can communicate. As **clinfoES** obtains its cluster status information from SNMP, it cannot populate the PowerHA SystemMirror MIB if it cannot communicate with this daemon. As a result, various intermittent problems can occur between SNMP and **clinfoES**.

Solution

Create an updated client-based `clhosts` file by running verification with automatic corrective actions enabled. This action creates a `clhosts.client` file on the server nodes. Copy this file to the `/usr/es/sbin/cluster/etc/` directory on the clients, renaming the `clhosts` file. The **clinfoES** daemon uses the addresses in this file to attempt communication with an SNMP process that is running on a PowerHA SystemMirror server.

Also, check the `/etc/hosts` file on the node on which the SNMP process is running and on the node that is having problems with **clstat** utility or other **clinfo** API programs.

clinfo daemon is not running

Problem

The service and boot addresses of the cluster node from which **clinfoES** was started do not exist in the client-based `clhosts` file.

Solution

Create an updated client-based `clhosts` file by running verification with automatic corrective actions enabled. The `clhosts.client` file is created on server nodes. Copy this file to the `/usr/es/sbin/cluster/etc/` directory on the clients, renaming the `clhosts` file. Then, run the **clstat** command.

clinfo daemon does not report that a node is down

Problem

Even though the node is down, the SNMP daemon and **clinfoES** daemon report that the node is up. All the node's interfaces are listed as down.

Solution

When one or more nodes are active and another node tries to join the cluster, the current cluster nodes send information to the SNMP daemon that the joining node is up. If for some reason, the node fails to join the cluster, **clinfoES** daemon does not send another message to the SNMP daemon the report that the node is down.

To correct the cluster status information, restart the SNMP daemon, by using the options on the PowerHA SystemMirror Cluster Services SMIT panel.

Related concepts

PowerHA SystemMirror startup issues
This section describes solutions for potential PowerHA SystemMirror startup issues.

Disk and file system issues
This section describes solutions for potential disk and file system issues.

PowerHA SystemMirror takeover issues
This section describes solutions for potential takeover issues.

Network and switch issues
This section describes solutions for potential network and switch issues.

Cluster communications issues

This section describes solutions for potential cluster communication issues.

SNMP-based status commands issues
This section describes solutions for problems that can cause SNMP-based status commands such as `clstat`, `cldump`, and `cldisp` to fail.

Miscellaneous issues
This section describes solutions for potential non-categorized PowerHA SystemMirror issues.

SNMP-based status commands issues

This section describes solutions for problems that can cause SNMP-based status commands such as `clstat`, `cldump`, and `cldisp` to fail.

The Simple Network Management Protocol (SNMP) provides access to a database of status and configuration variables that are referred as the Management Information Base (MIB). The SNMP subsystem included with base AIX provides a subset of the overall MIB, and can also work with peer daemons that provide access to other portions of the MIB. The PowerHA SystemMirror cluster manager daemon acts as such a peer and provides access to the PowerHA SystemMirror specific variables in the MIB.

When you experience problems with SNMP or the utilities that rely on it, you must first verify that the basic SNMP configuration is functioning, then proceed to check the SystemMirror specific function.

You can check for the basic function of SNMP by using the **snmpinfo** command. Use the **snmpinfo -m dump** command to display the default part of the MIB. If this command does not produce any output, there is a problem with the base setup of SNMP and the **snmpd** subsystem itself. Check to ensure that the **snmpd** subsystem is running and follow the steps in the following sections to make sure that the basic **snmpinfo** command is working.

After you verify the basic function, you can query the PowerHA SystemMirror specific portion of the MIB with the following command:

```
snmpinfo -m dump -v -o /usr/sbin/cluster/hacmp.defs risc6000clsmuxpd
```

If the preceding command does not produce an output (and **snmpinfo -m dump** does), the problem is specific to the SystemMirror portion of the MIB.

Complete the steps in the Troubleshooting common SNMP problems section to resolve the following common issues with the `snmpdv3.conf` file that is included with the AIX operating system:

- Access to the **internet** portion of the SNMP Management Information Base (MIB) is commented out.
- In PowerHA SystemMirror Version 7.1.2, there is no **COMMUNITY** entry for the IPv6 loopback address.

However, even after these two issues are fixed, other issues might still interfere with the proper working of the SNMP-based status commands. Complete the steps in the Troubleshooting SNMP status commands section to resolve these issues. If the status commands still fail, complete the steps in the Troubleshooting `snmpdv3.conf` file section to resolve the rest of the issues.

- Troubleshooting common SNMP problems
- Troubleshooting SNMP status commands
- Troubleshooting `snmpdv3.con` file

Troubleshooting common SNMP problems

1. Check for access permission to the PowerHA SystemMirror portion of the SNMP Management Information Base (MIB) in the SNMP configuration file. Find the **defaultView** entries in the `/etc/snmpdv3.conf` file:

```
# grep defaultView /etc/snmpdv3.conf
#VACM_VIEW defaultView         internet                - included -
VACM_VIEW defaultView         1.3.6.1.4.1.2.2.1.1.1.0   - included -
VACM_VIEW defaultView         1.3.6.1.4.1.2.6.191.1.6   - included -
VACM_VIEW defaultView         snmpModules             - excluded -
```

```
VACM_VIEW defaultView          1.3.6.1.6.3.1.1.4        - included -
VACM_VIEW defaultView          1.3.6.1.6.3.1.1.5        - included -
VACM_VIEW defaultView          1.3.6.1.4.1.2.6.191      - excluded -
VACM_ACCESS  group1 - - noAuthNoPriv SNMPv1  defaultView - defaultView -
```

