Reliable Scalable Cluster Technology
Version 3.1.5.0

Troubleshooting RSCT

Reliable Scalable Cluster Technology
Version 3.1.5.0

Troubleshooting RSCT

Note
Before using this information and the product it supports, read the information in "Notices" on page 241.

This edition applies to Reliable Scalable Cluster Technology Version 3.1.5.0 and to all subsequent releases and modifications until otherwise indicated in new editions.

Contents

About this document

This information describes how to diagnose and resolve problems related to the various components and subsystems of IBM® Reliable Scalable Cluster Technology (RSCT). The AIX® implementation of RSCT is part of the IBM AIX operating system. The Linux implementation of RSCT is included in these IBM licensed programs: Cluster Systems Management (CSM) for Linux and Tivoli® System Automation for Multiplatforms. The Solaris and Windows implementations of RSCT are also included in Tivoli System Automation for Multiplatforms.

Before using this information to diagnose RSCT problems, you should first verify that the RSCT components have been installed. To do this, see the RSCT installation and software verification chapter in *Administering RSCT* guide. This chapter also contains information about fixes required by various Linux distributions.

This information is a companion volume to *Messages for RSCT* guide, which lists the error messages that could be generated by each RSCT component. While this guide describes appropriate user responses to messages that are generated by RSCT components, this information contains additional and more detailed diagnostic procedures.

Highlighting

The following highlighting conventions are used in this document:

Table 1. Conventions

Convention	Usage
bold	**Bold** words or characters represent system elements that you must use literally, such as commands, flags, path names, directories, file names, values, PE component names (**poe**, for example), and selected menu options.
bold underlined	**Bold underlined** keywords are defaults. These take effect if you do not specify a different keyword.
`constant width`	Examples and information that the system displays appear in `constant-width` typeface.
italic	*Italic* words or characters represent variable values that you must supply. *Italics* are also used for information unit titles, for the first use of a glossary term, and for general emphasis in text.
<*key*>	Angle brackets (less-than and greater-than) enclose the name of a key on the keyboard. For example, <**Enter**> refers to the key on your terminal or workstation that is labeled with the word *Enter*.

\	In command examples, a backslash indicates that the command or coding example continues on the next line. For example:
	`mkcondition -r IBM.FileSystem -e "PercentTotUsed > 90" \`
	`-E "PercentTotUsed < 85" -m d "FileSystem space used"`

{*item*}	Braces enclose a list from which you must choose an item in format and syntax descriptions.
[*item*]	Brackets enclose optional items in format and syntax descriptions.
<**Ctrl-***x*>	The notation <**Ctrl-***x*> indicates a control character sequence. For example, <**Ctrl-c**> means that you hold down the control key while pressing <**c**>.
item...	Ellipses indicate that you can repeat the preceding item one or more times.
|	• In *synopsis* or *syntax* statements, vertical lines separate a list of choices. In other words, a vertical line means *Or*. • In the left margin of the document, vertical lines indicate technical changes to the information.

Entering commands

When you work with the operating system, you typically enter commands following the shell prompt on the command line. The shell prompt can vary. In the following examples, $ is the prompt.

To display a list of the contents of your current directory, you would type `ls` and press the **Enter** key:

```
$ ls
```

When you enter a command and it is running, the operating system does not display the shell prompt. When the command completes its action, the system displays the prompt again. This indicates that you can enter another command.

The general format for entering operating system commands is:

Command Flag(s) Parameter

The flag alters the way a command works. Many commands have several flags. For example, if you type the `-l` (long) flag following the **ls** command, the system provides additional information about the contents of the current directory. The following example shows how to use the `-l` flag with the **ls** command:

```
$ ls -l
```

A parameter consists of a string of characters that follows a command or a flag. It specifies data, such as the name of a file or directory, or values. In the following example, the directory named `/usr/bin` is a parameter:

```
$ ls -l /usr/bin
```

When entering commands in, it is important to remember the following items:

- Commands are usually entered in lowercase.
- Flags are usually prefixed with a - (minus sign).
- More than one command can be typed on the command line if the commands are separated by a ; (semicolon).
- Long sequences of commands can be continued on the next line by using the \ (backslash). The backslash is placed at the end of the first line. The following example shows the placement of the backslash:

  ```
  $ cat /usr/ust/mydir/mydata > \
  /usr/usts/yourdir/yourdata
  ```

When certain commands are entered, the shell prompt changes. Because some commands are actually programs (such as the telnet command), the prompt changes when you are operating within the command. Any command that you issue within a program is known as a subcommand. When you exit the program, the prompt returns to your shell prompt.

The operating system can operate with different shells (for example, Bourne, C, or Korn) and the commands that you enter are interpreted by the shell. Therefore, you must know what shell you are using so that you can enter the commands in the correct format.

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.

RSCT versions

This edition applies to RSCT version, release, modification, and fix number 3.1.5.0.

You can use the **ctversion** command to find out which version of RSCT is running on a particular AIX, Linux, Solaris, or Windows node. For example:

```
/usr/sbin/rsct/install/bin/ctversion
```

An example of the output follows:

```
# /usr/sbin/rsct/install/bin/ctversion
rlis1313a 3.1.5.0
```

where, `rlis1313a` is the RSCT build level.

On the AIX operating system, you can also use the **lslpp** command to find out which version of RSCT is running on a particular AIX node. For example:

```
lslpp -L rsct.core.utils
```

An example of the output follows:

```
  Fileset                      Level    State  Type  Description (Uninstaller)
  ----------------------------------------------------------------------------
  rsct.core.utils              3.1.5.0    C     F    RSCT Utilities

State codes:
 A -- Applied.
 B -- Broken.
 C -- Committed.
 E -- EFIX Locked.
 O -- Obsolete.  (partially migrated to newer version)
 ? -- Inconsistent State...Run lppchk -v.

Type codes:
 F -- Installp Fileset
 P -- Product
 C -- Component
 T -- Feature
 R -- RPM Package
```

On the Linux operating system, you can also use the **rpm** command to find out which version of RSCT is running on a particular Linux or Solaris node. For example:

```
rpm -qa | grep rsct.basic
```

On the Windows operating system, you can also perform the following steps to find out which version of RSCT is running on a particular Windows node:

1. Click the Windows **start** button.
2. Select **All Programs**.
3. Select **Tivoli SA MP Base**.
4. Click **SA MP Version**.

Related information

The following PDF documents that contain RSCT information can be found at Reliable Scalable Cluster Technology (RSCT) PDFs:

- *Administering RSCT*
- *Messages for RSCT*
- *Programming Group Services for RSCT*
- *Programming RMC for RSCT*
- *Technical Reference: RSCT for AIX*
- *Technical Reference: RSCT for Multiplatforms*

Troubleshooting RSCT

This information is designed for system programmers and administrators, but can be used by anyone responsible for diagnosing problems related to RSCT. To use this information, you need to be familiar with one or more of these operating systems: AIX, Linux, Solaris, and Windows, depending on which of them are in use at your installation. Where necessary, some background information relating to AIX, Linux, Solaris, or Windows is provided. More commonly, you are referred to the appropriate documentation.

What's new in Troubleshooting for RSCT

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

How to see what's new or changed

In this PDF file, you might see revision bars (|) in the left margin that identify new and changed information.

November 2013

The following information is a summary of the updates made to this topic collection:

- Added troubleshooting information about logical partition (LPAR) connection with management domain in **** MISSING FILE ****.

November 2012

The following information is a summary of the updates made to this topic collection:

- Added information about RSCT peer domain support on Cluster Aware AIX (CAA) linked clusters and about AIX node crash investigation in Diagnosing problems with group services in a CAA environment.

October 2011

The following information is a summary of the updates made to this topic collection:

- Added information about `DomainType` values in Operational test 1: verifying the configuration resource manager availability.
- Added a symptom and corresponding recovery action to diagnose problems with the configuration resource manager component of RSCT. For more information, see “Action 19: investigate the stoprpdomain command failure because of critical resources being active on nodes” on page 69.

Overview of diagnosing problems in RSCT

This section explains some general information on diagnosing RSCT problems, for example, how to access errors logged by the various RSCT subsystems, when and how to contact the IBM Support Center, and how to collect data for review by the IBM Support Center.

Before using this document to diagnose RSCT problems, verify that the RSCT components have been installed.

Accessing logged errors

The RSCT component subsystems write information about important errors. On AIX nodes, the RSCT component writes information to the AIX error log. On Linux, Windows and Solaris nodes, it writes information to the system log.

Log file location

Review the default locations for the AIX error log on AIX nodes, and the system log for Linux, Windows, and Solaris nodes.

Table 2 identifies the default locations of the AIX error log on AIX nodes, and the system log on Linux, Windows and Solaris nodes.

Table 2. Default locations of the AIX error log, and the Linux, Windows, and Solaris system logs

AIX error log	Linux system log	Windows system log	Solaris system log
By default, RSCT component subsystems store the error log file in **/var/adm/ras/errlog**. It logs one entry for each occurrence of the condition. It logs the condition on every node on which the event occurred.	By default, RSCT component subsystems store the system log messages in **/var/log/messages**. The system administrator, however, can change the default. It logs errors on the node(s) where the event occurred, unless the system administrator alters the default action of the system log to forward the errors to a remote system. Consult the file **/etc/syslog.conf** to determine if information has been redirected or filtered.	RSCT component subsystems store the system log messages in **/var/adm/log/messages**. It logs errors on the node(s) where the event occurred.	The system log messages are stored in **/var/adm/messages*** by default. Errors are logged on the node(s) where the event occurred.

Displaying logged errors

There are differences in the default locations of the AIX error log for AIX nodes, and the system logs for Linux, Windows, or Solaris nodes.

Note: In order for the RSCT subsystems on Linux, Windows, and Solaris nodes to record errors, the system log must be active and the **syslogd** daemon must be operational.

Table 3 on page 3 identifies the commands used to display logged errors for AIX, Linux, Windows and Solaris nodes.

Table 3. Displaying logged errors for AIX, Linux, Windows, and Solaris nodes

AIX nodes	Linux nodes	Windows nodes	Solaris nodes

To display a complete summary report, enter:
`errpt`
 	
To display a complete detailed report, enter:
 	
`errpt -a`
	Check the system log documentation for the Linux distribution used within the cluster for specifics on the log file behavior.
	
	Assuming that the system log messages are in directory **/var/log/messages**, the following command displays the error information:
	
	`fcslogrpt /var/log /messages`
	
	This command will display the error entries produced by RSCT in increasing timestamp value order.
		Check the log documentation for your Windows system to learn specifics on the log file behavior.
		
		To display the error information, see:
		
		`cat /var/adm/log/messages`
			The system log messages are stored in **/var/adm/messages*** by default.
			
			Errors are logged on the node(s) where the event occurred.

Log file size considerations

Consider managing the size of the AIX error log on AIX nodes, and the system log on Linux, Windows, and Solaris nodes.

Table 4 identifies the commands used to display logged errors for AIX, Linux, Windows, and Solaris nodes.

Table 4. Managing the size of the AIX error log, or the Linux, Windows, or Solaris system log

AIX error log file	Linux system log file	Windows system log file	Solaris system log file

The AIX error log file size is limited, and it operates as a circular file. When the log file reaches its maximum length, RSCT discards the oldest entries within the log in order to record newer entries.
 	
AIX installs a **cron** job that removes any hardware-related failure records within the log file after 90 days, and any software-related failure records or operator information records after 30 days.
 	
You can view and modify the error log file size through SMIT using the **smit error** command, or through the following commands:
• To display the error log file size:
 	
`/usr/lib/errdemon -l`
• To set the error log file size:
 	
`/usr/lib/errdemon -s`
 	
Both the **smit** and the **errdemon** commands require root user authority.
	The Linux system log is implemented as a text-based file. The exact behavior of this file varies among Linux distributions. Some Linux distributions archive this file and start a new log file at regular intervals, pruning the oldest archive to prevent consuming too much space in the **/var/file** system.
	
	Check the system log documentation for the Linux distribution used within the cluster for specifics on the log file behavior.
	
	Administrators may need to take additional steps to ensure that the system log files do not grow too large or remain on a system too long.
		The Windows system log is implemented as a text-based file. Check the system log documentation for the Windows platform used within the cluster for specifics on the log file behavior.
		
		Administrators may need to take additional steps to ensure that the system log files do not grow too large or remain on a system too long.
			The Solaris system log is implemented as a text-based file. Check the system log documentation for the Solaris platform used within the cluster for specifics on the log file behavior.Administrators may need to take additional steps to ensure that the system log files do not grow too large or remain on a system too long.

Configuring trace spooling

Applications use the common trace facility to write trace records to a logical trace file. Trace files, the internal logging format for RSCT, are fixed-size buffers. The fixed size of these trace files limits the amount of history that is available for debugging purposes. A mechanism called *trace spooling* exists to create new trace files as the existing ones become full and simultaneously move the completed files to another directory on the system.

Trace spooling is the copying of page files to a spool destination.

When trace spooling is *disabled*, the trace library writes trace records for a specific logical trace file to a single physical trace file and "wraps" to the beginning of this file when it is full.

When trace spooling is *enabled*, the trace library writes trace records for a specific logical trace file to a set of physical files with the same base name with suffixes of the form ***.number*.sp** appended to them. These files are referred to as trace "page" files. When the last page file associated with a logical trace file is full, writing "wraps" the first page file. Once enabled, trace spooling can additionally be either *on* or *off*.

It is important to distinguish between the distinct concepts of trace spooling being *enabled* or *disabled*, and trace spooling being *on* or *off*. The *on* and *off* states constitute a functional subset of trace spooling being enabled, but these states have no meaning when trace spooling is disabled.

Limited only by available disk space, trace spooling allows the system to maintain a continuous history of RSCT actions. You can configure trace spooling control at the daemon level, using the common trace facility configuration file (**/var/ct/cfg/trace.conf**) or the **CT_TR_SPOOL** environment variable.

When configured for trace spooling, the trace library manages a client's trace file as several page files and creates a spooling thread to copy full page files to the spooling area. If the copy thread cannot spool page files as quickly as the client can fill them, the trace library favors the retention of the most recent trace records by overwriting older spool page files that could not be spooled in time. When this occurs, the trace library adds a record that looks like this to the current page file:

```
SPOOL: page_file_name not spooled integer consecutive time[s]
```

This record documents the fact that a page file was not spooled, so its trace records were lost. This circumstance can be mitigated by configuring clients that write many trace records to have a larger page file footprint (that is, more pages, a larger size, or both).

Because the spooling area can become full, leading to a possible loss of long-term retention of trace data if unaddressed, consider setting up **IBM.Sensor** resources to monitor the disk usage in the spooling area. Use the **chkspool** script to manage spooling area storage.

Set up a **cron** job to run **chkspool** in either of the following modes:

1. `/usr/bin/chkspool --spool_dir /data/trc --days_limit 14`

   Result: Any trace files older than two weeks (14 days) will be deleted.
2. `/usr/bin/chkspool --spool_dir /data/trc --megabytes_limit 50`

   Result: Saved trace files are deleted, starting with the oldest ones, until no more than 50 MB total space is used under **/data/trc**.

Because trace spooling can result in a large number of spooled trace files, and the **rpttr** command can run out of system resources when attempting to show the content of too many, use the **showtr** command to narrow down the scope of spooled trace files to view. Run **showtr --help** to see its filtering options and usage examples.

Using the trace.conf configuration file:

Whenever it starts up, the trace library reads the configuration file. The **trace.conf** file specifies trace information in a global configuration file.

The common trace facility configuration file (**/var/ct/cfg/trace.conf**) can be modified to change aspects of configuration that pertain to trace spooling. Trace spooling is enabled through the presence and contents of the configuration file. The contents of the configuration file can enable zero or more logical trace files. Logical trace files should be considered independent from a trace spooling point of view.

Once trace spooling is enabled for a specific logical trace file,

- it cannot be disabled. Rather, it can be turned on or off. When trace spooling in on, the page files are used, and they are spooled as necessary. When trace spooling is off, the page files are also used, but not spooled.
- removing the configuration file does not disable it. Rather, it turns spooling off all logical trace files for which trace spooling was enabled.

While an application is running, there is no user action that results in the disabling of trace spooling for a logical trace file once it has been enabled for that logical trace file.

The trace configuration file is applied to separate application processes separately. Each application process has its own instance of the trace library, and thus its own spooling thread. While the configuration file potentially applies to all applications using the trace library (through multiple uses of the **pat** attribute, and any wild-card uses of it), each instance of the trace library reads it separately, and the behavior in each separate process depends on the contents of the configuration file and the previous state of trace spooling in each separate process.

The global configuration file, **/var/ct/cfg/trace.conf**, looks like this:

```
# comment
section_name:
pat        = source directory pattern
spooling   = [ OFF | ON ]
pages      = number of files
dest       = destination directory
size       = file size of trace pages
.
.
.
custom sections
.
.
.
```

Descriptions of the **trace.conf** fields follow:

Field	Description
section_name	Is an arbitrary string that indicates the start of a new stanza. The section name is not used to determine which daemon or process will have its trace files copied. This is determined solely by the regular expression that follows the **pat** attribute.
pat	Is the source directory pattern for trace files. This does not descend into subdirectories. You can use wild cards, for example: **/var/ct/.*/log/mc/.***. To match and save all RSCT trace files, use **/var/ct/.***.
spooling	**OFF** Indicates that spooling is disabled. This is the default. **ON** Copies the trace information to a user-defined location. Trace files are copied to the **dest** location when a trace file fills up.
pages	Indicates the number of files. A **pages** value that is less than 3 is forced to a value of 3. The recommended value is 9. Although there is no fixed upper limit, large values could degrade system performance.

Field	Description
dest	Indicates the base location of spooled files. While nothing will prevent setting the **dest** value to a relative path, this can lead to unexpected results. The **dest** value should always be specified as an absolute path that is not a directory under **/var**.
size	Is an optional attribute that sets the size, in bytes, of the individual trace pages. If unspecified, the size is determined automatically by the trace facility, based on the trace configuration of the client. The size of each page becomes the client's trace file size divided by the number of pages. A positive **size** value that is less than 4096 is forced to a value of 4096, unless it is less than 0, in which case it is forced to a value of 262144. However, if an application internally forces a size, this might override the **size** value in the configuration file.
custom sections	Indicates optional custom sections that describe how spooling is to be done for other directories.

If there is more than one configuration stanza, the first matching one will be applied to any given trace file. This means that they should be listed in order from the most specific pattern to the least specific pattern.

Dynamic reconfiguration:

With dynamic reconfiguration, a trace spooling administrator can change trace spooling configuration without having an impact on trace library clients. Reconfiguration is performed by the trace library's own spooling thread, which sleeps until a tracing thread's trace activity results in the need for a page to be spooled. This is called a *spool event*. Applications that are started without trace spooling enabled prevent further changes to the configuration file from being seen. Reconfiguration cannot occur until tracing that requires spooling occurs.

Some configuration file changes affect trace spooling being enabled, turned on, or turned off. For example, for a logical trace file named **/var/ct/IW/log/mc/IBM.GenericRM/trace**, a **pages** value of 3, and a **dest** value of **/tmp/TraceSpool**, the configuration possibilities follow:

1. No configuration file exists.
2. A configuration file exists, which does not have a stanza with a **pat** value that matches the logical trace file:

```
Generic:
pat      = /var/ct/IW/log/mc/IBM.ThisDoesNotMatch/.*
spooling = OFF
pages    = 3
dest     = /tmp/TraceSpool
```

3. A configuration file with a **spooling** value of **ON** exists, which has a stanza with a **pat** value that matches the logical trace file:

```
Generic:
pat =      /var/ct/IW/log/mc/IBM.GenericRM/.*
spooling = ON
pages =    3
dest =     /tmp/TraceSpool
```

4. A configuration file with a **spooling** value of **OFF** exists, which has a stanza with a **pat** value that matches the logical trace file:

```
Generic:
pat =      /var/ct/IW/log/mc/IBM.GenericRM/.*
spooling = OFF
pages =    3
dest =     /tmp/TraceSpool
```

How trace spooling occurs in response to configuration file changes differs depending on whether or not an application is running.

For applications that are not running and are started with one of the previous configuration possibilities:

- If there is no configuration file or if **pat** does not match the logical trace file, the trace data is sent to a file named **trace**.
- If **pat** matches the logical trace file, the trace data is sent to page files, for example: **trace.1.sp**, **trace.2.sp**, **trace.3.sp**. If the **spooling** value is **ON**, page files are spooled to **dest**. If the **spooling** value is **OFF**, page files are not spooled.

For applications that are already running, spooling is not already enabled, and the configuration file is created or changed: configuration file changes are irrelevant. Trace data continues to be sent to a file named **trace**.

For applications that are already running, spooling is already enabled, the configuration file is created or changed, and a spool event occurs in which the trace thread detects and reads the configuration file changes:

- **pat** has no effect on the files to which trace data is being sent. Trace data continues to be sent to page files, for example: **trace.1.sp**, **trace.2.sp**, **trace.3.sp**.
- If there is no configuration file or if **pat** does not the match logical trace file, page files are not spooled.
- If **pat** matches the logical trace file and the **spooling** value is **ON**, page files are spooled to **dest**. If the **spooling** value is **OFF**, page files are not spooled.

The following configuration file changes *do not* affect trace spooling being enabled, turned on, or turned off:

- If the **pages** attribute value is changed, the number of physical page files for logical trace files that match the **pat** attribute value is changed to the new value. If the number is decreased from the previous value, any page files outside the new value that were already in the process of being spooled are still spooled.
- If the **dest** attribute value is changed, the target spool file destination for logical trace files that match the **pat** attribute value is changed to the new value. If a page was in the process of being spooled to the previous destination, it still goes to the previous destination.
- If the **size** attribute is used, the value is meaningful only when logical trace files that match the **pat** attribute value are first enabled. After that, the value has no effect.

Using the CT_TR_SPOOL environment variable:

You can use the **CT_TR_SPOOL** environment variable to specify and enable spooling for a given daemon and to specify the location of the persistent spooling area.

If the **CT_TR_SPOOL** environment variable is set, the **trace.conf** configuration file is not used. Instead, the configuration is taken from the environment variable. **CT_TR_SPOOL** includes a pattern that is matched against the trace files. It has the following format:

pat:*spooling*:*pages*:*dest*,...

Descriptions of the **CT_TR_SPOOL** fields follow:

Field	Description
pat	Source directory pattern for trace files. This does not descend into subdirectories. You can use wild cards, for example: **/var/ct/.*/log/mc/.***.
spooling	**OFF** Indicates that spooling is disabled. This is the default. **ON** Copies the trace information to a user-defined location.
pages	Indicates the number of files. A **pages** value that is less than 3 is forced to a value of 3. The recommended value is 9. Although there is no fixed upper limit, large values could degrade system performance.

Field	Description
dest	Indicates the base location of spooled files. While nothing will prevent setting the **dest** value to a relative path, this can lead to unexpected results. The **dest** value should always be specified as an absolute path that is not a directory under **/var**.
,...	Specifies additional directories (optional).

Listing trace spooling status:

To display the trace spooling status of a resource manager's trace files, use the **lssrc -ls** *resource_manager* command.

For example, to display the trace spooling status of the configuration resource manager's trace files, run this command:

```
lssrc -ls IBM.ConfigRM
```

The following output is displayed:

```
(other resource manager information)
.
.
.
/var/ct/IW/log/mc/IBM.ConfigRM/trace.1.sp (7 pages) ->
  /admin/bubbly/Spool.ConfigRM/IW/25384144/c206bc1b07/f224ca38a699a2c/IBM.ConfigRMd
/var/ct/IW/log/mc/IBM.ConfigRM/trace.detail -> spooling not enabled
/var/ct/IW/log/mc/IBM.ConfigRM/trace.summary.1.sp (5 pages) -> /dev/null
```

The sample output shows that trace spooling is:

- *enabled and on* for the resource manager's **/var/ct/IW/log/mc/IBM.ConfigRM/trace** trace file, and is being spooled to the **/admin/bubbly/Spool.ConfigRM/IW/25384144/c206bc1b07/f224ca38a699a2c/IBM.ConfigRMd** directory.
- *not enabled* for the resource manager's **/var/ct/IW/log/mc/IBM.ConfigRM/trace.detail** trace file.
- *enabled and off* for the resource manager's **/var/ct/IW/log/mc/IBM.ConfigRM/trace.summary** trace file.

For enabled trace files, the listing shows the first page file name (including the page file **.***number***.sp** suffix) to emphasize the fact that spooling is enabled.

Examples

1. In this example, trace files produced by **rmcd** (those in **/var/ct/.*/log/mc/**) are each grouped into nine file sets, with trace records added globally, in a circular fashion:

   ```
   $ export CT_TR_SPOOL=\
   "/var/ct/.*/log/mc/.*:ON:9:/data/trc"
   ```

   As each file is completed, it is renamed (a time stamp is appended) and copied to **/data/trc/***cluster_name***/***cluster_ID*/host_name/node_ID**/rmcd**.
2. This example contains a second directory pattern in the environment variable, **/var/ct/.*/log/mc/IBM.StorageRM/.***, which instructs the trace library to create a nine-file spool group from the storage resource manager's trace files and to store the files under **/data/trc**:

   ```
   $ export CT_TR_SPOOL=\
   "/var/ct/.*/log/mc/IBM.StorageRM/.*:ON:9/data/trc,\
   /var/ct/.*/log/mc/.*:ON:9:/data/trc"
   ```
3. The following sample **/var/ct/cfg/trace.conf** file has the same effect as combining Examples 1 and 2:

   ```
   # IBM.StorageRM spooling
   IBM.StorageRM:
   pat = /var/ct/.*/log/mc/IBM.StorageRM/.*
   spooling   = ON
   pages      = 9
   dest       = /data/trc/
   ```

```
# RMCd spooling
rmcd:
pat        = /var/ct/.*/log/mc/.*
spooling   = ON
pages      = 9
dest       = /data/trc/
```

Message format

There are differences among the formats of error messages written to the AIX error log, or Linux, Windows, or Solaris system logs.

Table 5 describes the message formats on AIX, Linux, Windows, and Solaris system platforms.

Table 5. Error message formats in the AIX error log, and Linux, Windows, and Solaris system logs

System platform	Error message format sample
AIX	LABEL: TS_LOC_DOWN_ST IDENTIFIER: 173C787F Date/Time: Wed May 20 23: 34:55 EDT Sequence Number: 5434 Machine Id: 000123456A00 Node Id: c684n09 Class: S Type: INFO Resource Name: cthats Description Possible malfunction on local adapter : :
Linux	May 20 23:34:55 c117f1n1 cthats[10062]: (Recorded using libct_ffdc.a cv 2):::Error ID: 824....m/6V2/rEl176ba20.....:::Reference ID: :::Template ID: 0:::Details File: :::Location: rsct,nim_control.C,1.39.1.1,4147 :::TS_LOC_DOWN_ST Possible malfunction on local adapter ...
Windows	Jun 28 08:41:11 rsctvm2 RMCdaemon [26883]: (Recorded using libct_ffdc.a cv 2):::Error ID: 824....bluU4/B7B0..... .....:::Reference ID: :: :Template ID: 0:::Details File: :::Location: RSCT,rmcd.c,1.51, 209 :::RMCD_INFO_0_ST The daemon is started.
Solaris	Mar 10 01:09:03 e106e335n09.ppd.pok.ibm.com ConfigRM[3398]: [ID 489967 daemon.no tice] (Recorded using libct_ffdc.a cv 2):::Error ID: :::Reference ID: :::Templa te ID: 0:::Details File: :::Location: RSCT,ConfigRMDaemon.C,1.12,176 :::CONFIGRM_STOPPED_ST IBM.ConfigRM daemon has been stopped.

Table 6 on page 10 further describes the message formats shown in Table 5

Table 6. Error message format descriptions of the AIX error log , and Linux, Windows, and Solaris system logs

AIX error log	Linux system log	Windows system log	Solaris system log
The `LABEL` field contains a unique string identifying the error. In this manual, you can use the label to look up information on the error. The `Resource Name` field indicates the specific RSCT subsystem that generated the error. The error entry ends with a description of the error. For more information on any of the other fields of the error log entry, see the online man page for the **errupdate** command.	Individual fields within the error record are separated by three colons (:::). The first field of the error record contains a timestamp followed by the name of the node on which the error occurred, followed by the resource name. The resource name indicates the specific RSCT subsystem that generated the error. The `Location` field provides information about the code that detected and recorded the incident. The description of the incident follows the last set of colon separators. For most RSCT subsystems, the description will start with a label (in the preceding example, the label is TS_LOC_DOWN_ST). In this manual, you can use the label to look up information on the error.	Individual fields within the error record are separated by three colons (:::). The first field of the error record contains a timestamp followed by the name of the node on which the error occurred, followed by the resource name. The resource name indicates the specific RSCT subsystem that generated the error. The `Location` field provides information about the code that detected and recorded the incident. The description of the incident follows the last set of colon separators. For most RSCT subsystems, the description will start with a label (in the preceding example, the label is RMCD_INFO_0_ST). In this manual, you can use the label to look up information on the error.	Individual fields within the error record are separated by three colons (:::). The first field of the error record contains a timestamp followed by the name of the node on which the error occurred, followed by the resource name. The resource name indicates the specific RSCT subsystem that generated the error. The `Location` field provides information about the code that detected and recorded the incident. The description of the incident follows the last set of colon separators. For most RSCT subsystems, the description will start with a label (in the preceding example, the label is RMCD_INFO_0_ST). In this manual, you can use the label to look up information on the error.

Related reference:

"Additional information to collect if you suspect that a particular RSCT subsystem is at fault" on page 16
If you have already determined that the problem is with a particular RSCT subsystem, you may want to collect the errors reported by that subsystem.

Finding explanations for logged errors

This topic identifies tables that describe each of the possible errors that the various RSCT subsystems may log.

Identify your error prefix in Table 7 to determine which RSCT subsystem has logged the error. Then, consult the corresponding table for its detailed explanation.

Table 7. Where to find explanations for errors logged by RSCT subsystems

Error labels that contain or refer to prefix	Error labels that contain or refer to resource name	Related RSCT subsystem	Related information
RMCD_	RMCdaemon	Resource Monitoring and Control subsystem	Table 10 on page 20: Error Log templates for the Resource Monitoring and Control daemon
CONFIGRM_	ConfigRM	Configuration resource manager	Table 12 on page 40: Error log templates for the configuration resource manager
ctcasd daemon	ctcasd	Cluster Security Services	Table 17 on page 71: Error log templates for Cluster Security Services
TS_	hats, cthats, or topsvcs	Topology Services	Table 25 on page 138: Error Log templates for Topology Services
GS_	hags, cthags, or grpsvcs	Group Services	Table 35 on page 193: Error log templates for Group Services

Taking a snapshot

There are several snapshot tools available to help you collect data (such as configuration, trace, and log files) for review by the IBM Support Center. They prevent you from having to manually gather dozens of files and command responses. To determine the proper snapshot tool for data collection, consider the type of cluster or domain that you are using.

Resource monitoring and control (RMC) domains

The resource monitoring and control (RMC) subsystem runs in several different domains and can be running in more than one domain at a time.

The domain modes in which RMC can be running are:

no domain
: If it is not being used in either of the following domain modes, RMC will still be running, but only local scope data will be visible.

peer domain
: This is the mode used when RMC is running under an RSCT peer domain.

management domain
: This is the mode used when RMC is running under Cluster Systems Management (CSM) or supporting LPAR functionality. For more information about a CSM cluster, see the CSM product documentation.

Related information:

Understanding RMC and resource managers

Other cluster types

The cluster types cited in this topic are not mutually exclusive. For example, a node can simultaneously be an active member of an RSCT peer domain and a PowerHA® SystemMirror® (PowerHA SystemMirror) cluster.

Before using this document to diagnose RSCT problems, verify that the RSCT components have been installed. If so, determine which specific operating-system-related **rsct.basic** file set has been installed. See the *Verifying RSCT installation* chapter of the *Administering RSCT* guide for details on how to make these determinations.

RSCT peer domain:

An RSCT peer domain is a cluster of nodes with no central point of control. All nodes are peers of each other, sharing information and agreeing to the configuration. Although, you can have multiple peer domains defined, any given node can be active in only one domain at a time.

If you are running an RSCT peer domain, in addition to the **rsct.core** file sets, you will need to have the **rsct.basic** file sets installed:

```
Fileset                      Level  State      Description
----------------------------------------------------------------------
rsct.basic.rte             3.1.0.0  COMMITTED  RSCT Basic Function
```

To see how many domains are defined on a node and which (if any) are active, issue the command:

```
/usr/bin/lsrpdomain
```

The following output is displayed:

```
Name        OpState RSCTActiveVersion MixedVersions TSPort GSPort
my_peers1   Online  3.1.0.0           No            12347  12348
my_peers2   Offline 3.1.0.0           No            12347  12348
test_domain Offline 3.1.0.0           Yes           12347  12348
```

RSCT data related to each RSCT peer domain is found under **/var/ct/***domain_name***/**.

PowerHA SystemMirror:

PowerHA SystemMirror cluster is a software solution for keeping resources highly available. It uses some of the RSCT subsystems as backbone components, most notably Topology Services and Group Services. It has its own documentation, including a troubleshooting guide.

If you are running a PowerHA SystemMirror cluster, you will have **cluster.es** file sets installed:

```
Fileset                      Level  State      Description
----------------------------------------------------------------------
cluster.es.server.rte      5.2.0.6  APPLIED    ES Base Server Runtime
```

The name of the cluster can be found by issuing the command:

```
/usr/es/sbin/cluster/utilities/cltopinfo -c
```

The following output is displayed:

```
Cluster Name: my_cluster Cluster Connection Authentication Mode: Standard
Cluster Message Authentication Mode: None Cluster Message Encryption: None
Use Persistent Labels for Communication: No
```

RSCT data related to a PowerHA SystemMirror cluster is found under **/var/ha/**.

Parallel Systems Support Programs (PSSP):

PSSP software is used to manage clusters of nodes, often for high-scale computing power with licensed programs such as GPFS™ and IBM Tivoli Workload Scheduler (TWS) LoadLeveler® systems.

PSSP has its own set of documentation, including a diagnosis guide.

If you are running a PSSP cluster, you will have **ssp** file sets installed:

```
Fileset                       Level  State      Description
----------------------------------------------------------------------
ssp.basic               3.5.0.20  APPLIED    SP System Support Package
```

PSSP runs in partitions. The names of the partitions can be found by issuing the command:

```
/usr/lpp/ssp/bin/splstdata -p
```

The following output is displayed:

```
System Partitions:
------------------
production1

Syspar: production1
------------------------------------------------------------
syspar_name     production1
ip_address         x.x.x.x
```

RSCT data related to a PSSP cluster is found under **/var/ssp/**.

Snapshot commands

Data gathering requirements will vary on a case-by-case basis, but snapshot tools allow you to obtain the elements that are most often needed for each cluster type. They enable the IBM Support Center to either identify the problem at hand or refine a more specific data collection request.

There are different snapshot commands tailored to each of the cluster types described in "Resource monitoring and control (RMC) domains" on page 11 and "Other cluster types" on page 11.

You will need root user authority to run any of these commands.

ctsnap

The **ctsnap** command collects data only from the invoking node. Due to the distributed nature of RSCT and depending on the problem, it may be necessary for you to invoke the command from multiple nodes.

Basic information about the **ctsnap** command:

- Used for: RMC domains and RSCT peer domains
- Full path name: **/usr/sbin/rsct/bin/ctsnap**
- Default output location: **/tmp/ctsupt/**

See Table 8 for more information.

Table 8. Using **ctsnap** *to collect snapshot data*

Symptom	Recovery
A problem with RSCT on an AIX system	Run the **ctsnap -k stackdump_default** command to produce a stack dump for these RSCT subsystems: **cthags**, **cthats**, **ctrmc**, **IBM.AuditRM**, **IBM.CIMRM**, **IBM.ConfigRM**, **IBM.ERRM**, **IBM.FSRM**, **IBM.GblResRM**, **IBM.LPRM**, **IBM.MicroSensorRM**, **IBM.RecoveryRM**, **IBM.SensorRM**, and **IBM.StorageRM**.
A problem with a subsystem that is known to have trace spooling enabled	Run the **ctsnap** command with options that are relevant to trace spooling, so the resulting **ctsnap** file contains the desired trace spool files.
A problem with the Topology Services subsystem that is related to connectivity	If possible, run the **ctsnap** command on all nodes. If collecting data from all nodes is not feasible, run the the **ctsnap** command on at least the following nodes: • The node that exhibits the problem. • The problem node's *downstream neighbor*. The downstream neighbor is the node whose IP address is immediately lower than the address of the node on which you saw the problem. If the problem node is the one with the lowest IP address, its downstream neighbor is the node with the highest IP address. • The group leader node. This is the node with the highest IP address in the network. In addition to the data collected by **ctsnap**, run the **tcpdump** command to collect a sample of the traffic on the network: **tcpdump -n -x [-i** *interface_name*] **> output_file** Allow the command to run for at least 30 seconds and then terminate it with a signal. Collect this output file to send to the IBM Support Center along with the **ctsnap** output files.
A problem with the Group Services subsystem	Run the **ctsnap** command on these nodes: 1. The nodes that exhibit the problem. 2. The Group Services name server node. See "Finding the GS name server (NS) node" on page 202. 3. The group leader node, if the problem is related to a particular group. See "Finding the group leader (GL) node for a specific group" on page 205.

For complete syntax information on the **ctsnap** command, see the *Technical Reference: RSCT for AIX* or *Technical Reference: RSCT for Multiplatforms* guides.

Output will be:

- a compressed tar file (ctsnap.*host_name.nnnnnnnn*.tar.Z)
- a log file (ctsnap.*host_name.nnnnnnnn*.log)

In these file names, *nnnnnnnn* is a timestamp indicating when the command was run and *.host_name* identifies the host on which the command was run.

Note: The **ctsnap -x runrpttr** command will format the content of all RSCT resource manager trace files. This will increase the **ctsnap** output size, as well as the possibility that you will need to expand the file system containing its output directory.

snap -e

For PowerHA SystemMirror, the AIX **snap** command includes an **-e** flag which will gather all the necessary PowerHA SystemMirror data. In an PowerHA SystemMirror cluster, it will try to gather data from all defined nodes, whether the cluster is up or down. See the *AIX Commands Reference* for more information about the **snap** command and the **-e** flag in particular.

Basic information about the **snap -e** command:

- Used for: PowerHA SystemMirror clusters
- Full path name: **/usr/sbin/snap -e**
- Default output location: **/tmp/ibmsupt/**

Output is a compressed pax file (**snap.pax**), which contains the elements gathered in **/tmp/ibmsupt/hacmp/**.

Notes:

1. The **snap -e** command does not collect data about the RMC subsystem, which it uses instead of emsvcs starting in PowerHA SystemMirror 5.2. If an RMC problem is suspected, you should run **ctsnap** separately (see the previous explanation in "ctsnap" on page 13).
2. The **snap -e** command uses the **phoenix.snap** tool as part of its data collection process. So, if you have difficulty getting the **snap -e** command to complete successfully, you can also run **phoenix.snap** on any node in the cluster to ensure complete RSCT data collection.

phoenix.snap script

The **phoenix.snap** script is an as-is tool provided for gathering data for PSSP clusters.

As such, the *Troubleshooting for RSCT* guide contains the following disclaimer:

The **phoenix.snap** tool is a service tool and not a PSSP command. The tool is shipped with PSSP as is, that is, without documentation. For assistance on using **phoenix.snap** in a manner other than what is described in this section, contact the IBM Support Center.

Basic information about the **phoenix.snap** command:

- Used for: PSSP clusters
- Full path name: **/usr/sbin/rsct/bin/phoenix.snap**
- Default output location: **/tmp/phoenix.snapOut/**

The **phoenix.snap** tool collects data from different locations depending on where it is run. When run on the CWS, it automatically gathers data from the CWS and certain other nodes, such as the **hats** and **hags** group leaders. When run on a node, it only gathers data from that local node.

The output also varies depending on where and how the tool is run, as follows:

- When data from only one node is collected, the output consists of:
  - A compressed tar file (phoenix.snap *host_name.nnnnnnnn*.out.tar.Z)

  - A log file (phoenix.snap_info.*nnnnnnnn*.out)
  - An error file (phoenix.snap_err.*nnnnnnnn*.out)

  host_name identifies the host on which the script was run, and *nnnnnnnn* is a timestamp indicating when the script was run.
- When data from multiple nodes is collected, the output consists of:
  - A tar file (all.*nnnnnnnn*.tar) containing the compressed tar files from each node
  - A log file (phoenix.snap_info.*nnnnnnnn*.out)
  - An error file (phoenix.snap_err.*nnnnnnnn*.out)

IBM Support Center

There are several things that you need to know in order to make effective use of the IBM Support Center. You need to know when to call, how to contact, and what information to collect for the IBM Support Center before calling.

When to contact the IBM Support Center

Contact the IBM Support Center only under certain circumstances.

Contact the IBM Support Center when you experience the following situations:

- Node halt or crash not related to a hardware failure
- Node hang or response problems
- A repeated or persistent failure of specific RSCT software components

  These failures may not always occur on the same node, given the distributed nature of this software.
- Failure in other software supplied by IBM

A single node or infrequent software failure that is not mission-critical may not be a cause to contact the IBM Support Center immediately. These types of problems may be caused by conditions that can be remedied using administrative techniques. Investigate these failures using this manual as a guide for conducting the investigation. You should also:

- Determine what was active on the system at the time of the failure.
- Record the date and time of the failure.
- Determine what hardware was in use.
- Determine what specific RSCT software components were being used at the time that the failure was detected.