Beginning with AIX 7.1, as a security precaution, the snmpdv3.conf file is included with the **internet** access commented out. The preceding example shows the unmodified configuration file: the **internet** descriptor is commented out, which means that there is no access to most of the MIB, including the PowerHA SystemMirror information. (Other **included** entries provide access to other limited parts of the MIB.) By default in AIX 7.1 and later, the PowerHA SystemMirror SNMP-based status commands do not work, unless you edit the snmpdv3.conf file. There are two ways to provide access to the PowerHA SystemMirror MIB:

- Uncomment the following **internet** line in the snmpdv3.conf file to access the entire MIB:

```
VACM_VIEW defaultView  internet  -  included    -
```

- If you do not want access to the entire MIB, add the following line to the snmpdv3.conf file, which gives you access to the PowerHA SystemMirror MIB only:

```
VACM_VIEW defaultView  risc6000clsmuxpd  -  included    -
```

Note: After you edit the SNMP configuration file, you must stop and restart **snmpd**, and then refresh the cluster manager, by using the following commands:

```
stopsrc -s snmpd
startsrc -s snmpd
refresh -s clstrmgrES
```

Try the SNMP-based status commands again. If the commands work, you do not need to go through the rest of the section.

2. If you use PowerHA SystemMirror Version 7.1.2 or later, check for the correct IPv6 entries in the configuration files for **clinfoES** and **snmpd**. In PowerHA SystemMirror 7.1.2, an entry is added to the /usr/es/sbin/cluster/etc/clhosts file to support IPv6. However, the required corresponding entry is not added to the /etc/snmpdv3.conf file, which causes intermittent problems with the **clstat** command. To resolve the issue complete one of the following steps:

- If you do not plan to use IPv6, comment the line in the /usr/es/sbin/cluster/etc/clhosts file and restart **clinfoES**, by using the following commands:

```
# ::1     # PowerHA SystemMirror
stopsrc -s clinfoES
startsrc -s clinfoES
```

Try the SNMP-based status commands again. If the commands work, you do not need to go through the rest of the section.

- If you plan to use IPv6 in the future, add the following line to the /snmpdv3.conf file:

```
COMMUNITY public    public      noAuthNoPriv ::  0        -
```

If you are using a different community (other than public), substitute the name of that community for the word **public**.

Note: After you edit the SNMP configuration file, you must stop and restart **snmpd**, and then refresh the cluster manager, by using the following commands:

```
stopsrc -s snmpd
startsrc -s snmpd
refresh -s clstrmgrES
```

Try the SNMP-based status commands again. If the commands work, you do not need to go through the next section.

Troubleshooting SNMP status commands

1. Run the following command to check whether **snmpd** is running:

   ```
   lssrc -s snmpd
   ```

2. Run the following command to check whether cluster services are running:

   ```
   lssrc -ls clstrmgrES | grep state (looking for a state of ST_STABLE)
   ```

   If not, start the cluster services. None of the SNMP status commands work if the cluster services are not running.

3. If you are using the **clstat** command, check if the `/usr/es/sbin/cluster/etc/clhosts` file is correct. The `clhosts` file must contain a list of IP addresses of the PowerHA SystemMirror nodes with which the **clinfoES** daemon can communicate. (Persistent addresses are preferred. If the file contains addresses that do not belong to a cluster node, it might cause further problems.) If you edit the file on a system, you must restart **clinfoES** on that system.

   Cluster node

   - By default, the `clhosts` file is populated with the local host address. You can add entries for all the nodes in the cluster so that the **clstat** command works while the cluster services are running on the node.
   - Beginning with PowerHA SystemMirror Version 7.1.2, an entry for the IPv6 loopback address is added to the default `clhosts` file. As described in the Troubleshooting common SNMP problems section, you can either comment this line or add a line for the IPv6 loopback address to the SNMP configuration file.

   Client system
   By default the `clhosts` file is empty. You must add addresses for the cluster nodes.

4. If you are using the **clstat** command, run the following command to check whether **clinfoES** is running:

   ```
   lssrc -s clinfoES
   ```

   If not, run the following command to start it:

   ```
   startsrc -s clinfoES
   ```

   Tip: Start **clinfoES** every time you start cluster services to avoid this issue.

5. Check whether **snmpd** is listening at the **smux** port and if the cluster manager is connected. Run the following **netstat** command to list active sockets that use the **smux** port:

   ```
   # netstat -Aa | grep smux
   f1000e0002988bb8 tcp 0  *.smux *.* LISTEN
   f1000e00029d8bb8 tcp4 0 0 loopback.smux loopback.32776 ESTABLISHED
   f1000e00029d4bb8 tcp4 0 0 loopback.32776 loopback.smux ESTABLISHED
   f1000e000323fbb8 tcp4 0 0 loopback.smux loopback.34266 ESTABLISHED
   f1000e0001b86bb8 tcp4 0 0 loopback.34266 loopback.smux ESTABLISHED
   ```

   If you do not see a socket in the **LISTEN** state, use the following commands to stop and start **snmpd**:

   ```
   stopsrc -s snmpd; startsrc -s snmpd
   ```

6. After you have an **smux** socket in the **LISTEN** state, look for a socket pair in the **ESTABLISHED** state, with one of the sockets owned by the cluster manager. You can use the **rmsock** command to find which process owns the sockets. If you just restarted **snmpd**, ensure that there is a **LISTEN** socket at the **smux** port. If you do not see any **smux** socket in the **ESTABLISHED** state, you can either refresh the cluster manager (**refresh -s clstrmgrES**), or you can wait for a couple of minutes. Then, try the **netstat -Aa** command again. The cluster manager tries to connect to **snmpd** when services are started and then every few minutes after the services are started. The refresh command causes the cluster manager to try to connect to **snmpd** immediately. Do not use **stopsrc** and **startsrc** on the cluster manager.

7. Use **rmsock** to find the owners of the **smux** sockets in the **ESTABLISHED** state. Use the first field in the **netstat** output, which is the memory address of the socket, as an argument to **rmsock**. For example,

```
# rmsock f1000e00029d4bb8 tcpcb
The socket 0xf1000e00029d4808 is being held by proccess 4063356 (muxatmd).
# rmsock f1000e0001b86bb8 tcpcb
The socket 0xf1000e0001b86808 is being held by proccess 18546850 (clstrmgr).
```

In this example, there are two **ESTABLISHED** socket pairs. One between **snmpd** and **muxatmd** and one between **snmpd** and the cluster manager.

8. Try the SNMP-based status commands again. If the commands work, you do not need to go through the next section.

Troubleshooting snmpdv3.con file

1. Determine which version of **snmpd** is running, by using the following command:

```
#  ls -l /usr/sbin/snmpd
lrwxrwxrwx    1 root system 9 May 14 22:19 /usr/sbin/snmpd -> snmpdv3ne
```

snmpdv1 uses the /etc/snmpd.conf file and **snmpdv3** uses the /etc/snmpdv3.conf file.

Note: In the rest of these instructions, it is assumed that **snmpdv3** daemon, which is the default version, is running.

2. Check authentication and access control (authorization) settings for snmpdv3.conf file. **clinfoES**, **cldump**, and **cldisp** use community-based authentication. They use the first community that is listed in the configuration file. Although rare, it is possible to specify the community to **clinfoES**. To check this setting, use the following command:

```
odmget SRCsubsys | grep -p clinfo
```

Look for the value of the **cmdargs** field.

- If the field is empty, **clinfoES** uses the first **COMMUNITY** entry in the configuration file.
- If the field is set to -c **community_name**, **clinfoES** uses **community_name**.

    Note: If you want to change the community that is used by **clinfoES**, use the **chssys** command. After you change the community that is used by **clinfoES**, you must restart **clinfoES**.

3. Find the first SNMP community in the snmpdv3.conf file.

```
# grep -i comm /etc/snmpdv3.conf | grep -v ^#
COMMUNITY powerha powerha noAuthNoPriv 0.0.0.0    0.0.0.0        -
COMMUNITY test test noAuthNoPriv 0.0.0.0     0.0.0.0        -
```

In this example, the first community is **powerha**.

- If there are no uncommented community entries, you must add an entry in the snmpdv3.conf file. You can use these entries as a template. Use any text string as the community name (although **public** is not considered a good choice because it is common). The community name must be the second and third fields in the line.
- For the changes to take effect, you must restart **snmpd** after you edit the file.

4. The **snmpdv3** daemon uses view-based access control model (VACM) for access control. Find the **VACM_GROUP**, the **VACM_ACCESS**, and the **VACM_VIEW** entries that are associated with the community you are using.

    a. Find the group that is associated with the first community. Search in the configuration file for the community name. For example,

```
# grep powerha /etc/snmpdv3.conf
VACM_GROUP group1 SNMPv1   powerha  -
TARGET_PARAMETERS trapparms1 SNMPv1  SNMPv1  powerha  noAuthNoPriv -
COMMUNITY powerha    powerha    noAuthNoPriv 0.0.0.0   0.0.0.0        -
```

In this example the **VACM_GROUP** is **group1**.

b. Find the view that is associated with this group by searching for the group you identified. The view is listed in a **VACM_ACCESS** entry.

```
# grep group1 /etc/snmpdv3.conf
VACM_GROUP group1 SNMPv1  powerha  -
VACM_ACCESS  group1 - - noAuthNoPriv SNMPv1  defaultView - defaultView -
```

The syntax of a **VACM_ACCESS** entry is as follows:

```
VACM_ACCESS groupName contextPrefix contextMatch securityLevel
securityModel readView writeView notifyView storageType
```

Look for the name of the view for **readView** access. In this example, **defaultView** is used for **readView** and **notifyView** access for group **group1**. No access is provided for **writeView** and **storageType**.

c. Find the **VACM_VIEW** entries that are associated with this community by searching for the view you identified:

```
# grep defaultView /etc/snmpdv3.conf
#VACM_VIEW defaultView        internet                  - included -
VACM_VIEW defaultView        1.3.6.1.4.1.2.2.1.1.1.0   - included -
VACM_VIEW defaultView        1.3.6.1.4.1.2.6.191.1.6   - included -
VACM_VIEW defaultView        snmpModules               - excluded -
VACM_VIEW defaultView        1.3.6.1.6.3.1.1.4         - included -
VACM_VIEW defaultView        1.3.6.1.6.3.1.1.5         - included -
VACM_VIEW defaultView        1.3.6.1.4.1.2.6.191       - excluded -
VACM_ACCESS  group1 - - noAuthNoPriv SNMPv1  defaultView - defaultView -
```

i) Look for a **VACM_VIEW** entry that gives access to the PowerHA SystemMirror MIB. Locations in the MIB are identified either by a string of numbers (object identifier (OID)) or by a name (object descriptor). In this example, the first entry uses the object descriptor **internet**. That corresponds to the OID **1.3.6.1**. If this line is uncommented, it allows access to the entire MIB, that is 1.3.6.1 and everything that starts with 1.3.6.1, which is effectively the entire SNMP MIB.

ii) However, in this example, the **internet** descriptor is commented out, which means that there is no access at that level. Beginning with AIX 7.1, as a security precaution, the snmpdv3.conf file is included with the **internet** access commented out. By default in AIX 7.1 or later, the PowerHA SystemMirror SNMP-based status commands do not work, unless you edit the snmpdv3.conf file. Also, ensure that the relevant **VACM_VIEW** entry has the word **included** in the second last field and not **excluded**.

iii) As described in the Troubleshooting common SNMP problems section, there are two ways to provide access to the PowerHA SystemMirror MIB:

- Uncomment the **internet** line in snmpdv3.conf file to access the entire MIB.