Log information about the failures that you discover in the course of your investigations. This information can be used for your own future reference, and by the IBM Support Center should this failure occur frequently enough or become critical enough to require IBM Support Center assistance. The log information permits you to respond more quickly to similar failures in the future, and gives you a documented reference on how to resolve the problem.

You can also use the log information for pattern analysis. Problems and failures may appear to be unrelated at first, but may later evidence some relationship that was not immediately evident. Examine the conditions that were recorded for previous infrequent failures to see if there may be a pattern to them, even if the failure seem to be unrelated. Consider the following items when looking at the historical data on problems and failures:

- Do they happen when similar programs, procedures, or jobs are run?
- Do they happen when certain people or groups use the system?
- Do they happen at specific times (for example, at peak or off-peak hours), on particular days, or during certain shifts?
- Does the failure occur when specific hardware is in use?

- Are node reboots the only way to resolve the problem?

Contact the IBM Support Center when you discover any patterns in infrequent failures because:

- The system configuration may need repair.
- The IBM Support Center may have useful information about the problem you are experiencing.
- You may be experiencing a problem that no one else has ever encountered or reported.

Information to collect before contacting the IBM Support Center

There are things for you to do before you contact the IBM Support Center.

Before you contact the IBM Support Center, do the following:

1. Check the AIX error log, or the Linux, Solaris, or Windows system log (described in "Accessing logged errors" on page 2) on the node(s) manifesting the problem. If the instructions in this document do not enable you to resolve the problem, the error log or system log should help you to identify the RSCT subsystem experiencing the problem.
2. Issue the appropriate snapshot command, as prescribed in "Snapshot commands" on page 13.

Related reference:

"Snapshot" on page 157
A snapshot is a collection of configuration data, log and trace files, and other diagnostic data for the RSCT components used for problem determination.

Related information:

"ctsnap dump" on page 199
This dump contains diagnostic data used for RSCT problem determination. It is a collection of configuration data, log files, and other trace information for the RSCT components.

Additional information to collect if you suspect that a particular RSCT subsystem is at fault

If you have already determined that the problem is with a particular RSCT subsystem, you may want to collect the errors reported by that subsystem.

Although the ctsnap command will, depending on the node's operating system, collect either the AIX error log, or the Linux, Windows, or Solaris system log that you want to supply the IBM Support Center with the error listing for the specific RSCT subsystem you suspect is at fault, you may expedite problem resolution by doing so.

Table 9 on page 17 explains how to filter the contents of the AIX error log, and Linux, Windows, or Solaris system logs to extract information about a particular RSCT subsystem.

Table 9. Filtering the contents of the AIX error log, or Linux, Windows, or Solaris system logs for a particular RSCT subsystem

AIX error log	Linux system log	Windows system log	Solaris system log
You can filter the AIX error log using the **errpt** command's **-N** flag. Using the **-N** flag, simply indicate the resource name for the RSCT subsystem that reported the error(s). **Example:** The following command generates a detailed report for errors logged by Cluster Security Services (specifically, the **ctcasd** daemon). RSCT directs the output from this command to the file **ctcasderr.out**. `errpt -N ctcasd -a > ctcasderr.out` To determine the resource associated with an error in the AIX Error Log, see the Resource Name field of the error entry.	You can filter the Linux system log using the **grep** command. Supply the **grep** command with the resource name for the RSCT subsystem that reported the error(s). **Example:** The following command generates a detailed report for errors logged by Cluster Security Services (specifically, the **ctcasd** daemon). RSCT directs the output from this command to the file **ctcasderr.out**. Assuming that the system log has not been redirected from its default location, you would enter: `grep ctcasd / var/log/messages > ctcasderr.out` To determine the resource associated with an error in the Linux System Log, locate the resource name in the first field of the error entry.	You can filter the Windows system log using the **grep** command. Supply the **grep** command with the resource name for the RSCT subsystem that reported the error(s). **Example:** The following command generates a detailed report for errors logged by **cthags**. RSCT directs the output from this command to the file **cthagserr.out**. Assuming that the system log has not been redirected from its default location, you would enter: `grep cthags / var/adm/log /messages > cthagserr.out` To determine the resource associated with an error in the Windows System Log, locate the resource name in the first field of the error entry.	You can filter the Solaris system log using the **grep** command. Supply the **grep** command with the resource name for the RSCT subsystem that reported the error(s). **Example:** The following command generates a detailed report for errors logged by Cluster Security Services (specifically, the ctcasd daemon). RSCT directs the output from this command to the file **ctcasderr.out**. Assuming that the system log has not been redirected from its default location, you would enter: `grep ctcasd / var/adm/log /messages > ctcasderr.out` To determine the resource associated with an error in the Solaris System Log, locate the resource name in the first field of the error entry.

Related reference:

"Message format" on page 9
There are differences among the formats of error messages written to the AIX error log, or Linux, Windows, or Solaris system logs.

How to contact the IBM Support Center

IBM Customer Support Center assistance is readily available.

IBM support is available to:

- Customers without a SupportLine service contract.
- Customers with a SupportLine service contract.

Service for non-SupportLine customers:

You may access on-line support if you *do not* have an IBM SupportLine service contract.

If you *do not* have an IBM SupportLine service contract, access on-line support at:

www.ibm.com/support/

Service for SupportLine customers:

If you have an IBM SupportLine service contract, you may contact IBM by telephone in any of the following ways prescribed for your location.

1. In the United States:

- Contact IBM software support at: 1-800-237-5511.
- Contact IBM hardware support at: 1-800-IBM-SERV.

2. Outside the United States, contact your local IBM Service Center.

Contact the IBM Customer Support Center, for these problems:

- Node halt or crash not related to a hardware failure
- Node hang or response problems
- Failure in specific RSCT software subsystems
- Failure in other software supplied by IBM

The IBM representative will ask you for the information you collected from "Information to collect before contacting the IBM Support Center" on page 16.

The IBM representative will give you a time period during which another IBM representative will return your call.

For failures in non-IBM software, follow the problem reporting procedures documented for that product.

For assistance with IBM hardware failures, contact IBM Hardware Support by the means recommended above.

The IBM Customer Support Center creates a Problem Management Record (PMR) for all problems reported. A PMR is an online software record necessary for tracking software problems reported by IBM customers.

- The IBM Support Center representative will create the PMR and give you its number.
- Have the information you collected available as it will be required for completion of the PMR.
- Record the PMR number. You will need it to send any additional data to the IBM Support Center. You will also need it when on subsequent telephone calls with the IBM Support Center you may discuss this problem.

Ensure that the person you identified as your contact is accessible at the phone number that you provided in the PMR.

Diagnosing problems with the resource monitoring and control (RMC) subsystem

The Resource Monitoring and Control (RMC) subsystem is a generalized framework for managing, monitoring, and manipulating resources (physical or logical system entities).

The RMC subsystem runs as a daemon process on individual machines. You can use it to manage and monitor the resources of a single machine, or you can use it to manage and monitor the resources of a cluster's peer domain or management domain. In a peer domain or management domain, the RMC daemons on the various nodes work together to enable you to manage and monitor the domain's resources.

The term *peer domain* is defined as a set of nodes which have a consistent knowledge of the existence of each other and of the resources shared among them. On each node within the peer domain, RMC depends on a set of core cluster services, which include Topology Services, Group Services and Cluster Security Services.

The term *management domain* is defined as a set of nodes whose resources can be managed and monitored from one of the nodes, which is designated as the Management Control Point (MCP). All other nodes are considered to be Managed Nodes. Topology Services and Group Services are not used in a management domain.

When troubleshooting the RMC subsystem, it is important to note that, because of the dependencies of this subsystem on the core cluster services, problems that occur in the core cluster services may manifest in RMC. Because this is so, you should perform the diagnostic procedures for the core cluster services once you complete the initial verification tests for RMC. The most common problems caused by problems in the core cluster services are sundered or partitioned domains due to underlying network interface problems, and authentication or authorization errors due to incorrect security configuration.

Requisite function

The RMC subsystem directly uses required software components that may manifest problems as error symptoms in RMC.

If you perform all the diagnostic procedures and error responses listed in this topic, and still have problems with RMC, you should consider these components as possible sources of the error. The following list presents components in the order that they are most likely to introduce an error, from the most likely to the least likely.

- TCP/IP
- UDP/IP
- Cluster security services
- **/var** file system space, specifically the **/var/ct** directory
- **/usr/sbin/rsct** directory availability
- Topology Services/Group Services (peer domain)
- Cluster Utilities Library (libct_ct)
- System Resource Controller (SRC)

Error information

The RMC daemon writes information about important errors.

On AIX, the RSCT component subsystems write this information to the AIX error log. On Linux, Windows, and Solaris system platforms, it writes the information to the respective system log. For more information on the AIX error log, or the Linux, Windows, or Solaris system logs, see “Accessing logged errors” on page 2.

Error logs and templates

This topic lists Resource Monitoring and Control daemon error log labels and error log types with their associated explanations.

Table 10 on page 20 lists the messages that can be recorded by the RMC daemon.

Table 10. Error Log templates for the Resource Monitoring and Control daemon

Label	Type	Description
RMCD_INFO_0_ST	INFO	**Explanation:** The Resource Monitoring and Control daemon has started. **Cause:** The **startsrc -s ctrmc** command, or the **rmcctrl -s** command has been executed. **Recommended Action:** None.
RMCD_INFO_1_ST	INFO	**Explanation:** The Resource Monitoring and Control daemon has stopped. **Cause:** One of the following commands has been executed: • **stopsrc -s ctrmc** • **stopsrc -fs ctrmc** • **stopsrc -cs ctrmc** • **rmcctrl -k** **Recommended action:** Confirm that the daemon should be stopped. **Details:** A Detail Data field for this entry contains the number of the command that stopped the daemon.
RMCD_INFO_2_ST	INFO	**Explanation:** The default log file has been changed. **Cause:** The log file has become too large. For this reason, it has been renamed and a new log file has been created. **Recommended action:** None. **Details:** A Detail Data field for this entry contains the file name.
RMCD_2610_100_ER	PERM	**Explanation:** Incorrect command argument detected. **Cause:** An incorrect command argument was specified. **Recommended action:** When convenient, execute the following command: `rmcctrl -A` **Details:** A Detail Data field for this entry contains the incorrect command argument.
RMCD_2610_101_ER	PERM	**Explanation:** Internal error. **Cause:** An error in internal processing has occurred. **Recommended action:** 1. Verify that the RMC subsystem has restarted by executing the command: `lsrsrc -s ctrmc` 2. Contact the IBM Support Center. **Details:** Detail Data fields for this entry contain additional error data.

Table 10. Error Log templates for the Resource Monitoring and Control daemon (continued)

Label	Type	Description
RMCD_2610_102_ER	PERM	**Explanation:** Cannot execute with the current user ID. **Possible causes:** 1. The current user ID is not root. 2. The current user does not have the correct permissions for executing the RMC daemon. 3. The subsystem is not correctly configured in the SRC. **Recommended actions:** 1. Make sure the current user ID is root. 2. Make sure the permissions for `/usr/sbin/rsct/bin/rmcd` are set to 550. 3. Execute the command: `rmcctrl -A` **Details:** A Detail Data field for this entry contains the user ID under which the RMC daemon was executed.
RMCD_2610_103_ER	PERM	**Explanation:** Unexpected system call error. **Cause:** A system call returned an unexpected error. **Recommended action:** Verify that the RMC subsystem has restarted by executing the command: `lsrsrc -s ctrmc` **Details:** Detail Data fields for this entry contain the system call error number and the system call name.
RMCD_2610_104_ER	PERM	**Explanation:** Cannot open the Configuration Database. **Possible causes:** 1. The Configuration Database does not exist. 2. The subsystem is not configured correctly. **Recommended action:** Execute the command: `rmcctrl -A`
RMCD_2610_105_ER	PERM	**Explanation:** Error in the Configuration Database. **Possible causes:** 1. The Configuration Database is damaged. 2. The Configuration Database has been modified. **Recommended Action:** Execute the command: `rmcctrl -A`
RMCD_2610_106_ER	PERM	**Explanation:** Cannot create Configuration Database version file. **Possible cause:** The **/var** file system does not contain sufficient resources to create the Configuration Database version file. **Recommended action:** Make sure the **/var** file system contains free space and free i-nodes. Then restart the subsystem by executing the command: `rmcctrl -s` **Details:** A Detail Data field for this entry contains the Configuration Database version file name.

Table 10. Error Log templates for the Resource Monitoring and Control daemon (continued)

Label	Type	Description

RMCD_2610_107_ER	PERM	**Explanation:** Error in a Resource Manager definition file.
		
		Possible causes:
		1. The Resource Manager definition file is damaged and the associated Resource Manager is not used.
		2. The Resource Manager definition file has been modified.
		
		Recommended action:
		1. Reinstall the rsct.core fileset for the Resource Manager whose definition file had the error.
		2. When convenient, execute the following two commands:
		
		`rmcctrl -k`
		`rmcctrl -s`
		
		Details: Detail Data fields for this entry contain the system call error number, the error line, and the error position.

RMCD_2610_108_ER	PERM	**Explanation:** Cannot create default log file. **Possible cause:** The **/var** file system does not contain sufficient resources to create the default log file. **Recommended action:** Make sure the **/var** file system contains free space and free i-nodes, then restart the subsystem by executing the command: `rmcctrl -s` **Details:** A Detail Data field for this entry contains the default log file name.
RMCD_2610_109_ER	PERM	**Explanation:** Cannot create run directory. **Cause:** The **/var** file system does not contain sufficient resources to create the run directory. **Recommended Action:** Make sure the **/var** file system contains free space and free i-nodes, then restart the subsystem by executing the command: `rmcctrl -s` **Details:** A Detail Data field for this entry contains the run directory name.
RMCD_2610_110_ER	PERM	**Explanation:** Cannot create lock file. **Cause:** The **/var** file system does not contain sufficient resources to create the lock file. **Recommended action:** Make sure the **/var** file system contains free space and free i-nodes, then restart the subsystem by executing the command: `rmcctrl -s` **Details:** A Detail Data field for this entry contains the lock file name.

Table 10. Error Log templates for the Resource Monitoring and Control daemon (continued)

Label	Type	Description
RMCD_2610_111_ER	PERM	**Explanation:** Cannot start resource manager. **Cause:** The start command for the resource manager returned and error. **Recommended action:** Contact the IBM Support Center. **Details:** Detail Data fields for this entry contain the exit status of the start command, and the signal number.
RMCD_2610_112_ER	PERM	**Explanation:** Cannot create shared memory key file. **Possible cause:** The**/var** file system does not contain sufficient resources to create the key file. **Recommended action:** Make sure the **/var** file system contains free space and free i-nodes. **Details:** A Detail Data field for this entry contains the key file name.
RMCD_2610_113_ER	PERM	**Explanation:** Cannot create shared memory dump file. **Possible cause:** The **/var** file system does not contain sufficient resources to create the dump file. **Recommended action:** Make sure the **/var** file system contains free space and free i-nodes. **Details:** A Detail Data field for this entry contains the dump file name.
RMCD_2610_114_ER	PERM	**Explanation:** Error in shared memory. **Cause:** Shared memory is damaged **Recommended Action:** Contact the IBM Support Center. **Details:** Detail Data fields for this entry contain the shared memory ID and the name of the file containing a copy of the shared memory.
RMCD_2610_115_ER	PERM	**Explanation:** Cannot create message trace file. **Cause:** The **/var** file system does not contain sufficient resources to create the message trace file. **Recommended action:** Make sure the **/var** file system contains free space and free i-nodes. **Details:** A Detail Data field for this entry contains the message data trace file name.
RMCD_2610_116_ER	PERM	**Explanation:** Trace error. **Cause:** A trace function returned an unexpected error. **Recommended Action:** Contact the IBM Support Center. **Details:** Detail Data fields for this entry contain the trace error number and the trace argument.

Table 10. Error Log templates for the Resource Monitoring and Control daemon (continued)

Label	Type	Description
RMCD_2610_117_ER	PERM	**Explanation:** Cannot obtain service port number. **Possible causes:** 1. The port number is not in the file **/etc/services**. 2. The subsystem is not correctly configured. **Recommended Action:** Execute the command: `rmcctrl -A` **Details:** Detail Data fields for this entry contain the service name and protocol name.
RMCD_2610_118_ER	PERM	**Explanation:** Not responding to Group Services. **Possible cause:** The RMC daemon is not responding to Group Services in a timely manner. The RMC daemon cannot obtain system resources. **Recommended Action:** **Details:** Contact the IBM Support Center.
RMCD_2610_119_ER	PERM	**Explanation:** Cannot stop the resource manager. **Possible cause:** The resource manager stop command returned an error. **Recommended Action:** Contact the IBM Support Center and provide this information. **Details:** Detail Data fields for this entry contain the detecting module, error ID, reference code, exit status of stop command, signal number, and resource manager name.
RMCD_2610_120_ER	PEND	**Explanation:** RSCT has detected that the system time has moved backward. **Possible cause:** The system time has been set backward through a system operator or NTP action. **Recommended Action:** The RSCT components rely on system time to always increase. If the system time has moved backward, RSCT components may hang or present undefined behavior. If PowerHA SystemMirror is installed, refer to the PowerHA SystemMirror documentation. If SAMP is installed, refer to the SAMP documentation. Otherwise, if a peer domain is online, it should be forced offline using the **forcerpoffline** command. Once the peer domain is offline, or if there is no peer domain, run the **rmcctrl -z** command followed by the **rmcctrl -s** command. For more information about these commands, see their online man pages, the *RSCT for AIX: Technical Reference*, or the *RSCT for Multiplatforms: Technical Reference*. **Details:** Detail Data fields for this entry contain the detecting module, error ID, reference code, the current system time obtained by RSCT, and the last system time saved by RSCT.

While the preceding table denotes errors in the RMC subsystem itself, it writes operational errors to the file **/var/ct/IW/log/mc/default**. You may view this text file directly. The errors that the system writes to this file reflect problems in individual resource managers, RMC client connections, or resources upon

which RMC depends. Typically, the RMC daemon continues to run when it encounters such errors although, possibly, in a degraded mode. See the *Messages for RSCT* guide for explanations and recovery procedures for the errors written to this file.

Trace and core file information

Do not activate this trace facility until you have read this section completely and understand the material. If you are not certain how to properly use this facility, or if you are not under the guidance of IBM Service, **do not** activate this facility.

Activating this facility may result in degraded system performance. Activating this facility may also result in longer response times, higher processor loads, and the consumption of system disk resources. Activating this facility may also obscure or modify the symptoms of timing-related problems.

The RMC subsystem uses the common trace facility for tracking the internal activity of the daemon. You may select multiple levels of detail when diagnosing problems. You can activate additional tracing by issuing the **/usr/sbin/rsct/bin/rmctrace** command. RSCT writes the trace data to the **/var/ct/IW/log/mc/trace** file. You can view contents of the trace file with the **rpttr** command.

RSCT will write any core file that results from a program error in the RMC subsystem to the **/var/ct/IW/run/mc** directory. Upon restart, the RMC subsystem renames any core file to **core.last**. It also removes any prior core file named **core.last** . If it renames a core file to **core.last**, then it renames the trace file to **trace.last** and creates a new trace file. Thus, a trace file named **trace.last** contains a trace of the daemon activity at the time the **core.last** file is created.

Note: The **/var/ct/IW/run/mc** directory is the current working directory for the RMC daemon. If RMC abnormally terminates, the core dump file is placed in that directory, unless an alternate location has been designated for all core files. RSCT will not manage the core file retention if it is stored in the non-default location. The RSCT data gathering tool (ctsnap) will not collect the core files if they are not stored in the default location of **/var/ct/IW/run/mc**.

Diagnostic procedures

These procedures are used to verify the operation of the RMC subsystem.

To verify that RSCT has been installed, see the chapter entitled *Verifying RSCT installation* in the *Administering RSCT* guide.

Operational test 1: Checking the status of the RMC daemon

On the node, execute the **lssrc** command, as follows.

You should see output similar to the following:

```
# lssrc -s ctrmc
Subsystem         Group            PID     Status
  ctrmc           rsct             2388    active
```

If the daemon is inoperative, execute the following command on AIX system platforms to get a report from the Error Log and examine **err.out**. On Linux system platforms, examine the Syslog file `/var/log/messages`, and search for the token RMCD_2610. On Solaris system platforms, examine the `Syslog file /var/adm/messages*`. On Windows system platforms, examine the Syslog file `/var/adm/log/messages`, and search for the token RMCD_2610.

```
# errpt -a > err.out
```

If you find this token, it indicates a nonrecoverable error. The message may indicate a recovery action. If a recovery action is not specified, contact the IBM Support Center. If this token is not found, search for

the token RMCD_INFO. The *info* messages indicate start/stop status of the RMC daemon. Examine adjacent log entries to see if there are any from the SRC indicating the RMC daemon stopped with a resulting core file. Contact the IBM Support Center if a core file is found.

If there is an immediate need to have the RMC daemon running, then attempt to start it using the following command:

```
/usr/sbin/rsct/bin/rmcctrl -s
```

If the daemon does not stay active, check the Error Log or Syslog again. If there is no obvious error message, attempt to execute the RMC daemon from the command line:

```
/usr/sbin/rsct/bin/rmcd
```

If there are any problems in loading the daemon, messages should be written to the terminal indicating the problem. Correct any problems indicated by these messages. If no messages are written to the terminal, check the Error Log or Syslog and look for the error label **RMCD_2610_101_ER**. If Error data 3 is **DAE_EM_PWRONG_OTHER**, then the daemon started successfully and logged this error to indicate that it cannot be started from the command line. Contact the IBM Support Center.

Operational test 2: Checking the status of the management domain and the peer domain

Use the following procedure to check the status of the management domain and peer domain.

1. On any node, execute the **rmcdomainstatus** command.

   ```
   # /usr/sbin/rsct/bin/rmcdomainstatus -s ctrmc
   ```

   If there is no output, the node is not a member of a peer domain or a management domain. If the node is a member of a peer domain, a list similar to the following should be displayed.

   ```
   Peer Domain Status
      I A  0x09898b3065189db6  0002  c174tr6.ppd.pok.ibm.com
      S S  0x07e7287425d0becd  0001  c174tr5.ppd.pok.ibm.com
   ```

   If the node is an MCP, a list similar to the following should be displayed.

   ```
   Management Domain Status: Managed Nodes
      I a  0xbf1fb04e5b7d0b06  0001  C174tr4 !/+
      I a  0x3a75dd6c235c428e  0002  C174tr3  masMMtest/+ (1)
      I A  0x07e7287425d0becd  0003  C174tr5  masfive/+ (2)
      I A  0x09898b3065189db6  0004  C174tr6  masfive/+(2)
   ```

   If the node is a Managed Node, a list similar to the following should be displayed.

   ```
   Management Domain Status: Management Control Points
      I A  0xef889c809d9617c7 0001  9.57.24.139
   ```

   A node may be a member of a peer domain, the MCP of a management domain, and a managed node in a different management domain simultaneously. In this case, the output of the **rmcdomainstatus** would contain one or more of the preceding lists.

   Each line of output represents the status of a cluster node, relative to the node upon which the command is executed. The first token of the node status line is either **S**, **I**, **i**, **O**, **X**, or **Z**.

   S Indicates the line is the status of a peer node itself.

   I Indicates, in a management domain, that the node is Up as determined by the RMC heartbeat mechanism. In a peer domain, it indicates that the RMC daemon on the specified node is a member of the **rmc_peers** Group Services group and the node is online in the peer domain.

   i Indicates, in a management domain, the node is Pending Up. Communication has been established, but the initial handshake between two RMC daemons has not been completed. If this indicator is present upon successive executions of the **rmcdomainstatus** command, then message authentication is most likely failing. See "Diagnosing problems with cluster security services" on page 70 to validate proper security services configuration between the specified node and the node upon which the command is executed.

O Indicates, in a management domain, that the node is Down, as determined by the RMC heartbeat mechanism. In a peer domain, it indicates the RMC daemon on the specified node is no longer a member of the rmc_peers Group Services group. The most likely cause is that the node is down, as determined by the Topology Services component of RSCT. It may also indicate that the RMC daemon on the specified node is not functioning.

X Indicates, in a management domain, that a communication problem has been discovered, and the RMC daemon has suspended communications with the RMC daemon that is on the specified node. This is typically the result of a configuration problem in the network, such that small heartbeat packets can be exchanged between the RMC daemon and the RMC daemon that is on the specified node, but larger data packets cannot. This is usually the result of a difference in MTU sizes in the network adapters of the nodes. To recover, run the following command after correcting any communication problems.

```
refresh -s ctrmc
```

If the **rmcdomainstatus** output still indicates **X**, contact the IBM Support Center.

Z Indicates that the RMC daemon has suspended communication with the RMC daemon that is on the specified node because the up/down state of the node is changing too rapidly. This is typically the result of more than one node having the same node ID.

If the **rmcdomainstatus** output still indicates **Z**, contact the IBM Support Center.

The second token of the node status line is either **S**, **A**, **R**, **a**, or **r**.

S Indicates the line is the status of a peer node itself.

A Indicates that there are no messages queued to the specified node.

R Indicates that messages are queued to the specified node. If this indication persists upon repeated executions of the **rmcdomainstatus** command over several minutes, and your network is not operating under a heavy load, contact the IBM Support Center.

a Has the same meaning as **A**, but the specified node is executing a version of the RMC daemon that is at a lower code level than the local RMC daemon.

r Has the same meaning as **R**, but the specified node is executing a version of the RMC daemon that is at a lower code level than the local RMC daemon.

The third token of the status line is the ID of the specified node. The node ID is a 64-bit number that is created when RSCT is installed. It is derived using a True Random Number Generator and is used to uniquely identify a node to the RMC subsystem. The node ID is maintained in the **/var/ct/cfg/ct_node_id** file. A backup copy is maintained in the **/etc/ct_node_id** file. If this value is not unique among all systems where RSCT is installed, contact the IBM Support Center.

The fourth token of the status line is an internal node number that is used by the RMC daemon.

If the list is a list of Peer Nodes or Managed Nodes, the fifth token is the name of the node as known to the RMC subsystem. If the list is a list of MCPs, the fifth token is the first configured IP address of the specified MCP.

If the list is a list of Managed Nodes, the sixth token has the form:

```
<PD_name>/<PD_status> (n)
```

where *PD_name* is the name of the Peer Domain of which the Managed Node is an online member. Otherwise it is the **!** character and *(n)* is not present. *PD_status* is the **+** character if Peer Domain status has been received from the Managed Node. Otherwise it is the **-** character. ***n*** is the number of online nodes in the peer domain of which the specified Managed Node is a member.

If there is no status line for an expected node, contact the IBM Support Center.

2. If the following command is executed on a MCP or Managed Node,

```
# /usr/sbin/rsct/bin/rmcdomainstatus -s ctrmc -a ip
```

the fifth token in both the Managed Node list and the MCP list is the first configured IP address of the specified node. There is a subsequent line for each additional configured IP address for the specified node, consisting of only the IP address:

```
Management Domain Status: Managed Nodes
  I A  0x6dfa8e3206ff26c7  0003  9.114.113.233
  I A  0xe0870ff61109de87  0005  9.114.113.179
  I A  0xd7d2795c2516ecf8  0004  9.114.113.201
  I A  0x794697e35a3ab4c3  0006  9.114.113.200
  I A  0x7fb34d5799e489ad  0002  9.114.113.189
                                 192.160.2.1
  I A  0xd9b9a059686b4979  0001  9.114.113.188
                                 192.160.2.6
  Management Domain Status: Management Control Points
  I A  0xb2ae35236d8585bb  0001  192.160.2.8
                                 9.114.113.70
  I A  0x718336df238c7968  0002  192.160.2.10
                                 9.114.113.67
```

Operational test 3: Checking the status of microsensors

The validity of a microsensor is checked when it is created and when one of its supported attributes is monitored.

The microsensor module is loaded at creation time to make sure all necessary callbacks are present, to validate the custom dynamic attribute information (if applicable), and to calculate and store its binary signature. If any of the checks fails, the module is fenced and considered unusable. Otherwise, it is unloaded. The module is loaded again when the microsensor resource manager starts monitoring an attribute. At this point, the microsensor's binary signature is calculated again and compared to the previously-stored signature. If the signatures don't match, the microsensor is fenced.

The **IBM.MicroSensorRM** class uses two trace files, one for slow events and one for fast events. This method favors longer retention of more relevant events in separate trace files, reduces wrapping, and decreases the likelihood of information loss.

When the microsensor hangs during a callback, the entire microsensor process hangs, so it is very important that the callback functions defined within the microsensor do minimal processing and return control to the resource manager as soon as possible.

RMC usage on HMC

The Resource Monitoring and Control (RMC) subsystem running on Power Hardware Management Console (HMC) and logical partitions (LPAR) that are running AIX or Linux operating system, or Virtual I/O Server (VIOS) is responsible for establishing a management domain between the HMC and LPARs that has the HMC as its Management Control Point (MCP). This management domain formation can also be referred as RMC connection.

- Dynamic LPAR (DLPAR) operations
- Live partition mobility (LPM), hibernation, remote start
- VIOS management operations
- Operating system shutdown operation
- Capacity on Demand (CoD)
- Sending serviceable events from the operating system to the HMC

  For AIX and Linux partitions, these serviceable events can be reported to your service provider.
- Partition inventory
- AIX performance manager data collection
- Code update

An RMC connection is also used between the Power® HMC and each logical partition that is running IBM i operating system to facilitate a more limited set of functions:

- Sending serviceable events from the operating system to the HMC

  For IBM i partitions, these serviceable events are informational for the HMC. Reporting the event to the service provider is the responsibility of current versions of IBM i.
- Partition inventory
- Connection monitoring

The following figure illustrates the concept of the RMC network configuration.

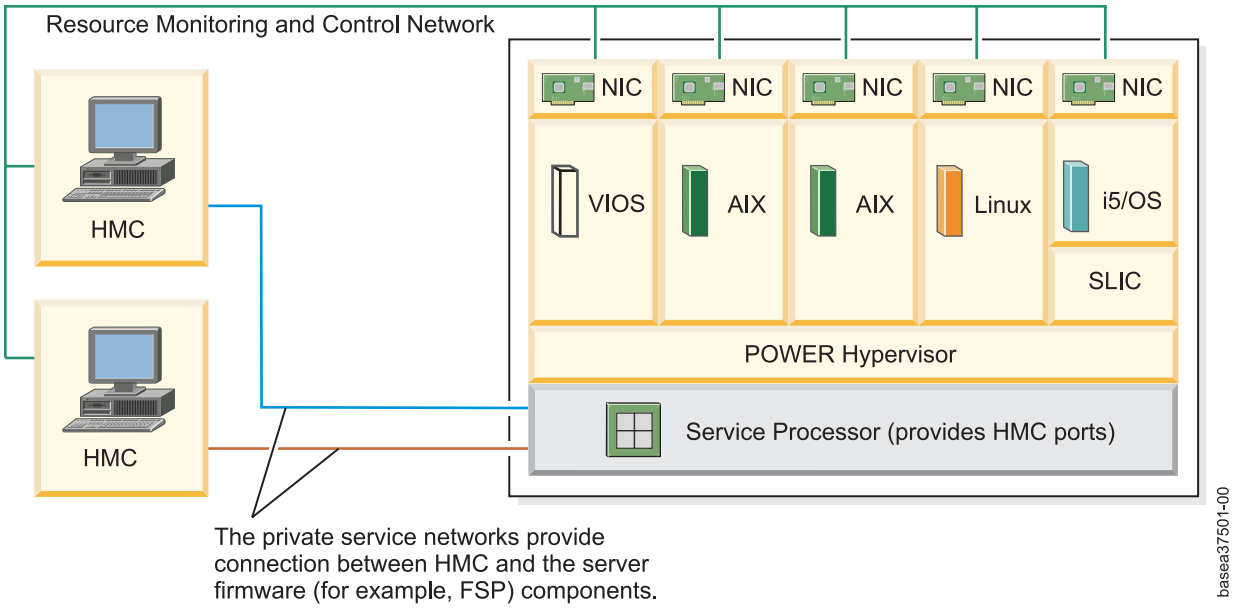

Figure 1. RMC network configuration for DLPAR support

As the number of partitions increases, it is impractical to assign a physical adapter to each partition. The following picture illustrates a more typical configuration that uses virtual Ethernet adapters (vEnet) in each client partition to implement the RMC network.

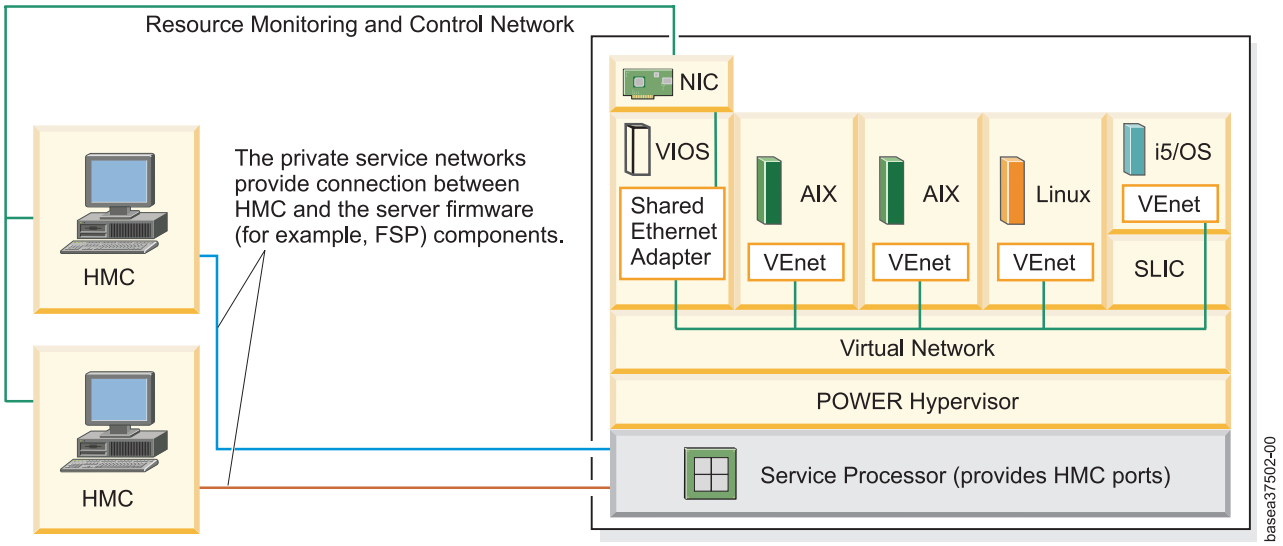

Figure 2. Typical RMC configuration for larger servers

RMC RPMs for Linux logical partitions

The RMC function is included in the AIX operating system. The Linux partitions require the RMC RPMs to be installed separately. The following Linux RMC RPMs must be installed:

```
# rpm -qa

src*.rpm
rsct.core.utils*.rpm
rsct.core*.rpm
devices.chrp.base.ServiceRM*
DynamicRM*.ppc.rpm
```

How management domain (RMC connection) is established between the HMC and the LPAR

1. For each server that is managed by an HMC, the HMC sends a packet of information that identifies the HMC (HMC name, machine type, model, and serial number), a list of the HMC's IP addresses, and a temporal key. This information is persisted in the server hypervisor. The temporal key changes each time the RMC subsystem gets restarted.
2. Every 5 minutes, the RMC component of the partition reads the HMC information packet from the hypervisor.
   - If the partition does not have a connection with the HMC, the partition's RMC component initiates the connection. The connecting sequence includes the identifying information to ensure that both parties are synchronized.
   - If the connection exists but is inactive, the partition requests the HMC to activate it.
3. After the HMC receives the connection or activation request, it verifies the identifying information of the LPAR and initiates a handshake protocol. if the handshake protocol is successful, an RMC connection is allowed and activated. This handshake protocol depends on an entry for the partition in the list of all active partitions in the `/tmp/actptnlist.txt` file.
4. Every 5 minutes, the RMC component of the HMC obtains a list of partitions for each server and updates the following partition lists:
   - List of active partitions in the `/tmp/actptnlist.txt` file
   - List of all defined (active or inactive) partitions in the `/tmp/ptnlist.txt` file

   If a partition does not exist in either list, any old connections corresponding to that partition are deleted.

5. After an RMC connection is established, HMC queries the partition for its capabilities (DLPAR, LPM, and so on) and the operating system information.
6. The RMC connection status, DLPAR capabilities, and so on, along with the operating system information are saved in the HMC (CIMOM repository) that can be displayed by using the **lssyscfg** command.
7. The RMC maintains a heartbeat from HMC to the partition via the connection. If there is an interruption in the heartbeat, both the HMC and partition attempt heartbeats to re-establish the connection. After the connection is re-established, the heartbeat flows only from the HMC with responses that are returned from the partition.

Checking the status of RMC connections

The **lssyscfg** and **lspartition** commands provides RMC connection status.

You can check the RMC connection status by running one of the following commands:

- **lssyscfg** -r lpar -m *frame-name* -F lpar_id,state, rmc_state,rmc_ipaddr, os_version,dlpar_mem_capable,dlpar_proc_capable,dlpar_io_capable --filter "lpar_ids=*LP_ID*"

  This command provides RMC connection status and the operating system capabilities. Example output follows:

```
hscroot@myhmc:~> lssyscfg -r lpar -m Frame3-top-9117-MMC-SN10364E7
-F lpar_id, state, rmc_state,rmc_ipaddr,os_version,dlpar_mem_capable,
dlpar_proc_capable,dlpar_io_capable
--filter "lpar_ids=3"

3,Running,active,10.32.244.214,AIX 6.1 6100-06-06-1140,1,1,1
```

- **lspartition** -dlpar

  This command is an internal command that is rarely used by customers. However, it is useful for RMC troubleshooting since it provides the raw RMC connection data. Example output follows:

```
hscroot@myMC:~> lspartition -dlpar | fgrep 214 -A1
<#4> Partition:<3*9117-MMC*10364E7, mycompany.com, 10.32.244.214>
Active:<1>, OS:<AIX, 6.1, 6100-06-06-1140>, DCaps:<0x2c5f>, CmdCaps:<0x1b, 0x1b>, PinnedMem:<1356>
```

Diagnosis

- If an active partition has RMC `Active<0>`, refer to the detailed diagnostics to address common RMC connection issues.
- If the **lspartition** command displays an RMC connection as `Active<1>` but the **lssyscfg** command displays none or inactive, the data that supports these two commands are not in agreement. In this event, perform a server rebuild operation on the server or restart the HMC. This operation brings the connection status data back in agreement.

Some of the common issues that cause an inactive RMC connection follow.

Notes:

- The diagnosis is explained considering that the RMC subsystem is using TCP and UDP ports 657 for the communication between HMC and partitions.
- There are typically more than one Ethernet adapters on the HMC. If an adapter is designated for partition communication on the HMC GUI panel, its IP addresses are ordered first in IP address list. The RMC component on the operating system tries to establish a single connection that starts with the first IP address on the list. If no connection is established with that IP address, the next IP address is tried until a successful connection is made.

Verifying server connection states

Verify all the managed servers on HMC have good connections to the service processor on the private service network by running the **lssyscfg** command.

```
hscroot@myMC:~> lssyscfg -r sys -F name,type_model,serial_num,state
 9.3.206.220,9179-MHD,1003EFP,No Connection
 9.3.206.223,9179-MHD,1038D0P,No Connection
```

The following states indicate good connections:

- `Operating`
- `Standby`
- `Power Off`
- `Error`
- Other transient states, for example, `Powering On`

The following states indicate problems:

- `Incomplete`
- `No Connection`
- `Recovery`

Note: In hmcV7.770.0/SP1, and earlier, a server in `No Connection`, `Incomplete`, or `Error` state have its existing connections removed and these servers prevent connections for newly activated partitions. This restriction is removed since hmcV7.770.0/SP2 release.

Diagnosis

- If server connection state is `Incomplete`, perform a server rebuild:

  ```
  hscroot@trucMC:~> chsysstate -r sys -o rebuild -m CEC_name
  ```

- If server connection state is `No Connection`, resolve it or remove it. Common issues that cause `No Connection` follow:
  - Improper firewall configuration on the network from HMC to the Fiber Service Platform (FSP).
  - More than two HMCs are attempting to manage the server.

Verifying the IP addresses used for RMC connections

List the HMC IP addresses by using the **lshmc** HMC command. In this example, the HMC has two network adapters with both IPv4 and Ipv6:

```
hscroot@myMC:~> lshmc -n -F ipaddrlpar,ipaddr,ipv6addrlpar
9.53.202.86,9.53.202.86,9.53.202.87,fe80:0:0:0:20c:29ff:fedb:4816,
fe80:0:0:0:20c:29ff:fedb:4817
```

This command lists the IP addresses that partitions use to establish RMC communication with the HMC. The **ipaddrlpr** parameter is the preferred IP address that is used to establish the connection. If a connection is not established with this IP address, RMC attempts connections on the other IP addresses in the listed order.

Diagnosis

If the IP addresses listed in this command are not correct, then one or more of the HMC network interfaces is configured incorrectly.

Verifying RMC port configuration

Verify that RMC is accepting requests from both TCP and UDP 675 ports by using the **netstat** HMC command:

```
hscroot@truchmc:~> netstat -tulpn | grep 657
tcp    0    0 :::657     :::*   LISTEN   -
udp    0    0 :::657     :::*            -
```

Diagnosis

If one of the entries is not listed, restart the HMC.