- Add a line that provides access to the PowerHA SystemMirror MIB only. The PowerHA SystemMirror MIB can be identified by the object descriptor or by the OID.

5. Edit the snmpdv3.conf file to ensure that the PowerHA SystemMirror MIB is accessible for the first community. You must make sure that the first **COMMUNITY** entry in the file maps to a **VACM_GROUP** entry that maps to a **VACM_ACCESS** entry that maps to a **VACM_VIEW** that includes the PowerHA SystemMirror MIB. In this example, the only change that is needed is to add a **VACM_VIEW** entry for the **risc6000clsmuxpd** object descriptor:

```
VACM_VIEW defaultView      risc6000clsmuxpd     - included -
```

6. If you edited the snnmpdv3.conf file, restart **snmpd**.

Note: You must use the **stopsrc** and **startsrc** commands, instead of the **refresh** command for **snmpd**.

```
stopsrc -s snmpd; startsrc -s snmpd
```

7. Repeat steps 5, 6, 7 as described in the Troubleshooting SNMP status commands section to ensure that the cluster manager is connected to **snmpd**.

8. Try the SNMP-based status commands again.

Related concepts

PowerHA SystemMirror startup issues
This section describes solutions for potential PowerHA SystemMirror startup issues.

Disk and file system issues
This section describes solutions for potential disk and file system issues.

PowerHA SystemMirror takeover issues
This section describes solutions for potential takeover issues.

Network and switch issues
This section describes solutions for potential network and switch issues.

Cluster communications issues
This section describes solutions for potential cluster communication issues.

Client issues
This section describes solutions for potential PowerHA SystemMirror client issues.

Miscellaneous issues
This section describes solutions for potential non-categorized PowerHA SystemMirror issues.

PowerHA SystemMirror LV encryption issue

The following section describes solution for potential PowerHA SystemMirror logical volume (LV) encryption issue.

System cannot mount specified file systems

Problem

When PowerHA SystemMirror acquires a resource group and if the associated logical volume is partially encrypted or decrypted, the file systems mount operation might fail and the resource group moves to an ERROR state. The `hacmp.out` log file displays an error message similar to the following example:

```
ERROR: cl_activate_fs(0.575):/<fsname>[fs_mount:86]
LC_ALL=C\nlsfs: No record matching '<fsname>' was found in /etc/filesystems.'
```

Solution

When you enable or disable the LV encryption by using the platform keystore (PKS) or key server authentication method, the LV encryption might fail and the logical volume gets locked because of the following reasons:

• A node got rebooted during LV encryption or decryption process.

• LV encryption or decryption process is forcefully stopped by using the Ctrl+C signal key.

During LV encryption or decryption process, conversion files (temporary files) are created and stored in the `/var/hdcrypt` directory on the node where the encrypt or decrypt process is running. These conversion files are automatically deleted after successful completion of encrypt or decrypt process. In case of failures, conversion files are used to continue the encrypt or decrypt process. However, you cannot continue encrypt or decrypt process on other nodes, which are part of the cluster.

If the node gets rebooted during an encrypt or decrypt process, you must copy the conversion files from the `/var/hdcrypt` directory on the failed node to other clusters nodes where you want to continue the encrypt or decrypt process.

To resolve the mount operation error, complete the following steps on the node where the failure is reported.

1. Run the **hdcryptmgr authunlock <lv name>** command.
2. Resume the encrypt or decrypt process by running the **clmgr modify lv** command.

To check the status of LV encryption, run the following commands:

```
hdcryptmgr showlv <lv name> -v
```

```
hdcryptmggr showmd <lv_name>
```

To check for LVM encryption failures, run the following command:

```
alog -t hdcrypt -o
```

Note: To run the **hdcryptmgr** command, the volume group that is associated with the logical volume must be in an active state or in the concurrent state.

Miscellaneous issues

This section describes solutions for potential non-categorized PowerHA SystemMirror issues.

If you are investigating resource group movement in PowerHA SystemMirror for why an **rg_move** event occurred, you must check the /var/hacmp/log/hacmp.out file. In general, given the recent changes in the way resource groups are handled and prioritized in fallover circumstances, particularly in PowerHA SystemMirror, the hacmp.out file and its event summaries are important in tracking the activity and resulting location of your resource groups. In addition, with parallel processing of resource groups, the hacmp.out file reports details that are not seen in the cluster history log or the clstrmgr.debug file. Check the hacmp.out log file early when you are investigating resource group movement after takeover activity.

- hacmp.out file displays limited output for the tail -f command
- Cluster verification displays message without configuring error notification
- Scenarios that display the config_too_long message
- Console displays SNMP messages
- Cluster nodes fail to fallover after system restart
- Deleted objects displayed in NetView map
- Pressing F1 in SMIT panel does not display help.
- Getting arithmetic syntax error with locale setting
- Event summaries file too large
- Application monitor issues
- Event summaries file does not display resource group information as expected
- Cluster disk replacement process fails
- rg_move event processes several resource groups at once
- File system fails to unmount
- Dynamic reconfiguration sets a lock
- Resource group fails to come online in a WPAR
- Nodes and repository disks fail simultaneously
- Commenting out the CAA entry in the /etc/inetd.conf file

hacmp.out file displays limited output for the tail -f command

Problem

Only script start messages appear in the /var/hacmp/log/hacmp.out file. The script that is specified in the message is not an executable, or the DEBUG level is set to low.