Verifying the RMC port for each partition

Verify the partition's firewall is open and authenticated for port 657 and accessible from the HMC by using the **telnet** or **ssh** commands from the HMC to establish a connection to the partition to verify network and authenticate the firewall.

```
hscroot@truchmc:~># ssh lpar_host name|IP
```

This verification must be repeated to each partition as necessary.

Diagnosis

From the HMC GUI, go to **HMC Management** → **Change Network Settings** → **LAN Adapter/Details** → **Firewall Settings** → select **Allow RMC**.

Verifying the HMC RMC port from each partition

Verify that the HMC firewall is open and authenticated for port 657 and accessible from one or more partitions.

From the partition, use the **telnet** command to verify the HMC port 657 is open for RMC use.

```
#telnet HMC_host name | IP 657
```

Diagnosis

The following problems can exhibit this symptom:

- RMC ports, specifically TCP 657, is not enabled in HMC firewall.
  Navigate to the HMC firewall as described earlier and enable the RMC port.
- RMC has a bug that it does not communicate to TCP 657.
  The only way to fix this problem is to restart HMC in order to restart RMC subsystem.

Verifying partition filesystems

Verify the partitions' `/var` and `/tmp` filesystems are not too full.

On each RS/6000® Platform Architecture (RPA) partition that does not have RMC connection to the HMC, use the **df** command to display filesystem usage.

```
# df
Filesystem   ... Use% Mounted on
/dev/hda2    ...  44% /
/dev/hda3    ...  23% /var
...
```

Diagnosis

If `/var` or `/tmp` is 100% full, remove unnecessary files or increase the filesystem sizes by using the **smitty** or equivalent Linux commands.

After changes are made to increase the space in `/var` filesystem, run the following commands to fix the potentially corrupted files.

```
# rmrsrc -s "Hostname!='t' " IBM.ManagementServer
# /usr/sbin/rsct/bin/rmcctrl -z
# rm /var/ct/cfg/ct_has.thl
# rm /var/ct/cfg/ctrmc.acls
# /usr/sbin/rsct/bin/rmcctrl -A
```

Checking for reused IP addresses

Similar to the `Duplicate NodeId` state, reused or recycled IP addresses among partitions can confuse the HMC if a new partition connection is made while the old (probably inactive) connection still exists.

The **lssyscfg -r lpar** HMC command can be used to list all the IP addresses for all RMC connections. When this list is sorted, duplicate RMC addresses stay adjacent and can be identified.

```
lssyscfg -r lpar -m CEC_name -F rmc_ipaddr,lpar_id,name,state,rmc_state | sort
```

When you scan the list, you can identify the duplicates as consecutive entries with the same first parameter (RMC IP address).

Diagnosis

If a duplicate is found, determine which is valid or expected and which IP address is invalid or stale. To correct the problem, perform the following actions:

1. On the HMC, unmanage the server corresponding to the stale RMC connection by running the following command:

   ```
   rmsysconn –ip CEC_IP
   ```

2. Wait for 6 minutes or more, then start managing the server again by running the following command:

   ```
   mksysconn -ip CEC_IP
   ```

Checking for MTU size mismatch

Most of the current versions of RMC require all parties to use the same maximum transmission unit (MTU) size. The recommended MTU setting for RMC on both HMC and partitions is 1500. If jumbo frames are required, all parties on that network must use jumbo frames.

It is entirely permissible to use different MTU sizes on other network interfaces. For example, if different HMC network adapters are used for the two networks, jumbo frames can be used on the HMC to server (Fiber Service Platform (FSP) network) while regular frames (MTU size = 1500) can be used for RMC communication.

A `No Connection` condition can be caused if there are different MTU settings between HMC and the partitions. Another possible symptom is an indefinite hang in the partition. This type of hang is recreatable by using VIOS **lsmap -all** command in a large system that produces a large output and requires multiple packages to be transferred between HMC and VIOS.

To check MTU size on partitions, run the following command:

```
#ifconfig | fgrep MTU
UP BROADCAST RUNNING MULTICAST  MTU:1500
```

To check whether jumbo frame is enabled on HMC, run the following command:

```
#lshmc -n
hostname=myhmc,...,jumboframe_eth0=off,lparcomm_eth0=off,..,jumboframe_eth1=on,lparcom_eth1=on
```

Diagnosis

This issue can be addressed by either changing the incorrect MTU sizes or by changing the HMC network interface that is used for RMC communication. To designate a different Ethernet adapter for partition communication, you can use one of the following options:

- Run the **chhmc** HMC command.
- Use the HMC GUI (**HMC Management** -> **Change Network Settings**).

Checking for duplicate node ID on the partitions

RMC uses a unique node ID to identify partitions. Having more than one partition with the same node ID confuses RMC.

If a partition is cloned improperly, it can have a duplicate node ID from the cloned partition, causing intermittent enabled or disabled connections between the partitions. It leads to disabled connections for all partition that share the duplicate node ID.

To determine whether duplicate Node IDs exist, consider the following options:

- For partitions with active RMC connections:

  From the HMC, as root user, run the **/usr/sbin/rsct/bin/rmcdomainstatus -s ctrmc** command and look for any duplicate entries. If HMC is managing a large number of partitions, it might be a difficult task.
- On partitions without an active RMC connection:

  A manual comparison can be done by comparing the `/etc/ct_node_id` file in each partition.

Diagnosis

To repair duplicate node IDs, perform the following actions on the partitions with duplicate node IDs:

1. Remove the `/etc/node_id` file, and then run the following commands to generate a new node ID.
2.

   Note: You must run the **recfgct** command only if you do not have any high availability clusters set up on this node that use the IBM PowerHA SystemMirror or IBM Tivoli System Automation for Multiplatforms (SA MP) products.

   If the LPARs are running AIX 6 with 6100-07, and later, run the following command:

   ```
   ⌂oemdelete -o CuAt -q name=cluster0 to remove 'cluster0' entry from the CuAt ODM.
   /usr/sbin/rsct/install/bin/recfgct
   ```
3. If the LPARs are running AIX version earlier than AIX 6 with 6100-07, run the following command:

   ```
   /usr/sbin/rsct/install/bin/recfgct
   ```

Related tasks:

"Operational test 2: Checking the status of the management domain and the peer domain" on page 26
Use the following procedure to check the status of the management domain and peer domain.

Related information:

Adding CEC management objects to the NIM environment

Error symptoms, responses, and recoveries

Use this information to diagnose problems with the Resource Monitoring and Control (RMC) subsystem component of RSCT.

Locate the symptom and perform the action described in Table 11 on page 36.

Table 11. RMC subsystem symptoms and recovery actions

Symptom	Recovery
RMC commands or client applications fail due to RMC subsystem session failure.	"Action 1: investigate RMC subsystem session failure"
The file **/var/ct/IW/log/mc/default** contains a message indicating the client connection was closed due to an incorrect message.	"Action 2: investigate closed client connection" on page 37

Action 1: investigate RMC subsystem session failure

Use this recovery procedure when RMC commands or client applications fail due to RMC subsystem session failure.

Symptom:

RMC commands or client applications fail due to RMC subsystem session failure.

Diagnosis:

If one of the following error messages (or a similar message indicating a session could not be established with the RMC subsystem or the session was interrupted) is displayed, either the RMC subsystem on the local node or on the node specified by *contact_name* terminated, the RMC subsystem closed the session due to a problem it discovered with the session, or the RMC subsystem rejected the connection.

```
2612-022 A session could not be established with the RMC daemon
  on contact_name.
2610-602 A session could not be established with the RMC subsystem.
2610-603 The session with the RMC subsystem has been interrupted.
2610-611 The command group has been sent, but the session was
  interrupted before all responses could be received.
```

Recovery procedure:

On the local node, the node specified by *contact_name*, or the node specified in a similar error message, perform "Operational test 1: Checking the status of the RMC daemon" on page 25. If the RMC subsystem is not operational, it is the likely cause of the command or application failure. Retry the command or restart the application after performing the recovery actions described in "Operational test 1: Checking the status of the RMC daemon" on page 25. If the RMC subsystem is operational:

1. Execute the following command:

   ```
   lssrc -ls ctrmc
   ```

2. Examine the command output for messages similar to the following:

   ```
   Daemon started on Wednesday 11/09/05 at 17:15:17
   Daemon has been running 83 days, 0 hours, 2 minutes and 30
       seconds
   ```

   These messages indicate when the RMC subsystem started and how long it has been running.

3. If the time of the RMC command or client application failure in the messages corresponds to the time the RMC subsystem last started, then either the RMC subsystem had not been running or it failed at the time the command or application executed. If this is the case, retry the command or restart the application.

   If, instead, the messages indicate that the RMC subsystem was operational at the time of the command or application failure:

   a. Examine the file **/var/ct/IW/log/mc/default** for the following message:

      ```
      2610-204 The client connection is closed due to incorrect
      message(incorrect message code)
      ```

b. If this message is found, and the timestamp associated with this message corresponds to the time of the command or application failure, then it is possible that the RMC subsystem closed the session before the command or application could complete the intended RMC operation. In this case, retry the command or restart the application. If the command or application fails with the same symptom, see the recovery procedure in "Action 2: investigate closed client connection."

   If this message is not found, then execute the following command:

   ```
   lssrc -ls ctrmc
   ```

c. If the value of this counter is not zero, the RMC subsystem has reached the limit of allowed client sessions at some point in time. Retry the command or restart the application. If the same failure occurs again, examine this counter. If it has incremented, the maximum number of client sessions has been reached. Again, execute the command **lssrc -ls ctrmc** and examine that part of the output similar to the following:

   ```
   Logical Connection Information for Local Clients
          LCID          FD          PID     Start Time
             0          40          18132    Tuesday
                        01/31/06 16:27:54
             2          39          13384    Tuesday
                        01/31/06 16:27:56

   Logical Connection Information for Remote Clients
          LCID          FD          PID     Start Time
            13          41          24024    Tuesday
                        01/31/06 17:40:05
       9.57.24.139
   ```

   The output will contain many such entries. For the local clients, verify that the process corresponding to each listed PID (process ID) is using the RMC subsystem. For remote clients, verify that the processes on the nodes specified by the listed IP address is using the RMC subsystem.

Action 2: investigate closed client connection

Use this recovery procedure when the file **/var/ct/IW/log/mc/default** contains a message indicating that the client connection was closed due to an incorrect message.

Symptom:

The file **/var/ct/IW/log/mc/default** contains the following message:

```
2610-204 The client connection is closed due to incorrect message
(incorrect message code)
```

Diagnosis:

The 2610-204 message is logged by the RMC subsystem, and the client session is closed, under the following circumstances:

- The client message policy has been violated (*incorrect message code* is 65536)
- An incorrectly formatted message has been received (*incorrect message code* is less than 65536)
- A time limit has been exceeded (*incorrect message code* is 131072)

Recovery procedure:

If the client message policy has been violated, see the description of the **rmcctrl** command's **-m** flag in *Technical Reference: RSCT for AIX* or *Technical Reference: RSCT for Multiplatforms* guides. See also the description of RMC network port usage, data flows and security in *Administering RSCT* guide.

If an incorrectly-formatted message has been received, then a program not using the RMC Application Programming Interface (RMC API) has connected to the RMC subsystem.

- If the incorrect message code is greater than 32768, then the program connected via the 657/tcp port. In this case, verify that connections to the RMC daemon are issued from trusted or permitted hosts.
- If the incorrect message code is less than or equal to 32768, the program connected via the local UNIX Domain Socket **/var/ct/IW/soc/mc/clsrv**. In this case, review usage of the local system with respect to the RMC subsystem.

If a time limit has been exceeded, verify that legitimate RMC commands or client applications connecting to the local RMC subsystem have failed. If so, then execute the following command:

```
lssrc -ls ctrmc
```

In the section of command output labeled Internal Daemon Counters, examine the value of the counters.

```
1st msg timeouts =          0  Message timeouts =          0
Start timeouts    =          0  Command timeouts =         0
```

In the command output:

- The **1st msg timeouts** counter indicates the number of times the RMC subsystem closed client connections that failed to send the first message of the start session protocol within the client message time-out limit.
- The **Message timeouts** counter indicates the number of times the RMC subsystem closed client connections that failed to send a complete message within the client message time-out limit. Use the **-t** option of the **rmcctrl** command to change the client message time-out limit.
- The **Start timeouts** counter indicates the number of times the RMC subsystem closed client connections that failed to complete the start session protocol within the start session time-out limit. Use the **rmcctrl** command with its **-u** option to change the start session time-out limit.
- The **Command timeouts** counter indicates the number of times the RMC subsystem closed client connections that failed to send the first command, subsequent to completion of start session processing, within the first command time-out limit. Use the **rmcctrl** command with its **-v** option to change the first command time-out limit.

If legitimate RMC command and client applications are failing and these counters are incrementing, the most likely cause is notwork congestion. The time-out limits may be increased. If RMC commands and client applications are failing as a result of reaching the limit on number of clients sessions, then the first command threshold may be decreased (using the **rmcctrl** command with its **-w** option), and first command time-outs for non-root authenticated client sessions can be enabled (using the **rmcctrl** command with its **-x** option). The result of these actions is to increase the number of client sessions subject to the first command timer.

If legitimate RMC command and client applications are not failing, but these counters are incrementing, it is likely that unknown applications are connecting to the 657/tcp port or the local UNIX Domain Socket. In the former case, verify that connections to the RMC daemon are issued from trusted or permitted hosts. In the latter case, review usage of the local system with respect to the RMC subsystem.

To view the current configuration of these time-out limits, execute the following command:

```
lssrc -ls ctrmc
```

Included in the output are messages similar to the following:

```
Client message timeout: 10
Client start session timeout: 60
Client first command threshold: 150
```

```
Client first command timeout: 10
Client first command timeout applies to unauthenticated and non-root
  authenticated users
```

If first command timers are not enabled (the threshold is 0), the last three messages are not present. The first command threshold can be modified using the **rmcctrl** command with its **-w** option. By default, first command timers do not apply to non-root authenticated users. To have non-root authenticated users subject to the first command time-out limits use the **rmcctrl** command with its **-x** option.

Diagnosing problems with the configuration resource manager

The configuration resource manager offers facilities to configure multiple machines (nodes) into an integrated peer domain, detects changes in the peer domain configuration, and synchronizes the configuration changes across the members of the peer domain.

A *peer domain* is a "realm", a set of *peer nodes* that have a consistent knowledge of the existence of each other and of the devices shared among them. On each node within the realm, the configuration resource manager depends on a set of core cluster services, which include Topology Services, Group Services, cluster security services, and the resource monitoring and control (RMC) subsystem.

Because of configuration resource manager subsystem dependencies on the core cluster services, problems that actually occur in the core cluster services may manifest in the configuration resource manager. To address these conditions as you trouble shoot configuration resource manager subsystems, complete the initial verification tests for the configuration resource manager before you perform diagnostic procedures for the core cluster services. Problems in the core cluster services most commonly introduce sundered or partitioned domains due to the underlying network interface problems, and authentication or authorization errors due to incorrect security configuration.

Requisite function

The configuration resource manager directly uses required software components that may manifest problems as error symptoms in the configuration resource manager.

If you perform all the diagnostic procedures and error responses listed in this topic, and still have problems with the configuration resource manager, you should consider these components as possible sources of the error. The following list presents components in the order that they are most likely to introduce an error, from the most likely to the least likely.

- TCP/IP
- UDP/IP
- UNIX Domain Sockets
- **/var** file system space, specifically the **/var/ct** directory
- **/usr/sbin/rsct** directory availability
- Topology Services/Group Services/RMC/Security
- First Failure Data Capture Library (libct_ffdc)
- Cluster Utilities Library (libct_ct)
- System Resource Controller (SRC)

Error information

The configuration resource manager writes information about important errors. On AIX system platforms, the RSCT component subsystems write this information to the AIX error log. On Linux, Windows, and Solaris system platforms, it writes the information to the respective system log.

For more information on the AIX error log, or on the Linux, Windows, or Solaris system logs, see "Accessing logged errors" on page 2.

Error logs and templates

This topic lists configuration resource manager error log labels and error log types with their associated explanations.

Table 12 lists the messages that can be recorded by the configuration resource manager.

Table 12. Error log templates for the configuration resource manager

Label	Type	Description
CONFIGRM_STARTED_ST	INFO	**Explanation:** IBM.ConfigRM daemon has started. **Cause:** The RSCT configuration resource manager (IBM.ConfigRMd) has been started. **Recommended actions:** None.
CONFIGRM_INFO_1_ST	PERM	**Explanation:** IBM.ConfigRM daemon has been stopped. **Cause:** The RSCT configuration resource manager (IBM.ConfigRMd) has been stopped. The **stopsrc -s IBM.ConfigRM** command has been executed. **Recommended actions:** Confirm that the daemon should be stopped. Normally, this daemon should not be stopped explicitly by the user.
CONFIGRM_NOQUORUM_ER	PERM	**Explanation:** The operational quorum state of the active peer domain has changed to NO_QUORUM. This indicates that recovery of cluster resources can no longer occur and that the node may be rebooted or halted in order to ensure that critical resources are released so that they can be recovered by another sub-domain that may have operational quorum. **Possible causes:** 1. One or more nodes in the active peer domain have failed. 2. One or more nodes in the active peer domain have been taken offline by the user. 3. A network failure has disrupted communication among the cluster nodes. **Recommended actions:** 1. Ensure that more than half of the nodes of the domain are online. 2. Ensure that the network that is used for communication among the nodes is functioning correctly.

Table 12. Error log templates for the configuration resource manager (continued)

Label	Type	Description
CONFIGRM_PENDINGQUORUM_ER	PERM	**Explanation:** The operational quorum state of the active peer domain has changed to PENDING_QUORUM. This state usually indicates that exactly half of the nodes that are defined in the peer domain are online. In this state, cluster resources cannot be recovered although none will be stopped explicitly. **Possible causes:** 1. One or more nodes in the active peer domain have failed. 2. One or more nodes in the active peer domain have been taken offline by the user. 3. A network failure is disrupted communication among the cluster nodes. **Recommended actions:** 1. Ensure that more than half of the nodes of the domain are online. 2. Ensure that the network that is used for communication among the nodes is functioning correctly. 3. Ensure that the active tiebreaker device is operational and, if it set to 'Operator', then resolve the tie situation by granting ownership to one of the active sub-domains. See the *Administering RSCT* guide for more information.
CONFIGRM_HASQUORUM_ST	INFO	**Explanation:** The operational quorum state of the active peer domain has changed to HAS_QUORUM. In this state, cluster resources may be recovered and controlled as needed by management applications. **Cause:** One or more nodes have come online in the peer domain. **Recommended actions:** None.
CONFIGRM_REBOOTOS_ER	PERM	**Explanation:** The operating system is being rebooted to ensure that critical resources are stopped so that another sub-domain that has operational quorum may recover these resources without causing corruption or conflict. **Cause:** Critical resources are active and the active sub-domain does not have operational quorum. **Recommended actions:** After node finishes rebooting, resolve problems that caused the operational quorum to be lost.
CONFIGRM_HALTOS_ER	PERM	**Explanation:** The operating system is being halted to ensure that critical resources are stopped so that another sub-domain that has operational quorum may recover these resources without causing corruption or conflict. **Cause:** Critical resources are active and the active sub-domain does not have operational quorum. **Recommended actions:** Boot the operating system and resolve any problems that caused the operational quorum to be lost.

Table 12. Error log templates for the configuration resource manager (continued)

Label	Type	Description
CONFIGRM_EXITCS_ER	PERM	**Explanation:** The cluster software will be forced to recycle the node through an offline/online transition to recover from an error. Note that this will not guarantee that critical cluster resources are stopped, and therefore does not prevent corruption or conflict if another sub-domain attempts to recover these resources. **Cause:** Critical resources are active and the active sub-domain does not have operational quorum. **Recommended actions:** 1. Manually stop any critical resources so that another sub-domain may recover them. 2. Resolve any problems preventing other nodes of the cluster from being brought online or resolve any network problems preventing the cluster nodes from communicating.
CONFIGRM_EXIT_CONFIG_ST	INFO	**Explanation:** The peer domain configuration manager daemon (IBM.ConfigRMd) is exiting due to the local node's configuration version being different from that of the active domain. The daemon will be restarted automatically and the configuration of the local node will be synchronized with the domain. **Cause:** The domain configuration changed while the node was coming online. **Recommended actions:** None.
CONFIGRM_EXIT_COMMIT_ER	PERM	**Explanation:**A configuration change was applied, but could not be committed. For this reason, the node will be taken offline and back online. During the online processing , the configuration will be synchronized if the problem as been cleared. **Cause:** Insufficient free space in the **/var** filesystem. **Recommended actions:** Ensure there is sufficient free space in the **/var** filesystem.
CONFIGRM_EXIT_GS_ER	PERM	**Explanation:** The peer domain configuration manager daemon (IBM.ConfigRMd) is exiting due to the Group Services subsystem terminating. The configuration resource manager daemon will restart automatically, synchronize the node's configuration with the domain, and rejoin the domain if possible. **Possible causes:** 1. The Group Services subsystem detected another sub-domain and is attempting to merge with it. 2. The Group Services subsystem has failed. **Recommended actions:** No action is necessary. Recovery should be automatic.
CONFIGRM_MERGE_ST	INFO	**Explanation:** The sub-domain containing the local node is being dissolved because another sub-domain has been detected that takes precedence over it. Group services will be ended on each node of the local sub-domain. This will cause the configuration resource manager daemon (IBM.ConfigRMd) to force the node offline and then bring it back online in the surviving domain. **Cause:** A merge of two sub-domains is usually caused by a network outage being repaired, enabling the nodes of the two sub-domains to communicate. **Recommended actions:** No action is necessary since the nodes will be automatically synchronized and brought online in the surviving domain.

Table 12. Error log templates for the configuration resource manager (continued)

Label	Type	Description
CONFIGRM_ONLINE_ST	INFO	**Explanation:** The node is online in the domain indicated in the detail data. **Possible causes:** 1. A user ran the **startrpdomain** or **startrpnode** commands. 2. The node rebooted while the node was online. 3. The configuration resource manager recycled the node through an offline/online transition to synchronize the domain configuration, or to recover from some other failure. **Recommended actions:** None.
CONFIGRM_OFFLINE_ST	INFO	**Explanation:** The node is offline. **Possible causes:** 1. A user ran the **stoprpdomain** or **stoprpnode** command. 2. There was a failure while trying to bring the node online. **Recommended actions:** If the node is offline due to a failure, try to resolve the failure and then run the **startrpdomain** or **startrpnode** command to bring the node online.
CONFIGRM_ONLINEFAILED_ER	PERM	**Explanation:** An error was encountered while the node was being brought online. The configuration resource manager daemon (IBM.ConfigRMd) will attempt to return the node to an offline state. **Possible causes:** **Cause:** Failure in a dependent subsystem such as RMC. See the detailed error fields for the specific error. **Recommended actions:** Resolve the problem indicated in the detailed data fields and try bringing the node online using the **startrpnode** or **startrpdomain** command.
CONFIGRM_OFFLINEFAILED_ER	PERM	**Explanation:** An error was encountered while the node was being taken offline. The configuration resource manager daemon (IBM.ConfigRMd) will exit and restart in an attempt to recover form this error. **Possible causes:** **Cause:** Failure in a dependent subsystem such as RMC. See the detailed error fields for the specific error. **Recommended actions:** If the configuration resource manager daemon (IBM.ConfigRMd) fails to restart after attempting to recover from this error, contact your software service organization.

Trace and core file information

Do not activate this trace facility until you have read this section completely, and understand this material.

If you are not certain how to properly use this facility, or if you are not under the guidance of IBM Service, **do not** activate this facility.

Activating this facility may result in degraded performance of your system. Activating this facility may also result in longer response times, higher processor loads, and the excessive consumption of system disk resources. Activating this facility may also obscure or modify the symptoms of timing-related problems.

The configuration resource manager uses the Common Trace Facility for tracking the internal activity of the daemon. Multiple levels of detail may be selected when diagnosing problems. Additional tracing can be activated by the **traceon** or **ctsettrace** utility. All trace files are written to **/var/ct/IW/log/mc/IBM.ConfigRM**. Each file in this directory named trace *n* will correspond to a separate execution of the resource manager. The latest file which corresponds to the current execution of the resource manager is called trace. Trace files for prior runs have a suffix of *.n* where *n* starts at 0 and increases for older runs. The trace files may be viewed with the **rpttr** command.

Any core files that result from a program error in this resource manager will be written to **/var/ct/IW/run/mc/IBM.ConfigRM**. As for the trace files, older core files will have an *.n* suffix which increases with age. Core files and trace files with the same suffix correspond to the same execution instance. The configuration resource manager manages the space in **log** and **run** directories described above so that the total amount of disk space used is less than 10 megabytes. Trace files without corresponding core files will be removed first if the resource manager is over its limit. Then pairs of core files and trace files will be removed starting with the oldest. At least one core/trace file pair will always be retained.

Note: The **/var/ct/IW/run/mc/IBM.ConfigRM** directory is the current working directory for the configuration resource manager. If IBM.ConfigRM terminates abnormally, the core dump file is placed in that directory, unless an alternate location has been designated for all core files. RSCT will not manage the core file retention if it is stored in the non-default location. The RSCT data gathering tool (ctsnap) will not collect the core files if they are not stored in the default location of **/var/ct/IW/run/mc/IBM.ConfigRM**.

Diagnostic procedures

These procedures are used to verify the operation of the configuration resource manager.

To verify that RSCT has been installed, see the *Verifying RSCT installation* chapter in the *Administering RSCT* guide.

Operational test 1: verifying the configuration resource manager availability

This test verifies if the configuration resource manager is active on a node and the peer domain is online.

To determine if the peer domain is active, issue the **lssrc** command:

```
lssrc -a|grep ConfigRM
```

If there are no errors, the following output is displayed:

```
 IBM.ConfigRM     rsct_rm          553016       active
```

If there are errors, the following output is displayed:

```
IBM.ConfigRM     rsct_rm          553016       inoperative
```

If the configuration resource manager is inactive, see "Operational test 2: determining why the configuration resource manager is inactive" on page 48 and "Error symptoms, responses, and recoveries" on page 49.

If the configuration resource manager subsystem is active, check if the domain is Online. This can be done by issuing the **lsrpdomain** command:

```
lsrpdomain
```

If there are no errors, the following output is displayed:

```
Name       OpState RSCTActiveVersion MixedVersions TSPort GSPort
IBMCluster Online  3.1.4.0           No            12347  12348
```

Error results:

1. If the domain is offline, the following output is displayed:

   ```
   Name       OpState  RSCTActiveVersion MixedVersions TSPort GSPort
   IBMCluster Offline  3.1.4.0           No            12347  12348
   ```

2. If there are problems interacting with the RMC daemon, the following output is displayed:

   ```
   /usr/sbin/rsct/bin/lsrsrc-api: 2612-022 A session could not be
   established with the RMC daemon on "local_node".
   ```

   This is due to momentary interruption of RMC monitoring. The RMC daemon and all resource managers except the configuration resource manager need to be shut down and restarted on peer domain nodes whenever a transition is made between IW and peer domain mode on a node. This will discontinue the monitoring activities momentarily on the nodes until the RMC daemon is restarted and resumes monitoring.

   If however, the error persists then the node is experiencing start up problems.

3. If the peer domain shows it is in pending state, the following output is displayed:

   ```
   Name       OpState         RSCTActiveVersion MixedVersions TSPort GSPort
   IBMCluster Pending Online  3.1.4.0           No            12347  12348
   ```

   The preceding output indicates a transitional state, and is generally not an error. However, if the Domain continues to show a **Pending Online** or **Pending Offline** state, see "Operational test 2: determining why the configuration resource manager is inactive" on page 48.

If the peer domain is online, issue the **lsrpnode** command to check if all the nodes in the peer domain are also online.

lsrpnode

If there are no errors, the following output is displayed:

```
Name      OpState RSCTVersion
davrosp01 Online  3.1.4.0
davrosp04 Online  3.1.4.0
```

If there are errors, the following output is displayed:

```
Name      OpState  RSCTVersion
davrosp01 Offline  3.1.4.0
davrosp04 Online   3.1.4.0
```

If the error persists, then the **Offline** node is experiencing start up problems. In order to get detail status of the configuration resource manager and its resources or resource classes, issue the following commands from a node that is Online in the peer domain:

1. In order to get the status of the configuration resource manager, issue:

   ```
   lssrc -ls IBM.ConfigRM
   ```

   The following output is displayed:

```
Subsystem         : IBM.ConfigRM
PID               : 553016
Cluster Name      : IBMCluster
Node Number       : 1
Daemon start time : Fri Oct  8 17:15:47 EDT 2004
Daemon State: Online in IBMCluster  <- points to the peer node state
ConfigVersion: 0x14167efaf
Group IBM.ConfigRM:
    Providers: 1
    GroupLeader: davrosp01, 0xebf461dcb6d2479a, 1  <- points to the
                                                      Group Leader Node
Information from malloc about memory use:
    Total Space    : 0x012d02b0 (19727024)
    Allocated Space: 0x00cb6018 (13328408)
    Unused Space   : 0x00616eb0 (6385328)
    Freeable Space : 0x00000000 (0)
```

2. In order to check the peer domain resource class, issue:

   ```
   lsrsrc IBM.PeerDomain
   ```

   The following output is displayed:

   ```
   Resource Persistent Attributes for IBM.PeerDomain resource 1:
            Name              = "my_caa"
            RSCTActiveVersion = "3.1.4.0"
            MixedVersions     = 0
            TSPort            = 12347
            GSPort            = 12348
            RMCPort           = 657
            ResourceClasses   = {}
            QuorumType        = 4
            DomainType        = 1
            ActivePeerDomain  = "my_caa"
   ```

   Note: A `DomainType = 1` indicates that the RPD is created in the Cluster-Aware AIX (CAA) environment by using the **mkrpdomain** command with the **-C** option. A `DomainType = 0` indicates that the RPD is created by using the **mkrpdomain** command without the **-C** option.

3. In order to check the peer Node resource class, issue:

   ```
   lsrpnode -i
   ```

   The following output is displayed:

   ```
   Name         OpState RSCTVersion NodeNum NodeID
   node02.ppd.pok.ibm.com Online  3.1.4.0     4       2199b8719d8ac4e2
   node03.ppd.pok.ibm.com Online  3.1.4.0     3       18d9d0c39e2380e2
   node04.ppd.pok.ibm.com Online  3.1.4.0     2       3bc2b0f98ff83e2
   node01.ppd.pok.ibm.com Online  3.1.4.0     1       da5902cf92db57e2
   ```

   Note: In a peer domain containing a large number of nodes, the **lsrsrc** command output will be easier to read if the information is returned in a tabular format. To have the information returned in a tabular format specify the **-t** flag on the **lsrsrc** command.

4. In order to check the Network Interface resource class, issue:

   ```
   lsrsrc -A b IBM.NetworkInterface
   ```

The following output is displayed:

```
Resource Persistent and Dynamic Attributes for IBM.NetworkInterface
resource 1:
         Name             = "en0"
         DeviceName       = ""
         IPAddress        = "9.112.55.74"
         SubnetMask       = "255.255.255.128"
         Subnet           = "9.112.55.0"
         CommGroup        = ""
         HeartbeatActive  = 1
         Aliases          = {}
         DeviceSubType    = 1
         LogicalID        = 0
         NetworkID        = 0
         NetworkID64      = 0
         PortID           = 0
         HardwareAddress  = "4e:b5:3f:9c:46:02"
         DevicePathName   = ""
         IPVersion        = 4
         Role             = 0
         ActivePeerDomain = "my_caa"
         NodeNameList     = {"e105n2ec01.ppd.pok.ibm.com"}
         OpState          = 1
         ConfigChanged    = 0
resource 2:
         Name             = "en0"
         DeviceName       = ""
         IPAddress        = "9.114.55.76"
         SubnetMask       = "255.255.255.128"
         Subnet           = "9.114.55.0"
         CommGroup        = ""
         HeartbeatActive  = 1
         Aliases          = {}
         DeviceSubType    = 1
         LogicalID        = 0
         NetworkID        = 0
         NetworkID64      = 0
         PortID           = 0
         HardwareAddress  = "4e:b5:33:64:91:02"
         DevicePathName   = ""
         IPVersion        = 4
         Role             = 0
         ActivePeerDomain = "my_caa"
         NodeNameList     = {"e105n2ec03.ppd.pok.ibm.com"}
         OpState          = 1
         ConfigChanged    = 0
```

An Opstate of 1 signifies that the interface is configured and up. An Opstate of 2 signifies that the interface is down. If this is the case and the HeartBeatActive flag is set to 1, the problem is likely related to Topology Services. See “Diagnosing problems with Topology Services” on page 133 for more information.

In order to get detailed Network Interface information for all the nodes, issue the **lsrsrc** command with the CT_MANAGEMENT_SCOPE environment variable set to 2:

```
export CT_MANAGEMENT_SCOPE=2  lsrsrc -A b IBM.NetworkInterface
```

5. In order to check the communication groups of a peer domain, issue:

   lscomg

   The following output is displayed:

```
Name Sensitivity Period Priority Broadcast SourceRouting
          NIMPathName NIMParameters
CG1  4           1      1        Yes       Yes
CG2  4           1      1        Yes       Yes
CG3  4           1      1        Yes       Yes
```

   The preceding output shows that the Peer Domain has three communication groups.
   For details regarding the interfaces that each communication group refers to, issue:

```
lscomg -i  communication_group_name
```

   For example, to list the interfaces in the communication group named *CG1* in the preceding output, you would enter:

```
lscomg -i CG1
```

   The following output is displayed:

```
Name NodeName                   IPAddress  Subnet     SubnetMask
en0  davrosp02.ppd.pok.ibm.com 9.224.30.2 9.224.30.0 255.255.255.192
en0  davrosp04.ppd.pok.ibm.com 9.224.30.4 9.224.30.0 255.255.255.192
en0  davrosp01.ppd.pok.ibm.com 9.224.30.1 9.224.30.0 255.255.255.192
```

Operational test 2: determining why the configuration resource manager is inactive

Use this test to determine why the configuration resource manager is inactive.

Table 13 on page 49 details those tests useful in determining why the configuration resource manager may be inactive. These tests vary by node type, as follows:

Table 13. Determine CRM inactivity cause on Linux, AIX, Windows, and Solaris nodes

On Linux nodes:	On AIX nodes:	On Windows nodes:	On Solaris nodes:
Issue the command: `fcslogrpt /var/log /messages` ...and look for entries for subsystem **ConfigRM**. The syslog entries produced by this command, together with their descriptions in Table 12 on page 40, explain why the subsystem is inactive. If no entry exists that explains why the subsystem went down or could not start, it is possible that the daemon exited abnormally. In this case, issue the **fcslogrpt /var/log/messages** command and look for an error. Look for an error entry with a LABEL: of **CORE_DUMP** and PROGRAM NAME of **ConfigRM**. If you find such an entry, see "Information to collect before contacting the IBM Support Center" on page 16 and contact the IBM Support Center.	For an RSCT peer domain, issue the command: `errpt -N ConfigRM -a` The AIX error log entries produced by this command, together with their descriptions in Table 12 on page 40, explain why the subsystem is inactive. If no entry that explains why the subsystem went down or could not start exists, it is possible that the daemon exited abnormally. In this case, issue the **errpt -a** command and look for an error. Look for an error entry with a LABEL: of CORE_DUMP and PROGRAM NAME of **ConfigRM**. (Issue the command: **errpt -J CORE_DUMP -a.**) If you find such an entry, see "Information to collect before contacting the IBM Support Center" on page 16 and contact the IBM Support Center.	For an RSCT peer domain, issue the command: `grep ConfigRM /var/adm/log /messages > ConfigRM.out` The Windows error log entries produced by this command, together with their descriptions in Table 12 on page 40, explain why the subsystem is inactive. If no entry that explains why the subsystem went down or could not start exists, it is possible that the daemon exited abnormally. In this case, issue the **cat /var/adm/log/messages** command and look for an error entry with a LABEL: of **CORE_DUMP** and PROGRAM NAME of **ConfigRM** If you find such an entry, see "Information to collect before contacting the IBM Support Center" on page 16 and contact the IBM Support Center.	For an RSCT peer domain, issue the command: `grep ConfigRM /var/adm/messages > ConfigRM.out` The Solaris system log entries produced by this command, together with their descriptions in Table 12 on page 40, explain why the subsystem is inactive. If no entry that explains why the subsystem went down or could not start exists, it is possible that the daemon exited abnormally. In this case, issue the cat **/var/adm/messages** command and look for an error. Look for an error entry with a LABEL: of **CORE_DUMP** and PROGRAM NAME of **ConfigRM**. If you find such an entry, see "Information to collect before contacting the IBM Support Center" on page 16 and contact the IBM Support Center.

Error symptoms, responses, and recoveries

Use this information to diagnose problems with the configuration resource manager component of RSCT.

Use Table 14 to diagnose problems with the configuration resource manager component of RSCT. Locate the symptom and perform the action described in the following table.

Table 14. Configuration resource manager symptoms and recovery actions

Symptom	Recovery
Configuration resource manager commands fail due to insufficient space in the file system.	"Action 1: check the /var file system" on page 50
The **mkrpdomain** command fails with authentication or authorization errors.	"Action 2: investigate authentication and authorization problems when creating a domain, starting a domain, or adding a node to a domain" on page 51
The **startrpdomain** or **startrpnode** command fails with authorization errors.	"Action 2: investigate authentication and authorization problems when creating a domain, starting a domain, or adding a node to a domain" on page 51
The **addrpnode** command fails with authentication errors.	"Action 2: investigate authentication and authorization problems when creating a domain, starting a domain, or adding a node to a domain" on page 51
An authorization error for the IBM.PeerDomain resource class appears in the configuration resource manager trace file.	"Action 2: investigate authentication and authorization problems when creating a domain, starting a domain, or adding a node to a domain" on page 51
Configuration changes are rejected by the configuration resource manager due to insufficient quorum.	"Action 3: investigate quorum problems" on page 55
The configuration resource manager reports a duplicate IP address error.	"Action 4: investigate duplicate IP address problems" on page 56
A peer node is unable to rejoin the cluster.	"Action 5: investigate node startup failure" on page 57

Table 14. Configuration resource manager symptoms and recovery actions (continued)

Symptom	Recovery
A peer domain has been partitioned into two domains.	"Action 6: respond to cluster partitioning" on page 58
Interfaces on a node or set of nodes are not part of the heartbeat ring of the configuration resource manager.	"Action 7: add an interface to the heartbeat ring of the configuration resource manager" on page 60
Unable to add a node to a peer domain.	"Action 8: investigate failure to add a node" on page 60
Peer domain operations fail. Errors indicate there was a problem in establishing a session with the RMC Subsystem.	"Action 9: investigate accessibility of peer domain nodes" on page 62 "Action 10: investigate responsiveness of RMC subsystem" on page 63
The configuration resource manager is inoperative or a node cannot be brought online in the peer domain.	"Action 11: check the root file system" on page 64
Configuration resource manager commands, RMC commands, or RMC client operations fail.	"Action 12: check the /tmp file system" on page 64
A peer node remains in the pending online state indefinitely.	"Action 16: force a peer domain offline" on page 67, "Action 17: synchronize the time-of-day clocks in the peer domain" on page 67
The **stoprpdomain** command fails with an error indicating that no resources are online.	"Action 19: investigate the stoprpdomain command failure because of critical resources being active on nodes" on page 69

Action 1: check the /var file system

Common operational problems in the configuration resource manager occur when the **/var** file system runs out of space. For example, the following information describes a typical symptom seen during the peer domain setup. Take this action when configuration resource manager commands fail due to insufficient space in the file system.

Symptom:

Configuration resource manager commands fail due to insufficient space in the file system.

Diagnosis:

There is likely insufficient space in the **/var** file system if the system returns one or both of the following errors:

```
2650-943 ctsthl Failure: Insufficient space in file system.
The file system where the trusted host list file is stored has
insufficient space available.  The modification attempted by this
command has failed.
         Trusted Host List File name: /var/ct/cfg/ct_has.thl
Contact the system administrator and report this problem.  System
administrators should extend the size of the file system where this
file is stored, remove unnecessary files from this file system, or
compress files residing in this file system to regain storage.

preprpnode: 2602-342 Trusted host list file update for
zagreus.ppd.ibm.com  failed with return code 21.
```

If the system returns either of the preceding errors, issue the **df** command to check the amount of free space available in the **/var** file system.

Recovery procedure:

Increase the size of the **/var** file system.

Action 2: investigate authentication and authorization problems when creating a domain, starting a domain, or adding a node to a domain

Multiple symptoms suggest this action. Take this action when you encounter any of the following symptoms.

Symptom 1: the mkrpdomain command fails with an authentication error:

This symptom suggests the following diagnosis and recovery.

Diagnosis:

There is likely an authentication error if the system returns one or both of the following errors:

```
2632-044 The domain cannot be created due to the following
errors that were detected while harvesting information from the
target nodes:

davrosp67: 2645-061 The requesting node cannot be authenticated
by the target node.
```

If you get either of the preceding errors, check the **/etc/hosts** file on the problem node. The preceding message identifies the problem node as *davrosp67*. A good entry will have a format like the following:

```
127.0.0.1       localhost.localdomain   localhost
```

An example of an erroneous entry that can cause the host name to resolve to the loopback address, resulting in an authentication error, is:

```
127.0.0.1  zagreus1.ibm.com  zagreus1    localhost.localdomain
   localhost
```

Recovery procedure:

In case it is determined that the **/etc/hosts** entry is incorrect and the host name is being resolved with the loopback address, correct the entry in the **/etc/hosts** file. In case the authentication problem persists, recovery procedure at the end of this Action section.

Symptom 2: the mkrpdomain command fails with authorization errors:

This symptom suggests the following diagnosis.

Diagnosis:

There is likely an authentication error if the system returns one or both of the following errors:

```
2632-044 The domain cannot be created due to the following errors
that were detected while harvesting information from the target
nodes:

davrosp02: 2610-418 Permission is denied to access the resources
or resource class specified in this command.
```

Symptom 3: the startrpdomain or startrpnode command fails because of authorization errors:

This symptom suggests the following diagnosis.

Diagnosis:

There is likely an authorization error if the system returns any or all of the following errors:

```
2632-046 The following errors were detected while attempting to
find the latest configuration for the domain.  The domain cannot be
brought online.

2632-024 The following error was returned from the RMC subsystem
while attempting to contact node 9.222.30.1 during a start domain
operation.

2610-418 Permission is denied to access the resources or resource
class specified in this command.
```

Symptom 4: the addrpnode command fails because of authentication errors:

This symptom suggests the following diagnosis and recovery.

Diagnosis:

There is likely an authentication error if the system returns either or both of the following errors:

```
2632-077 The following problems were detected while adding nodes
 to the domain. As a result, no nodes will be added to the domain.

davrosp04: 2610-418 Permission is denied to access the resources
or resource class specified in this command.
```

The messages in the preceding symptoms indicate that the underlying security setup is faulty. In case of an authorization error it implies the originator credential is authenticated (originator is in the Cluster Security Services Trusted Host List file: **/var/ct/cfg/ct_has.thl**) but the originator IP address is not in **/var/ct/cfg/ctrmc.acls** or it is not in **/var/ct/cfg/ctsec.nodeinfo**. In case of authentication errors, it points to the fact that the entry for an interface is probably missing from the Trusted Host List file.

1. Check if the **/var/ct/cfg/ctsec.nodeinfo** file has the appropriate entries for all nodes in the peer domain. If this file is missing, then it basically points to the fact that this node has either never been configured to be part of the peer domain or the node has been removed from the peer domain. If the file exists and entries for any node are missing, it could result in domain startup problems for the node.

   Pick any node in the peer domain and issue the following command. In our example, we are on a three-node peer domain *IBMCluster* with nodes *davrosp01*, *davrosp02* and *davrosp04*. We enter the command on *davrosp04*.

   ```
   cat /var/ct/cfg/ctsec.nodeinfo
   ```

   Output will show the entries in the **ctsec.nodeinfo** file, and should be similar to the following:

   ```
   NODE:
        NAME:  davrosp02.ppd.pok.ibm.com davrosp02
               0xcb3de83d2c7f84a4
        ADDRESS: 10.10.11.2 9.222.30.2 192.169.1.2 192.169.0.2
            0xcb3de83d2c7f84a4 CLUSTER: IBMCluster

   NODE:
        NAME:  davrosp01.ppd.pok.ibm.com davrosp01
               0xebf461dcb6d2479a
        ADDRESS: 10.10.11.1 9.222.30.1 192.169.1.1 192.169.0.1
            0xebf461dcb6d2479a CLUSTER: IBMCluster

   NODE:
        NAME: LOCALHOST  davrosp04.ppd.pok.ibm.com davrosp04
   ```

```
    0x7f4b34c8852def94
ADDRESS: 10.10.11.4 9.222.30.4 192.169.0.4
      0x7f4b34c8852def94
CLUSTER: IBMCluster
```

A typical entry for a three node peer domain IBMCluster with nodes has the following entries in the **ctsec.nodeinfo** file.

2. Check the **/var/ct/cfg/ctrmc.acls** ACL file. An entry for IBM.PeerDomain should exist. A typical entry is shown below:

   Issue the command:

   ```
   cat /var/ct/cfg/ctrmc.acls
   ```

   Output should be similar to the following:

   ```
   IBM.PeerDomain
        none:root      *   rw // root on any node of active
                       cluster
        none:any_root  *   rw // root on any node of any cluster
                           that this node is defined to
        root@davrosp01.ppd.pok.ibm.com *   rw   // cluster node
        root@davrosp02.ppd.pok.ibm.com *   rw   // cluster node
        root@davrosp04.ppd.pok.ibm.com *   rw   // cluster node
        root@9.222.30.2 *   rw   // cluster node
        root@9.222.30.1 *   rw   // cluster node
        root@9.222.30.4 *   rw   // cluster node
   ```

3. Perform cluster security services diagnosis to ensure that the initiating system is recognized as a trusted host by the intended target system. See "Diagnosing problems with cluster security services" on page 70 for more information.

Recovery procedure:

To recover from authorization or authentication errors, you need to introduce the missing interface into the ACL files or the trusted host list file as needed. The best way to do this is to execute the **preprpnode** command with the IP address of the missing interface(s) and the IP address of the configuration resource manager Group Leader node. This will add the missing IP address to the ACL/Trusted host list file on each node.

The **preprpnode** command on each node performs the following steps:

1. Establishes trust with the node names/IP addresses of the interfaces specified on the command by adding their public keys to the trusted host list.
2. Modifies the resource monitoring and control (RMC) access control list (ACL) file to enable access to peer domain resources on this node from the other nodes in the peer domain. This allows peer domain operations to occur on the node. The RMC subsystem is refreshed so that these access changes will take effect.
3. Enables RMC remote connections

Symptom 5: an authorization error for the IBM.PeerDomain resource class appears in the configuration resource manager trace file:

This symptom suggests the following diagnosis and recovery.

Diagnosis:

Errors similar to the following seen in the configuration resource manager trace file can result from an inconsistency in the resolution of host names on different nodes. Such an inconsistency could be caused by a mismatch between the short and long (fully-qualified) host names in the trusted host lists used by different nodes.

```
06/26/06 13:52:32.774795 T(1084024048) _CFD id=0xffffffffError
262160 was returned from "UpdateConfigOp::handleCallback" on
line 189 in file "/project/spreldeb/build/rdebs002a/src/rsct/rm
/ConfigRM/Update ConfigOp.C".
Message=2610-441 Permission is denied to access the resource
class specified in this command.
Network Identity UNAUTHENT requires 's' permission for the
resource class IBM.PeerDomain on node c701f1sq03.
```

For example, consider the following inconsistency in the contents of **/etc/hosts** on two different nodes:

On node c701f1sq02:

```
c701f1sq02:~ # grep c701f1sq02 /etc/hosts
192.168.8.2     c701f1sq02ib0 c701f1sq02ib0.ppd.pok.ibm.com
192.168.9.2     c701f1sq02ib1 c701f1sq02ib1.ppd.pok.ibm.com
192.168.14.2    c701f1sq02eth2 c701f1sq02eth2.ppd.pok.ibm.com
9.114.187.2     c701f1sq02.ppd.pok.ibm.com c701f1sq02
```

On node c701f1sq03:

```
c701f1sq03:/var/ct/cfg # grep c701f1sq02 /etc/hosts
9.114.187.2     c701f1sq02 c701f1sq02.ppd.pok.ibm.com
192.168.8.2     c701f1sq02ib0 c701f1sq02ib0.ppd.pok.ibm.com
192.168.9.2     c701f1sq02ib1 c701f1sq02ib1.ppd.pok.ibm.com
192.168.14.2    c701f1sq02eth2 c701f1sq02eth2.ppd.pok.ibm.com
```

Recovery procedure:

Use the **ctsvhbal** and **ctsvhbar** tools to help identify inconsistencies in host name resolution and make the appropriate corrections to the trusted host list.

Continuing the preceding example, the output from these tools resembles the following:

On node c701f1sq02:

```
c701f1sq02:~ # /usr/sbin/rsct/bin/ctsvhbal
ctsvhbal: The Host Based Authentication (HBA) mechanism identities
for the local system are:
                Identity:  c701f1sq02.ppd.pok.ibm.com
                Identity:  c701f1sq02eth2
                Identity:  192.168.14.2
                Identity:  9.114.187.2
                Identity:  c701f1sq02ib1
                Identity:  192.168.9.2
                Identity:  c701f1sq02ib0
                Identity:  192.168.8.2

ctsvhbal: In order for remote authentication to be successful,
at least one of the above identities for the local system must
appear in the trusted host list on the remote node where a service
application resides.
Ensure that at least one host name and one network address identity
from the above list appears in the trusted host list on any remote
systems that act as servers for applications executing on this
local system.
```

On node c701f1sq03:

```
c701f1sq03:/var/ct/cfg # /usr/sbin/rsct/bin/ctsvhbar c701f1sq02
          Host name or network address: c701f1sq02
          Fully qualified host name
               used for authentication: c701f1sq02
```

Action 3: investigate quorum problems

Take this action when configuration changes are rejected by the configuration resource manager due to insufficient quorum.

Symptom:

Configuration changes are rejected by the configuration resource manager because quorum cannot be established.

Diagnosis:

The following error indicates an insufficient quorum exists:

```
2632-072 The operation cannot be performed because a majority
of nodes or configuration daemons is not currently active in the
domain, or because the quorum of the domain is not currently
satisfied.
```

Configuration quorum is needed when the latest cluster configuration is to be determined. It ensures the integrity of the cluster definition. The configuration quorum of most peer domain operations follows the majority rule of ceil(n/2+1)

The configuration quorum majority rule is applied only to nodes defined as quorum nodes, which can be a subset of all cluster nodes. Non-quorum nodes do not figure into the majority rule. See the *Administering RSCT* guide for a precise description of the quorum nodes. Quorum and non-quorum nodes are listed with the **lsrpnode** command's **-Q** option. For complete syntax information on the **lsrpnode** command, see its man page in the *Technical Reference: RSCT for AIX* or the *Technical Reference: RSCT for Multiplatforms* guides.

The quorum rules that are applied to the peer domain operations are summarized below.

1. All the operations that may change the cluster definition follow the majority rule. For example, adding or removing nodes, adding, changing or removing communication groups, RSCT parameters or tiebreaker resources. However, there are two exception cases for the **rmrpnode** command.
   - Nodes may also be removed if exactly half of the nodes are online (in a tie situation) and if the configuration can be successfully removed from at least one of the offline nodes.
   - An **-f** option to override the majority rule and forcefully remove the node. Although this option is applicable for clusters of all sizes, it is especially useful for 2-node clusters.
2. By default, the quorum rule for the **startrpdomain** command is ceil(n/2). But the rule can be overridden by specifying the **-A** (all nodes) option or the **-L** (local node) option on the **startrpdomain** command.
   - Quorum (default): Each node to be started must be connected to a subcluster of nodes of which at least *ceil*(*n*/2) nodes have a cluster definition and *n* is the size of the most recent cluster definition in that subcluster.
   - **-A** option: All nodes option, where all the nodes defined in the peer domain must be contacted to locate the latest configuration in the peer domain. The all nodes option is useful if the quorum has been overridden by a previous **rmrpnode** command and it is not certain which node or nodes have the latest configuration.
   - -L option: The local node option, the configuration on the node where the **startrpdomain** command is executed is used to bring the peer domain online.
3. For all other operations (for example, **mkrpdomain**, **rmrpdomain**, **stoprpdomain**, **startrpnode**, **stoprpnode**) quorum rules are not applied.

Table 15 lists the configuration quorum rule for each cluster operation.

Table 15. Configuration quorum rules

Configuration resource manager command	Configuration quorum rule
mkrpdomain	No quorum rules are applied to this operation.
startrpdomain	Three online criteria options: 1. Quorum (default): *ceil*(*n*/2). 2. **-A** option: All nodes. 3. **-L** option: Local node configuration.
stoprpdomain	No quorum rules are applied to this operation.
rmrpdomain	No quorum rules are applied to this operation.
addrpnode	Majority
rmrpnode	Majority except for the following two cases: • Nodes may also be removed if exactly half of the nodes are online (tie) and if the configuration can be successfully removed from at least one of the offline nodes • Use the **-f** option to override the majority rule and forcefully remove a node. Useful for 2 node cluster.
startrpnode	No quorum rules are applied to this operation.
stoprpnode	No quorum rules are applied to this operation.
mkcomg	Majority
rmcomg	Majority
chcomg	Majority
mkrsrc / **chrsrc** / **rmrsrc** that may change cluster definition (For example, **chrsrc -c IBM.RSCTParameters**, **mkrsrc /chrsrc/rmrsrc IBM.TieBreaker**, **chrsrc -c IBm.PeerNode**)	Majority

Recovery procedure:

Ensure that the configurational quorum, as described in the diagnosis, exists.

Action 4: investigate duplicate IP address problems

Take this action when the configuration resource manager reports a duplicate IP address error.

Symptom:

The configuration resource manager reports an error when duplicate IP addresses are encountered.

Diagnosis:

The configuration resource manager checks for duplicate IP addresses when attempting to:

- create a peer domain with one or more nodes (using the **mkrpdomain** command);
- add nodes to an existing peer domain (using the **addrpnode** command)

If the configuration resource manager finds a duplicate IP address on the nodes to be added to a peer domain:

1. The configuration resource manager reports an error with the duplicate IP address and the node(s) with the duplicate IP address.
2. If the **-c** option is not specified, the operation fails entirely. For the **mkrpdomain** command, the domain will not be created. For the **addrpnode** command, none of the new nodes will be added.
3. If the **-c** option is specified, the operation will continue even when duplicate IP addresses or other errors are encountered. A node will be added as long as it has at least one valid IP address that is unique within the set of nodes to be added to the domain.

Recovery procedure:

- Correct the network configurations on the nodes to have unique IP addresses and retry the operation. The configuration resource manager will automatically harvest the modified configuration and will proceed with the operation using the corrected IP addresses.
- If the network configuration is not correctable for some reason, resubmit the operation with the **-c** option.

Action 5: investigate node startup failure

Take this action when a peer node is unable to rejoin the cluster.

Symptom:

A peer node is unable to rejoin the cluster.

Diagnosis:

Check if the **rmcd** subsystem is up using the **lssrc -a** command.

```
lssrc -a | grep ctrmc
```

If the **rmcd** subsystem is down, the**lssrc** command will return the following output:

```
ctrmc             rsct                          inoperative
```

If the **rmcd** subsystem is not up, it is possible that a service is probably using the reserved RMC port number 657. To determine if another service has taken the reserved port number 657:

1. Check the **errpt** file (on AIX), **/var/log/messages** (on Linux), **/var/adm/log/messages** (on Windows), or **/var/adm/messages** (on Solaris). A typical error record in **/var/log/messages** would be:

   ```
   Dec  1 15:22:40 elmo RMCdaemon[4494]:
              (Recorded using libct_ffdc.a cv 2)
   :::Error ID:822....EAumz.zmx0MRa47...................
   :::Reference ID:
   :::Template ID: 0:::Details File:
   :::Location: RSCT,rmcd_pci.c,1.46,393
   :::RMCD_2610_101_ER Internal error. Error data 1 ffffffff
               Error data 2
      000003f3 Error data 3 rmc
   ```

   The key indicator is the text starting with *RMCD_2610_101_ER*

2. Check **/var/ct/IW/log/mc/default** file. If a service is using the reserved RMC port number 657, a typical entry in the **/var/ct/IW/log/mc/default** file would be:

   ```
   ../../../../../src/rsct/rmc/mcdaemon/rmcd.c/00339/1.43
   2610-223 Cannot bind to the port specified by service name rmc,
   using the udp protocol.
   ```

   The problem often happens when a service obtains port 657, which is reserved for RMC. If such a service gets started prior to RMC, then RMC fails. In case of a peer domain, RMC does not bind to port 657 until after it has joined its peer group. At boot time, there is a possibility of a service binding to port 657 first.

Recovery procedure:

Once the problem is detected, the process using the **rmcd** port needs to be stopped and **rmcd** needs to be restarted. Stopping the problem process frees up the RMC port. The easiest way find the process using the port is by using the **lsof** tool. The **lsof** utility is available on most Linux and Solaris distributions. For AIX users, it is included on the AIX Toolbox for Linux Applications CD. If the **lsof** tool is not present on the system, you can use the **rmsock** command for the same purpose.

The following services are known to exhibit the problem: **biod**, **rpc.statd**, **xntpd**, **rpc.mountd**, **ypbind**.

Issue the following command:

```
netstat -an | grep 657
```

If a service is using port number 657, the following output is displayed:

```
udp4   19512      0  *.657                  *.*
```

Issue the **lsof** command to discover the service using the port 657. To circumvent this problem, stop the service using the port 657, and restart RMC. For example, assume that **rpc.statd** is the service that has obtained the port 657. This particular service is started by the **nfslock** script in **/etc/init.d** . The following commands are issued to free up the reserved port

1. `/etc/init.d/nfslock stop`
2. `startsrc -s ctrmc`
3. `/etc/init.d/nfslock start`

An alternative in case **lsof** is not present is to use **rmsock**.

Attention: The **rmsock** command actually removes a socket that does not have a file descriptor. See the online man page for the **rmsock** command for more information.

For example:

1. Issue the following command:

   ```
   netstat -Aan | grep 657
   ```

   If a service is using port number 657, the following output is displayed:

   ```
   f10000f000127358 tcp4    0    0  *.657      *.*     LISTEN
   f10000f000064600 udp4    0    0  *.657      *.*
   ```

2. Issue the **rmsock** command:

   ```
   rmsock f10000f000127358 tcpcb
   ```

   If port 657 is being used by the RMC daemon, the following output is displayed:

   ```
   The socket 0x127000 is being held by proccess 483448 (rmcd).
   ```

Action 6: respond to cluster partitioning

Take this action when a peer domain has been partitioned into two domains.

Symptom:

The peer domain has been partitioned into two sub-domains. Both partitions are running, but neither give you a complete cluster view.

Diagnosis:

Issue the **lsrpnode** command on nodes in each partition.

```
lsrpnode
```

If domain partitioning has occurred, the following example shows the output:

- On Node 1 in one partition:

```
Name   OpState RSCTVersion
Node1  Online  2.5.4.0
Node2  Offline 2.5.4.0
```

- On Node 2 in another partition:

```
Name   OpState RSCTVersion
Node1  Offline 2.5.4.0
Node2  Online  2.5.4.0
```

Typical reasons for such a view would be:

- Mis-configuration of network mask.
- The cluster definition of the peer domain may be out of sync among nodes of different partitions.

To check if the network mask is mis-configured, issue the following **ifconfig** command on all the nodes:

```
ifconfig -a
```

When examining the output from the **ifconfig** command, keep in mind that:

- Interfaces belonging to the same subnet must have the same subnet address and broadcast mask.
- On each given interface, the following rules must be observed:
  - Subnet masks must have the format 11...1100..00. (All ones, followed by all zeros.)
  - bcast_address = address | ~(subnet mask)

Table 16 shows example output of a misconfigured network mask:

Table 16. Example of **ifconfig** *output for a misconfigured network mask*

Node information	Output text
On Node 1:	en0: flags=5e080863,c0>UP **inet 9.43.241.84 netmask 0xffff9b00 broadcast** 9.43.245.255 tcp_sendspace 131072 tcp_recvspace 65536 lo0: flags=e08084b>UP **inet 127.0.0.1 netmask 0xff000000 broadcast 127.255.255.255** inet6 ::1/0 tcp_sendspace 65536 tcp_recvspace 65536
On Node 2:	en0: flags=5e080863,c0>UP **inet 9.43.241.85 netmask 0xffffff00 broadcast 9.43.241.255** tcp_sendspace 131072 tcp_recvspace 65536 lo0: flags=e08084b>UP **inet 127.0.0.1 netmask 0xff000000 broadcast 127.255.255.255** inet6 ::1/0 tcp_sendspace 65536 tcp_recvspace 65536

Recovery procedure:

As shown in the preceding sample output, because the network mask is mis-configured on Node1, the broadcast address for 9.43.241 is incorrect. The correct broadcast address for 9.43.241 should be 9.43.241.255. Changing the network mask to 0xffffff00 will correct the problem. Broadcast addresses apply to IPv4 networks only.

If however, the network setup is correct, it is possible that the cluster definition is out of sync among the nodes. In this case, select one partition and issue the **stoprpdomain** command on one of its nodes. All nodes in the partition will be taken offline.

Go to an online node in the other partition, and execute a command that may cause change(s) on the cluster configuration. For example:

```
chcomg -s 5 communication_group
```

The cluster configuration will be rebuilt with a newer version number for the change, and then populated to all online nodes in the partition via group protocol.

Issue the **startrpdomain** command on an online node. Since the cluster configuration in the online partition has a newer version number than the offline partition, the cluster configuration of the online partition will replace the older version in the offline partition. Now, both partitions have the same version of the cluster configuration.

In case the cluster view continues to remain inconsistent, see "Diagnosing problems with Topology Services" on page 133 and "Diagnosing problems with Group Services" on page 192.

Action 7: add an interface to the heartbeat ring of the configuration resource manager

Take this action when interfaces on a node or set of nodes are not part of the heartbeat ring of the configuration resource manager.

Symptom:

Interfaces on a node or set of nodes are not part of the heartbeat ring of the configuration resource manager.

Diagnosis:

Issue the **lssrsrc** command to check the HeartBeatActive flag. If it is 0, the interface is not in the heartbeat ring.

```
lsrsrc -t IBM.NetworkInterface Name NodeNameList IPAddress
    CommGroup HeartbeatActive
```

The following output is displayed. In this example, the interface is not in the heartbeat ring.

```
Resource Persistent and Dynamic Attributes for
     IBM.NetworkInterface Name
NodeNameList    IPAddress     CommGroup HeartbeatActive
"en0" {"k1n11e.ibm.com"} "192.224.0.1" "CG3"     0 ← set to zero
```

Recovery procedure:

With the CT_MANAGEMENT_SCOPE environment variable set to 2, use the **chrsrc** command to reset the HeartBeatActive flag to 1.

```
export CT_MANAGEMENT_SCOPE=2;
chrsrc -s'Name=="en0" && NodeNameList=={"k1n11e"}'
     IBM.NetworkInterface
HeartbeatActive = 1
```

Action 8: investigate failure to add a node

Multiple symptoms suggest this action.

Take this action when you encounter any of the following symptoms:

Symptom 1: the addrpnode command fails when the domain is offline:

This symptom suggests the following diagnosis and recovery.

Diagnosis:

The **addrpnode** command fails with the following message:

```
addrpnode: 2602-021 There are no nodes in the peer domain or
an online peer domain does not exist.
```

This message indicates that either the domain is offline or that the node on which you are executing the **addrpnode** command is not part of the peer domain. To determine if the peer domain is offline, issue the **lsrpdomain** command.

```
lsrpdomain
```

The following output is displayed. In this example, the domain is offline.

```
Name       OpState RSCTActiveVersion MixedVersions TSPort GSPort
IBMCluster Offline  3.1.4.0           No            12347  12348
```

Recovery procedure:

If the domain is Offline, bring the domain Online and issue the **addrpnode** command again. An alternative will be to invoke the command on a node that is online in the peer domain.

Symptom 2: unable to add a node to an existing online peer domain:

This symptom suggests the following diagnosis and recovery.

Diagnosis:

The **addrpnode** command fails with the following message:

```
2632-077 The following problems were detected while adding nodes
to the domain. As a result, no nodes will be added to the domain.
davrosp03: 2632-071 The node cannot be added to the domain because
the version of RSCT on the node is earlier than the version that
is active in the domain.
```

Check the version of the nodes in the current domain and the RSCT version of the node that is being added. Adding a node with an older version of RSCT to an online peer domain that has a more recent active RSCT version is not allowed.

On a node already in the domain, issue the following command:

```
/usr/sbin/rsct/install/bin/ctversion
```

The following output is displayed:

```
rjop1135a 3.1.4.0
```

The preceding output shows that the existing RSCT peer domain has an **RSCTActiveVersion** value of 3.1.4.0.

On the node indicated in the error message, check which RSCT version is installed using the **ctversion** command.

```
/usr/sbin/rsct/install/bin/ctversion
```

Output shows whether the RSCT version is at the same level as the rest of the cluster.

```
rhay002a 2.5.5.0
```

The preceding output shows that an attempt to add an older node (version 2.5.5.0) was being made.

Recovery procedure:

Either the peer domain must be created with the older node first and then migrate to the newer version or node to be added should first be migrated with the new code and then added to the Online domain.

Symptom 3: the rmrpnode command needs to be run to clear an addrpnode command failure:

This symptom suggests the following diagnosis and recovery.

Diagnosis:

The **addrpnode** command returns an error similar to the following:

```
2632-074 The following problems were detected while successfully
adding nodes to the domain. Nodes that could not be harvested
were not added to the domain.
ml0f1rp01: 2645-000 Operation failed due to error 0 returned from
   rm -rf.
```

Subsequently, issuing the **addrpnode** command to add the nodes indicated in the preceding error results in the following error:

```
addrpnode: 2602-162 ml0f1rp01 is already defined in the online
peer domain.
```

Recovery procedure:

After the failure, remove the node using the **rmrpnode** command, and then invoke the **addrpnode** command again.

Symptom 4: the addrpnode command fails when an alias host name is used:

This symptom suggests the following diagnosis and recovery.

Diagnosis:

If the **addrpnode** command fails when an alias host name is used, go to the Group Leader node (as described in "Finding the group leader (GL) node for a specific group" on page 205 and issue the host command to see if the alias host name can be resolved. For example, if the alias host name is **colt1**, enter:

```
host colt1
```

Output similar to the following indicates that the alias host name could not be resolved.

```
Host colt1. not found: 3(NXDOMAIN).
```

Recovery procedure:

Make sure the host name can be resolved on all nodes.

Action 9: investigate accessibility of peer domain nodes

Take this action when peer domain operations fail and errors indicate there was a problem in establishing a session with the RMC Subsystem.

Symptom:

Error messages indicate that peer domain operations have failed because of problems in establishing session with the RMC subsystem.

Diagnosis:

The problem could be the result of one or more peer domain nodes being inaccessible. To determine whether the peer domain nodes are accessible.

1. Obtain a file that lists the peer domain nodes. This can be a working collective file, or can be obtained by issuing the following **lsrpnode** command on a node that is online in the peer domain. In this example, the **lsrpnode** command output is redirected to the file **/tmp/nodes**.

   ```
   lsrpnode -xd | cut -f1 -d: > /tmp/nodes
   ```

2. On a host machine that you expect to have connectivity with the nodes in the peer domain, issue the following shell commands to check the connectivity of each node in the file. In this example, the file listing the peer domain nodes is **/tmp/nodes**.

   ```
   for node in 'cat /tmp/nodes'
   do
     ping -c 1 -w 2 $node
   done
   ```

   The preceding shell commands will ping each node in the list and wait 2 seconds for a reply. Nodes that are unresponsive will show a 100 percent packet loss rate, while nodes that do respond will show a 0 percent packet loss rate.

Recovery procedure:

If nodes are unresponsive to the **ping** command, check your basic networking software and hardware configurations for errors. If all nodes are responsive to the **ping** command, then the problem may be that the RMC subsystem is unresponsive. See the instructions in "Action 10: investigate responsiveness of RMC subsystem."

Action 10: investigate responsiveness of RMC subsystem

Take this action when peer domain operations fail and errors indicate there was a problem in establishing a session with the RMC Subsystem.

Symptom:

Error messages indicate that peer domain operations have failed because of problems in establishing session with the RMC subsystem.

Diagnosis:

The problem could mean that the RMC subsystem is unresponsive. To determine whether the RMC subsystem is responsive:

1. Obtain a file that lists the peer domain nodes. This can be a working collective file, or can be obtained by issuing the following **lsrpnode** command on a node that is online in the peer domain. In this example, the **lsrpnode** command output is redirected to the file **/tmp/nodes**.

   ```
   lsrpnode -xd | cut -f1 -d: > /tmp/nodes
   ```

2. On a host with RSCT installed, issue the following shell commands:

   ```
   for node in `cat /tmp/nodes`
   do
     echo contacting RMC on $node
     CT_CONTACT=$node lsrsrc done
   ```

   The preceding shell commands will run the **lsrsrc** command against all RMC daemons on all nodes. This should return a list of the resource classes that RMC supports on each node.

Recovery procedure:

If the **lsrsrc** command on any node does not return output from the RMC subsystem, then contact the IBM Support Center for assistance.

Action 11: check the root file system

Common operational problems in the configuration resource manager occur when the root file system runs out of space. Take this action when the configuration resource manager is inoperative or a node cannot be brought online in the peer domain.

Symptom:

The configuration resource manager is inoperative or a node cannot be brought online in the peer domain.

Diagnosis:

There is likely insufficient space in the root file system if the system returns the following error:

```
2523-638 Cannot set port number into /etc/services
```

If the system returns the preceding error, issue the **df** command to check the amount of free space available in the root file system.

Recovery procedure:

Increase the size of the root file system.

Action 12: check the /tmp file system

Common operational problems in the configuration resource manager occur when the **/tmp** file system runs out of space. Take this action when configuration resource manager commands, RMC commands, or RMC client operations fail.

Symptom:

Configuration resource manager commands, RMC commands, or RMC client operations fail.

Diagnosis:

There is likely insufficient space in the **/tmp** file system if the system returns the following error:

```
2610-637 The security library routine sec_setup_socket()
returned error 10:
"2650-008 A socket operation failed.".
```

If the system returns the preceding error, issue the **df** command to check the amount of free space available in the **/tmp** file system.

Recovery procedure:

Increase the size of the **/tmp** file system.

Action 13: restore nodes to a peer domain with their original node numbers

There are situations that will require you to re-add a node or set of nodes to a peer domain using the original node numbers. Take this action to restore nodes to a peer domain with their original node numbers.

There are situations that will require you to re-add a node or set of nodes to a peer domain using the original node numbers. For example, you will need to do this if:

- A node or a set of nodes is re-installed using an image from another node. On an AIX node, it is common to use the **mksysb** command to create an image of an existing AIX installation to either restore the entire system or to install a new node using an image from an existing node for subsequent install using NIM.
- Nodes are erroneously removed from the peer domain cluster.

As described in *Administering RSCT* guide, we recommend that, once a peer domain is created and the peer nodes are online, you save a record of the node to node number mapping. To save a record of the node to node number mapping, issue the following command from a node that is online in the peer domain.

```
lsrsrc -x -D' ' IBM.PeerNode Name NodeList | sed
      's/{/ /g' | sed 's/}/ /g'|sed 's/"//g' >
      rpdNodeMap.save
```

The **lsrsrc** command output will be piped into the **rpdNodeMap.save** file. The contents of this file will be similar to the following:

```
c18n01  1
c18n02  2
c17n06  3
c17n07  4
```

If you have saved a record of the original node to node number mapping, you will be able to restore a node or set or nodes to the peer domain with the required node number(s). To do this:

1. Create a file that lists the node name to node number mappings specifying the new/required node numbers.

   For example, suppose nodes **c17n06** and **c17n07** were removed from the domain and two new nodes, say **nodeX** and **nodeY** were then added. If there was a need for the two original nodes, **c17n06** and **c17n07** to now be re-added to the domain and reassigned their original node numbers 3 and 4, create a file with the required node name to node number mapping. For this example, we create a file named **newIDMap.in** that contains the following two entries:

   ```
   c17n07  4
   c17n06  3
   ```

2. Locate any nodes that may already be using the required node numbers. *This crucial step is needed to make sure that you only add nodes with unique node numbers.* To do this, save a record of the node to node number mapping, by issuing the following **lsrsrc** command from a node that is online in the peer domain.

   ```
    lsrsrc -x -D' ' IBM.PeerNode Name NodeList | sed
      's/{/ /g' | sed 's/}/ /g'|sed 's/"//g' > \
      rpdNodeMap.current
   ```

   The **lsrsrc** command output will be piped into the *rpdNodeMap.current* file. In our example, the contents of this file are:

   ```
   nodex   3
   nodeY   4
   c18n01  1
   c18n02  2
   ```

   Search the nodes in this file for any that are using the required new numbers. In our example, **nodeX** and **nodeY** now have the node numbers originally used by **c17n06** and **c17n07**.

3. Create an input file for the **addrpnode** command containing node name/node number pairs that identify the nodes you want to re-add to the peer domain. If the preceding step showed that no other nodes were using the original node numbers, the file need only contain the node name to node number mappings for the nodes you are re-adding. If we had not found any nodes that were using the node numbers 3 and 4, our file would only require the following mappings:

   ```
   c17n06  3
   c17n07  4
   ```

   If, however, the preceding step did show that other nodes were currently using the required node numbers, then the file will also need to reassign these nodes to new numbers. Since, in our example the preceding step showed that **nodeX** and **nodeY** now have the node numbers originally used by

c17n06 and **c17n07**, we will have to assign **nodeX** and **nodeY** some other unique node numbers. Since, in our example, there are no nodes in the peer domain with node number 5 and 6, we will assign these numbers to the nodes. In this case our file will have the following mappings.

```
c17n06   3
c17n07   4
nodeX    5
nodeY    6
```

4. If the input file you created in the preceding step will be swapping nodes (reassigning existing nodes in the domain to new node numbers so that the nodes you are re-adding to the domain can be assigned their original node numbers), take any nodes that will be reassigned a node number offline and remove the nodes from the peer domain. To take the nodes offline, run the **stoprpnode** command on the nodes. To remove the nodes from the peer domain, use the **rmrpnode** command. In our example, we will need to run the **stoprpnode** on **nodeX** and **nodeY**. Once the nodes are offline, we will use the **rmrpnode** command to remove them from the peer domain. Wait at least 2 minutes (120 seconds) after issuing the **rmrpnode** command before issuing the subsequent **addrpnode** command as described in Step 5.)

   Note: If an image of another node was installed on a node, it may be necessary to run the following **recfgct** command on the installed node to cleanup the configuration information of the node. On an AIX node, the **recfgct** command should have been run automatically when the **mksysb** image was restored on the node.

   ```
   /usr/sbin/rsct/install/bin/recfgct
   ```

5. Re-add the nodes to the peer domain by issuing the **addrpnode** command from the peer domain's Group Leader node. To identify the Group Leader node, see “Finding the group leader (GL) node for a specific group” on page 205. Use the **addrpnode** command's **-f** option to specify the input file you created containing the node name to node number mappings. In our example, we named our input file **NodeFile.in**.

   ```
   addrpnode -f NodeFile.in
   ```

6. Use the **lsrsrc** command to ensure that the node have been added to the domain and the node numbers are as desired.

   ```
   lsrsrc -x -D' ' IBM.PeerNode Name NodeList | sed
          's/{/ /g' | sed 's/}/ /g'|sed 's/"//g' > \
          newNodeMap.save
   ```

7. Use the **startrpnode** command to bring the new nodes online.

Action 14: change a node's public or private key

A node's public key is usually not expected to change. Take this action, however, when such a change becomes necessary.

If, however, a situation arises that requires you to change the public or the private key of a node already defined in a peer domain, you should follow these steps.

1. Take the node offline if it is online.
2. Use the **rmrpnode** command to remove the node from all the peer domains in which it is defined.
3. Use the **ctskeygen** command to generate new public and private keys.
4. Execute the **preprpnode** command on the node, specifying all the other cluster nodes.
5. On a node that is online in the peer domain, execute the **addrpnode** command to add the new node to the online peer domain. The new keys will be distributed to other nodes in the domain during the execution of the **addrpnode** command provided the automatic key exchange option is not disabled.

Related tasks:

"Diagnosing problems with cluster security services" on page 70
Cluster security services software components grant or deny access to resources.

Action 15: respond to a changed or missing public key

If a node's public key is changed unintentionally or is missing, problems may arise. Take this action to address problems arising from a changed or missing public encryption key.

For example, if a node's public key changed unintentionally, nodes may fail to authenticate when cluster security services' UNIX Host Based Authentication is used. Specifically:

- If the public and private key files were accidentally removed from a node and the cluster security services' daemon (**ctcasd**) is restarted, **ctcasd** will create new keys for the node. These new keys will not match the keys stored on the other nodes defined in the peer domain.
- If the public key file is missing but the private key file is detected, **ctcasd** will terminate.

If a node's public key is changed unintentionally or is missing, you should:

1. Remove the node from the peer domain.
2. Ensure that the node's security has not been compromised.
3. Generate the key if missing or regenerate the key if needed.
4. Add the node back to the peer domain.

During the node addition process, the new key will be exchanged with other nodes in the peer domain if the automatic key exchange option is not disabled.

Related tasks:

"Diagnosing problems with cluster security services" on page 70
Cluster security services software components grant or deny access to resources.

Action 16: force a peer domain offline

Take this action when a peer node remains in the pending online state indefinitely.

Symptom:

A peer domain remains in the pending online state indefinitely.

Diagnosis:

There is likely a failure in which, as soon as the configuration resource manager is started and the online process is initiated, the process fails and maintenance on its configuration with respect to the peer domain cannot take place to address the failure. The **forcerpoffline** command can be used to modify the **/var/ct/cfg/current_cluster** and **/var/ct/cfg/default_cluster** files, recycle the configuration resource manager and the RMC subsystem, and allow the node to come up in IW mode.

Recovery procedure:

If the cause of the failure keeping the node in the pending online state is unknown, capture a ctsnap. Then run the **forcerpoffline** command on the affected node, as follows:

```
forcerpoffline domain_name
```

Action 17: synchronize the time-of-day clocks in the peer domain

If you have enabled the use of a shared secret key in the peer domain, you must ensure that the time-of-day clocks on all nodes in the peer domain are synchronized. Take this action when a peer node remains in the pending online state indefinitely.

Symptom:

After enabling the use of a shared secret key in the peer domain, the domain remains in the pending online state after you issue the **startrpdomain** command.

Diagnosis:

The time-of-day clocks on all nodes within the peer domain must be synchronized to within a

reasonable tolerance of each other – typically, only a few seconds. Too great a variance among the clock settings on the nodes in the peer domain can cause control messages to time out. This will prevent the proper operation of shared secret key authentication.

Recovery procedure:

Use the **date** command on each of the nodes in the peer domain to verify that their time-of-day clocks are synchronized and reset the time on any nodes whose current time setting differs from the other nodes by more than a few seconds. After the clocks have been synchronized on all nodes, try to start the peer domain again.

To prevent a reoccurrence of the problem, consider using a network time synchronization protocol, such as NTP. This will allow nodes in the peer domain can maintain a common agreement on the current time of day.

Action 18: respond to cluster partition when adding a node

When adding a new node to a running peer domain in which the peer domain has been partitioned into two sub-domains, times may be out-of-sync among the nodes. You must synchronize the times among all the nodes.

Symptom:

The peer domain has been partitioned into two sub-domains when adding a new node to a running peer domain. Both partitions are running, but neither gives you a complete cluster view.

Diagnosis:

Issue the **lsrpnode** command on nodes in each partition.

```
lsrpnode
```

If domain partitioning has occurred, output will be similar to the following examples:

- On Node 1 in one partition:

```
Name    OpState RSCTVersion
Node1   Online  3.1.4.0
Node2   Offline 3.1.4.0
```

- On Node 2 in another partition:

```
Name    OpState RSCTVersion
Node1   Offline 3.1.4.0
Node2   Online  3.1.4.0
```

The typical reason for such a view would be:

- times are out-of-sync among the nodes

Check the time/date stamp on all the nodes.

Recovery procedure:

Synchronize the times among all the nodes. To do so, issue the **startrpnode** command on the new node. Then verify that the peer domain exists and that all nodes are seen in the cluster view by issuing the **lsrpnode** command on all nodes. Their OpState should all be Online.

On Node 1:

```
Name  OpState RSCTVersion
Node1 Online 3.1.4.0
Node2 Online 3.1.4.0
```

On Node 2:

```
Name  OpState RSCTVersion
Node1 Online 3.1.4.0
Node2 Online 3.1.4.0
```

Action 19: investigate the stoprpdomain command failure because of critical resources being active on nodes

Take this action when the **stoprpdomain** command fails and errors indicate it cannot determine if any resources are online.

Symptom:

The **stoprpdomain** command fails with errors indicating that it cannot determine whether any resources are online or not.

Diagnosis:

Note: You can decide whether to stop the domain when a critical resource is online. In many cases, you might decide not to stop the domain when a critical resource is online.
The **stoprpdomain** command fails with the following message:

```
# stoprpdomain rpd1
2632-110 The operation was rejected by one or more nodes,
probably because one or more resources are online
or there was an error encountered in determining
if any resources are online.
node01: 2668-014 The ConfigCoordination action to
class IBM.Disk of request 0 is rejected
because one or more resources are currently active.
```

This problem could mean that the **stoprpdomain** command fails because critical resources are active on some nodes. To identify the nodes and to stop the active resources on the node, complete the following steps:

1. Identify the nodes owning the critical resource by issuing the following command:

   ```
   # CT_MANAGEMENT_COPE=2 lsrsrc IBM.PeerNode CritRsrcActive NodeNameList
   ```

   The following output is displayed:

   ```
   Resource Persistent and Dynamic Attributes for IBM.PeerNode
   resource 1:
     CritRsrcActive = 1
     NodeNameList = {"node01.ppd.pok.ibm.com"}
   ```

2. Identify the resource class that owns the critical resource on the node identified in step 1. The **stoprpdomain** command operation was rejected because one of the nodes (`node01.ppd.pok.ibm.com`, in this case) had a critical resource running.

   To identify the Owner resource class for the critical resource, issue the following command:

   ```
   # lssrc -ls IBM.ConfigRM|grep CritRsrc
   ```

   The following output is displayed:

   ```
     CritRsrcActive:1
     CritRsrcOwner:IBM.AgFileSystem
   ```

3. Take the necessary action to stop the resource. The output in step 2 shows that one of the `IBM.AgFileSystem` resources is active.

   The error initially pointed to the `IBM.Disk` class in the previous error message because a disk cannot be stopped until the file systems on the disk are offline.