Solution

Add permission to run the script by using the `chmod` command, and ensure that the DEBUG level is set to `high`.

Cluster verification displays message without configuring error notification

Problem

The following message is displayed regardless of whether the Auto Error Notification is configured:

```
"Remember to redo automatic error notification if configuration
has changed."
```

Solution

Ignore this message if the Auto Error Notification is not configured.

Scenarios that display the config_too_long message

The **config_too_long** message appears each time a cluster event takes more time to complete than a specified timeout period.

In PowerHA SystemMirror versions earlier than 4.5, the timeout period was fixed for all cluster events and set to 360 seconds by default. If a cluster event, such as a **node_up** or a **node_down** event, lasted longer than 360 seconds, then every 30 seconds PowerHA SystemMirror displays a **config_too_long** warning message that was logged in the `hacmp.out` file.

In PowerHA SystemMirror you can customize the time period that is allowed for a cluster event to complete before PowerHA SystemMirror issues a system warning for it.

If the **config_too_long** message is logged in the `hacmp.out` file, the message might be similar to the following sample:

```
config_too_long $sec $event_name $argument<
```

where,

- `$event_name`: Indicates the reconfiguration event that failed.
- `$argument`: Indicates the parameter that is used by the event.
- `$sec`: Indicates the number of seconds before the message was sent.

In PowerHA SystemMirror Version 4.5 or earlier, **config_too_long** messages continued to be appended to the `hacmp.out` file every 30 seconds until action was taken.

Starting with PowerHA SystemMirror Version 4.5, for each cluster event that does not complete within the specified event duration time, **config_too_long** messages are logged in the `hacmp.out` file and sent to the console according to the following pattern:

- The first five **config_too_long** messages appear in the `hacmp.out` file at 30-second interval.
- The next set of five messages appears at interval that is double the previous interval until the interval reaches 1 hour.
- These messages are logged every hour until the event is complete or the event is ended on that node.

Problem 1

Activities that the script is performing take longer than the specified time to complete. For example, this scenario might occur with events that involve many disks or complex scripts.

Solution

– Determine what is taking so long to run, and correct or streamline that process if possible.
– Increase the wait time before you run **config_too_long**.

You can customize **Event Duration Time** by using the **Change/Show Time Until Warning** panel in SMIT. Access this panel through the **Extended Configuration** > **Extended Event Configuration** SMIT panel.

Problem 2

A command is not responding and event script is waiting before you resume execution. If so, you can probably see the command in the AIX process table (**ps -ef**). It is most likely the last command in the `/var/hacmp/log/hacmp.out` file before the **config_too_long** script output.

Solution

End the command that is not responding.

Problem 3

The foreground startup process is specified for an application controller start script. However, the start script does not exist.

Note: This problem exists if you are using PowerHA SystemMirror Version 7.1.1, or later.

Solution

Examine the start script to see whether it is functioning correctly. If the script is not responding, use the startup option along with a startup monitor instead of foreground startup.

Console displays SNMP messages

Problem

The `/etc/syslogd` file is changed to send the **daemon.notice** output to the `/dev/console` file.

Solution

Edit the `/etc/syslogd` file to redirect the **daemon.notice** output to `/usr/tmp/snmpd.log`. The `snmpd.log` file is the default location for logging messages.

Cluster nodes fail to fallover after system restart

Problem

Cluster nodes did not fallover after the system is restarted.

Solution

To prevent unplanned system restart from disrupting a fallover in your cluster environment, all nodes in the cluster must either have the **Automatically REBOOT a system after a crash** field on the **Change/Show Characteristics of Operating System** SMIT panel set to `false`, or you must keep the IBM System p key-in Secure mode during normal operation.

Both measures prevent a system from rebooting if the **shutdown** command is run inadvertently. Without one of these measures in place, if an unplanned restart occurs the activity against the disks on the rebooting node can prevent other nodes from successfully acquiring the disks.

Deleted objects displayed in NetView map

Problem

Previously deleted or extraneous object symbols appeared in the Tivoli NetView for z/OS map.

Solution

Rebuild the Tivoli NetView for z/OS database.

To rebuild the Tivoli NetView for z/OS database, perform the following steps on the Tivoli NetView for z/OS server:

1. Stop all Tivoli NetView for z/OS daemons:

   ```
   /usr/OV/bin/ovstop -a
   ```

2. Remove the database from the Tivoli NetView for z/OS server:

   ```
   rm -rf /usr/OV/database/*
   ```

3. Start the Tivoli NetView for z/OS object database:

   ```
   /usr/OV/bin/ovstart ovwdb
   ```

4. Restore the Tivoli NetView for z/OS or HAView fields:

   ```
   /usr/OV/bin/ovw -fields
   ```

5. Start all Tivoli NetView for z/OS daemons.

   ```
   /usr/OV/bin/ovstart -a
   ```

Pressing F1 in SMIT panel does not display help

Problem

Pressing F1 in SMIT panel does not display help.

Solution

Help can be displayed only if the LANG variable is set to one of the languages supported by PowerHA SystemMirror, and if the associated PowerHA SystemMirror message catalogs are installed. The following languages are supported by PowerHA SystemMirror:

- en_US
- ja_JP

To list the installed locales (the bsl LPPs), enter the following command:

```
 locale -a
```

To list the active locale, enter the following command:

```
 locale
```

Since the LANG environment variable determines the active locale, if LANG=en_US, the locale is en_US.