4. Identify the resource among all of the `IBM.AgFileSystem` resources, and stop it by using the **stoprsrc** command.

   You can now stop the domain by using the **stoprpdomain** command.

Recovery procedure:

If any other critical resources are online, repeat steps 1 - 4, and take the necessary action.

Diagnosing problems with cluster security services

Cluster security services software components grant or deny access to resources.

Related tasks:

"Action 14: change a node's public or private key" on page 66
A node's public key is usually not expected to change. Take this action, however, when such a change becomes necessary.

"Action 15: respond to a changed or missing public key" on page 67
If a node's public key is changed unintentionally or is missing, problems may arise. Take this action to address problems arising from a changed or missing public encryption key.

Requisite function

This is a list of the software directly used by the cluster security services component of RSCT. Problems within the requisite software may manifest themselves as error symptoms in the cluster security services.

If you perform all the diagnostic procedures and error responses listed in this topic, and still have problems with the cluster security services component of RSCT, you should consider these components as possible sources of the error. The following list presents components in the order that they are most likely to introduce an error, from the most likely to the least likely.

- TCP/IP
- UDP/IP
- UNIX Domain Sockets
- **/var** file system space, specifically the **/var/ct/cfg** directory
- **/usr/sbin/rsct** directory availability
- First Failure Data Capture Library (libct_ffdc)
- Cluster Utilities Library (libct_ct)
- System Resource Controller (SRC)

Note: RSCT cluster security services are disabled when running on a Windows platform. On the Windows platform, authentication is verified by the standard Windows login process and authorization is verified by the standard Windows file permissions on the RSCT commands. A logged in Windows user must have `execute` permission on the appropriate files in order to run RSCT commands.

Error information

The Host Based Authentication service daemon **ctcasd** records failure information. On AIX system platforms, the RSCT component subsystems write this information to the AIX error log. On Linux, and Solaris system platforms, it writes the information to the respective system log. For compatibility, the RSCT component subsystems record any **ctcasd** failures to the system log on AIX nodes, if the system log is active.

Error logs and templates

This topic lists Cluster Security Services error log labels and error log types with their associated explanations.

For more information on the AIX error log, or Linux or Solaris system logs, see "Accessing logged errors" on page 2.

Table 17 lists the messages that can be recorded by the **ctcasd** daemon. On AIX system platforms, the message is identified by an error log label. On Linux and Solaris system platforms, the entire message will appear in the respective system log.

Table 17. Error log templates for Cluster Security Services

AIX Error Log Label	AIX Error Log Type	Linux System Log Label	Linux System Log Selector Value	Description
ARG_INT_ER	PERM	ctcasd Daemon Internal Failure, Terminating: Failing routine *name*, Positional parameter in error *position*, Value *parameter_value*, Caller of failing routine *name*	daemon.err	**Explanation:** An unexpected internal failure condition was detected by the **ctcasd** daemon. The daemon has shut itself down. Authentication attempts using the Host Based Authentication mechanism will not be successful on this node. **Details:** Note the information recorded in this entry and contact the Cluster Security software service provider.
CASD_INT_ER	PERM	ctcasd Daemon Internal Failure, Terminating: Failing routine *name*, Failure code from routine *error_code* , Caller of failing routine *name*	daemon.err	**Explanation:** An unexpected internal failure condition was detected by the **ctcasd** daemon. The daemon has shut itself down. Authentication attempts using the Host Based Authentication mechanism will not be successful on this node. **Details:** Note the information recorded in this entry and contact the Cluster Security software service provider.
CASD_DN_IN	INFO	ctcasd Daemon Stopped	daemon.info	**Explanation:** The **ctcasd** daemon has been shut down on the node. Authentication attempts using the Host Based Authentication mechanism will no longer be successful until the daemon is restarted. This is a normal operational message. **Details:** The **ctcasd** daemon may have been forcibly shut down.
CASD_ENV_VAR_ER	INFO	ctcasd Daemon trace Environment Variable has incorrect value. Trace Settings: CT_TR_TRACE=**value**, CT_TR_TRACELEVELS=**value**, CT_TR_TRACEFILE=**value**, CT_TR_SIZE=**value**	daemon.info	**Explanation:** The **ctcasd** daemon detected that it was being invoked in an incorrect environment or configuration. Authentication attempts using the Host Based Authentication mechanism (HBA) or the Enhanced Host Based Authentication mechanism (HBA2) will not be successful on this node. **Details:** When the **ctcasd** daemon is started, it checks the values of environment variables. If the environment variables are set to unsupported values, the daemon shuts itself down. The daemon will not start until the environment is corrected. The Detail Data section of this record contains the names of the environment variables and the values that were detected by the daemon. The following environment variables may trigger this condition: • **CT_TR_TRACE** must be set to the values "on" or "off" only. Mixed case may be used when specifying these values. • **CT_TR_SIZE** must not be set to an empty string. • **CT_TR_FILENAME** must not be set to an empty string. • **CT_TR_TRACELEVELS** must not be set to an empty string. Verify that none of these environment variables are incorrectly set in the **/etc/environment** file or explicitly set by the System Resource Controller. Verify that the **ctcasd** daemon was not started from the command line.
CASD_TRACE_ERR	INFO	ctcasd Daemon Trace Error	daemon.info	**Explanation:** The **ctcasd** daemon was unable to start the trace facility. **Details:** Examine the **ctcasd.cfg** file and the CT_TR_TRACE, CT_TR_SIZE, CT_TR_TRACE_LEVELS, and CT_TR_FILENAME environment variable settings to determine why the trace could not be enabled. See "Tracing the ctcasd daemon" on page 82. For information on configuring the **ctcasd** daemon on a node, see the *Administering RSCT* guide.
CASD_UP_IN	INFO	ctcasd Daemon Started	daemon.info	**Explanation:** The **ctcasd** daemon has been started on the node. Authentication is now possible, using the Host Based Authentication mechanism. This is a normal operational message. **Details:** The **ctcasd** daemon is started automatically when first contacted for authentication.
CTS_DCFG_ER	PERM	ctcasd Demon Initialization Failure, error in configuration file - file does not exist, cannot be accessed, or the contents of the file are incorrect. Verify that the file exists and the contents are correct.	daemon.err	**Explanation:** The **ctcasd** daemon received invalid startup options, or was unable to correctly process its configuration information. The daemon on this node has shut itself down. Authentication attempts using the Host Based Authentication mechanism will not be successful on this node. **Details:** The **ctcasd** daemon is started upon demand when authentication is attempted using the Host Based Authentication mechanism. This daemon can be started manually, using the **startsrc -s ctcasd** command. An attempt to start the daemon directly from the command line can result in this failure. This failure can also result when the configuration information for the **ctcasd** daemon is missing, corrupted, or invalid. The default location for this data is the **/usr/sbin/rsct/cfg/ctcasd.cfg** file. The default configuration can be overridden by the file **/var/ct/cfg/ctcasd.cfg**. If this failure occurs, one of these files is missing, corrupted, or contains invalid information. The error log entry indicates the configuration file used by this instance of the daemon. If the daemon was correctly started, examine this file for problems.

Table 17. Error log templates for Cluster Security Services (continued)

AIX Error Log Label	AIX Error Log Type	Linux System Log Label	Linux System Log Selector Value	Description
CTS_ENV_ERR	PERM	ctcasd Initialization Failure, incorrect execution environment detected by routine *name* - cannot find or create socket directory *pathname*, or cannot change to working directory *pathname*, or cannot submit to System Resource Controller control.	daemon.err	**Explanation:** The - daemon detected that it was being invoked in an incorrect environment or configuration. Authentication attempts using the Host Based Authentication mechanism will not be successful on this node. **Details:** The - daemon attempts to change to a specific working directory, submit itself to System Resource Controller (SRC) control, and create a UNIX Domain Socket to interface with the cluster security services library. During the startup of the daemon, one of these efforts failed. The Detail Data section will list the intended working directory for the process and the socket file name that the daemon was to create. The daemon has shut itself down.
CTS_ISVR_ER	PERM	ctcasd Daemon Initialization Failure, cannot set up Internet Domain Socket server - *subroutine_name* returned *error_code*.	daemon.err	**Explanation:** The **ctcasd** daemon was unable to set up the service to handle requests via an Internet Domain Socket. Authentication attempts using the Host Based Authentication mechanism will not be successful on this node. **Details:** The **ctcasd** daemon interfaces with certain cluster security services library requests through an Internet Domain Socket. The daemon was unable to set up a service thread to handle these requests because of a failure condition detected with the Internet Domain Socket. The daemon is unable to set up a service thread for this socket as a result. The daemon has shut itself down
CTS_MEM_ERR	UNKN	ctcasd Daemon Failure, unable to allocate *size* bytes of memory in routine *name* - retry this operation at a later time, identify processes using large amounts of memory and consider terminating them.	daemon.err	**Explanation:** The **ctcasd** daemon was unable to dynamically allocate memory. Authentication attempts using the Host Based Authentication mechanism will not be successful on this node. **Details:** The daemon dynamically allocates memory to construct Host Based Authentication credentials and to authenticate these credentials. During one of these attempts, the daemon was unable to obtain dynamic memory. The internal routine that attempted to allocate this memory, and the amount of memory requested, are listed in the Detail Data section of this record. The daemon has shut itself down.
CTS_QUE_ER	PERM	ctcasd Daemon Failure, unable to allocate size bytes of memory for internal queue in routine name - retry this operation at a later time, identify processes using large amounts of memory and consider terminating them.	daemon.err	**Explanation:** The **ctcasd** daemon was unable to create an internal process thread queue for organizing and dispatching working threads. Authentication attempts using the Host Based Authentication mechanism will not be successful on this node. **Details:** This error log entry will provide internal diagnostic information on the cause of the failure. Make note of this information and contact the Cluster Security software service provider.
CTS_THRD_ER	PERM	ctcasd Daemon Initialization Failure, cannot create or detatch from thread in *subroutine_name -subroutine_name* return code *error_code*. The daemon may be reaching a per-process or system thread limit. Reduce thread limits in the ctcasd configuration file. Consider reducing thread usage by other processes.	daemon.err	**Explanation:** The **ctcasd** daemon detected an unexpected failure in the execution of one of its process threads. Authentication attempts using the Host Based Authentication mechanism will not be successful on this node. **Details:** This error log entry will provide internal diagnostic information on the cause of the failure. Make note of this information and contact the Cluster Security software service provider.
CTS_THRDI_ER	PERM	ctcasd Daemon Initialization Failure, thread initialization failure in *subroutine_name* - Contact the cluster software service provider and report this failure condition.	daemon.err	**Explanation:** The **ctcasd** daemon was unable to create and initialize process threads. Authentication attempts using the Host Based Authentication mechanism will not be successful on this node. **Details:** The files **/usr/sbin/rsct/cfg/ctcasd.cfg** (default) or **/var/ct/cfg/ctcasd.cfg** (override) provide configuration information to the **ctcasd** daemon, including the number of threads to create. The daemon encountered a failure while creating and initializing at least one thread. The number of available threads on the system may need to be increased, or the number of active processes and threads on the system may need to be decreased. Consult the error log entry for specific responses to take.
CTS_USVR_ER	PERM	ctcasd Daemon Initialization Failure, cannot set up UNIX Domain Socket server. Check permissions on the directory for file *filename*, and verify that this file is not being removed explicitly by another system user.	daemon.err	**Explanation:** The **ctcasd** daemon was unable to set up the service to handle requests via its UNIX Domain Socket. Authentication attempts using the Host Based Authentication mechanism will not be successful on this node. **Details:** The **ctcasd** daemon interfaces with the cluster security services library through a UNIX Domain Socket. This socket may have been removed, or permissions on the file or directory may have been altered. The name of the socket file is provided in the Detail Data section of this record. The daemon is unable to set up a service thread for this socket as a result. The daemon has shut itself down.
HID_MEM_ER	PERM	ctcasd Daemon Failure, Unable to create a host identifier in routine *name* - memory may not be available, retry request at a later time, identify processes using large amounts of memory and consider terminating them	daemon.err	**Explanation:** The **ctcasd** daemon was unable to allocate dynamic memory while creating the Host Based Authentication host identifier token for the local system. Authentication attempts using the Host Based Authentication mechanism will not be successful on this node. **Details:** To authenticate remote clients using Host Based Authentication, the local host must posses a Trusted Host List file, which associates known trusted host names to the node's associated public key value. The trusted host list file is created by the installation process, or by the **ctcasd** daemon when it is executed for the first time after installation. The initial Trusted Host List file is populated with the local node's names, IP addresses, and public key. This file is stored by default in **/var/ct/cfg/ct_has.thl**. The default path name can be overridden by the files **/usr/sbin/rsct/cfg/ctcasd.cfg** (default) or **/var/ct/cfg/ctcasd.cfg** (override).

Table 17. Error log templates for Cluster Security Services (continued)

AIX Error Log Label	AIX Error Log Type	Linux System Log Label	Linux System Log Selector Value	Description
I18N_MEM_ERR	PERM	ctcasd Daemon Failure, unable to construct internationalization control information in routine *name* - memory may be temporarily unavailable, or the process may be using a locale that does not support internationalization.	daemon.err	**Explanation:** The **ctcasd** daemon was unable to convert Host Based Authentication host identifier token data either to or from a locale independent format. Authentication attempts using the Host Based Authentication mechanism will not be successful on this node. **Details:** To authenticate remote clients using Host Based Authentication, the local host must posses a Trusted Host List file, which associates known trusted host names to the node's associated public key value. The trusted host list file is created by the installation process, or by the **ctcasd** daemon when it is executed for the first time after installation. The initial Trusted Host List file is populated with the local node's names, IP addresses, and public key. This file is stored by default in **/var/ct/cfg/ct_has.thl**. The default path name can be overridden by the files **/usr/sbin/rsct/cfg/ctcasd.cfg** (default) or **/var/ct/cfg/ctcasd.cfg** (override).
KEYF_ACC_ER	PERM	ctcasd Daemon cannot access key file *filename*, file may be removed or access to file or directory restricted - verify that file exists, recreate file if necessary, verify permissions on the file and directory	daemon.err	**Explanation:** The **ctcasd** daemon was unable to access the files containing either the local system's public or private key. Authentication attempts using the Host Based Authentication mechanism will not be successful on this node. **Details:** For Host Based Authentication to succeed, each host must possess both a private key, and a public key derived from that private key. These keys are created by the installation process, or by the **ctcasd** daemon when it is executed for the first time after installation. They are stored by default in the following locations: • **/var/ct/cfg/ct_has.qkf** (private key) • **/var/ct/cfg/ct_has.pkf** (public key) The defaults specified in **/usr/sbin/rsct/cfg/ctcasd.cfg** can be overridden by values specified in **/var/ct/cfg/ctcasd.cfg**. The daemon was unable to access at least one of these files. The files may not exist, or may have permissions set that do not permit processes running with *root* authority to access them. The name of the specific file causing the failure is named in the Detail Data section of this record. The daemon has shut itself down.
KEYF_CFG_ER	PERM	ctcasd Daemon Configuration Failure, key file *filename* not present - recreate public and private key files for this system, verify that the file was not intentionally removed, monitor the file for removal attempts	daemon.err	**Explanation:** The **ctcasd** daemon was unable to locate the local node's public or private key file. The daemon has shut itself down. Authentication attempts using the Host Based Authentication mechanism will not be successful on this node. **Details:** For Host Based Authentication to succeed, each host must possess both a private key, and a public key derived from that private key. These keys are created by the installation process, or by the **ctcasd** daemon when it is executed for the first time after installation. They are stored by default in the following locations: • **/var/ct/cfg/ct_has.qkf** (private key) • **/var/ct/cfg/ct_has.pkf** (public key) The defaults specified in **/usr/sbin/rsct/cfg/ctcasd.cfg** can be overridden by values specified in **/var/ct/cfg/ctcasd.cfg**. Upon startup, the daemon was unable to locate one of the key files. Concluding that this is a configuration failure, the daemon shut itself down. The identity of the missing file is recorded in the Detail Data section of this error log entry.
KEYF_PCREA_ER	PERM	ctcasd Daemon unable to create public key file *filename* - verify that directory exists and has correct permissions	daemon.err	**Explanation:** The **ctcasd** daemon was unable to create a public key for the local node, or was unable to store the public key to a file. The daemon has shut itself down. Authentication attempts using the Host Based Authentication mechanism will not be successful on this node. **Details:** For Host Based Authentication to succeed, each host must possess both a private key, and a public key derived from that private key. These keys are created by the installation process, or by the **ctcasd** daemon when it is executed for the first time after installation. They are stored by default in the following locations: • **/var/ct/cfg/ct_has.qkf** (private key) • **/var/ct/cfg/ct_has.pkf** (public key) The defaults specified in **/usr/sbin/rsct/cfg/ctcasd.cfg** can be overridden by values specified in **/var/ct/cfg/ctcasd.cfg**. The daemon was unable to create or store the public key for this host in the intended file. The intended file is named in the Detail Data section of this error log record. The daemon has shut itself down.

Table 17. Error log templates for Cluster Security Services (continued)

AIX Error Log Label	AIX Error Log Type	Linux System Log Label	Linux System Log Selector Value	Description
KEYF_PDIR_ER	PERM	ctcasd Daemon unable to create public key file *filename* because of directory access failure - verify existence and permissions on the directory	daemon.err	**Explanation:** The **ctcasd** daemon could not access the directory where the public key file for the local system is stored. Authentication attempts using the Host Based Authentication mechanism will not be successful on this node. **Details:** For Host Based Authentication to succeed, each host must possess both a private key, and a public key derived from that private key. These keys are created by the installation process, or by the **ctcasd** daemon when it is executed for the first time after installation. They are stored by default in the following locations: • **/var/ct/cfg/ct_has.qkf** (private key) • **/var/ct/cfg/ct_has.pkf** (public key) The defaults specified in **/usr/sbin/rsct/cfg/ctcasd.cfg** can be overridden by values specified in **/var/ct/cfg/ctcasd.cfg**. The daemon was unable to access the directory where the public key file is supposed to reside on the local system. The directory may be missing, or permissions may have been altered on one or more elements of the directory path to prevent access from root authority processes. The Detail Data section of this record contains the path name of the directory used by the daemon when the failure was detected. The daemon has shut itself down.
KEYF_PLCK_ER	PERM	ctcasd Daemon unable to lock public key file *filename* - verify that file is not locked by system management applications, delete and recreate the file only if the problem cannot be identified and cleared.	daemon.err	**Explanation:** The **ctcasd** daemon was unable to lock the public key file on the local node for exclusive use. Authentication attempts using the Host Based Authentication mechanism will not be successful on this node. **Details:** For Host Based Authentication to succeed, each host must possess both a private key, and a public key derived from that private key. These keys are created by the installation process, or by the **ctcasd** daemon when it is executed for the first time after installation. They are stored by default in the following locations: • **/var/ct/cfg/ct_has.qkf** (private key) • **/var/ct/cfg/ct_has.pkf** (public key) The defaults specified in **/usr/sbin/rsct/cfg/ctcasd.cfg** can be overridden by values specified in **/var/ct/cfg/ctcasd.cfg**. The daemon was unable to obtain exclusive use of the public key file. The file is named in the Detail Data section of this error log record. Another process making use of this file may be hung, or may not have released its exclusive use lock on this file. The daemon has shut itself down.
KEYF_PSPC_ER	PERM	ctcasd Daemon cannot create public key file *filename*, no space in file system - remove obsolete files or extend the file system space	daemon.err	**Explanation:** The **ctcasd** daemon was unable to create a file to store the local node's public key because sufficient file system space was not available. Authentication attempts using the Host Based Authentication mechanism will not be successful on this node. **Details:** For Host Based Authentication to succeed, each host must possess both a private key, and a public key derived from that private key. These keys are created by the installation process, or by the **ctcasd** daemon when it is executed for the first time after installation. They are stored by default in the following locations: • **/var/ct/cfg/ct_has.qkf** (private key) • **/var/ct/cfg/ct_has.pkf** (public key) The defaults specified in **/usr/sbin/rsct/cfg/ctcasd.cfg** can be overridden by values specified in **/var/ct/cfg/ctcasd.cfg**. The daemon detected that neither the public nor the private key file existed on this system. Assuming this to be the initial execution of the daemon, **ctcasd** attempted to create these files. The public key could not be stored because there is not sufficient space in the file system where the public key file – either **/var/ct/cfg/ct_has.pkf** or whatever override value was used in the **ctcasd.cfg** file – was to be stored. The name of the intended target file is provided in the Detail Data section of this record. The daemon has shut itself down.
KEYF_QCREA_ER	PERM	ctcasd Daemon unable to create private key file *filename* - verify that directory exists and has correct permissions	daemon.err	**Explanation:** The **ctcasd** daemon was unable to create a private key for the local node, or was unable to store the private key to a file. The daemon has shut itself down. Authentication attempts using the Host Based Authentication mechanism will not be successful on this node. **Details:** For Host Based Authentication to succeed, each host must possess both a private key, and a public key derived from that private key. These keys are created by the installation process, or by the **ctcasd** daemon when it is executed for the first time after installation. They are stored by default in the following locations: • **/var/ct/cfg/ct_has.qkf** (private key) • **/var/ct/cfg/ct_has.pkf** (public key) The defaults specified in **/usr/sbin/rsct/cfg/ctcasd.cfg** can be overridden by values specified in **/var/ct/cfg/ctcasd.cfg**. The daemon was unable to create or store the private key for this host in the intended file. The intended file is named in this record. The daemon has shut itself down.

Table 17. Error log templates for Cluster Security Services (continued)

AIX Error Log Label	AIX Error Log Type	Linux System Log Label	Linux System Log Selector Value	Description
KEYF_QDIR_ER	PERM	ctcasd Daemon unable to create private key file *filename* because of directory access failure - verify existence and permissions on the directory	daemon.err	**Explanation:** The **ctcasd** daemon could not access the directory where the private key file for the local system is stored. Authentication attempts using the Host Based Authentication mechanism will not be successful on this node. **Details:** For Host Based Authentication to succeed, each host must possess both a private key, and a public key derived from that private key. These keys are created by the installation process, or by the **ctcasd** daemon when it is executed for the first time after installation. They are stored by default in the following locations: • **/var/ct/cfg/ct_has.qkf** (private key) • **/var/ct/cfg/ct_has.pkf** (public key) The defaults specified in **/usr/sbin/rsct/cfg/ctcasd.cfg** can be overridden by values specified in **/var/ct/cfg/ctcasd.cfg**. The daemon was unable to access the directory where the private key file is supposed to reside on the local system. The directory may be missing, or permissions may have been altered on one or more elements of the directory path to prevent access from root authority processes. The Detail Data section of this record contains the path name of the directory used by the daemon when the failure was detected. The daemon has shut itself down.
KEYF_QLCK_ER	PERM	ctcasd Daemon unable to lock private key file *filename* - verify that file is not locked by system management applications, delete and recreate the file ONLY if the problem cannot be identified and cleared.	daemon.err	**Explanation:** The **ctcasd** daemon was unable to lock the private key file on the local node for exclusive use. Authentication attempts using the Host Based Authentication mechanism will not be successful on this node. **Details:** For Host Based Authentication to succeed, each host must possess both a private key, and a public key derived from that private key. These keys are created by the installation process, or by the **ctcasd** daemon when it is executed for the first time after installation. They are stored by default in the following locations: • **/var/ct/cfg/ct_has.qkf** (private key) • **/var/ct/cfg/ct_has.pkf** (public key) The defaults specified in **/usr/sbin/rsct/cfg/ctcasd.cfg** can be overridden by values specified in **/var/ct/cfg/ctcasd.cfg**. The daemon was unable to obtain exclusive use of the private key file. The file is named in the Detail Data section of this error log record. Another process making use of this file may be hung, or may not have released its exclusive use lock on this file. The daemon has shut itself down.
KEYF_QSPC_ER	PERM	ctcasd Daemon cannot create private key file *filename*, no space in file system - remove obsolete files or extend the file system space	daemon.err	**Explanation:** The **ctcasd** daemon was unable to create a file to store the local node's private key because sufficient file system space was not available. Authentication attempts using the Host Based Authentication mechanism will not be successful on this node. **Details:** For Host Based Authentication to succeed, each host must possess both a private key, and a public key derived from that private key. These keys are created by the installation process, or by the **ctcasd** daemon when it is executed for the first time after installation. They are stored by default in the following locations: • **/var/ct/cfg/ct_has.qkf** (private key) • **/var/ct/cfg/ct_has.pkf** (public key) The defaults specified in **/usr/sbin/rsct/cfg/ctcasd.cfg** can be overridden by values specified in **/var/ct/cfg/ctcasd.cfg**. The daemon detected that neither the public nor the private key file existed on this system. Assuming this to be the initial execution of the daemon, **ctcasd** attempted to create these files. The private key could not be stored because there is not sufficient space in the file system where the public key file – either **/var/ct/cfg/ct_has.qkf** or whatever override value was used in the **ctcasd.cfg** file – was to be stored. The name of the intended target file is provided in the Detail Data section of this record. The daemon has shut itself down.

Table 17. Error log templates for Cluster Security Services (continued)

AIX Error Log Label	AIX Error Log Type	Linux System Log Label	Linux System Log Selector Value	Description
KEYF_STAT_ER	PERM	ctcasd Daemon failure, unexpected failure in stat() of file *filename* (error code *error_code*) - The operating system may need additional memory resources	daemon.err	**Explanation:** The **ctcasd** daemon failed while issuing the C library stat() call on either the local system's public or private key files. The presence of these files cannot be confirmed by the daemon. Authentication attempts using the Host Based Authentication mechanism will not be successful on this node. **Details:** For Host Based Authentication to succeed, each host must possess both a private key, and a public key derived from that private key. These keys are created by the installation process, or by the **ctcasd** daemon when it is executed for the first time after installation. They are stored by default in the following locations: • **/var/ct/cfg/ct_has.qkf** (private key) • **/var/ct/cfg/ct_has.pkf** (public key) The defaults specified in **/usr/sbin/rsct/cfg/ctcasd.cfg** can be overridden by values specified in **/var/ct/cfg/ctcasd.cfg**. The daemon was unable to determine if at least one of these files is missing from the local system. The file causing this failure is named in the Detail Data section of this record, along with the **errno** value set by the C library **stat()** routine. Examining the documentation for the **stat()** routine and determining what could cause the generation of the specific **errno** value may assist in determining the root cause of the failure. The daemon has shut itself down.
RPLYINIT_CHMOD_ER	PERM	ctcasd Daemon cannot change the permission of the replay log file **pathname** to read and write by owner only (errno from chmod(): **error_code**) -verify the following: that the directory path exists and has correct permissions; that the daemon is running as root; that the file system where the file resides is not read-only.	daemon.err	**Explanation:** Authentication logging files could not be created for the Enhanced Host Based Authentication mechanism (HBA2) and the mechanism could not be initialized. Authentication attempts using the Enhanced Host Based Authentication mechanism will not be successful on this node. The Host Based Authentication mechanism (HBA) is not affected by this condition and can still be used for authentication, if the local system has enabled this mechanism. **Details:** When the **ctcasd** daemon is started, it will attempt to initialize the Enhanced Host Based Authentication mechanism if the mechanism is configured in the security subsystem. The configuration information for the security subsystem is recorded in the file **/usr/sbin/rsct/cfg/ctsec.cfg** (default) or **/var/ct/cfg/ctsec.cfg** (override). This condition occurs when the **ctcasd** daemon is unable to create an authentication log file and set the file permissions to permit reading and writing only to the *root* user. This condition may occur if the **ctcasd** daemon is started from the command line by a user other than *root*. The Detail Data section of this record contains the path name of the log file that cannot be created. The Enhanced Host Based Authentication mechanism is disabled. The **ctcasd** daemon remains active, and the Host Based Authentication mechanism may be functional if it is configured in the security subsystem. Verify that the **ctcasd** daemon was not started from the command line and verify that the System Resource Controller is running as the *root* user.
RPLYINIT_CHOWN_ER	PERM	ctcasd Daemon cannot change the ownership of the replay log file **pathname** to root (errno from chown(): **error_code**) - verify the following: that the directory path exists and has correct permissions; that the daemon is running as root; that the file system where the file resides is not read-only.	daemon.err	**Explanation:** Authentication logging files could not be created for the Enhanced Host Based Authentication mechanism (HBA2) and the mechanism could not be initialized. Authentication attempts using the Enhanced Host Based Authentication mechanism will not be successful on this node. The Host Based Authentication mechanism (HBA) is not affected by this condition and can still be used for authentication, if the local system has enabled this mechanism. **Details:** When the **ctcasd** daemon is started, it will attempt to initialize the Enhanced Host Based Authentication mechanism if the mechanism is configured in the security subsystem. The configuration information for the security subsystem is recorded in the file **/usr/sbin/rsct/cfg/ctsec.cfg** (default) or **/var/ct/cfg/ctsec.cfg** (override). This condition occurs when the **ctcasd** daemon is unable to create an authentication log file and change the owner of the file to the *root* user. This condition may occur if the **ctcasd** daemon is started from the command line by a user other than *root*. The Detail Data section of this record contains the path name of the log file that cannot be created. The Enhanced Host Based Authentication mechanism is disabled. The **ctcasd** daemon remains active and the Host Based Authentication mechanism may be functional if it is configured in the security subsystem. Verify that the **ctcasd** daemon was not started from the command line and verify that the System Resource Controller is running as the *root* user.

Table 17. Error log templates for Cluster Security Services (continued)

AIX Error Log Label	AIX Error Log Type	Linux System Log Label	Linux System Log Selector Value	Description
RPLYINIT_CREAT_ER	PERM	ctcasd Daemon unable to create replay log file **pathname** (errno from stat(): **error_code**) - verify that the directory path exists and has correct permissions.	daemon.err	**Explanation:** Authentication logging files could not be created for the Enhanced Host Based Authentication mechanism (HBA2) and the mechanism could not be initialized. Authentication attempts using the Enhanced Host Based Authentication mechanism will not be successful on this node. The Host Based Authentication mechanism (HBA) is not affected by this condition and can still be used for authentication, if the local system has enabled this mechanism. **Details:** When the **ctcasd** daemon is started, it will attempt to initialize the Enhanced Host Based Authentication mechanism if the mechanism is configured in the security subsystem. The configuration information for the security subsystem is recorded in the file **/usr/sbin/rsct/cfg/ctsec.cfg** (default) or **/var/ct/cfg/ctsec.cfg** (override). This condition occurs when the **ctcasd** daemon is unable to create an authentication log file. The Detail Data section of this record contains the path name of the log file that cannot be created. The Enhanced Host Based Authentication mechanism is disabled. The **ctcasd** daemon remains active and the Host Based Authentication mechanism may be functional if it is configured in the security subsystem. Verify that the file system containing the path name listed in the Detail Data section is not configured as a read-only file system. Also verify that the path name does not indicate a file that resides in a read-only directory.
RPLYINIT_FILE_ER	PERM	ctcasd Daemon cannot use the existing replay log file - verify that the replay log file is a regular file; that it is owned by root; and that its file system permission allows read/write by root.	daemon.err	**Explanation:** Authentication logging files could not be used by the Enhanced Host Based Authentication mechanism (HBA2). Authentication attempts using the Enhanced Host Based Authentication mechanism will not be successful on this node. The Host Based Authentication mechanism (HBA) is not affected by this condition and can still be used for authentication, if the local system has enabled this mechanism. **Details:** When the **ctcasd** daemon is operational, it will attempt to record information concerning authentication attempts that use the Enhanced Host Based Authentication mechanism if the mechanism is configured in the security subsystem. The configuration information for the security subsystem is recorded in the file **/usr/sbin/rsct/cfg/ctsec.cfg** (default) or **/var/ct/cfg/ctsec.cfg** (override). This condition occurs when the **ctcasd** daemon cannot read or modify the authentication log file. The **ctcasd** daemon may have been started from the command line by a user other than *root*. This condition may also occur if the contents of the file were corrupted due to storage hardware failure, by another application running with *root* privilege modifying the file, or by a user running as *root* modifying the contents of the file. The Detail Data section of this record contains the path name of the log file that cannot be modified. The Enhanced Host Based Authentication mechanism is disabled and any pending authentication requests that make use of this mechanism will fail. The Host Based Authentication mechanism may also be functional if it is configured in the security subsystem. Verify that the file named in the Detail Data section of this report exists, has a size greater than zero bytes, is owned by the *root* user, and has permissions set so only the *root* user may modify the file. Verify that the **ctcasd** daemon was not started from the command line , and verify that the System Resource Controller is running as the *root* user. If these tests show no anomalies, attempt to remove the file and restart the **ctcasd** daemon. If the condition persists, contact the IBM Support Center.
RPLYINIT_FSTATFS_ER	PERM	ctcasd Daemon cannot determine the size of the file system where the replay log file **pathname** resides (errno from fstatfs(): **error_code**) - verify that the directory path exists and has correct permissions.	daemon.err	**Explanation:** Authentication logging files could not be opened by the Enhanced Host Based Authentication mechanism (HBA2) and the mechanism could not be initialized. Authentication attempts using the Enhanced Host Based Authentication mechanism will not be successful on this node. The Host Based Authentication mechanism (HBA) is not affected by this condition and can still be used for authentication, if the local system has enabled this mechanism. **Details:** When the **ctcasd** daemon is started, it will attempt to initialize the Enhanced Host Based Authentication mechanism ("HBA2") if the mechanism is configured in the security subsystem. The configuration information for the security subsystem is recorded in the file **/usr/sbin/rsct/cfg/ctsec.cfg** (default) or **/var/ct/cfg/ctsec.cfg** (override). This condition occurs when the **ctcasd** daemon cannot obtain information for the file system that will store the authentication log file. The Detail Data section of this record contains the path name of the log file that cannot be opened. The Enhanced Host Based Authentication mechanism is disabled. The **ctcasd** daemon remains active and the Host Based Authentication mechanism may be functional if it is configured in the security subsystem. Check the path name listed in the Detail Data section for characters not typically found in file names. Verify that the named file exists. Verify that the **ctcasd** daemon was not started from the command line and verify that the System Resource Controller is running as the *root* user. If all these tests show no anomalies, contact the IBM Support Center.

Table 17. Error log templates for Cluster Security Services (continued)

AIX Error Log Label	AIX Error Log Type	Linux System Log Label	Linux System Log Selector Value	Description
RPLYINIT_MMAP_ER	PERM	ctcasd Daemon cannot memory map the replay log file **pathname** (errno from mmap(): **error_code**) - verify that the directory path exists and has correct permissions.	daemon.err	**Explanation:** Authentication logging files could not be mapped to memory by the Enhanced Host Based Authentication mechanism (HBA2). The Enhanced Host Based Authentication mechanism remains active on this node. The Host Based Authentication mechanism (HBA) is not affected by this condition and can still be used for authentication, if the local system has enabled this mechanism. **Details:** When the **ctcasd** daemon is operational, it will attempt to record information concerning authentication attempts that use the Enhanced Host Based Authentication mechanism if the mechanism is configured in the security subsystem. The configuration information for the security subsystem is recorded in the file **/usr/sbin/rsct/cfg/ctsec.cfg** (default) or **/var/ct/cfg/ctsec.cfg** (override). This condition occurs when the **ctcasd** daemon cannot map the authentication log file to memory. Another application may be attempting to open or map the authentication log file; this should not occur under normal operations. The Detail Data section of this record contains the path name of the log file that could not be mapped. The Enhanced Host Based Authentication mechanism and the **ctcasd** daemon remain active. The Host Based Authentication mechanism may also be functional if it is configured in the security subsystem. Verify that the file named in the Detail Data section of this report exists, has a size greater than zero bytes, is owned by the *root* user, and has permissions set so only the *root* user may modify the file. Verify that the **ctcasd** daemon was not started from the command line and verify that the System Resource Controller is running as the *root* user. If these tests show no anomalies, contact the IBM Support Center.
RPLYINIT_MUTEX_ER	PERM	ctcasd Daemon unable to initialize any of the mutexes used for the replay protection mechanism (error code from **pthread_library_routine** is **error_code**) - verify that the system has sufficient pthread resources available.	daemon.err	**Explanation:** Multiprocessing locks could not be established for the Enhanced Host Based Authentication mechanism (HBA2) and the mechanism could not be initialized. Authentication attempts using the Enhanced Host Based Authentication mechanism will not be successful on this node. The Host Based Authentication mechanism (HBA) is not affected by this condition and can still be used for authentication, if the local system has enabled this mechanism. **Details:** When the **ctcasd** daemon is started, it will attempt to initialize the Enhanced Host Based Authentication mechanism if the mechanism is configured in the security subsystem. The configuration information for the security subsystem is recorded in the file **/usr/sbin/rsct/cfg/ctsec.cfg** (default) or **/var/ct/cfg/ctsec.cfg** (override). This condition occurs when the **ctcasd** daemon is unable to establish multiprocessing locks for the Enhanced Host Based Authentication mechanism. The Detail Data section of this record contains information about the failure condition that should be reported to the IBM Support Center. The Enhanced Host Based Authentication mechanism is disabled. The **ctcasd** daemon remains active and the Host Based Authentication mechanism may be functional if it is configured in the security subsystem.
RPLYINIT_NOSPC_ER	PERM	ctcasd Daemon unable to create the replay log file **pathname** because there is not sufficient space in the file system - create space in the file system by removing unnecessary files.	daemon.err	**Explanation:** Authentication logging files could not be reserved for the Enhanced Host Based Authentication mechanism (HBA2), and the mechanism could not be initialized. Authentication attempts using the Enhanced Host Based Authentication mechanism will not be successful on this node. The Host Based Authentication mechanism (HBA) is not affected by this condition and can still be used for authentication, if the local system has enabled this mechanism. **Details:** When the **ctcasd** daemon is started, it will attempt to initialize the Enhanced Host Based Authentication mechanism if the mechanism is configured in the security subsystem. The configuration information for the security subsystem is recorded in the file **/usr/sbin/rsct/cfg/ctsec.cfg** (default) or **/var/ct/cfg/ctsec.cfg** (override). This condition occurs when the **ctcasd** daemon is unable to reserve file system space to store an authentication log file. The Detail Data section of this record contains the path name of the log file that cannot be reserved. The Enhanced Host Based Authentication mechanism is disabled. The **ctcasd** daemon remains active and the Host Based Authentication mechanism may be functional if it is configured in the security subsystem. To remove this condition, create additional space in the file system that contains the path name listed in the Detail Data section of this report.

Table 17. Error log templates for Cluster Security Services (continued)

AIX Error Log Label	AIX Error Log Type	Linux System Log Label	Linux System Log Selector Value	Description
RPLYINIT_OPEN_ER	PERM	ctcasd Daemon cannot open the replay log file **pathname** for readng and writing (errno from open(): **error_code**) - verify the following: that the directory path exists and has correct permissions; that the daemon is running as root; that the file system where the file resides is not read-only.	daemon.err	**Explanation:** Authentication logging files could not be opened by the Enhanced Host Based Authentication mechanism (HBA2) and the mechanism could not be initialized. Authentication attempts using the Enhanced Host Based Authentication mechanism will not be successful on this node. The Host Based Authentication mechanism (HBA) is not affected by this condition and can still be used for authentication, if the local system has enabled this mechanism. **Details:** When the **ctcasd** daemon is started, it will attempt to initialize the Enhanced Host Based Authentication mechanism (HBA2) if the mechanism is configured in the security subsystem. The configuration information for the security subsystem is recorded in the file **/usr/sbin/rsct/cfg/ctsec.cfg** (default) or **/var/ct/cfg/ctsec.cfg** (override). This condition occurs when the **ctcasd** daemon is unable to open an existing authentication log file. This condition may occur if the **ctcasd** daemon is started from the command line by a user other than *root*, or if the log file was removed by another application or the *root* user during the daemon start procedure. The Detail Data section of this record contains the path name of the log file that cannot be opened. The Enhanced Host Based Authentication mechanism is disabled. The **ctcasd** daemon remains active and the Host Based Authentication mechanism may be functional if it is configured in the security subsystem. Verify that the named file exists. If the file exists and has a zero length, remove the file and attempt to restart the **ctcasd** daemon. Verify that the **ctcasd** daemon was not started from the command line and verify that the System Resource Controller is running as the *root* user. If all these tests show no anomalies, contact the IBM Support Center.
RPLYINIT_READ_ER	PERM	ctcasd Daemon cannot read the replay log file **pathname** (errno from read(): **error_code**) - verify that the directory path exists and has correct permissions.	daemon.err	**Explanation:** Authentication logging files could not be used by the Enhanced Host Based Authentication mechanism (HBA2). Authentication attempts using the Enhanced Host Based Authentication mechanism will not be successful on this node. The Host Based Authentication mechanism (HBA) is not affected by this condition and can still be used for authentication, if the local system has enabled this mechanism. **Details:** When the **ctcasd** daemon is operational, it will attempt to record information concerning authentication attempts that use the Enhanced Host Based Authentication mechanism if the mechanism is configured in the security subsystem. The configuration information for the security subsystem is recorded in the file **/usr/sbin/rsct/cfg/ctsec.cfg** (default) or **/var/ct/cfg/ctsec.cfg** (override). This condition occurs when the **ctcasd** daemon could not read the authentication log file. The **ctcasd** daemon may have been started from the command line by a user other than *root*. This condition may also occur if the contents of the file were corrupted due to storage hardware failure, by another application running with *root* privilege modifying the file, or by a user running as *root* modifying the contents of the file. The Detail Data section of this record contains the path name of the log file that cannot be read. The Enhanced Host Based Authentication mechanism is disabled and any pending authentication requests that make use of this mechanism will fail. The Host Based Authentication mechanism may also be functional if it is configured in the security subsystem. Verify that the file named in the Detail Data section of this report exists, has a size greater than zero bytes, is owned by the *root* user, and has permissions set so only the *root* user may modify the file. Verify that the **ctcasd** daemon was not started from the command line and verify that the System Resource Controller is running as the *root* user. If these tests show no anomalies, attempt to remove the file and restart the **ctcasd** daemon. If the condition persists, contact the IBM Support Center.
RPLYINIT_STAT_ER	PERM	ctcasd Daemon unable to check replay log file **pathname** (errno from stat(): **error_code**) - verify that the directory path exists and has correct permissions.	daemon.err	**Explanation:** Authentication logging files could not be verified for the Enhanced Host Based Authentication mechanism (HBA2) and the mechanism could not be initialized. Authentication attempts using the Enhanced Host Based Authentication mechanism will not be successful on this node. The Host Based Authentication mechanism (HBA) is not affected by this condition and can still be used for authentication, if the local system has enabled this mechanism. **Details:** When the **ctcasd** daemon is started, it will attempt to initialize the Enhanced Host Based Authentication mechanism if the mechanism is configured in the security subsystem. The configuration information for the security subsystem is recorded in the file **/usr/sbin/rsct/cfg/ctsec.cfg** (default) or **/var/ct/cfg/ctsec.cfg** (override). This condition occurs when the **ctcasd** daemon is unable to verify the status of an authentication log file. The Detail Data section of this record contains the path name of the log file that cannot be verified. The Enhanced Host Based Authentication mechanism is disabled. The **ctcasd** daemon remains active and the Host Based Authentication mechanism may be functional if it is configured in the security subsystem. This condition is likely caused by a file system corruption or by another application attempting to modify the file. Perform diagnostic procedures to verify that the file system is not experiencing failures, and monitor the system for other applications that may attempt to modify the file. The condition can be cleared by removing the authentication log file but this action should only be taken after the file system is checked for problems. If this failure persists, contact the IBM Support Center.

Table 17. Error log templates for Cluster Security Services (continued)

AIX Error Log Label	AIX Error Log Type	Linux System Log Label	Linux System Log Selector Value	Description
RPLYINIT_UNLINK_ER	PERM	ctcasd Daemon cannot use the existing replay log file - ctcasd deletes the exiting file and create a new one.	daemon.err	**Explanation:** Authentication logging files were removed by the Enhanced Host Based Authentication mechanism (HBA2). The Enhanced Host Based Authentication mechanism remains active on this node. The Host Based Authentication mechanism (HBA) is not affected by this condition and can still be used for authentication, if the local system has enabled this mechanism. **Details:** When the **ctcasd** daemon is operational, it will attempt to record information concerning authentication attempts that use the Enhanced Host Based Authentication mechanism if the mechanism is configured in the security subsystem. The configuration information for the security subsystem is recorded in the file **/usr/sbin/rsct/cfg/ctsec.cfg** (default) or **/var/ct/cfg/ctsec.cfg** (override). This condition occurs when the **ctcasd** daemon can no longer use an authentication log file and the daemon has removed the log file. The contents of the file may have been corrupted due to storage hardware failure, by another application running with *root* privilege modifying the file, or by a user running as *root* modifying the contents of the file. The Detail Data section of this record contains the path name of the log file that was removed. The Enhanced Host Based Authentication mechanism and the **ctcasd** daemon remain operational. The Host Based Authentication mechanism may also be functional if it is configured in the security subsystem.
RPLYINIT_WRITE_ER	PERM	ctcasd Daemon cannot write to the replay log file **pathname** (errno from write() of **number** bytes: **error_code**) - verify the following that the directory path exists and has correct permissions; and that the daemon is running as root.	daemon.err	**Explanation:** Authentication logging files could not be modified by the Enhanced Host Based Authentication mechanism (HBA2). Authentication attempts using the Enhanced Host Based Authentication mechanism will not be successful on this node. The Host Based Authentication mechanism (HBA) is not affected by this condition and can still be used for authentication, if the local system has enabled this mechanism. **Details:** When the **ctcasd** daemon is operational, it will attempt to record information concerning authentication attempts that use the Enhanced Host Based Authentication mechanism if the mechanism is configured in the security subsystem. The configuration information for the security subsystem is recorded in the file **/usr/sbin/rsct/cfg/ctsec.cfg** (default) or **/var/ct/cfg/ctsec.cfg** (override). This condition occurs when the **ctcasd** daemon cannot record information to the authentication log file. The **ctcasd** daemon may have been started from the command line by a user other than *root*. This condition may also occur if the contents of the file were corrupted due to storage hardware failure, by another application running with root privilege modifying the file, or by a user running as *root* modifying the contents of the file. The Detail Data section of this record contains the path name of the log file that cannot be modified. The Enhanced Host Based Authentication mechanism is disabled and any pending authentication requests that make use of this mechanism will fail. The **ctcasd** daemon remains active and the Host Based Authentication mechanism may be functional if it is configured in the security subsystem. Verify that the file named in the Detail Data section of this report exists, has a size greater than zero bytes, is owned by the *root* user, and has permissions set so only the *root* user may modify the file. Verify that the **ctcasd** daemon was not started from the command line and verify that the System Resource Controller is running as the *root* user. Perform diagnostics on the file system to check for file system or disk hardware errors. If all these tests show no anomalies, attempt to remove the file and restart the **ctcasd** daemon. If the condition persists, contact the IBM Support Center.
THL_ACC_ER	PERM	ctcasd Daemon cannot access trusted host list file *filename*, file may be removed or access to file or directory restricted - verify that file exists, recreate file if necessary, verify permissions on the file and directory	daemon.err	**Explanation:** The **ctcasd** daemon was unable to access the Authentication Trusted Host List for the local system. Authentication attempts using the Host Based Authentication mechanism will not be successful on this node. **Details:** To authenticate remote clients using Host Based Authentication, the local host must posses a Trusted Host List file, which associates known trusted host names to the node's associated public key value. The trusted host list file is created by the installation process, or by the **ctcasd** daemon when it is executed for the first time after installation. The initial Trusted Host List file is populated with the local node's names, IP addresses, and public key. This file is stored by default in **/var/ct/cfg/ct_has.thl**. The default path name can be overridden by the files **/usr/sbin/rsct/cfg/ctcasd.cfg** (default) or **/var/ct/cfg/ctcasd.cfg** (override). The daemon was unable to access the initial Trusted Host List file. The file may not exist, or may have permissions altered to prevent access to the file. The intended name of the Trusted Host List file is provided in the Detail Data section of this record. The daemon has shut itself down.

Table 17. Error log templates for Cluster Security Services (continued)

AIX Error Log Label	AIX Error Log Type	Linux System Log Label	Linux System Log Selector Value	Description
THL_CREAT_ER	PERM	ctcasd Initialization Failure, cannot created trusted host list file *filename* - verify that the directory exists and has correct permissions	daemon.err	**Explanation:** The **ctcasd** daemon was unable to create the initial Host Based Authentication Trusted Host List for the local system. This error can occur if the local node does not have any IP interfaces configured and active at the time the daemon attempts to create the initial trusted host list file. Authentication attempts using the Host Based Authentication mechanism will not be successful on this node. **Details:** To authenticate remote clients using Host Based Authentication, the local host must posses a Trusted Host List file, which associates known trusted host names to the node's associated public key value. The trusted host list file is created by the installation process, or by the **ctcasd** daemon when it is executed for the first time after installation. The initial Trusted Host List file is populated with the local node's names, IP addresses, and public key. This file is stored by default in **/var/ct/cfg/ct_has.thl**. The default path name can be overridden by the files **/usr/sbin/rsct/cfg/ctcasd.cfg** (default) or **/var/ct/cfg/ctcasd.cfg** (override). The daemon was unable to create the initial Trusted Host List file. The intended name of the Trusted Host List file is provided in the Detail Data section of this record. The daemon has shut itself down.
THL_DIR_ER	PERM	ctcasd Daemon unable to create trusted host list file *filename* because of directory access failure - verify existence and permissions on the directory	daemon.err	**Explanation:** The **ctcasd** daemon could not access the directory where the Host Based Authentication Trusted Host List file for the local system is stored. Authentication attempts using the Host Based Authentication mechanism will not be successful on this node. **Details:** To authenticate remote clients using Host Based Authentication, the local host must posses a Trusted Host List file, which associates known trusted host names to the node's associated public key value. The trusted host list file is created by the installation process, or by the **ctcasd** daemon when it is executed for the first time after installation. The initial Trusted Host List file is populated with the local node's names, IP addresses, and public key. This file is stored by default in **/var/ct/cfg/ct_has.thl**. The default path name can be overridden by the files **/usr/sbin/rsct/cfg/ctcasd.cfg** (default) or **/var/ct/cfg/ctcasd.cfg** (override). The daemon was unable to access the directory where the Trusted Host List file is supposed to reside on the local system. The directory may be missing, or permissions may have been altered on one or more elements of the directory path to prevent access from root authority processes. The Detail Data section of this record contains the path name of the directory used by the daemon when the failure was detected. The daemon has shut itself down.
THL_KEY_ER	INFO	ctcasd Daemon check of the Trusted Host List **pathname** to ensure that the public key for the host name matches the public key in the HBA_PUBKEYFILE **pathname** failed.	daemon.info	**Explanation:** The public key value for the local host does not match the public key value recorded for the local host in the trusted host list file on this host. Client applications on this host may not be able to authenticate to service applications that are operating on this host. Service applications on this host may be able to successfully authenticate clients from other hosts. **Details:** To authenticate remote clients using Host Based Authentication, the local host must posses a Trusted Host List file, which associates known trusted host names to the node's associated public key value. The trusted host list file is created by the installation process, or by the **ctcasd** daemon when it is executed for the first time after installation. The initial Trusted Host List file is populated with the local node's names, IP addresses, and public key. This file is stored by default in **/var/ct/cfg/ct_has.thl**. The default path name can be overridden by the files **/usr/sbin/rsct/cfg/ctcasd.cfg** (default) or **/var/ct/cfg/ctcasd.cfg** (override). When the **ctcasd** daemon is started, the daemon examines the Trusted Host List file to ensure that the host name or IP address entries for the local host use the same public key value that is recorded in the public key file. This file is stored by default in **/var/ct/cfg/ct_has.pkf**, and this default path name can be overridden by the files **/usr/sbin/rsct/cfg/ctcasd.cfg** (default) or **/var/ct/cfg/ctcasd.cfg** (override). The **ctcasd** daemon has detected that at least one host name or IP address entry in the Trusted Host List file uses a public key value that does not match the current value recorded in the public key file. This condition can occur if the public and private keys were modified since the Trusted Host List file was last modified. The Detail Data section of this record contains the names of the Trusted Host List file and the public key file used by the daemon when this condition was detected. The **ctcasd** daemon remains operational. Issuing the **ctsthl -s** command usually rectifies this condition.

Table 17. Error log templates for Cluster Security Services (continued)

AIX Error Log Label	AIX Error Log Type	Linux System Log Label	Linux System Log Selector Value	Description
THL_SPC_ER	PERM	ctcasd Daemon cannot create trusted host list file *filename*, no space in file system - remove obsolete files or extend the file system space	daemon.err	**Explanation:** The **ctcasd** daemon was unable to create a file to store the local node's Trusted Host List because sufficient file system space was not available. Authentication attempts using the Host Based Authentication mechanism will not be successful on this node. **Details:** To authenticate remote clients using Host Based Authentication, the local host must posses a Trusted Host List file, which associates known trusted host names to the node's associated public key value. The trusted host list file is created by the installation process, or by the **ctcasd** daemon when it is executed for the first time after installation. The initial Trusted Host List file is populated with the local node's names, IP addresses, and public key. This file is stored by default in **/var/ct/cfg/ct_has.thl**. The default path name can be overridden by the files **/usr/sbin/rsct/cfg/ctcasd.cfg** (default) or **/var/ct/cfg/ctcasd.cfg** (override). The daemon detected that the Trusted Host List file did not exist on this system. Assuming this to be the initial execution of the daemon, **ctcasd** attempted to create this file. The file data could not be stored because there is not sufficient space in the file system where the Trusted Host List file was to be stored. The name of the intended file is provided in the Detail Data section of this record. The daemon has shut itself down.

Trace information

Do *not* activate this trace facility until you have read this section completely, and understand this material. If you are not certain how to properly use this facility, or if you are not under the guidance of IBM Service, do *not* activate this facility.

Activating this facility may result in degraded performance of your system. Activating this facility may also result in longer response times, higher processor loads, and the consumption of system disk resources. Activating this facility may also obscure or modify the symptoms of timing-related problems.

The cluster security services libraries exploit the Cluster Trace facility. By default, these libraries do not generate trace information. Trace information can be obtained by activating one or more of the available Cluster Trace tracing levels and specifying a trace output file. Any trace output generated is specific to events and processing that occurs on the local system; security events on remote nodes within the cluster are not reflected within this trace output. To trace authentication and authorization related processing within the cluster, it may be necessary to activate tracing on multiple nodes within the cluster, and for IBM Customer Support personnel to consolidate these traces and detect patterns within the trace files.

Tracing the ctcasd daemon

Tracing of the **ctcasd** daemon is controlled by a set of four environment variables.

For each of the environment variables, there is a corresponding keyword that can be set in the **ctcasd** daemon's configuration file (**ctcasd**). If set, however, the environment variables always override the settings in the **ctcasd.cfg** file. For more information on the **ctcasd.cfg** file, see the *Administering RSCT* guide.

The environment variables that control the tracing of the **ctcasd** daemon are:

CT_TR_TRACE
: Indicates whether or not tracing of the **ctcasd** daemon is enabled. Valid values are *on* and *off*. If not set, the CT_TR_TRACE environment variable's associated keyword in the **ctcasd.cfg** file (the TRACE keyword) can specify whether or not tracing is on. If not specified in either of these ways, the default is "ON" with a minimal level of tracing.

CT_TR_FILENAME
: When tracing of the **ctcasd** daemon is enabled (either by the CR_TR_TRACE environment variable or its associated **ctcasd.cfg** file keyword TRACE), this environment variable indicates the location of the trace file. If not set, the CT_TR_FILENAME environment variable's associated keyword in the **ctcasd.cfg** file (the TRACEFILE keyword) can specify the location of the trace file. If not specified in either of these ways, the default location is **/var/ct/IW/log/ctsec/ctcasd/trace**. The default directory will be created automatically by the **ctcasd** daemon. However, if you specify

another location using this environment variable or its associated keyword TRACEFILE, you must ensure that the directory you specify exists. If it does not, the default location is used instead, and an error is logged in the trace.

CT_TR_TRACE_LEVELS
When tracing of the **ctcasd** daemon is enabled (either by the CT_TR_TRACE environment variable or its associated **ctcasd.cfg** file keyword TRACE), this environment variable indicates the level of the trace.

The format of this environment variable is *component*:*category*=*level*. For example, to activate tracing of all information messages:

```
export CT_TR_TRACE_LEVELS="_SEC:Info=8"
```

To enable multiple trace levels, separate the trace level specifications with a comma:

```
export CT_TR_TRACE_LEVELS="_SEC:Info=4,_SEC:Errors=8"
```

Table 18 lists the supported trace categories and levels for tracing the **ctcasd** daemon.

Table 18. Trace categories supported for tracing the ctcasd daemon

Component	Category	Level	Description
_SEC	Info	0	no tracing
_SEC	Info	1	trace minimum informational messages
_SEC	Info	4	trace additional informational messages
_SEC	Info	8	trace all informational messages
_SEC	Errors	0	no tracing for errors
_SEC	Errors	1	trace all errors causing daemon termination
_SEC	Errors	2	trace all call errors and errors causing termination
_SEC	Errors	4	trace failed requests, call errors, and errors causing daemon termination
_SEC	Errors	8	trace all errors

If not set, the CT_TR_TRACE_LEVELS environment variable's associated keyword in the **ctcasd.cfg** file (TRACELEVELS) can specify the trace levels. If not specified in either of these ways, the default is "_SEC:Info=1,_SEC:Errors=1"

CT_TR_SIZE
When tracing of the **ctcasd** daemon is enabled (either by the CT_TR_TRACE environment variable or its associated **ctcasd.cfg** file keyword TRACE), this environment variable indicates the size of the trace file. The minimum size is 4096, and the number specified will be rounded up to the nearest 4096 multiple. If not set, the CT_TR_SIZE environment variable's associated keyword in the **ctcasd.cfg** file (the TRACESIZE keyword) can specify the trace file size. If not specified in either of these ways, the default trace-file size is 1003520.

Tracing cluster security services libraries

Do not activate the tracing of cluster security services libraries without instruction or guidance from IBM Customer Support Center personnel.

Trace is activated by setting two environment variables for a process using the cluster security services libraries:

CT_TR_TRACE_LEVELS
This environment variable is used to control what tracing points and levels of detail are activated. The format of this environment variable is *component:category=level*.

For example, to activate the trace points within the cluster security services library **libct_sec** to trace the entry and exit of routines:

```
export CT_TR_TRACE_LEVELS="_SEA:API=1"
```

To enable multiple trace levels, separate the trace level specifications with a comma:

```
export CT_TR_TRACE_LEVELS="_SEA:API=1,_SEU:API=1"
```

CT_TR_FILENAME
This environment variable names the output file where trace information is to be stored. To avoid confusion, specify a fully qualified path name for this variable.

Trace output files are recorded in binary format. The **rpttr** command reads trace output files and converts them to text readable forms.

Table 19 lists the supported trace categories and levels for tracing cluster security services libraries.

Table 19. Trace categories supported for tracing cluster security services libraries

Library	Component	Category	Level	Description
libct_sec	_SEA	Errors	1	Records incidents of failure detected by the cluster security services **libct_sec** library.
libct_sec	_SEA	API	1	Records the entry and exit points of **libct_sec** library and subroutine calls. This level is used to trace which routines are invoked to handle an application request. No data is displayed.
libct_sec	_SEA	API	8	Records the entry and exit points of internal cluster security services library and subroutine calls. Entry points display the parameter values provided by the calling routine. Exit points display the return code value being passed to the caller.
libct_sec	_SEA	SVCTKN	4	Traces status changes in a cluster security services security services token – required by any exploiter of the cluster security services library – through the **libct_sec** library.
libct_sec	_SEA	CTXTKN	4	Traces status changes in a cluster security services security context token – which defined a secured context between a service requestor and a service provider – through the **libct_sec** library.
libct_sec	_SEU	Errors	1	Records incidents of failure detected by the Host Based Authentication (HBA) Mechanism Pluggable Module.
libct_sec	_SEU	API	1	Records entry and exit points within the Host Based Authentication (HBA) Mechanism Pluggable Module that were invoked in response to an application request. No data is displayed.
libct_sec	_SEU	API	8	Records entry and exit points within the Host Based Authentication (HBA) Mechanism Pluggable Module that were invoked in response to an application request. Entry points display the parameter values provided by the calling routine. Exit points display the return code value being passed to the caller.
libct_sec	_SEU	SVCTKN	4	Traces status changes in a cluster security services security services token – required by any exploiter of the cluster security services library – by the Host Based Authentication (HBA) Mechanism Pluggable Module.
libct_sec	_SEU	CTXTKN	4	Traces status changes in a cluster security services security context token – which defined a secured context between a service requestor and a service provider – by the Host Based Authentication (HBA) Mechanism Pluggable Module.
libct_sec	_SEH	Auth	1	Records successful and unsuccessful authentications performed by the Enhanced Host Based Authentication (HBA2) Mechanism Pluggable Module. No identity information is provided at this level.
libct_sec	_SEH	Auth	8	Records successful and unsuccessful authentications performed by the Enhanced Host Based Authentication (HBA2) Mechanism Pluggable Module. The identities of parties requesting authentication are listed in the trace information, as well as the time of the attempt.

Table 19. Trace categories supported for tracing cluster security services libraries (continued)

Library	Component	Category	Level	Description
libct_sec	_SEH	Errors	1	Records incidents of failure detected by the Enhanced Host Based Authentication (HBA2) Mechanism Pluggable Module.
libct_sec	_SEH	API	1	Records entry and exit points within the Enhanced Host Based Authentication (HBA2) Mechanism Pluggable Module that were invoked in response to an application request. No data is displayed.
libct_sec	_SEH	API	8	Records entry and exit points within the Enhanced Host Based Authentication (HBA2) Mechanism Pluggable Module that were invoked in response to an application request. Entry points display the parameter values provided by the calling routine. Exit points display the return code value being passed to the caller.
libct_sec	_SEH	SvcTkn	1	Traces status changes in a cluster security services security services token–required by any exploiter of the cluster security services library–by the Enhanced Host Based Authentication (HBA2) Mechanism Pluggable Module.
libct_sec	_SEH	SvcTkn	8	Traces details of status changes in a cluster security services security services token–required by any exploiter of the cluster security services library–by the Enhanced Host Based Authentication (HBA2) Mechanism Pluggable Module.
libct_sec	_SEH	CtxTkn	1	Traces status changes in a cluster security services security context token–required by any exploiter of the cluster security services library–by the Enhanced Host Based Authentication (HBA2) Mechanism Pluggable Module.
libct_sec	_SEH	CtxTkn	8	Traces details of status changes in a cluster security services security context token–required by any exploiter of the cluster security services library–by the Enhanced Host Based Authentication (HBA2) Mechanism Pluggable Module.
libct_sec	_SEH	Cred	1	Traces creation and destruction of identity tokens in the Enhanced Host Based Authentication (HBA2) Mechanism Pluggable Module.
libct_sec	_SEH	Cred	8	Traces details of the creation and destruction of identity tokens in the Enhanced Host Based Authentication (HBA2) Mechanism Pluggable Module.
libct_sec	_SEH	IDM	1	Traces operating system mapped identities assigned to authenticated parties by the Enhanced Host Based Authentication (HBA2) Mechanism Pluggable Module. This level indicates the mapped identity assigned, if any.
libct_sec	_SEH	IDM	8	Traces the processing of operating system mapped identities assigned to authenticated parties by the Enhanced Host Based Authentication (HBA2) Mechanism Pluggable Module. This level details the execution of the mapped identity assignment.
libct_sec	_SEH	ACL	1	Traces access control list (ACL) identifier expansion for the security services library performed by the Enhanced Host Based Authentication (HBA2) Mechanism Pluggable Module. This level indicates the expanded ACL identity match.
libct_sec	_SEH	ACL	8	Traces access control list (ACL) identifier expansion for the security services library performed by the Enhanced Host Based Authentication (HBA2) Mechanism Pluggable Module. This level details the execution of the expanded ACL identity processing.
libct_mss	_SEM	Errors	1	Records incidents of failure detected by the cluster security services **libct_mss** library.
libct_mss	_SEM	API	1	Records the entry and exit points of **libct_mss** library and subroutine calls. This level is used to trace which routines are invoked to handle an application request. No data is displayed.
libct_mss	_SEM	API	8	Records the entry and exit points of **libct_mss** library and subroutine calls. Entry points display the parameter values provided by the calling routine. Exit points display the return code value being passed to the caller.
libct_mss	_SEM	Perf	1	Records data used to monitor the overall performance of the **libct_mss** functions. Performance assessments should only be made by IBM Customer Support Center personnel.

Table 19. Trace categories supported for tracing cluster security services libraries (continued)

Library	Component	Category	Level	Description
libct_idm	_SEI	Error	1	Records incidents of failure detected by the cluster security services **libct_idm** library.
libct_idm	_SEI	API	1	Records the entry and exit points of **libct_idm** library and subroutine calls. This level is used to trace which routines are invoked to handle an application request. No data is displayed.
libct_idm	_SEI	API	8	Records the entry and exit points of **libct_idm** library and subroutine calls. Entry points display the parameter values provided by the calling routine. Exit points display the return code value being passed to the caller.
libct_idm	_SEI	Mapping	1	Records the identity mapping rule utilized by cluster security services to map a network security identity to a local user identity.
libct_idm	_SEI	Mapping	2	Records the local identity that was mapped to a security network identity by the **libct_idm** library.
libct_idm	_SEI	Mapping	8	Records both the identity mapping rule utilized by cluster security services to map a network security identity to a local user identity, and the local identity obtained from applying this rule.
libct_idm	_SEI	Milestone	1	Generates a record to indicate that a specific internal checkpoint has been reached. This record contains only the name of the checkpoint.
libct_idm	_SEI	Milestone	8	Generates a record to indicate that a specific internal checkpoint has been reached. This record contains the name of the checkpoint and some diagnostic data that IBM Customer Support Center personnel may need in tracing internal failures.
libct_idm	_SEI	Diag	1	Records diagnostic information about the identity mapping definition file input and output processing. This information is meaningful only to IBM Customer Support Center personnel.

Diagnostic procedures

Diagnostic procedures are divided into those oriented towards either of the two primary security functions: authentication and authorization.

Authentication troubleshooting procedures

There are several procedures for troubleshooting authentication problems.

Procedures for troubleshooting authentication problems include:

- Troubleshooting procedures that are independent of the authentication mechanism being used.
- Troubleshooting procedures for host-based authentication mechanisms.

Mechanism independent authentication troubleshooting procedures:

When troubleshooting the RSCT Security subsystem, these procedures can be used regardless of the specific security mechanisms employed throughout the cluster.

Perform these diagnostic procedures first, before attempting to troubleshoot specific security mechanisms.

These diagnostic procedures should be performed by the **root** user on AIX, Linux, or Solaris systems.

Procedure 1: verifying the location of the cluster security services configuration file:

The cluster security services libraries may be unable to locate configuration information for the node.

Purpose:

To ensure that the cluster security services libraries can locate configuration information for the node.

Instructions:

The cluster security services library employs a configuration file that informs the library which security mechanisms are currently available on the local system. By default, this information resides in the file **/usr/sbin/rsct/cfg/ctsec.cfg**. Should a system administrator care to modify or extend this configuration information, the file must be copied to the override location of **/var/ct/cfg/ctsec.cfg** before any modifications are made. If a configuration file exists as **/var/ct/cfg/ctsec.cfg** on the local node, the cluster security services library will ignore the default configuration file and use this one. Under normal circumstances, when all nodes within the cluster employ the same software levels of RSCT, all nodes should use either the default or the override file; there should not be a set of nodes using the default configuration while others use an override. Verify that at least one of these files is present on the local system, and that any such files are not zero-length files:

```
ls -l /usr/sbin/rsct/cfg/ctsec.cfg /var/ct/cfg/ctsec.cfg
```

Verifying the diagnostic:

On AIX nodes, normal configurations will yield a result similar to:

```
ls: 0653-341 The file /var/ct/cfg/ctsec.cfg does not exist
-r--r--r--   1 bin   bin  630  Apr 09 14:29
                        /usr/sbin/rsct/cfg/ctsec.cfg
```

On Linux or Solaris nodes, normal configurations will yield results similar to:

```
ls: /var/ct/cfg/ctsec.cfg: No such file or directory
  -r--r--r--   1 bin   bin  630  Apr 09 14:29
                        /usr/sbin/rsct/cfg/ctsec.cfg
```

At least one of the files should be detected, and any detected file should show read-only permissions and a size greater than zero bytes.

Failure actions:

Restore the default cluster security services configuration file **/usr/sbin/rsct/cfg/ctsec.cfg** from either a system backup or from the RSCT installation media. Monitor the system to ensure that the file is not removed by another user or process.

Next diagnostic procedure:

Proceed to "Procedure 2: verifying the contents of the cluster security services configuration file."

Procedure 2: verifying the contents of the cluster security services configuration file:

The configuration information for the node may not be valid.

Purpose:

To ensure that the configuration information for the node is valid.

Instructions:

Examine the configuration file that will be used by cluster security services. If an override file is in place (as described in Procedure 1), examine that file with a text editor; otherwise, examine the default file with a text editor. The format of the cluster security services configuration file is:

```
#Prior  Mnemonic   Code        Path                Flags
#------------------------------------------------------------
  1      unix       0x00001     /usr/lib/unix.mpm    i
  2      hba2       0x00002     /usr/lib/hba2.mpm    iz[unix]
```

Each line within the file constitutes an entry for a security mechanism. Any blank lines or lines beginning with a # character are ignored. Each entry not commented should posses an unique mnemonic for the security mechanism, code for the mechanism, and priority.

Verifying the diagnostic:

Examine the contents of the file to ensure that none share a priority value, a mnemonic name, or a code number. For any entries that are not commented, verify that a binary file exists on the system in the location specified in the **Path** column.

Failure actions:

If the file being examined is the override configuration file, consider moving it so that the default cluster security services configuration file will be used until problems with this file are corrected.

If any priority or code numbers are shared, modify the file to makes these values unique for each entry. It is best to examine other **ctsec.cfg** files elsewhere within the cluster and to choose values for the priority and code that agree with those used by the other cluster members. Do **not** alter the value for the mechanism mnemonic unless instructed to do so by the IBM Customer Support Center.

Next diagnostic procedure:

Proceed to "Procedure 3: verifying that mechanism-pluggable modules are installed."

Procedure 3: verifying that mechanism-pluggable modules are installed:

The cluster security services library **libct_sec** may be unable to locate the mechanism pluggable modules (MPMs) required to use the security mechanisms configured in the **ctsec.cfg** file.

Purpose:

To ensure that the cluster security services library **libct_sec** can locate the mechanism-pluggable modules (MPMs) required to use the security mechanisms configured in the **ctsec.cfg** file.

Instructions:

The **ctsec.cfg** configuration file provides the location of the MPM that is loaded by the cluster security services library to interface with that security mechanism. This location is specified in the Path column of each entry:

```
#Prior  Mnemonic   Code        Path                Flags
 #-------------------------------------------------------
  1      unix       0x00001     /usr/lib/unix.mpm    i
  2      hba2       0x00002     /usr/lib/hba2.mpm    iz[unix]
```

MPMs shipped by RSCT reside in the **/usr/sbin/rsct/lib** directory and have an extension of ***.mpm**. RSCT places symbolic links to these modules in the **/usr/lib** directory so that the cluster security services library can find them as part of the default library path search. Verify that any MPM files listed in the configuration exist and are binary files. For example:

```
file /usr/lib/unix.mpm
```

If the file proves to be a symbolic link, check the type of file referenced by that link. For example:

```
file /usr/sbin/rsct/lib/unix.mpm
```

Verifying the diagnostic:

For AIX operating systems, the mechanism pluggable module should appear as:

```
/usr/sbin/rsct/bin/unix.mpm: executable (RISC System 6000)
or object module
```

For Intel-based Linux systems, the mechanism pluggable module should appear as:

```
/usr/sbin/rsct/bin/unix.mpm: ELF 32-bit LSB shared object.
Intel 80386,  version 1
```

For PowerPC®-based Linux systems, the mechanism pluggable module should appear as:

```
/usr/sbin/rsct/bin/unix.mpm: ELF 32-bit MSB shared object,
PowerPC  or cisco 4500, version 1 (SYSV)
```

For Intel-based Solaris systems, the mechanism pluggable module should appear as:

```
/usr/sbin/rsct/lib/unix.mpm: ELF 32-bit LSB dynamic lib
 80386 Version 1, dynamically linked, not stripped
```

For SPARC-based Solaris systems, the mechanism pluggable module should appear as:

```
/usr/sbin/rsct/lib/unix.mpm: ELF 32-bit MSB dynamic lib
SPARC32PLUS Version 1, V8+ Required, dynamically linked, not
stripped
```

Failure actions:

If the default Cluster Security Services configuration is currently not in use, consider restoring the default configuration until problems with the Cluster Security Services are resolved.

If mechanism pluggable modules exist in the **/usr/sbin/rsct/lib** directory but not the **/usr/lib** directory, make symbolic links to these files in the **/usr/lib** directory, or alter the default library search path setting (LIBPATH on AIX systems, LD_LIBRARY_PATH on Linux or Solaris systems) to include the **/usr/sbin/rsct/lib** directory.

If MPMs are not found in either location, restore them from a system backup or from the RSCT installation media.

Next diagnostic procedure:

Proceed to "Procedure 4: verifying consistent cluster security services configuration throughout the cluster."

Procedure 4: verifying consistent cluster security services configuration throughout the cluster:

All cluster security services libraries within the cluster may not be using consistent configurations.

Purpose:

To ensure that all cluster security services libraries within the cluster are using consistent configurations.

Instructions:

Unless the cluster consists of nodes at differing RSCT software levels, all nodes within the cluster should employ either the default cluster security services library configuration file, or they should use the override location for this file. Nodes would only use a mix of these files when the cluster contains back-level RSCT nodes that have been modified to operate within a cluster containing more recent RSCT nodes.

The exact content of this file will depend on the RSCT Cluster setup.

- In a management domain, each node must share at least one security mechanism in common with the Management Server. Verify this by examining the active cluster security services configuration files on the Management Server and any nodes that the Management Server controls.
- In an RSCT peer domain, each node must share all security mechanisms, since each node can be considered a fail-over replacement for each other node within the peer domain. Verify this by examining the active cluster security services configuration files on each node within the peer domain.

Verifying the diagnostic:

Examine the cluster security services configuration files on all nodes within the cluster using a

text editor. Verify that these files are consistent, using the criteria stated in the preceding "Instructions" subsection. These files are **/usr/sbin/rsct/cfg/ctsec.cfg** or the override file **/var/ct/cfg/ctsec.cfg**.

Failure actions:

If modifications must be made to the configurations on specific nodes to make them consistent with the configurations on the remaining cluster nodes, **make modifications to the override configuration file instead of the default configuration file**. Edit the configuration files to be consistent. However, do **not** add entries to these files **unless** the system contains the mechanism pluggable module for any security mechanism that is to be added **and** that node is configured to make use of that security mechanism.

Next diagnostic procedure:

Determine which security mechanism would be used by an application, and proceed to the diagnostic procedures specific to that security mechanism.

Troubleshooting procedures for host-based authentication mechanisms:

The host-based authentication mechanisms – Host-Based Authentication (HBA) and Enhanced Host-Based Authentication (HBA2) – rely upon the ability to resolve the IP address of a host to a host name, and to obtain a consistent host name value for a system throughout the cluster.

The local system's host based authentication mechanism trusted host list is searched to find an entry matching the host name or IP address, obtain the public key associated with it, and use this key in the verification of credentials. Authentication failures can result if the host based authentication Mechanism Pluggable Modules or the **ctcasd** daemon are unable to resolve IP addresses, if the addresses are resolved in inconsistent ways throughout the cluster, or if differing host name values are obtained for the same system in different locations within the cluster.

These troubleshooting procedures are designed to be used between two separate nodes of a cluster that are experiencing authentication problems. These procedures use the terms **nodeA** and **nodeB** to generically to refer to these nodes, where **nodeA** is initiating a request to **nodeB**, and an authentication problem occurs as a result. If the problem involves more than two nodes in the cluster, repeat these steps for each pairing of nodes that is experiencing the problem.

When performing these procedures, connect to the systems as the **root** user.

Procedure 1: verifying the ctcasd daemon configurations:

You may need to verify basic configuration information for the host-based authentication mechanisms.

Purpose:

To verify basic configuration information for the host based authentication mechanisms. This procedure indicates what configuration is in use by this node, whether private and public keys have been established for this node and appear to be valid, and whether the node has any entries for itself in its own trusted host list.

Instructions:

To perform the basic configuration check, issue the following command on both systems:

```
/usr/sbin/rsct/bin/ctsvhbac
```

Verifying the diagnostic:

Normal output for this command is similar to the following:

```
-------------------------------------------------------
Host Based Authentication Mechanism Verification Check

Private and Public Key Verifications
```

```
        Configuration file:  /usr/sbin/rsct/cfg/ctcasd.cfg
                    Status:  Available
                  Key Type:  rsa512
                             RSA key generation method, 512-bit
                             key

          Private Key file:  /var/ct/cfg/ct_has.qkf
                    Source:  Configuration file
                    Status:  Available
                  Key Type:  rsa512
                             RSA key generation method, 512-bit
                             key

           Public Key file:  /var/ct/cfg/ct_has.pkf
                    Source:  Configuration file
                    Status:  Available
                  Key Type:  rsa512
                             RSA key generation method, 512-bit
                             key

               Key Parity:  Public and private keys are in pair

  Trusted Host List File Verifications

    Trusted Host List file:  /var/ct/cfg/ct_has.thl
                    Source:  Configuration file
                    Status:  Available

                  Identity:  mimbar.ialliance.org
                    Status:  Trusted host

                  Identity:  9.194.78.145
                    Status:  Trusted host

                  Identity:  127.0.0.1
                    Status:  Trusted host

                   Identity:  localhost
                     Status:  Trusted host
                   Identity:  ::1
                     Status:  Trusted host

Host Based Authentication Mechanism Verification Check completed
------------------------------------------------------------
```

Make note of the configuration file currently in use on this system; this file will be used in later procedures. Also, make note of the public key file name listed in the `Private and Public Key Verifications` section; this information will be used in several of the procedures that follow.

If the command detects any problems, messages will be displayed to indicate these problems. Critical problems are accompanied by messages to assist the user in resolving the problem. For example, if a mismatch exists between the private and public keys for this system, the output generated by the command will appear as follows:

```
-------------------------------------------------------
Host Based Authentication Mechanism Verification Check

Private and Public Key Verifications

        Configuration file:  /var/ct/cfg/ctcasd.cfg
                    Status:  Available
                  Key Type:  rsa512
                             RSA key generation method, 512-bit
                             key

          Private Key file:  /var/ct/cfg/badpvt
                    Source:  Configuration file
                    Status:  Available
                  Key Type:  rsa512
                             RSA key generation method, 512-bit
                             key

            Public Key file:  /var/ct/cfg/ct_has.pkf
                      Source:  Configuration file
                      Status:  Available
                    Key Type:  rsa512
                               RSA key generation method, 512-bit
                               key

                  Key Parity:  Configuration Error - Public and
                               private keys are not in pair
ctsvhbac: Private and public key parity test failed.  The private
and public keys tested were found to be not in pair.  This can
cause authentication failures between the local system and other
systems in the cluster.  These keys were obtained from the
following files:
          Private key file: /var/ct/cfg/badpvt
           Public key file: /var/ct/cfg/ct_has.pkf
If the -q or -p options were specified, ensure that the correct
private and public key file path names were used.  If the correct
file path names were used, the system administrator should
consider generating a new pair of private and public keys using
the ctskeygen command and replacing the entries for the local
system in the trusted host list file using the ctsthl command.
System administrators should remember that when these keys are
regenerated for a node, all systems that consider the local
system a trusted host must be informed of the public key value
change and update their trusted host lists accordingly.

Host Based Authentication Mechanism Verification Check completed
-------------------------------------------------------------------
```

Failure actions:

Perform any suggested actions recommended in the command output. Assistance for resolving any critical problems that the command might detect are provided in the "Error symptoms, responses, and recoveries" on page 123.

If problems are detected using the override configuration file **/var/ct/cfg/ctcasd.cfg**, consider removing this file temporarily and making use of the default configuration file **/usr/sbin/rsct/bin/ctcasd.cfg** .

If none of the network interfaces for the local system appear in the `Trusted Host List File Verifications` output section, re-seed the trusted host list with the local system interface data by using the **ctsthl -s** command.

Next diagnostic procedure:
Proceed to "Procedure 2: verifying permissions of the ctcas daemon start-up program."

Procedure 2: verifying permissions of the ctcas daemon start-up program:

You may need to verify that the **ctcas** service can be started in response to an authenticate request by any system user.

Purpose:
To verify that the **ctcas** service can be started in response to an authenticate request by any system user.

The **ctcas** service is implemented as an on-demand subservice of the System Resource Controller. When the operating system starts, the **ctcas** service remains inactive until the first host-based authentication request for the HBA or HBA2 mechanism is received by the cluster security services. The System Resource Controller will attempt to start the **ctcas** subservice to handle the request, but the ability to start subservices is restricted to system super users. To permit an authentication request from a non-root system user to start the **ctcas** subservice, the cluster security services provide a binary set-user-on-execution command that grants sufficient privilege to non-root users to start the **ctcas** subservice.

If the system administrator chooses to alter the set-user-on-execution permissions for this startup command, non-root users may experience authentication failures when using the host-based authentication mechanisms. These users will not be able to use the HBA or HBA2 mechanisms for authentication unless the **ctcas** service is already active.

This procedure identifies whether the file permissions on the **ctcas** startup command have been altered, which may cause authentication failures for non-root users.

Instructions:
Issue the following command to obtain the file permissions on the **ctcas** startup command:

```
ls -l /usr/sbin/rsct/bin/ctstrtcasd
```

Normal output for this command is similar to the following:

```
-r-sr-xr-x   1 root     bin    130822 Aug 17 15:18
          /usr/sbin/rsct/bin/ctstrtcasd
```

Verifying the diagnostic:
In the preceding sample output, note that the set-user-on-execution bit (`-r-sr-xr-x`) is displayed for the command and that the command is owned by the *root* system user. These are the default settings for this command.

- If the file permissions and ownership are the same as shown in the preceding sample, proceed to "Procedure 3: verifying that the ctcasd daemon is functional" on page 94.
- If the default permissions have been altered, the set-user-on-execution bit may no longer be active or the file owner may have been changed to a non-root system user. This will make it impossible for non-root users to initiate the **ctcas** service and may result in authentication failures if the service is not already active.

Failure actions:
The system administrator must decide whether to restore the default ownership and permissions on this file, or whether to provide a workaround for the permission change.

- **Restore default ownership and permissions**

Restore the default file system permissions and ownership on the **/usr/sbin/rsct/bin/ctstrtcasd** command. See Verifying the diagnostic for the proper default file ownership and permission settings.