Getting arithmetic syntax error with locale setting

Problem

Following arithmetic syntax issue may occur while working with PowerHA cluster with locale setting as pt_BR, fr_FR or, sl_SI etc. For example

```
# export LANG=pt_BR
# typeset -F4 var=0.00
ksh93: typeset: 0.00: arithmetic syntax error
# (( sum = sum + 0.5 ))
ksh93:  sum = sum + 0.5 : arithmetic syntax error
```

This issue can be avoided if the LANG is set to one of the languages like en_Us and ja_JP.

Solution

The problem occurs due to inappropriate `LC_NUMERIC` environment variable setting. When the `LANG` environmental variable is set to any of the languages like `pt_BR`, `fr_FR` or, `sl_SI` etc., `LC_NUMERIC` variable by default considers it's decimal delimiter as comma ("`,`").

```
# locale
LANG=fr_FR
LC_COLLATE="fr_FR"
LC_CTYPE="fr_FR"
LC_MONETARY="fr_FR"
LC_NUMERIC="fr_FR"
LC_TIME="fr_FR"
LC_MESSAGES="fr_FR"
LC_ALL=
# locale -k LC_NUMERIC
decimal_point=","
thousands_sep=" "
grouping="3"
```

In order to perform any numeric operations with decimals, `LC_NUMERIC` variable should have decimal delimiter as period (".").

To avoid this problem, set `LC_NUMERIC=C` along with `LANG` setting.

Event summaries file too large

Problem

In PowerHA SystemMirror event summaries are pulled from the **hacmp.out** file and can be viewed by using the **Problem Determination Tools** > **PowerHA SystemMirror Viewing and Management** > **View/Save/Delete Event Summaries** > **View Event Summaries** option in SMIT. This display includes resource group status and location information at the end. The resource group information is gathered by the **clRGinfo** command, and might take extra time if the cluster is not running when you run the **View Event Summaries** option.

Solution

The **clRGinfo** command displays resource group information quickly when the cluster is running. If the cluster is not running, there is a delay in displaying the resource group information.

Application monitor issues

If you are running application monitors, you might encounter occasional problems or situations in which you want to check the state or the configuration of a monitor. Here are some possible problems and ways to diagnose and act on them.

Problem 1

Checking the State of an Application Monitor. In some circumstances, it might not be clear whether an application monitor is running or not. To check on the state of an application monitor, run the following command:

```
ps -ef | grep <application controller name> | grep clappmond
```

This command produces a long line of verbose output if the application is being monitored.

If there is no output, the application is not being monitored.

Solution

The application monitor might not if:

- No monitor is configured for the application controller.
- The monitor is not started because the stabilization interval is not completed.
- The monitor is in a suspended state.
- The monitor is not configured properly.

- An error occurred.

Check to see whether a monitor is configured, the stabilization interval is passed, and the monitor is not placed in a suspended state. If any configuration is incorrect, reexamine the original configuration of the monitor in SMIT and reconfigure as needed.

Problem 2

Application monitor does not perform specified failure action. The specified failure action does not occur when an application fails.

Solution

Check the **Restart Interval**. If the **Restart Interval** is set too short, the **Restart Counter** might be reset to zero too quickly, resulting in an endless series of restart attempts and no other action taken.

Problem 3

Application monitor does not always indicate that the application is working correctly.

Solution

- Check that the monitor is written to return the correct exit code in all cases. The return value must be zero if the application is working fine, and it must be a nonzero value if the application fails.
- Check all possible paths through the code, including error paths to make sure that the exit code is consistent with the application state.

Problem 4

Unable to determine whether and when the monitor is run.

Solution

Check the log files that are created by the monitor. The monitor can log messages by printing them to the standard output **stdout** file. For long running monitors, the output is stored in the **/var/hacmp/log/clappmond**application monitor name.resource group name.monitor**.log** file. For startup monitors, this output is stored in the **/var/hacmp/log/clappmond.** application server name.resource group name.monitor**.log** file. The monitor log files are overwritten, each time the application monitor runs.

Event summaries file does not display resource group information as expected

Problem

PowerHA SystemMirror fails to selectively move the affected resource group to another cluster node when a volume group quorum loss occurs.

Solution

If quorum is lost for a volume group that belongs to a resource group on a cluster node, the system checks whether the LVM_SA_QUORCLOSE error is displayed in the node's AIX error log file. The system informs the Cluster Manager to selectively move the affected resource group. PowerHA SystemMirror uses this error notification method only for mirrored volume groups with quorum enabled.

If fallover does not occur, check that the LVM_SA_QUORCLOSE error appeared in the AIX error log. When the AIX error log buffer is full, new entries are discarded until buffer space becomes available and an error log entry informs you of this problem. To resolve this issue, increase the size of the AIX error log internal buffer for the device driver.

Cluster disk replacement process fails

Problem

The disk replacement process fails while the **replacepv** command was running.

Solution

Be sure to delete the `/tmp/replacepv` directory, and attempt the replacement process again.

You can also try running the process on another disk.

rg_move event processes several resource groups at once

Problem

In the `hacmp.out` file, an **rg_move** event processes multiple non-concurrent resource groups in one operation.

Solution

In clusters with dependencies, PowerHA SystemMirror processes all resource groups on **node_up** events, though **rg_move** events. During a single **rg_move** event, PowerHA SystemMirror can process multiple non-concurrent resource groups within one event.

File system fails to unmount

Problem

A file system is not unmounted properly during an event such as when you stop cluster services with the option to bring resource groups offline.

Solution

One of the more common reasons for a file system to fail being unmounted when you stop cluster services with the option to bring resource groups offline is because the file system is busy. To unmount a file system successfully, no processes, or users can be accessing it at the time. If a user or process is accessing the file system, the file system cannot be unmounted. The same issue might result if a file is deleted but is still open.

The script to stop an application must include a check to ensure that the shared file systems are not in use or deleted and in the open state by using the **fuser** command. The script must use the **fuser** command to see what processes or users are accessing the file systems in question. The PIDs of these processes can then be acquired and ended to free the file system so it can be unmounted. For more information, see **fuser** Command.

Dynamic reconfiguration sets a lock

Problem

When you run a dynamic reconfiguration (DARE) operation, an error message might be displayed on a DARE lock if another DARE operation is in process, or if a previous DARE operation did not complete. The error message indicates that lock must be cleared if a DARE operation is not started or if a previous DARE operation did not complete.

Solution

Examine the `/var/hacmp/log/hacmp.out` logs on the cluster nodes to determine the reason for the previous DARE failure. A **config_too_long** entry is displayed in `hacmp.out` file where an operation in an event script took too long to complete. If **hacmp.ou** t indicates that a script failed to complete due to some error, correct this problem and manually complete the remaining steps that are necessary to complete the event.

Run the PowerHA SystemMirror SMIT **Problem Determination Tools** > **Recover from PowerHA SystemMirror Script Failure** option to bring the nodes in the cluster to the next complete event state.

You can clear the DARE lock by selecting the PowerHA SystemMirror SMIT option **Problem Determination Tools** > **Release Locks Set by Dynamic Configuration** if the PowerHA SystemMirror SMIT **Recover from PowerHA SystemMirror Script Failure** step did not do so.

Resource group fails to come online in a WPAR

Problem

Resource Group fails to come online in a WPAR on a particular node.

Solution

1. Verify that the node in question is WPAR-capable. You must install the **bos.wpars** fileset on an AIX node with WPAR capability. If the node is not WPAR-capable, then the resource group does not run in the WPAR. To check whether the **bos.wpars** fileset is installed, enter the following command:

   ```
   lslpp -L "bos.wpars"
   ```

2. On the specified node, verify that there is a WPAR with the same name as the WPAR-enabled resource group. Use the `lswpar <resource group name>` command to verify. If there is no WPAR with the specified name, create it using the **mkwpar** command. After you create a WPAR, make sure that all the user-defined scripts that are associated with the WPAR-enabled resource group are accessible within the WPAR.

3. Ensure that the file systems on the node are not full. If so, free up some disk space by moving some files to external storage.

4. Verify that the **rsh** service is enabled in the corresponding WPAR by completing the following steps:

   • Check that the **inetd** service is running in WPAR by entering the following command in WPAR:

   ```
   lssrc -s inetd
   ```

   If the **inetd** service is not active, then start the service by using the **startsrc** command.

   • Ensure that rsh is listed as a known service in `/etc/inetd.conf` file in the WPAR.

Nodes and repository disks fail simultaneously

Problem

Nodes and repository disks fail simultaneously during an event such as a data center failure.

Solution

In a simultaneous node and repository disk failure, such as when a data center fails, it might be necessary to replace the repository disk before all nodes restart.

1. To replace the repository disk, use the following System Management Interface Tool (SMIT) path:

   ```
   $ smitty sysmirror
   >Problem Determination Tools > Replace the Primary Repository Disk
   ```

   Note: A node that is in the **DOWN** state while the repository disk is being replaced continues to access the 'original' repository disk even after the restart. If the 'original' repository disk becomes available again, Cluster Aware AIX (CAA) cluster services start to use that disk. The node remains in the **DOWN** state.

2. To check the status of a node, enter the following command:

   ```
   lscluster -m
   ```

   This command produces an output that is similar to the following output:

   ```
   Calling node query for all nodes...
   Node query number of nodes examined: 2
         Node name: ha1clA
         Cluster shorthand id for node: 1
         UUID for node: 1ab63438-d7ed-11e2-91ce-46fc4000a002
         State of node: DOWN  NODE_LOCAL
         ...
        -----------------------------------------------------
   ```

```
        Node name: ha2clA
        Cluster shorthand id for node: 2
        UUID for node: 1ac309e2-d7ed-11e2-91ce-46fc4000a002
        State of node: UP
        ...
        Points of contact for node: 2
        ----------------------------------------
        Interface    State  Protocol    Status
        ----------------------------------------
        en0          UP     IPv4        none
        en1          UP     IPv4        none
```

3. To force a previously failed node to use the 'new' repository disk, enter the following commands at the affected node:

   a. **$ export CAA_FORCE_ENABLED=true**

   b. **$ clusterconf -fu**

4. To check whether the CAA cluster services are not active, enter the following command:

   ```
   lscluster -c
   ```

   Note: You might need to wait up to 10 minutes for the node to join the CAA cluster again, by using the 'new' repository disk.

5. To verify whether the CAA cluster services are started successfully, enter the following command:

   a. **lscluster -c**

   b. **lscluster -m**

6. Before you restart PowerHA SystemMirror at the affected node, the PowerHA SystemMirror configuration needs to be synchronized. The synchronization needs to be started at a node, which was in the **UP** state while the repository disk was replaced. To start the verification and synchronization process at a node, use the following SMIT path:

   ```
   $ smitty sysmirror
   >Cluster Nodes and Networks > Verify and Synchronize Cluster Configuration
   ```

   Note: If there are multiple nodes available and PowerHA SystemMirror is not running on all of them, you need to choose an active node to start the synchronization.

After the verification and synchronization is successfully completed in Step 6, you can restart PowerHA SystemMirror at the previously failed node, by using the following SMIT path:

```
$ smitty sysmirror
>System Management (C-SPOC) > PowerHA SystemMirror Services > Start Cluster Services
```

Commenting out the CAA entry in the /etc/inetd.conf file

Problem

A commented Cluster Aware AIX (CAA) entry is causing an error during cluster creation process.