- **Manually start the ctcas service**

  When the set-user-on-execution permission is not present or when the file is not owned by the *root* system user, an administrator will need to start the **ctcas** service manually as the system superuser. This can be accomplished by issuing the following command:

  ```
  startsrc -s ctcas
  ```

  Proceed to the next diagnostic procedure to verify the state of the **ctcas** service.

Next diagnostic procedure:

Proceed to "Procedure 3: verifying that the ctcasd daemon is functional."

Procedure 3: verifying that the ctcasd daemon is functional:

You may need to verify that the local system can create and validate host-based authentication mechanism credentials.

Purpose:

To verify that the local system can create and validate host-based authentication mechanism credentials.

The **ctcasd** daemon is controlled by the System Resource Controller (SRC) and operates as a standalone daemon. The daemon is started on demand when any applications on the local nodes needs to obtain credentials to send to a remote server, or when an application attempts to validate these credentials on the local system. If no such requests have been made on the local system, the **ctcasd** daemon will not be active. The daemon may also be inactive if a failure condition caused the daemon to shut down.

Instructions:

Verify that the **ctcasd** daemon is active on both systems using the following SRC query on each system:

```
lssrc -s ctcas
```

Verifying the diagnostic:

If the daemon is active, the command will respond:

```
Subsystem    Group      PID     Status   ctcas      rsct
       120248         active
```

If the daemon is not active, the command will respond:

```
Subsystem    Group      PID     Status   ctcas      rsct
                      inoperative
```

If the daemon has not been properly installed, an error message will be displayed.

Failure actions:

If **ctcasd** is not active, verify that the **ctcasd** daemon has not recorded any failure information from previous start attempts in the AIX Error Log (on AIX nodes) or the System Log (on Linux or Solaris nodes). If any failures are indicated, proceed to "Error symptoms, responses, and recoveries" on page 123 and perform the action associated with abnormal termination of the **ctcasd** daemon. If no failures are indicated, attempt to activate it using the SRC command:

```
startsrc -s ctcasd
```

Wait about five seconds, and then reissue the query instruction listed in the "Instructions" subsection above. If the daemon is not reported as active, examine the error information logs on the system to determine a possible cause of failure. See the section entitled “Error information” on page 19 for assistance in finding this information.

Next diagnostic procedure:

Proceed to “Procedure 4: verifying nodeA registration in the trusted host list residing on nodeB”

Procedure 4: verifying nodeA registration in the trusted host list residing on nodeB:

You may need to verify that the initiating system is recognized as a trusted host by the intended target system.

Purpose:

To verify that the initiating system is recognized as a trusted host by the intended target system.

For authentication to be successful, the intended target service node must "trust" the initiating node, and in most cases, the initiating node must also "trust" the intended target service node. This "trust" is established by recording the identity of the host in the other host's trusted host list.

When the identity is recorded, the public key for the node is also recorded, so that the host can obtain this key whenever it attempts to authenticate host based authentication mechanism credentials from that host.

Instructions:

To determine if the intended target service system trusts the initiating node, first obtain the network identities for the initiating node. On *nodeA*, issue the **ctsvhbal** command to get the list of identities for this system:

```
/usr/sbin/rsct/bin/ctsvhbal
```

A failure will occur if no active network interfaces could be found on the system. This will cause problems in the authentication process. Enable at least one network interface for this system.

Successful output from this command is similar to the following:

```
ctsvhbal: The Host Based Authentication (HBA) mechanism identities
for the local system are:
                  Identity:  mimbar.ialliance.org
                  Identity:  9.194.78.145

ctsvhbal: At least one of the above identities must appear in the
trusted host list on the node where a service application resides
in order for client applications on the local system to authenticate
successfully.
Ensure that at least one host name and one network address identity
from the above list appears in the trusted host list on the service
systems used by applications on this local system.
```

Next, obtain the public key value for *nodeA*. To obtain the key, obtain the name of the currently active public key file as it was displayed in the **ctsvhbac** command executed in “Procedure 1: verifying the ctcasd daemon configurations” on page 90:

```
   Public Key file:  /var/ct/cfg/ct_has.pkf
            Source:  Configuration file
            Status:  Available
          Key Type:  rsa512
                     RSA key generation method, 512-bit key
```

Use this file name as the argument to the **ctskeygen -d -p** command to obtain the current public key value. Using the above sample output as an example, the proper **ctskeygen** command would be:

```
/usr/sbin/rsct/bin/ctskeygen -d -p /var/ct/cfg/ct_has.pkf
```

Successful output from this command is similar to the following:

```
[mimbar/]> ctskeygen -d -p /var/ct/cfg/ct_has.pkf
120200e1235018b7dc7ca24bd6da7fbf508f9eb48a65e40e3e5a685a88c
e9514e5cfd4ff4238d88e8dc095e7e957e9d3ee042ee600d0a508acfe49
e2d5a4995b0f95330103 (generation method: rsa512)
```

Record this information in a location where it can be easily obtained when executing instructions on a remote system.

Switch to *nodeB*. Examine the contents of the trusted host list on *nodeB* to verify that *nodeA* is among its list of trusted hosts. This is done by issuing the **ctsthl -l** command on *nodeB*:

```
/usr/sbin/rsct/bin/ctsthl -l
```

Successful output from this command is similar to the following:

```
[epsilon3][/]> ctsthl -l
--------------------
Host Identity:                127.0.0.1
Identifier Generation Method: rsa512
Identifier Value:
120200a2247fc16279baa24225cdcc101522505655a298b0a48cc792f7
350d2c2d9dd983833c662e9a3f5c0d9114cbdd1486b474b6d3abe89b10
950f329d8fd693de7b0103
--------------------
Host Identity:                9.194.78.149
Identifier Generation Method: rsa512
Identifier Value:
120200a2247fc16279baa24225cdcc101522505655a298b0a48cc792f73
50d2c2d9dd983833c662e9a3f5c0d9114cbdd1486b474b6d3abe89b1095
0f329d8fd693de7b0103
--------------------
Host Identity:                epsilon3.ialliance.org
Identifier Generation Method: rsa512
Identifier Value:
120200a2247fc16279baa24225cdcc101522505655a298b0a48cc792f735
0d2c2d9dd983833c662e9a3f5c0d9114cbdd1486b474b6d3abe89b10950f
329d8fd693de7b0103
--------------------
Host Identity:                9.194.78.145
Identifier Generation Method: rsa512
Identifier Value:
120200e1235018b7dc7ca24bd6da7fbf508f9eb48a65e40e3e5a685a88ce9
514e5cfd4ff4238d88e8dc095e7e957e9d3ee042ee600d0a508acfe49e2d5
a4995b0f95330103
--------------------
Host Identity:                mimbar.ialliance.org
Identifier Generation Method: rsa512
Identifier Value:
```

```
120200e1235018b7dc7ca24bd6da7fbf508f9eb48a65e40e3e5a685a88ce9
514e5cfd4ff4238d88e8dc095e7e957e9d3ee042ee600d0a508acfe49e2d5
a4995b0f95330103
--------------------
```

An exact match must be found for the host name values returned by the **ctsvhbal** command executed on *nodeA* and a host identity listed in the **ctsthl -l** output on *nodeB*. When the matching entry is found, the public key value associated with that entry must match exactly to the value displayed by the **ctskeygen -d** command executed previously on *nodeA*. Also, at least one network address associated with *nodeA* should be listed in the trusted host list on *nodeB* as well. The above example demonstrates such an case.

The following demonstrates a case where the public key values match but an exact host name match does not exist. In this case, problems can occur with the authentication process between *nodeA* and *nodeB*:

```
[mimbar][/]> ctsvhbal
ctsvhbal: The Host Based Authentication (HBA) mechanism
identities for the local system are:
        Identity:  mimbar.ialliance.org  <--- Note the
                                              name displayed
                                              here
        Identity:  9.194.78.145

ctsvhbal: At least one of the above identities must appear in
the trusted host list on the node where a service application
residesin order for client applications on the local system to
authenticate successfully.
Ensure that at least one host name and one network address
identity from the above list appears in the trusted host list
on the service systems used by applications on this local
system.
[mimbar][/]> ctskeygen -d -p /var/ct/cfg/ct_has.pkf
120200e1235018b7dc7ca24bd6da7fbf508f9eb48a65e40e3e5a685a88ce9
514e5cfd4ff4238d88e8dc095e7e957e9d3ee042ee600d0a508acfe49e2d5
a4995b0f95330103
(generation method: rsa512)

[epsilon3][/]> ctsthl -l
--------------------
Host Identity:                127.0.0.1
Identifier Generation Method: rsa512
Identifier Value:
120200a2247fc16279baa24225cdcc101522505655a298b0a48cc792f7350
d2c2d9dd983833c662e9a3f5c0d9114cbdd1486b474b6d3abe89b10950f32
9d8fd693de7b0103
--------------------
Host Identity:                9.194.78.149
Identifier Generation Method: rsa512
Identifier Value:
120200a2247fc16279baa24225cdcc101522505655a298b0a48cc792f7350
d2c2d9dd983833c662e9a3f5c0d9114cbdd1486b474b6d3abe89b10950f32
9d8fd693de7b0103
--------------------
Host Identity:                epsilon3.ialliance.org
Identifier Generation Method: rsa512
```

```
Identifier Value:
120200a2247fc16279baa24225cdcc101522505655a298b0a48cc792f7350
d2c2d9dd983833c662e9a3f5c0d9114cbdd1486b474b6d3abe89b10950f32
9d8fd693de7b0103
--------------------
Host Identity:                9.194.78.145
Identifier Generation Method: rsa512
Identifier Value:
120200e1235018b7dc7ca24bd6da7fbf508f9eb48a65e40e3e5a685a88ce
9514e5cfd4ff4238d88e8dc095e7e957e9d3ee042ee600d0a508acfe49e2
d5a4995b0f95330103
--------------------
Host Identity:                mimbar  <--- Note how name
                                          differs here
Identifier Generation Method: rsa512
Identifier Value:
120200e1235018b7dc7ca24bd6da7fbf508f9eb48a65e40e3e5a685a88ce
9514e5cfd4ff4238d88e8dc095e7e957e9d3ee042ee600d0a508acfe49e2
d5a4995b0f95330103
--------------------
```

The following demonstrates a case where the host identities match but the public key values do not match. This will also inject problems in the authentication process between these systems:

```
[mimbar][/] ctsvhbal
ctsvhbal: The Host Based Authentication (HBA) mechanism
identities for the local system are:
                  Identity:  mimbar.ialliance.org

                  Identity:  9.194.78.145

ctsvhbal: At least one of the above identities must appear
in the trusted host list on the node where a service application
resides in order for client applications on the local system to
authenticate successfully.
Ensure that at least one host name and one network address
identity from the above list appears in the trusted host list
on the service systems used by applications on this local system.
[mimbar][/]> ctskeygen -d -p /var/ct/cfg/ct_has.pkf
120200c75d8cab600c151cd60902a12c430768ee3189cf946d688138356
306b064fd30720b2d37a4b21c0ab2e7092298697d973ce76eb27480b0a8
42daa4f59596e6410103
(generation method: rsa512)

[epsilon3][/]> ctsthl -l
--------------------
Host Identity:                127.0.0.1
Identifier Generation Method: rsa512
Identifier Value:
120200a2247fc16279baa24225cdcc101522505655a298b0a48cc792f735
0d2c2d9dd983833c662e9a3f5c0d9114cbdd1486b474b6d3abe89b10950f
329d8fd693de7b0103
--------------------
Host Identity:                9.194.78.149
Identifier Generation Method: rsa512
Identifier Value:
```

```
120200a2247fc16279baa24225cdcc101522505655a298b0a48cc792f7350
d2c2d9dd983833c662e9a3f5c0d9114cbdd1486b474b6d3abe89b10950f32
9d8fd693de7b0103
--------------------
Host Identity:                epsilon3.ialliance.org
Identifier Generation Method: rsa512
Identifier Value:
120200a2247fc16279baa24225cdcc101522505655a298b0a48cc792f7350
d2c2d9dd983833c662e9a3f5c0d9114cbdd1486b474b6d3abe89b10950f32
9d8fd693de7b0103
--------------------
Host Identity:                9.194.78.145
Identifier Generation Method: rsa512
Identifier Value:
120200e1235018b7dc7ca24bd6da7fbf508f9eb48a65e40e3e5a685a88ce9
514e5cfd4ff4238d88e8dc095e7e957e9d3ee042ee600d0a508acfe49e2d5
a4995b0f95330103
--------------------
Host Identity:                mimbar.ialliance.org
Identifier Generation Method: rsa512
Identifier Value:
120200e1235018b7dc7ca24bd6da7fbf508f9eb48a65e40e3e5a685a88ce9
514e5cfd4ff4238d88e8dc095e7e957e9d3ee042ee600d0a508acfe49e2d5
a4995b0f95330103
--------------------
```

The following demonstrates a case where the network address for *nodeA* is not listed in the trusted host list for *nodeB*. This can inject problems into the authentication process, especially in RSCT Peer Domains.

```
[mimbar][/]> ctsvhbal
ctsvhbal: The Host Based Authentication (HBA) mechanism
identities for the local system are:
          Identity:  mimbar.ialliance.org
          Identity:  9.194.78.145      <--- Note that no
                                            entry will exist
                                            for this address
ctsvhbal: At least one of the above identities must appear in
the trusted host list on the node where a service application
resides in order for client applications on the local system to
authenticate successfully.
Ensure that at least one host name and one network address
identity from the above list appears in the trusted host list
on the service systems used by applications on this local
system.
[mimbar][/]> ctskeygen -d -p /var/ct/cfg/ct_has.pkf
120200e1235018b7dc7ca24bd6da7fbf508f9eb48a65e40e3e5a685a88ce
9514e5cfd4ff4238d88e8dc095e7e957e9d3ee042ee600d0a508acfe49e2
d5a4995b0f95330103
(generation method: rsa512)

[epsilon3][/]> ctsthl -l
--------------------
Host Identity:                127.0.0.1
Identifier Generation Method: rsa512
Identifier Value:
```

```
120200a2247fc16279baa24225cdcc101522505655a298b0a48cc792f7350
d2c2d9dd983833c662e9a3f5c0d9114cbdd1486b474b6d3abe89b10950f32
9d8fd693de7b0103
--------------------
Host Identity:                9.194.78.149
Identifier Generation Method: rsa512
Identifier Value:
120200a2247fc16279baa24225cdcc101522505655a298b0a48cc792f7350
d2c2d9dd983833c662e9a3f5c0d9114cbdd1486b474b6d3abe89b10950f32
9d8fd693de7b0103
--------------------
Host Identity:                epsilon3.ialliance.org
Identifier Generation Method: rsa512
Identifier Value:
120200a2247fc16279baa24225cdcc101522505655a298b0a48cc792f7350d
2c2d9dd983833c662e9a3f5c0d9114cbdd1486b474b6d3abe89b10950f329d
8fd693de7b0103
--------------------
Host Identity:                mimbar.ialliance.org
Identifier Generation Method: rsa512
Identifier Value:
120200e1235018b7dc7ca24bd6da7fbf508f9eb48a65e40e3e5a685a88ce95
14e5cfd4ff4238d88e8dc095e7e957e9d3ee042ee600d0a508acfe49e2d5a4
995b0f95330103
--------------------
```

Failure actions:

If the **ctsvhbal** command failed to find any active network interfaces on the system, enable at least one network connection.

If any entries for *nodeA* in the trusted host list for *nodeB* use incorrect host name, network address, or public key values, remove these entries from the trusted host list by using the **ctsthl -d -n** command on *nodeB*. For example:

```
/usr/sbin/rsct/bin/ctsthl -d -n mimbar
```

After the incorrect entries are removed, add new entries that make use of the correct host name, network address, and public key by using the **ctsthl -a -n** command on *nodeB*. For example:

```
/usr/sbin/rsct/bin/ctsthl -a -n
mimbar.ialliance.org -m rsa512 -p
  120200e1235018b7dc7ca24bd6da7fbf508f9eb48a65e40e3e5a685a
88ce9514e5cfd4ff4238d88e8dc095e7e957e9d3ee042ee600d0a508ac
fe49e2d5a4995b0f95330103
```

Consider adding entries for any omitted host names or network addresses ued by *nodeA* in the trusted host list on *nodeB*. These entries should only remain omitted if the system administrator explicitly chooses not to "trust" clients that connect to *nodeB* that make use of that host identity. Entries are added using the same **ctsthl -a -n** command demonstrated above.

Next diagnostic procedure:

Proceed to “Procedure 5: verifying nodeB registration in the trusted host list residing on nodeA.”

Procedure 5: verifying nodeB registration in the trusted host list residing on nodeA:

You may need to verify that the target service system is recognized as a trusted host by the initiating system, and to ensure that mutual authentication processing can succeed.

Purpose:

To verify that the target service system is recognized as a trusted host by the initiating system, and to ensure that mutual authentication processing can succeed.

For authentication to be successful, the intended target service node must "trust" the initiating node, and in most cases, the initiating node must also "trust" the intended target service node. This "trust" is established by recording the identity of the host in the other host's trusted host list. When the identity is recorded, the public key for the node is also recorded, so that the host can obtain this key whenever it attempts to authenticate host based authentication mechanism credentials from that host.

Instructions:

This procedure is the reverse of "Procedure 4: verifying nodeA registration in the trusted host list residing on nodeB" on page 95.

To determine if the initiating system trusts the intended target service node for mutual authentication processing, first obtain the network identities for the target service node. On *nodeB*, issue the **ctsvhbal** command to get the list of identities for this system:

```
/usr/sbin/rsct/bin/ctsvhbal
```

A failure will occur if no active network interfaces could be found on the system. This will cause problems in the authentication process. Enable at least one network interface for this system.

Successful output from this command is similar to the following:

```
ctsvhbal: The Host Based Authentication (HBA) mechanism
identities for the local system are:
                  Identity:  epsilon3.ialliance.org
                  Identity:  9.194.78.149

  ctsvhbal: At least one of the above identities must appear in
the trusted host list on the node where a service application
resides in order for client applications on the local system to
authenticate successfully.
Ensure that at least one host name and one network address identity
from the above list appears in the trusted host list on the service
systems used by applications on this local system.
```

Next, obtain the public key value for *nodeB*. To obtain the key, obtain the name of the currently active public key file as it was displayed in the **ctsvhbac** command executed in "Procedure 2: verifying permissions of the ctcas daemon start-up program" on page 93:

```
      Public Key file:  /var/ct/cfg/ct_has.pkf
               Source:  Configuration file
               Status:  Available
             Key Type:  rsa512
                        RSA key generation method, 512-bit key
```

Use this file name as the argument to the **ctskeygen -d -p** command to obtain the current public key value. Using the above sample output as an example, the proper **ctskeygen** command would be:

```
/usr/sbin/rsct/bin/ctskeygen -d -p /var/ct/cfg/ct_has.pkf
```

Successful output from this command is similar to the following:

```
[epsilon3][/]> ctskeygen -d -p /var/ct/cfg/ct_has.pkf
120200a2247fc16279baa24225cdcc101522505655a298b0a48cc792f7350d2c
```

```
2d9dd983833c662e9a 3f5c0d9114cbdd1486b474b6d3abe89b10950f329d8fd
693de7b0103
(generation method: rsa512)
```

Record this information in a location where it can be easily obtained when executing instructions on a remote system.

Switch to *nodeA*. Examine the contents of the trusted host list on *nodeA* to verify that *nodeB* is among its list of trusted hosts. This is done by issuing the **ctsthl -l** command on *nodeA*:

```
/usr/sbin/rsct/bin/ctsthl -l
```

Successful output from this command is similar to the following:

```
[mimbar][/]> ctsthl -l
--------------------
Host Identity:                 127.0.0.1
Identifier Generation Method: rsa512
Identifier Value:
120200e1235018b7dc7ca24bd6da7fbf508f9eb48a65e40e3e5a685a88ce9514
e5cfd4ff4238d88e8d c095e7e957e9d3ee042ee600d0a508acfe49e2d5a4995
b0f95330103
--------------------
Host Identity:                 9.194.78.145
Identifier Generation Method: rsa512
Identifier Value:
120200e1235018b7dc7ca24bd6da7fbf508f9eb48a65e40e3e5a685a88ce9514
e5cfd4ff4238d88e8d c095e7e957e9d3ee042ee600d0a508acfe49e2d5a4995
b0f95330103
--------------------
Host Identity:                 mimbar.ialliance.org
Identifier Generation Method: rsa512
Identifier Value:
120200e1235018b7dc7ca24bd6da7fbf508f9eb48a65e40e3e5a685a88ce9514
e5cfd4ff4238d88e8d c095e7e957e9d3ee042ee600d0a508acfe49e2d5a4995
b0f95330103
--------------------
Host Identity:                 9.194.78.149
Identifier Generation Method: rsa512
Identifier Value:
120200a2247fc16279baa24225cdcc101522505655a298b0a48cc792f7350d2c2
d9dd983833c662e9a 3f5c0d9114cbdd1486b474b6d3abe89b10950f329d8fd69
3de7b0103
--------------------
Host Identity:                 epsilon3.ialliance.org
Identifier Generation Method: rsa512
Identifier Value:
120200a2247fc16279baa24225cdcc101522505655a298b0a48cc792f7350d2c2
d9dd983833c662e9a 3f5c0d9114cbdd1486b474b6d3abe89b10950f329d8fd69
3de7b0103
--------------------
```

An *exact* match must be found for the host name values returned by the **ctsvhbal** command executed on *nodeB* and a host identity listed in the **ctsthl -l** output on *nodeA*. When the matching entry is found, the public key value associated with that entry must match exactly to the value

displayed by the **ctskeygen -d** command executed previously on *nodeB*. Also, at least one network address associated with *nodeA* should be listed in the trusted host list on *nodeA* as well. The above example demonstrates such an case.

The following demonstrates a case where the public key values match but an exact host name match does not exist. In this case, problems can occur with the authentication process between *nodeA* and *nodeB*:

```
[epsilon3][/]> ctsvhbal
ctsvhbal: The Host Based Authentication (HBA) mechanism identities
for the local system are:

        Identity:  epsilon3.ialliance.org    <--- Note name
                                                  displayed here

        Identity:  9.194.78.149

ctsvhbal: At least one of the above identities must appear in the
trusted host list on the node where a service application resides
in order for client applications on the local system to authenticate
successfully.
Ensure that at least one host name and one network address identity
from the above list appears in the trusted host list on the service
systems used by applications on this local system.
[epsilon3][/]> ctskeygen -d -p /var/ct/cfg/ct_has.pkf
120200a2247fc16279baa24225cdcc101522505655a298b0a48cc792f7350d2c2
d9dd983833c662e9a 3f5c0d9114cbdd1486b474b6d3abe89b10950f329d8fd69
3de7b0103
(generation method: rsa512)

[mimbar][/]> ctsthl -l
--------------------
Host Identity:                127.0.0.1
Identifier Generation Method: rsa512
Identifier Value:
120200e1235018b7dc7ca24bd6da7fbf508f9eb48a65e40e3e5a685a88ce9514e5
cfd4ff4238d88e8d c095e7e957e9d3ee042ee600d0a508acfe49e2d5a4995b0f9
5330103
--------------------
Host Identity:                9.194.78.145
Identifier Generation Method: rsa512
Identifier Value:
120200e1235018b7dc7ca24bd6da7fbf508f9eb48a65e40e3e5a685a88ce9514e5
cfd4ff4238d88e8d c095e7e957e9d3ee042ee600d0a508acfe49e2d5a4995b0f9
5330103
--------------------
Host Identity:                mimbar.ialliance.org
Identifier Generation Method: rsa512
Identifier Value:
120200e1235018b7dc7ca24bd6da7fbf508f9eb48a65e40e3e5a685a88ce9514e5
cfd4ff4238d88e8d c095e7e957e9d3ee042ee600d0a508acfe49e2d5a4995b0f9
5330103
--------------------
Host Identity:                9.194.78.149
Identifier Generation Method: rsa512
Identifier Value:
```

```
120200a2247fc16279baa24225cdcc101522505655a298b0a48cc792f7350d2c2d
9dd983833c662e9a 3f5c0d9114cbdd1486b474b6d3abe89b10950f329d8fd693d
e7b0103
--------------------
Host Identity:                epsilon3  <--- Note how name
                                              differs here
Identifier Generation Method: rsa512
Identifier Value:
120200a2247fc16279baa24225cdcc101522505655a298b0a48cc792f7350d2c2d
9dd983833c662e9a 3f5c0d9114cbdd1486b474b6d3abe89b10950f329d8fd693d
e7b0103
--------------------
```

The following demonstrates a case where the host identities match but the public key values do not match. This will also inject problems in the authentication process between these systems:

```
[epsilon3][/]> ctsvhbal
ctsvhbal: The Host Based Authentication (HBA) mechanism identities
for the local system are:
                  Identity:  epsilon3.ialliance.org
                  Identity:  9.194.78.149

ctsvhbal: At least one of the above identities must appear in the
trusted host list on the node where a service application resides
in order for client applications on the local system to authenticate
successfully.
Ensure that at least one host name and one network address identity
from the above list appears in the trusted host list on the service
systems used by applications on this local system.
[epsilon3][/]> ctskeygen -d -p /var/ct/cfg/ct_has.pkf
120200a2247fc16279baa24225cdcc101522505655a298b0a48cc792f7350d2c2d
9dd983833c662e9a 3f5c0d9114cbdd1486b474b6d3abe89b10950f329d8fd693d
e7b0103
(generation method: rsa512)

[mimbar][/]> ctsthl -l
[epsilon3][/]> ctsthl -l
--------------------
Host Identity:                127.0.0.1
Identifier Generation Method: rsa512
Identifier Value:
120200e1235018b7dc7ca24bd6da7fbf508f9eb48a65e40e3e5a685a88ce9514e5
cfd4ff4238d88e8d c095e7e957e9d3ee042ee600d0a508acfe49e2d5a4995b0f9
5330103
--------------------
Host Identity:                9.194.78.145
Identifier Generation Method: rsa512
Identifier Value:
120200e1235018b7dc7ca24bd6da7fbf508f9eb48a65e40e3e5a685a88ce9514e5
cfd4ff4238d88e8d c095e7e957e9d3ee042ee600d0a508acfe49e2d5a4995b0f9
5330103
--------------------
Host Identity:                mimbar.ialliance.org
Identifier Generation Method: rsa512
Identifier Value:
120200e1235018b7dc7ca24bd6da7fbf508f9eb48a65e40e3e5a685a88ce9514e5c
```

```
fd4ff4238d88e8d c095e7e957e9d3ee042ee600d0a508acfe49e2d5a4995b0f953
30103
--------------------
Host Identity:                 9.194.78.149
Identifier Generation Method: rsa512
Identifier Value:
120200a2247fc16279baa24225cdcc101522505655a298b0a48cc792f7350d2c2d9
dd983833c662e9a 3f5c0d9114cbdd1486b474b6d3abe89b10950f329d8fd693de7
b0103
--------------------
Host Identity:                 epsilon3.ialliance.org
Identifier Generation Method: rsa512
Identifier Value:
120200e1235018b7dc7ca24bd6da7fbf508f9eb48a65e40e3e5a685a88ce9514e5c
fd4ff4238d88e8d c095e7e957e9d3ee042ee600d0a508acfe49e2d5a4995b0f953
30103
--------------------
```

The following demonstrates a case where the network address for *nodeB* is not listed in the trusted host list for *nodeA*. This can inject problems into the authentication process, especially in RSCT Peer Domains.

```
[epsilon3][/]> ctsvhbal
ctsvhbal: The Host Based Authentication (HBA) mechanism identities
for the local system are:
         Identity:  epsilon3.ialliance.org
         Identity:  9.194.78.149    <-- Note that no entry will
                                        exist for this address
ctsvhbal: At least one of the above identities must appear in the
trusted host list on the node where a service application resides
in order
for client applications on the local system to authenticate
successfully.
Ensure that at least one host name and one network address identity
from the above list appears in the trusted host list on the service
systems used by applications on this local system.
[epsilon3][/]> ctskeygen -d -p /var/ct/cfg/ct_has.pkf
120200a2247fc16279baa24225cdcc101522505655a298b0a48cc792f7350d2c2d9
dd983833c662e9a 3f5c0d9114cbdd1486b474b6d3abe89b10950f329d8fd693de7
b0103
(generation method: rsa512)
[mimbar][/]> ctsthl -l
--------------------
Host Identity:                 127.0.0.1
Identifier Generation Method: rsa512
Identifier Value:
120200e1235018b7dc7ca24bd6da7fbf508f9eb48a65e40e3e5a685a88ce9514e5c
fd4ff4238d88e8d c095e7e957e9d3ee042ee600d0a508acfe49e2d5a4995b0f953
30103
--------------------
Host Identity:                 9.194.78.145
Identifier Generation Method: rsa512
Identifier Value:
120200e1235018b7dc7ca24bd6da7fbf508f9eb48a65e40e3e5a685a88ce9514e5c
fd4ff4238d88e8d c095e7e957e9d3ee042ee600d0a508acfe49e2d5a4995b0f953
30103
```

```
--------------------
Host Identity:                mimbar.ialliance.org
Identifier Generation Method: rsa512
Identifier Value:
120200e1235018b7dc7ca24bd6da7fbf508f9eb48a65e40e3e5a685a88ce9514e5c
fd4ff4238d88e8d c095e7e957e9d3ee042ee600d0a508acfe49e2d5a4995b0f953
30103
--------------------
Host Identity:                epsilon3.ialliance.org
Identifier Generation Method: rsa512
Identifier Value:
120200a2247fc16279baa24225cdcc101522505655a298b0a48cc792f7350d2c2d9
dd983833c662e9a 3f5c0d9114cbdd1486b474b6d3abe89b10950f329d8fd693de7
b0103
--------------------
```

Failure actions:

If the **ctsvhbal** command failed to find any active network interfaces on the system, enable at least one network connection.

If any entries for *nodeB* in the trusted host list for *nodeA* use incorrect host name, network address, or public key values, remove these entries from the trusted host list by using the **ctsthl -d -n** command on *nodeA*. For example:

```
/usr/sbin/rsct/bin/ctsthl -d -n epsilon3
```

After the incorrect entries are removed, add new entries that make use of the correct host name, network address, and public key by using the **ctsthl -a -n** command on *nodeA*. For example:

```
 /usr/sbin/rsct/bin/ctsthl -a -n
epsilon3.ialliance.org -m rsa512 -p
120200a2247fc16279baa24225cdcc101522505655a298b0a48cc792f7350
d2c2d9dd983833c662e9a 3f5c0d9114cbdd1486b474b6d3abe89b10950f3
29d8fd693de7b0103
```

Consider adding entries for any omitted host names or network addresses used by *nodeB* in the trusted host list on *nodeA*. These entries should only remain omitted if the system administrator explicitly chooses not to "trust" clients that connect to *nodeA* that make use of that host identity. Entries are added using the same **ctsthl -a -n** command demonstrated above.

Next diagnostic procedure:

Proceed to “Procedure 6: verifying that credential expiration checking is active.”

Procedure 6: verifying that credential expiration checking is active:

You may need to determine whether the credential expiration time interval is injecting authentication problems.

Purpose:

To determine if the credential expiration time interval may be injecting authentication problems.

The Host Based Authentication mechanism (HBA) and the Enhanced Host Based Authentication mechanism (HBA2) provide a control to allow the system to reject outdated credentials that might be replayed at a later time by applications seeking to get unwarranted access to the system. By default, these controls are disabled. For HBA, the control is enabled by specifying a count, in seconds or minutes, in the HBA_CRED_TIMETOLIVE field of the override configuration file **/var/ct/cfg/ctcasd.cfg**. For HBA2, the control is enabled by specifying a count, in seconds or minutes, in the HBA2_CRED_TIMETOLIVE field of the same file. These counts are used in conjunction with the time-of-day clock value by the **ctcasd** daemon to determine if it is

processing an outdated credential. Authentication failures can result if the HBA_CRED_TIMETOLIVE or HBA2_CRED_TIMETOLIVE values are not large enough to account for time-of-day clock differences (in Universal Time Coordinated or UTC) between the systems and any latency added by network speed and processor loads.

HBA_CRED_TIMETOLIVE is an option available starting in RSCT version 2.3.2.0. HBA2_CRED_TIMETOLIVE is an option available starting in RSCT version 2.3.10.0 and 2.4.6.0. Earlier versions of RSCT do not support these options.

Instructions:

On each system, examine the contents of the currently active configuration file. This file is listed in the **/var/ct/cfg/ctcasd.cfg** command output generated for that system in "Procedure 1: verifying the ctcasd daemon configurations" on page 90. For example:

```
Configuration file:  /var/ct/cfg/ctcasd.cfg
             Status:  Available
           Key Type:  rsa512
                      RSA key generation method, 512-bit key
```

Examine this file with a text editor and make note of any values listed for the HBA_CRED_TIMETOLIVE and HBA2_CRED_TIMETOLIVE options. The file contents may appear as follows:

```
TRACE= ON
TRACEFILE= /var/ct/IW/log/ctsec/ctcasd/trace
TRACELEVELS= _SEC:Info=1,_SEC:Errors=1
TRACESIZE= 1003520
RQUEUESIZE=
MAXTHREADS=
MINTHREADS=
THREADSTACK= 131072
HBA_USING_SSH_KEYS= false
HBA_PRVKEYFILE=
HBA_PUBKEYFILE=
HBA_THLFILE=
HBA_KEYGEN_METHOD= rsa512
HBA_CRED_TIMETOLIVE=90
HBA2_CRED_CTX_LIFETIME= -1
HBA2_CRED_TIMETOLIVE= 300
HBA2_NONCE_FILEMIN=
SERVICES=hba CAS
```

Details:

For more information about the **ctcasd.cfg** file and about using the HBA_CRED_TIMETOLIVE and HBA2_CRED_TIMETOLIVE options, see the *Administering RSCT* guide.

Note that the HBA_CRED_TIMETOLIVE option should never be set on a Hardware Management Console (HMC) device, even if other systems in the cluster have this option set. Leaving the option blank on an HMC will not inject problems into the authentication process.

The values of HBA_CRED_TIMETOLIVE and HBA2_CRED_TIMETOLIVE are not required to be the same. However, most cluster configurations will use the same values for these options because the values are calculated using the same method. A separate option is provided for each security mechanism to allow system administrators to set either a more lenient or a more restrictive expiration time for an individual mechanism.

Verifying the diagnostic:

If the cluster consists of systems using various levels of RSCT and any system within the cluster uses a level of RSCT earlier than 2.3.2.0, it is recommended that the

HBA_CRED_TIMETOLIVE option be left disabled. Consider leaving this option disabled until all systems within the cluster are upgraded to RSCT 2.3.2.0 or greater and proceed to "Procedure 9: checking host name resolution for nodeB" on page 112. Continue with the rest of this test if both systems being tested are using RSCT 2.3.2.0 or greater.

If the HBA_CRED_TIMETOLIVE or HBA2_CRED_TIMETOLIVE options are not set on both systems, no credential life span is being enforced by either the HBA or HBA2 mechanism, respectively, on this system and the credential life span is not injecting any problems into authentication processing. Proceed to "Procedure 9: checking host name resolution for nodeB" on page 112

If either of these options is set, the values should be consistent between the two systems: if one system has these options set, so should the other system, and the values should be the same. Inconsistent setting of these options can inject problems into the authentication processing. The most typical result is that authentication requests succeed when initiated by one of the nodes, but fail when initiated by the other node.

The only exception to this general consistency rule concerns the HBA mechanism and the Hardware Management Console (HMC). HMC devices should never set the HBA_CRED_TIMETOLIVE option, even if the other systems have the option set. Leaving the option blank on an HMC will not inject problems into the authentication process.

For example, the values of HBA_CRED_TIMETOLIVE and HBA2_CRED_TIMETOLIVE are considered to be consistent if both nodes have the following entries for these values:

```
HBA_CRED_TIMETOLIVE=90 HBA2_CRED_TIMETOLIVE=90
```

Make a note of these values because they will be used in "Procedure 7: testing for time-of-day clock skew" on page 109.

However, these values would be considered inconsistent if the entries differed in value on each node in this test. For instance:

```
Value from nodeA:  HBA_CRED_TIMETOLIVE=90
                    HBA2_CRED_TIMETOLIVE=90
Value from nodeB:  HBA_CRED_TIMETOLIVE=180
                    HBA2_CRED_TIMETOLIVE=180
```

In this case, authentication requests for either the HBA or HBA2 mechanism may succeed when *nodeA* initiates the process, but may fail when **nodeB** initiates the process.

However, these values would be considered inconsistent if the entries differed in value on each node in this test. For instance:

```
Value from nodeA:  HBA_CRED_TIMETOLIVE=90
                    HBA2_CRED_TIMETOLIVE=90
Value from nodeB:  HBA_CRED_TIMETOLIVE=180
                    HBA2_CRED_TIMETOLIVE=180
```

In this case, authentication requests for either the HBA or HBA2 mechanism may succeed when *nodeA* initiates the process, but may fail when *nodeB* initiates the process.

The values would also be considered inconsistent if a value was set on one system and not on the other system (assuming the system that has not set this option is not an HMC device). For instance:

```
Value from nodeA:  HBA_CRED_TIMETOLIVE=
                     HBA2_CRED_TIMETOLIVE=
Value from nodeB:  HBA_CRED_TIMETOLIVE=90
                    HBA2_CRED_TIMETOLIVE=90
```

In this case, authentication processing will always succeed for either the HBA or HBA2 mechanism when initiated by *nodeB* because *nodeA* never performs an expiration check. Authentication requests may fail when initiated by *nodeA* if the network is sufficiently slow or the time-of-day clock values on these systems differ by close to 90 seconds.

The HBA_CRED_TIMETOLIVE and HBA2_CRED_TIMETOLIVE values should be set in excess of the expiration time desired. Additional time must be allowed for network latency, processor load factors, and time-of-day clock value differences between the systems.

Note that the default configuration file **/usr/sbin/rsct/bin/ctcasd.cfg** should have no value set for HBA_CRED_TIMETOLIVE and a value of 300 set for HBA2_CRED_TIMETOLIVE. If the default configuration file does not reflect these values, the **ctcasd** configuration has been improperly altered. Consider restoring the original default configuration from the installation media and use the override configuration file **/var/ct/cfg/ctcasd.cfg** to make any local system modifications to this configuration.

Failure actions:

If the HBA_CRED_TIMETOLIVE and HBA2_CRED_TIMETOLIVE values are not consistent between these systems, modify the configurations to make these values consistent.

If the time-of-day clock values of each system within the cluster cannot be reasonably synchronized or if time-of-day clock value drift is a known problem on some cluster systems, consider turning off the HBA_CRED_TIMETOLIVE option or setting the value sufficiently large. The HBA2_CRED_TIMETOLIVE option should not be disabled except on the advice of the IBM Support Center; instead, set the value of this option sufficiently large to account for network latency and time-of-day clock skew. For more information on using these configuration options, see the *Administering RSCT* guide.

Make a note of the values used for the new HBA_CRED_TIMETOLIVE and HBA2_CRED_TIMETOLIVE settings. These values will be needed in "Procedure 7: testing for time-of-day clock skew."

If any modifications to the HBA_CRED_TIMETOLIVE or HBA2_CRED_TIMETOLIVE options are made on a system, stop and restart the **ctcasd** daemon on the node for the configuration change to take effect, as follows:

```
stopsrc -s ctcas
startsrc -s ctcas
```

Next diagnostic procedure:

If the HBA_CRED_TIMETOLIVE or HBA2_CRED_TIMETOLIVE option is enabled for either system, proceed to "Procedure 7: testing for time-of-day clock skew"

If neither of these options are set on both systems, credential expiration is not injecting any problems in the authentication process. Proceed to "Procedure 8: checking for host name resolution to an inactive address" on page 110.

Procedure 7: testing for time-of-day clock skew:

You may need to determine whether time-of-day clock value differences between systems are injecting authentication problems, in configurations where a credential life span is active on one or more of the systems.

Requisite information:

The HBA_CRED_TIMETOLIVE and HBA2_CRED_TIMETOLIVE values verified (or set) as a result of "Procedure 6: verifying that credential expiration checking is active" on page 106.

Instructions:

Using a distributed shell or similar utility, issue simultaneous **date -u** commands on both *nodeA* and *nodeB* to obtain their current time of day in Universal Time Coordinated (UTC) format. For example:

```
dsh -w epsilon3,mimbar date -u
```

If successful, the following command output is displayed:

```
[epsilon3][/] dsh -w epsilon3,mimbar date -u
epsilon3: Wed Oct 29 21:59:43 UTC 2003
mimbar:   Wed Oct 29 21:59:29 UTC 2003
```

Compare any difference in the time of day clocks to the HBA_CRED_TIMETOLIVE and HBA2_CRED_TIMETOLIVE values resulting from "Procedure 6: verifying that credential expiration checking is active" on page 106. The HBA_CRED_TIMETOLIVE and HBA2_CRED_TIMETOLIVE values should be selected using the following general formula:

```
 desired        greatest       network
credential  +  time-of-day  +  latency  +  system
           =  HBA_CRED_TIMETOLIVE
expiration     clock value      time        load
             HBA2_CRED_TIMETOLIVE
  time         difference
```

In the above example output, the HBA_CRED_TIMETOLIVE and HBA2_CRED_TIMETOLIVE values must be set to a value of at least 14 seconds to allow for the time of day clock value differences between the two systems. A value of less than 14 seconds for HBA_CRED_TIMETOLIVE or HBA2_CRED_TIMETOLIVE in this case will result in authentication problems between these two systems.

For more information on using the HBA_CRED_TIMETOLIVE and HBA2_CRED_TIMETOLIVE options and determining their values, see the *Administering RSCT* guide.