Solution

After you install PowerHA SystemMirror, verify that the following Cluster Aware AIX (CAA) entry is commented out in the /etc/inetd.conf file before creating PowerHA SystemMirror cluster.

```
caa_cfg stream tcp6 nowait root /usr/sbin/clusterconf
clusterconf >>/var/adm/ras/clusterconf.log 2>&1
```

Starting with AIX 7.3, CAA entry in the /etc/inetd.conf file is already commented. During PowerHA SystemMirror installation, this CAA entry in the /etc/inetd.conf file is commented out for the following releases:

- PowerHA SystemMirror Version 7.2.3 SP6, or later
- PowerHA SystemMirror Version 7.2.4 SP4, or later

- PowerHA SystemMirror Version 7.2.5 SP2, or later
- PowerHA SystemMirror Version 7.2.6

If AIX 7.3 is installed with an earlier version of PowerHA SystemMirror than the versions listed earlier, the CAA entry is not commented out automatically. You must manually comment out the CAA entry in the /etc/inetd.conf file.

To uncomment the CAA entry, run the following command on all the nodes of the cluster before creating the cluster:

```
chsubserver -a -r inetd -v caa_cfg -p tcp6
```

Related concepts

PowerHA SystemMirror startup issues
This section describes solutions for potential PowerHA SystemMirror startup issues.

Disk and file system issues
This section describes solutions for potential disk and file system issues.

PowerHA SystemMirror takeover issues
This section describes solutions for potential takeover issues.

Network and switch issues
This section describes solutions for potential network and switch issues.

Cluster communications issues
This section describes solutions for potential cluster communication issues.

Client issues
This section describes solutions for potential PowerHA SystemMirror client issues.

SNMP-based status commands issues
This section describes solutions for problems that can cause SNMP-based status commands such as clstat, cldump, and cldisp to fail.

Notices

This information was developed for products and services offered in the US.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

The performance data and client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work must include a copyright notice as follows:

© (your company name) (year).

Portions of this code are derived from IBM Corp. Sample Programs.

© Copyright IBM Corp. _enter the year or years_.

Privacy policy considerations

IBM Software products, including software as a service solutions, ("Software Offerings") may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering's use of cookies is set forth below.

This Software Offering does not use cookies or other technologies to collect personally identifiable information.

If the configurations deployed for this Software Offering provide you as the customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, see IBM's Privacy Policy at http://www.ibm.com/privacy and IBM's Online Privacy Statement at http://www.ibm.com/privacy/details the section entitled "Cookies, Web Beacons and Other Technologies" and the "IBM Software Products and Software-as-a-Service Privacy Statement" at http://www.ibm.com/software/info/product-privacy.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at Copyright and trademark information at www.ibm.com/legal/copytrade.shtml.

Index

Special Characters

.info 36
.odm 35
/usr/es/sbin/cluster/history/cluster.mmddyyyy
    understanding 22
/usr/es/sbin/cluster/snapshots/ clsnapshot.log 11
/usr/es/sbin/cluster/wsm/logs/ wsm_smit.log 11
/var/ha/log/grpglsm 11
/var/ha/log/grpsvcs 11
/var/hacmp/adm/cluster.log 11, 13
/var/hacmp/adm/history/cluster.mmddyyyy 11
/var/hacmp/clverify/clverify.log 11
/var/hacmp/log/ cl_testtool.log 11
/var/hacmp/log/ clconfigassist.log 11
/var/hacmp/log/ clstrmgr.debug.long 11
/var/hacmp/log/autoverify.log 11
/var/hacmp/log/clavan.log 11
/var/hacmp/log/clinfo.log 11
/var/hacmp/log/clstrmgr.debug 11
/var/hacmp/log/clutils.log 11
/var/hacmp/log/cspoc.log 11
/var/hacmp/log/cspoc.log.long/var/hacmp/log/
cspoc.log.remote 11
/var/hacmp/log/hacmp.out
    event preamble 15
    event summary 15
    setting level of information recorded 19
    viewing compiled event summaries 20
/var/hacmp/log/migration.log 11
/var/hacmp/log/oraclesa.log 11
/var/hacmp/log/sa.log 11

A

AIX operating system
    checking 44
application
    checking 31

C

checking
    AIX operating system 44
    applications 31
    cluster communications daemon 46
    cluster configuration 33
    cluster snapshot 34
    disk adapters 45
    disks 45
    file system information 40
    file systems 40
    IP address 43
    logical volume manager 37
    logical volumes 39
    mount points 40
    netmask 43

checking *(continued)*
    permissions 40
    physical networks 45
    physical volumes 38
    point-to-point connectivity 43
    PowerHA SystemMirror components 32
    system hardware 46
    TCP/IP subsystem 41
    volume group
        definitions 37
        varyon state 38
client issues 71
cluster
    checking communications daemon 46
    checking configuration 33
    checking snapshot 34
    collecting log files 22
    communication issues 65
    reviewing message log files 11
    stopping 3
    tracking resource group 23
    understand log files 13, 14
    viewing log files 11
cluster history log
    understanding 22
cluster log
    managing 23
cluster state information file 36
cluster.log 13
collecting
    cluster log files 22
configuration database data file 35
cron job
    making highly available 9

D

diagnostic utilities
    using 4
disk
    checking 45
disk adapter
    checking 45
disk issues 53

E

event preamble 15
event summary
    parallel processing order 23
    saving 21
    viewing compiled hacmp.out 20

F

file system