Failure actions:

If the distributed shell utility fails, troubleshoot this utility and retry the distributed **date -u** command after the necessary repairs have been made.

Adjust the time of day clocks on the systems to be in closer agreement if their values are too divergent. Time of day clock differences may not only inject authentication problems, but can also cause difficulties in other problem determination efforts. If possible, establish a network time service for the cluster and configure all systems in the cluster to make use of the service.

Adjust the HBA_CRED_TIMETOLIVE value to account for any time of day clock differences, network latency, and system loads. Modify the configurations on each node to use the same HBA_CRED_TIMETOLIVE value. Stop and restart the **ctcasd** daemon on the system where the configuration was adjusted for the change to take effect:

```
stopsrc -s ctcas  startsrc -s ctcas
```

Next diagnostic procedure:

Proceed to "Procedure 8: checking for host name resolution to an inactive address."

Procedure 8: checking for host name resolution to an inactive address:

You may need to determine if a host resolves its host name to an IP address that is not currently active on that host.

Purpose:

To determine if a host resolves its host name to an IP address that is not currently active on that host.

Instructions:

On each host, examine the contents of the **/etc/hosts** file and search for entries that associate the name of the host to an IP address. For the host based authentication mechanisms to function properly, the first such entry must associate the name of the host to an IP address that is currently active on the host.

The **ctsvhbal** command will indicate the host name and the active IP addresses on the host. For example:

```
[epsilon3][/]> ctsvhbal
ctsvhbal: The Host Based Authentication (HBA) mechanism identities
for the local system are:

                 Identity: epsilon3.ialliance.org

                 Identity: 9.194.78.149

ctsvhbal: At least one of the above identities must appear in the
trusted host list on the node where a service application resides
in order for client applications on the local system to authenticate
successfully.
Ensure that at least one host name and one network address identity
from the above list appears in the trusted host list on the service
systems used by applications on this local system.
```

The contents of the **/etc/hosts** file should associate the name of the host to the addresses displayed in the results of the **ctsvhbal** command before they are associated with other addresses not shown in these results.

Example: Based on the results of the **ctsvhbal** command shown above, the following is an acceptable **/etc/hosts** file:

```
127.0.0.1       localhost.localdomain localhost
9.194.78.149    epsilon3.ialliance.org epsilon3
127.0.0.2       epsilon3.ialliance.org epsilon3  <-- Note
                                          address was not
                                        displayed by
                                        ctsvhbal results
                                          above
```

Example: Based on the results of the **ctsvhbal** command shown above, the following **/etc/hosts** file can cause failures for the host based authentication mechanisms. Because the name of the host is first associated with an IP address that is not currently active, attempts by service applications on this host to verify clients using either the HBA or HBA2 mechanisms can fail.

```
127.0.0.1       localhost.localdomain localhost
127.0.0.2       epsilon3.ialliance.org epsilon3  <-- Note
                                          address was not
                                        displayed by
                                        ctsvhbal results
                                          above
9.194.78.149    epsilon3.ialliance.org epsilon3
```

Failure actions:

Modify the **/etc/hosts** file to place any entries that associate the name of the local host with currently inactive IP addresses *after* entries that associate the name with active IP addresses. Shut down and restart the **ctcasd** daemon for the daemon to obtain the revised host name resolution mappings.

Next diagnostic procedure:

If authentication failures persist, proceed to "Procedure 9: checking host name resolution for nodeB."

Procedure 9: checking host name resolution for nodeB:

You may need to determine if host name resolution differences are injecting problems into the initiating phase of the authentication process.

Purpose:

To determine if host name resolution differences are injecting problems into the initiating phase of the authentication process.

Instructions:

On *nodeA*, issue the following command to get the perceived network identity for *nodeB*:

```
/usr/sbin/rsct/bin/ctsvhbar nodeB
```

On *nodeB*, issue the following instruction to obtain the values that *nodeB* would use to verify its own identity:

```
/usr/sbin/rsct/bin/ctsvhbal
```

Verifying the diagnostic:

If the command could not resolve the host name, the following output is displayed:

```
[epsilon3][/]> ctsvhbar mimbar
          Host name or network address: mimbar
          Fully qualified host name
               used for authentication: [Cannot determine
                                         host name]
```

Verify that the correct host name was used as an argument to the **ctsvhbar** command. If the correct name was used, the host is not known to either the local system's host name resolution files, or it is not known to the network domain name services. This will cause problems in the authentication process.

Successful output from the **ctsvhbar** command is similar to the following:

```
[epsilon3][/]> ctsvhbar mimbar
          Host name or network address: mimbar
          Fully qualified host name
               used for authentication: mimbar.ialliance.org
```

Successful output from the **ctsvhbal** command is similar to the following:

```
[mimbar][/]> ctsvhbal
ctsvhbal: The Host Based Authentication (HBA) mechanism
identities for the local system are:

                Identity:  mimbar.ialliance.org

                Identity:  9.194.78.145
```

```
ctsvhbal: At least one of the above identities must appear in the
trusted host list on the node where a service application resides
in order for client applications on the local system to
authenticate successfully.
Ensure that at least one host name and one network address identity
from the above list appears in the trusted host list on the service
systems used by applications on this local system.
```

The fully qualified host name obtained for *nodeB* in the **ctsvhbar** command output from *nodeA* must match exactly to one of the identities displayed for *nodeB* in the **ctsvhbal** command output. In the above examples of successful outputs, an exact match is found for the host identity value `mimbar.ialliance.org` .

In the following example, an exact match is **not** found, which would indicate that host name resolution can inject problems into the authentication process:

```
[epsilon3][/]> ctsvhbar mimbar
          Host name or network address: mimbar
          Fully qualified host name
               used for authentication: mimbar

[mimbar][/]> ctsvhbal
ctsvhbal: The Host Based Authentication (HBA) mechanism
identities for the local system are:

                 Identity:  mimbar.ialliance.org

                 Identity:  9.194.78.145

ctsvhbal: At least one of the above identities must appear in the
trusted host list on the node where a service application resides
in order for client applications on the local system to
authenticate successfully.
Ensure that at least one host name and one network address identity
from the above list appears in the trusted host list on the service
systems used by applications on this local system.
```

Note that in this example, an exact match is not found. A match on the shortened version of the host name is insufficient, and can cause problems in the authentication process.

Failure actions:

If *nodeA* is unable to resolve the name for *nodeB*, modify either the network domain name services or the host definition files on *nodeA* to include the host name for *nodeB*.

If *nodeA* obtains a different name for *nodeB* than *nodeB* obtains for itself, host name resolution is inconsistent between the nodes and must be repaired.

For assistance in both efforts, see “Error symptoms, responses, and recoveries” on page 123.

Next diagnostic procedure:

Proceed to “Procedure 10: checking host name resolution for nodeA.”

Procedure 10: checking host name resolution for nodeA:

You may need to determine if host name resolution differences are injecting problems into the mutual authentication phase of the authentication process.

Purpose:

To determine if host name resolution differences are injecting problems into the mutual authentication phase of the authentication process.

Instructions:

This test reverses the instructions from "Procedure 9: checking host name resolution for nodeB" on page 112

On *nodeB*, issue the following command to get the perceived network identity for *nodeA*:

```
/usr/sbin/rsct/bin/ctsvhbar nodeA
```

On *nodeA*, issue the following instruction to obtain the values that *nodeA* would use to verify its own identity:

```
/usr/sbin/rsct/bin/ctsvhbal
```

Verifying the diagnostic:

If the command could not resolve the host name, the following output is displayed:

```
[mimbar][/]> ctsvhbar epsilon3
          Host name or network address: epsilon3
          Fully qualified host name
               used for authentication: [Cannot determine
                                         host name]
```

Verify that the correct host name was used as an argument to the **ctsvhbar** command. If the correct name was used, the host is not known to either the local system's host name resolution files, or it is not known to the network domain name services. This will cause problems in the authentication process.

Successful output from the **ctsvhbar** command is similar to the following:

```
[mimbar][/]> ctsvhbar epsilon3
          Host name or network address: epsilon3
          Fully qualified host name
               used for authentication: epsilon3.ialliance.org
```

Successful output from the **ctsvhbal** command is similar to the following:

```
[epsilon3][/]> ctsvhbal
ctsvhbal: The Host Based Authentication (HBA) mechanism
identities for the local system are:

                 Identity:  epsilon3.ialliance.org

                 Identity:  9.194.78.149

ctsvhbal: At least one of the above identities must appear in the
trusted host list on the node where a service application resides
in order for client applications on the local system to
authenticate successfully.
Ensure that at least one host name and one network address
identity from the above list appears in the trusted host list on
the service systems used by applications on this local system.
```

The fully qualified host name obtained for *nodeA* in the **ctsvhbar** command output from *nodeB* must match exactly to one of the identities displayed for *nodeA* in the **ctsvhbal** command output. In the above examples of successful outputs, an exact match is found for the host identity value `epsilon3.ialliance.org` .

In the following example, an exact match is not found, which would indicate that host name resolution can inject problems into the authentication process:

```
[mimbar][/]> ctsvhbar epsilon3
          Host name or network address: epsilon3
          Fully qualified host name
               used for authentication: epsilon3

[epsilon3][/]> ctsvhbal
ctsvhbal: The Host Based Authentication (HBA) mechanism
identities for the local system are:

                 Identity:  epsilon3.ialliance.org

                 Identity:  9.194.78.149

ctsvhbal: At least one of the above identities must appear in the
trusted host list on the node where a service application resides
in order for client applications on the local system to
authenticate successfully.
Ensure that at least one host name and one network address
identity from the above list appears in the trusted host list on
the service systems used by applications on this local system.
```

Note that in this example, an exact match is not found. A match on the shortened version of the host name is insufficient, and can cause problems in the authentication process.

Failure actions:

If *nodeB* is unable to resolve the name for *nodeA*, modify either the network domain name services or the host definition files on *nodeB* to include the host name for *nodeA*.

If *nodeA* obtains a different name for *nodeB* than *nodeB* obtains for tself, host name resolution is inconsistent between the nodes and must be repaired.

For assistance in both efforts, see "Error symptoms, responses, and recoveries" on page 123.

Next diagnostic procedure:

If host name resolution appears consistent between *nodeA* and *nodeB*, no further procedures are necessary. Consider troubleshooting the management domain or peer domain configuration to ensure that the two systems are members of the same cluster configuration. Consider troubleshooting the RMC authorization facility to ensure that the appropriate users from *nodeA* are granted the necessary permissions on *nodeB* if RMC commands or applications such as Cluster Systems Management (CSM) are unable to function properly.

If host name resolution appears inconsistent between *nodeA* and *nodeB*, proceed to "Procedure 11: verifying domain name service setup."

Procedure 11: verifying domain name service setup:

You may need to ensure that the security library can resolve host IP addresses and names to the correct host name equivalent.

Purpose:

To ensure that the security library can resolve host IP addresses and names to the correct host name equivalent.

The host based authentication mechanism associates public keys to host names. Host name resolution must be consistent, or authentication attempts can fail.

Instructions:

Examine the **/etc/resolv.conf** file on each systems to determine if any name servers have been set up for these systems. If a name server has been established, and entry with the label `nameserver` will appear at least once within this file.

Verifying the diagnostic:

Using a text file viewer, examine the **/etc/resolv.conf** file and search for `nameserver` entries. It is not necessary for a node to have established a name server for host name resolution, but make note of any host names or addresses if a name server is specified. These names will be used in "Procedure 13: verifying access to the domain name servers" on page 117

Failure actions:

It is not necessary for a node to have established a name server for host name resolution. However, it is likely that if any one host within a cluster configuration makes use of a domain name server, the rest of the systems should also be making use of the domain name server. If one system makes use of a name server and the other does not, or if the systems use differing name servers, this may cause inconsistent results in host name resolution on these two systems, leading to problems in the authentication process. Modify the system configurations to use the same name server, or to not use any name server. Keep in mind that if neither host uses a name server, the host will have to record all the host names that it requires in its local host configuration files.

Next diagnostic procedure:

Proceed to "Procedure 12: verifying host name resolution order"

Procedure 12: verifying host name resolution order:

You may need to ensure that the security library can resolve host IP addresses and names to the correct host name equivalent.

Purpose:

To ensure that the security library can resolve host IP addresses and names to the correct host name equivalent. The host based authentication mechanism associates public keys to host names. Host name resolution must be consistent, or authentication attempts can fail.

Instructions:

Check if both systems specify the name resolution order through the configuration files **/etc/irc.conf** or **/etc/netsvc.conf** . Neither of these files should exist if a name server entry was not found on the local host in "Procedure 11: verifying domain name service setup" on page 115. If neither of these files exist, the host is using the default name resolution order. Otherwise, note the order of name resolution as specified in these files.

Verifying the diagnostic:

If a name server entry was not found while performing "Procedure 11: verifying domain name service setup" on page 115, ensure that neither the **/etc/netsvc.conf** nor the **/etc/irc.conf** file exists on either system.

Both systems should make use of a consistent ordering scheme. The files used in the ordering scheme differ between AIX, Linux, and Solaris systems, but the same general resolution scheme should be used. If *nodeA* resolves host names by first examining local host configuration files and then checking through the domain name services, *nodeB* should behave in the same manner. If both systems use differing host name resolution schemes, each system may resolve the same host name to a different value, which will inject problems into the authentication process.

Failure actions:

If a name server is not specified but either the **/etc/netsvc.conf** or the **/etc/irc.conf** files exist, the system may have an incorrect network configuration. Troubleshoot the system's network configuration to make sure it is correct.

If a name server is in use, the **/etc/netsvc.conf** or the **/etc/irc.conf** files should be in place on both systems, and should specify the same host resolution order scheme for both systems. If both systems do not use a consistent host resolution order, update the configuration on these systems to make use of a consistent host resolution order.

Next diagnostic procedure:

If a name server is not configured for either system, no further procedures are necessary. Consider troubleshooting the management domain or peer domain configuration to ensure that the two systems are members of the same cluster configuration. Consider troubleshooting the RMC authorization facility to ensure that the appropriate users from *nodeA* are granted the necessary permissions on *nodeB* if RMC commands or applications such as Cluster Systems Management (CSM) are unable to function properly.

If a name server is configured for at lest one of the systems, proceed to "Procedure 13: verifying access to the domain name servers."

Procedure 13: verifying access to the domain name servers:

You may need to ensure that the security library can resolve host IP addresses and names to the correct host name equivalent through a name server.

Purpose:

To ensure that the security library can resolve host IP addresses and names to the correct host name equivalent through a name server.

The inability to contact a domain name server can inject significant performance degradation to the host based authentication mechanism, and can inject problems into the authentication process.

Instructions:

If the cluster nodes are not making use of name servers, skip this procedure. Verify that both *nodeA* and *nodeB* can access the name servers discovered in "Procedure 11: verifying domain name service setup" on page 115 by issuing a **ping** command from each system to the name servers. For example:

```
ping -c1 9.199.1.1
ping -c1 129.90.77.1
```

Verifying the diagnostic:

If the name server can be reached, you will get results similar to the following:

```
PING 9.114.1.1: (9.199.1.1): 56 data bytes
64 bytes from 9.199.1.1:icmp_seq=0 ttl=253 time=1 ms

----9.199.1.1 PING Statistics----
1 packets transmitted, 1 packets received, 0% packet loss round-trip
min/avg/max = 1/1/1 ms
```

If the name server cannot be reached, an error message will be displayed:

```
PING 9.114.1.1: (9.199.1.1): 56 data bytes

----9.199.1.1 PING Statistics----
1 packets transmitted, 0 packets received, 100% packet loss
```

Failure actions:

Verify that the correct name or address is being used for the domain name server. Troubleshoot the network connectivity between any failing node and the name server. Consider changing to a backup or alternate name server.

Next diagnostic procedure:
None.

Authorization troubleshooting procedures

There are several procedures for troubleshooting authentication problems.

Identity mapping troubleshooting procedures:

The cluster security services identity mapping facility permits administrators to associate an operating system user identity on the local system to a security network identity. Future versions of the cluster security services library will permit group based authorization making use of such mapped identities.

Procedure 1: verifying the default global mapping file:

It may become necessary to verify that the cluster security services library can locate the correct identity mapping definition files for the local system.

Purpose:
To verify that the cluster security services library can locate the correct identity mapping definition files for the local system. Two input files are supported: a global mapping file intended to contain identity maps for network identities that are intended to be consistent throughout the cluster; and a local mapping file that defines identity maps intended to be used on the local node alone. The local definition file resides in the file **/var/ct/cfg/ctsec_map.local**. A default global definition file is shipped with RSCT in the file **/usr/sbin/rsct/cfg/ctsec_map.global**. If system administrators wish to extend the contents of this file, the file should be copied to its override position of **/var/ct/cfg/ctsec_map.global** and modifications made to that version of the file.

Instructions:
Test for the presence of the default global identity map file:

```
file /usr/sbin/rsct/cfg/ctsec_map.global
```

Verifying the diagnostic:
On AIX nodes, the following output is displayed:

```
/usr/sbin/rsct/cfg/ctsec_map.global: commands text
```

On Linux and Solaris nodes, the following output is displayed:

```
/usr/sbin/rsct/cfg/ctsec_map.global: ASCII text
```

Failure actions:
Restore the default global map definition file from either a system backup or from the RSCT installation media.

Next diagnostic procedure:
Proceed to "Procedure 2: verifying the override global mapping file."

Procedure 2: verifying the override global mapping file:

It may become necessary to verify that the cluster security services library can locate the correct identity mapping definition files for the local system.

Purpose:
To verify that the cluster security services library can locate the correct identity mapping definition files for the local system. Two input files are supported: a global mapping file intended to contain identity maps for network identities that are intended to be consistent throughout the cluster; and a local mapping file that defines identity maps intended to be used on the local node alone. The local definition file resides in the file **/var/ct/cfg/ctsec_map.local**. A default global

definition file is shipped with RSCT in the file **/usr/sbin/rsct/cfg/ctsec_map.global** . If system administrators wish to extend the contents of this file, the file should be copied to its override position of **/var/ct/cfg/ctsec_map.global** and modifications made to that version of the file.

Instructions:

Test for the presence of the override global identity map file:

```
file /var/ct/cfg/ctsec_map.global
```

Verifying the diagnostic:

The absence of an override global identity map file does not necessarily constitute a failure condition. On AIX nodes, if the file is present, the following output is displayed:

```
/var/ct/cfg/ctsec_map.global: commands text
```

On Linux nodes, if the file is present, the following output is displayed:

```
/var/ct/cfg/ctsec_map.global: ASCII text
```

Next diagnostic procedure:

Proceed to "Procedure 3: verifying the local mapping file."

Procedure 3: verifying the local mapping file:

It may become necessary to verify that the cluster security services library can locate the correct identity mapping definition files for the local system.

Purpose:

To verify that the cluster security services library can locate the correct identity mapping definition files for the local system. Two input files are supported: a global mapping file intended to contain identity maps for network identities that are intended to be consistent throughout the cluster; and a local mapping file that defines identity maps intended to be used on the local node alone. The local definition file resides in the file **/var/ct/cfg/ctsec_map.local**. A default global definition file is shipped with RSCT in the file **/usr/sbin/rsct/cfg/ctsec_map.global** . If system administrators wish to extend the contents of this file, the file should be copied to its override position of **/var/ct/cfg/ctsec_map.global** and modifications made to that version of the file.

Instructions:

Test for the presence of the local identity map file:

```
file /var/ct/cfg/ctsec_map.local
```

Verifying the diagnostic:

The absence of an override global identity map file does not necessarily constitute a failure condition.

On AIX nodes, if the file is present, the following output is displayed:

```
/var/ct/cfg/ctsec_map.local: commands text
```

On Linux nodes, if the file is present, the following output is displayed:

```
/var/ct/cfg/ctsec_map.local: ASCII text
```

Next diagnostic procedure:

Proceed to "Procedure 4: checking the mapping for a network identity on a node" on page 120.

Procedure 4: checking the mapping for a network identity on a node:

It may become necessary to verify that the cluster security services library will find the correct local user map for a network identity.

Purpose:

To verify that the cluster security services library will find the correct local user map for a network identity.

Instructions:

Select a network identity from a specific security mechanism supported by cluster security services. Examine the cluster security services configuration file –**/usr/sbin/rsct/cfg/ctsec.cfg** or **/var/ct/cfg/ctsec.cfg** – to determine the correct mnemonic to be used for that security mechanism. Provide both the network identity and the security mnemonic as arguments to the **ctsidmck** command.

Example: To test the mapping for the Host Based Authentication (HBA) network identity `zathras@epsilon3.org`, enter:

```
ctsidmck -dm -munix zathras@epsilon3.org
```

To test the mapping for the Enhanced Host Based Authentication (HBA2) network identity `ranger1@ialliance.gov`, enter:

```
ctsidmck -dm -mhba2 ranger1@ialliance.gov
```

Result: The **ctsidmck** command displays any map that was obtained as well as the mapping file entry that resulted in the map.

Verifying the diagnostic:

Verify that the resulting map – if any – was the intended mapping for the network identifier.

Failure actions:

If a mapping was intended and not found, extend the identity mapping definition files to include a mapping entry to form this mapping. Add the definition either to the local definition file (if the map is intended for this node only) or the override version of the global mapping file (if the map is intended to eventually be used on all nodes within the cluster). Do **not** make modifications to the default global identity mapping definition file **/usr/sbin/rsct/cfg/ctsec_map.global**. After making the necessary modifications, repeat "Procedure 4: checking the mapping for a network identity on a node" to ensure that the correct modifications were made. .

Next diagnostic procedure:

If a mapping was intended and an incorrect mapping was displayed, proceed to "Procedure 6: adding mapping definitions" on page 121.

If a mapping was not intended an a map was found, proceed to "Procedure 5: modifying incorrect mapping definitions."

Procedure 5: modifying incorrect mapping definitions:

It may become necessary to ensure that a local operating system user identity map is not granted to a network identity that should not receive such a map.

Purpose:

To ensure that a local operating system user identity map is not granted to a network identity that should not receive such a map.

Instructions:

Find the mapping definition file that specifies the rule in error that was displayed in "Procedure 4: checking the mapping for a network identity on a node." For example, if that procedure

indicated that the rule `*@epsilon3.org=draal` mapped `zathras@epsilon3.org` to `draal`, issue the following command to locate the file that specifies this rule:

```
grep -l "*@epsilon3.org=draal" \
/usr/sbin/rsct/cfg/ctsec_map.global \
/var/ct/cfg/ctsec_map.global \
/var/ct/cfg/ctsec_map.local
```

This command will display the name of the file that contains the rule. Modify this file using a text editor to correct the mapping rule to yield the correct result.

Verifying the diagnostic:

Return to "Procedure 4: checking the mapping for a network identity on a node" on page 120 and repeat the test.

Next diagnostic procedure:

None.

Procedure 6: adding mapping definitions:

It may become necessary to ensure that a local operating system user identity map is granted to a network identity that should receive it.

Purpose:

To ensure that a local operating system user identity map is granted to a network identity that should receive it.

Instructions:

Determine whether the identity mapping is unique to the local node, or will apply to all nodes within the cluster configuration.

- If the mapping is intended to be used only on this node, ensure that the local mapping definition file **/var/ct/cfg/ctsec_map.local** exists. If not, issue the following commands to bring it into being:

  ```
  touch /var/ct/cfg/ctsec_map.local
  chmod 644 /var/ct/cfg/ctsec_map.local
  ```

- If the mapping is intended to be used on all nodes within the cluster configuration, ensure that the override global mapping file **/var/ct/cfg/ctsec_map.global** exists. If not, issue the following command to bring it into being:

  ```
  cp /usr/sbin/rsct/cfg/ctsec_map.global \
     /var/ct/cfg/ctsec_map.global
  ```

  Using a text editor, modify the correct file to include a mapping rule to yield the desired map. Remember, order is important within these files. The interactions of new rules with existing rules must be considered carefully. For more information, see the entries for the **ctsec_map.global** and **ctsec_map.local** files in *Technical Reference: RSCT for AIX* or *Technical Reference: RSCT for Multiplatforms* guides.

Verifying the diagnostic:

Return to "Procedure 4: checking the mapping for a network identity on a node" on page 120 and repeat the test.

Next diagnostic procedure:

Proceed to "Procedure 7: checking for an alternate authorization mechanism in use" on page 122.

Procedure 7: checking for an alternate authorization mechanism in use:

It may become necessary to determine if the security services are using the expected mechanism pluggable module (MPM) to authorize a user.

Purpose:

To determine if the security services are using the expected mechanism pluggable module (MPM) to authorize a user.

Beginning in RSCT version 2.3.10.0 and 2.4.6.0, cluster security services allows the system administrator to specify an *alternate authorization mechanism* to be used for all authorization processing for parties that are authenticated using a specific mechanism. This feature allows cluster security services to *authenticate* a party using one security mechanism and then to *authorize* the same party using a different security mechanism.

Alternate authorization mechanisms are specified in the cluster security services configuration file: **/var/ct/cfg/ctsec.cfg** or **/usr/sbin/rsct/cfg/ctsec.cfg** . If an authentication mechanism is configured to use an alternate mechanism for authorization, the entry for that mechanism will contain **z[***mnemonic***]** flag in its entry, where *mnemonic* is the mnemonic for the mechanism to be used for authorization. For example, the default cluster security services configuration for RSCT 2.4.6.0 is configured to use the Host Based Authentication (HBA) mechanism for authorizing any parties authenticated through the Enhanced Host Based Authentication (HBA2) mechanism, as follows:

```
#Prior Mnemonic Code    Path                Flags
#-------------------------------------------------------
   1    unix     0x00001 /usr/lib/unix.mpm   i
   2    hba2     0x00002 /usr/lib/hba2.mpm   iz[unix]
```

Note that the default configuration does not use an alternate authorization mechanism for the Host Based Authentication (HBA) mechanism `unix`; parties authenticated through the HBA mechanism will also be authorized using that mechanism.

Instructions:

Determine the network identity of the party that is being denied access and the security mechanism that is being used to authenticate the party. Identify the service application that is denying the client that access. On the node where the service application executes, examine the cluster security service's configuration file – **/var/ct/cfg/ctsec.cfg** or **/usr/sbin/rsct/cfg/ctsec.cfg** – to determine if an alternate authorization mechanism is in use for the authentication mechanism. An alternate authorization mechanism is indicated by the **z[***mnemonic***]** flag for that security mechanism's entry.

After obtaining this information, examine the access controls for the service application. If no alternate authorization mechanism was specified in the cluster security services configuration file, the identity of the party should be listed in the access controls for the authentication mechanism. If an alternate authorization mechanism was specified in the configuration file, the identity of the party should be listed in the access controls for the alternate authorization mechanism.

Example: Consider the case where a service application on *nodeB* denies access to the Enhanced Host Based Authentication (HBA2) identity `zathras@epsilon3.org`. Examining the cluster security services configuration files on *nodeB* shows that the HBA2 mechanism on *nodeB* is using the Host Based Authentication (HBA) mechanism as the alternate authorization mechanism:

```
#Prior Mnemonic Code    Path                Flags
#-------------------------------------------------------
   1    unix     0x00001 /usr/lib/unix.mpm   i
   2    hba2     0x00002 /usr/lib/hba2.mpm   iz[unix]
```

In order for the service application on this node to grant access to its resources to any users authenticated through the HBA2 mechanism, the user identities need to be listed in its access controls as HBA identities, not HBA2 identities. The service application's access controls should be checked to see if they list the user `zathras@epsilon3.org` in the "unix" mechanism section and, if this entry is missing, the service application administrator should consider adding an entry for that user to grant the user access.

Example: Now consider the case where a service application on *nodeB* denies access to the Host Based Authentication (HBA) identity `zathras@epsilon3.org`. Examining the cluster security services configuration files on *nodeB* shows that the HBA mechanism on *nodeB* is not using an alternate authorization mechanism:

```
#Prior Mnemonic Code    Path                Flags
#-----------------------------------------------------
1    unix     0x00001 /usr/lib/unix.mpm   i
2    hba2     0x00002 /usr/lib/hba2.mpm   iz[unix]
```

In order for the service application on this node to grant access to its resources to any users authenticated through the HBA mechanism, the user identities need to be listed in its access controls as HBA identities. The service application's access controls should be checked to see if they list the user `zathras@epsilon3.org` in the "unix" mechanism section and, if this entry is missing, the service application administrator should consider adding an entry for that user to grant the user access.

Details:

For more information about the cluster security services configuration file and using alternate authorization mechanisms, see the *Administering RSCT* guide.

Failure actions:

If an alternate authorization mechanisms is used for a specific security mechanism, ensure that network identities for the security mechanism using the alternate authorization mechanism are listed in the access controls for the service application using the alternate authorization mechanism, instead of using the mechanism that was used to authenticate the party.

If an alternate authorization mechanisms is *not* used for a specific security mechanism, ensure that network identities for the security mechanism are listed in the access controls for the service application using that same mechanism.

Next diagnostic procedure:

None.

Error symptoms, responses, and recoveries

Use this information to diagnose problems with cluster security services. Locate the symptom and perform the specified recovery action.

Use the information in Table 20 to diagnose problems with cluster security services. Locate the symptom and perform the specified action.

Table 20. Cluster security services error conditions and recovery actions

Error condition	Action
Private or public key file missing on a node	“Action 1: correct host-based authentication configuration errors” on page 124
Private and public key files mismatch on a node	“Action 1: correct host-based authentication configuration errors” on page 124
ctcasd daemon abnormally terminates	“Action 2: identify, rectify, or report ctcasd daemon failures” on page 126
Cannot add entries to Trusted Host List File	“Action 3: compress the trusted host list file” on page 127
Trusted Host List File size too large	“Action 3: compress the trusted host list file” on page 127

Table 20. Cluster security services error conditions and recovery actions (continued)

Error condition	Action
Authentication Failures	"Action 4: identify the cause of authentication-related failures" on page 130
Host Name Resolution and Short Host Name Support	"Action 5: set consistent host name resolution" on page 131
Private key becomes compromised	"Action 6: recover from a security breach" on page 131
Trusted Host List on local node must be reset because it is missing or incorrectly populated	"Action 7: create an initial trusted host list" on page 132

Action 1: correct host-based authentication configuration errors

Generate new private and public keys to correct host-based authentication mechanism configuration errors. Take this action when a private or public key file is missing or when there is a private and public key mismatch on a node.

Description:

Used to correct Host Based Authentication mechanism configuration errors where one of the necessary key files is missing, or to recover from a mismatch between the node's private and public keys. This action involves the generation of new private and public keys.

Repair action:

Follow these steps:

1. Log onto the local system as **root**.
2. Shut down all trusted services on the local node.
3. On each node within the cluster configuration (including the local node), remove the public key for this node from the Trusted Host List files on these nodes using the **ctsthl -d** command. Be sure to remove all entries for every name and IP address that can be used by this node.
4. Remove the trusted host list from this node.
5. On the local node, determine the parameters for private and public keys on the node. Examine the Host Based Authentication configuration file – **/var/ct/cfg/ctcasd.cfg** or **/usr/sbin/rsct/cfg/ctcasd.cfg** – and find the values for the following entries:

   ```
   HBA_PRVKEYFILE
   HBA_PUBKEYFILE
   HBA_KEYGEN_METHOD
   ```

   If no explicit values are provided for these entries, the defaults used by the **ctcasd** daemon are:

   ```
   HBA_PRVKEYFILE=/var/ct/cfg/ct_has.qkf
   HBA_PUBKEYFILE=/var/ct/cfg/ct_has.pkf
   HBA_KEYGEN_METHOD=rsa512
   ```

6. Issue the **ctskeygen -n -d** command to create new private and public keys for the local node and store them in the appropriate files. The command will display the new public key value to standard output, so redirect standard output to a file. The new key value will be needed in later steps. If the default **ctcasd** settings are used by the configuration file, issue the command:

   ```
   ctskeygen -n -mrsa512 -p/var/ct/cfg/ct_has.pkf \
       -q/var/ct/cfg/ct_has.qkf -l > /tmp/pubk.out
   ```

7. See "Action 7: create an initial trusted host list" on page 132 to reset the contents of a trusted host list. Proceed to Step 8 below when that action is complete.
8. Manually distribute the new public key to the cluster nodes. For information on how to do this, see the *Administering RSCT* guide. The key was stored in **/tmp/pubk.out** in Step 6.

9. Restart the trusted services on the local node.
10. Remove the temporary file created in Step 6.
11. Log off from the node.

Repair test:

Perform the troubleshooting procedures for the Host Based Authentication mechanism listed earlier in this section to validate the repair.

Recovery action:

Read this paragraph in its entirety. A recovery action exists that can help avoid triggering failures related to private and public key mismatches. This recovery action will *disable* the Host Based Authentication (HBA) mechanism and the Enhanced Host Based Authentication (HBA2) mechanism on the local node. Applications on the local node will not be able to authenticate with other applications using either the HBA or HBA2 mechanisms. If no other mechanism is available, then *all applications on the local node will be unauthenticated* if this recovery action is taken. Do not use this recovery action if this solution is not acceptable.

1. Log on to the node as **root**.
2. Shut down all trusted services on the node.
3. If an override for the cluster security services configuration file does not exist in the file **/var/ct/cfg/ctsec.cfg**, create this file using the following command:

   ```
   cp /usr/sbin/rsct/cfg/ctsec.cfg  /var/ct/cfg/ctsec.cfg
   ```

4. Using a text editor, insert a comment character (#) at the start of the entries for the Host Based Authentication and Enhanced Host Based Authentication mechanisms, as follows:

   ```
   #Prior Mnemonic Code    Path               Flags
   #-------------------------------------------------------
   # 1    unix     0x00001 /usr/lib/unix.mpm  i
   # 2    hba2     0x00002 /usr/lib/hba2.mpm  iz[unix]
   ```

5. Restart the trusted services on this node
6. Log off the node.

Recovery removal:

To remove the above recovery action:

1. Log on to the node as **root**.
2. Shut down all trusted services on the node.
3. Using a text editor, edit the override cluster security services configuration file **/var/ct/cfg/ctsec.cfg**. Delete the comment character (#) from the start of the entries for the Host Based Authentication and Enhanced Host Based Authentication mechanisms:

   ```
   #Prior Mnemonic Code    Path               Flags
   #-------------------------------------------------------
   1     unix     0x00001 /usr/lib/unix.mpm  i
   2     hba2     0x00002 /usr/lib/hba2.mpm  iz[unix]
   ```

4. Compare the override configuration file to the default configuration file using the **diff** command:

   ```
   diff /var/ct/cfg/ctsec.cfg  /usr/sbin/rsct/cfg/ctsec.cfg
   ```

5. If the files are not different, remove the override file **/var/ct/cfg/ctsec.cfg** from this system; it is no longer required.
6. Restart the trusted services on this node.
7. Log off the node.

Action 2: identify, rectify, or report ctcasd daemon failures

It may become necessary to identify, rectify, or report failures in the **ctcasd** daemon. Take this action when the **ctcasd** daemon abnormally terminates.

Description:

Used to identify, rectify, or report failures in the **ctcasd** daemon.

Repair action:

Examine the AIX Error Log (on AIX nodes) or the System Log (on Linux or Solaris nodes) for any entries made by the **ctcasd** daemon. Consult the earlier section on "Error information" on page 70 for assistance in locating these entries. Perform any recommended actions indicated in the entry for the failure condition.

Repair test:

Restart the **ctcasd** daemon. If the daemon will not restart or stay operational, examine the AIX Error Log (on AIX nodes) or the System Log (on Linux, or Solaris nodes) for any new failure records recorded by the daemon. Contact the IBM Support Center for assistance if the problem cannot be rectified on site.

Recovery action:

Read this paragraph in its entirety. A recovery action exists that can help avoid triggering failures related to private and public key mismatches. This recovery action will *disable* the Host Based Authentication (HBA) mechanism and the Enhanced Host Based Authentication (HBA2) mechanism on the local node. Applications on the local node will not be able to authenticate with other applications using either the HBA or HBA2 mechanisms. If no other mechanism is available, then *all applications on the local node will be unauthenticated* if this recovery action is taken. Do not use this recovery action if this solution is not acceptable.

1. Log on to the node as **root**.
2. Shut down all trusted services on the node.
3. If an override for the cluster security services configuration file does not exist in the file **/var/ct/cfg/ctsec.cfg**, create this file using the following command:

   ```
   cp /usr/sbin/rsct/cfg/ctsec.cfg  /var/ct/cfg/ctsec.cfg
   ```

4. Using a text editor, insert a comment character (#) at the start of the entries for the Host Based Authentication and Enhanced Host Based Authentication mechanisms, as follows:

   ```
   #Prior Mnemonic Code    Path               Flags
   #-------------------------------------------------------
   # 1    unix     0x00001 /usr/lib/unix.mpm  i
   # 2    hba2     0x00002 /usr/lib/hba2.mpm  iz[unix]
   ```

5. Restart the trusted services on this node
6. Log off the node.

Recovery removal:

To remove the preceding recovery action:

1. Log on to the node as **root**.
2. Shut down all trusted services on the node.
3. Using a text editor, edit the override cluster security services configuration file **/var/ct/cfg/ctsec.cfg**. Delete the comment character (#) from the start of the entries for the Host Based Authentication and Enhanced Host Based Authentication mechanisms:

   ```
   #Prior Mnemonic Code    Path               Flags
   #-------------------------------------------------------
   1     unix     0x00001 /usr/lib/unix.mpm  i
   2     hba2     0x00002 /usr/lib/hba2.mpm  iz[unix]
   ```

4. Compare the override configuration file to the default configuration file using the **diff** command:

   ```
   diff /var/ct/cfg/ctsec.cfg  /usr/sbin/rsct/cfg/ctsec.cfg
   ```

5. If the files are not different, remove the override file **/var/ct/cfg/ctsec.cfg** from this system; it is no longer required.
6. Restart the trusted services on this node.
7. Log off the node.

Action 3: compress the trusted host list file

It may become necessary to compress the file space used by the Host Based Authentication mechanism's trusted host list file. Take this action when you cannot add entries to Trusted Host List File or when the Trusted Host List File size is too large.

Description:

Used to compress the file space used by the Host Based Authentication mechanism's trusted host list file.

Repair action:

Perform the following steps:

1. Select a time when system activity is low, and RMC clients will not be attempting to authenticate to the RMC subsystem.
2. Log onto the system as **root**.
3. Examine the Host Based Authentication mechanism configuration file –**/usr/sbin/rsct/cfg/ctcasd.cfg** or **/var/ct/cfg/ctcasd.cfg** – to determine what file is being used as the trusted host list file. This value is given in the following entry:

   ```
   HBA_THLFILE
   ```

   If no value is given for this entry, the default file location of **/var/ct/cfg/ct_has.thl** is in use. Make note of the correct file name; it will be required in subsequent steps of this action.
4. Issue the following command to compress the contents of the trusted host list file:

   ```
   /usr/sbin/rsct/bin/ctsthl -z -f trusted_host_list_file
   ```

   If this command completes successfully, then the repair action is complete. You do not need to perform the remaining steps for this action.

   The **ctsthl -z** command option may not exist on older versions of RSCT. If your system does not support this command option, proceed to Step 5 of this action.
5. Copy the trusted host list file to a backup. For example:

   ```
   cp /var/ct/cfg/ct_has.thl /var/ct/cfg/ct_has.thl.orig
   ```

6. Display the current contents of the trusted host list file, redirecting the output to a file. This file will be used to verify the actions of a shell script used in the subsequent steps. For example:

   ```
   /usr/sbin/rsct/bin/ctsthl -l -f /var/ct/cfg/ct_has.thl >\
        /tmp/thlorig.out
   ```

   The contents of this file will be similar to the following example:

   ```
   ------------------- Host name: avenger.pok.ibm.com
   Identifier Generation Method: rsa1024
   Identifier Value:
   120400a25e168a7eafcbe44fde48799cc3a88cc177019100
   ```

```
09587ea7d9af5db90f29415db7892c7ec018640eaae9c6bd
a64098efaf6d4680ea3bb83bac663cf340b5419623be80ce
977e153576d9a707bcb8e8969ed338fd2c1df4855b233ee6
533199d40a7267dcfb01e923c5693c4230a5f8c60c7b8e67
9eb313d926beed115464cb0103
--------------------
Host name: ppsclnt16.pok.ibm.com
Identifier Generation Method: rsa1024
Identifier Value:
120400a25e168a7eafcbe44fde48799cc3a88cc177019100
09587ea7d9af5db90f29415db7892c7ec018640eaae9c6bd
a64098efaf6d4680ea3bb83bac663cf340b5419623be80ce
977e153576d9a707bcb8e8969ed338fd2c1df4855b233ee6
533199d40a7267dcfb01e923c5693c4230a5f8c60c7b8e67
9eb313d926beed115464cb0103
--------------------
Host name: sh2n04.pok.ibm.com
Identifier Generation Method: rsa1024
Identifier Value:
120400a25e168a7eafcbe44fde48799cc3a88cc177019100
09587ea7d9af5db90f29415db7892c7ec018640eaae9c6bd
a64098efaf6d4680ea3bb83bac663cf340b5419623be80ce
977e153576d9a707bcb8e8969ed338fd2c1df4855b233ee6
533199d40a7267dcfb01e923c5693c4230a5f8c60c7b8e67
9eb313d926beed115464cb0103
--------------------
```

7. Copy this file to a new file. This new file will be used as the shell script to clean up the trusted host list file. For example:

   ```
   cp /tmp/thlorig.out /tmp/cleanthl
   ```

8. Select a name for a new trusted host list file. This is going to be the "compressed" or "cleaned up" trusted host list file. It will not become the "active" trusted host list file for a few steps yet. To ensure that the later step is as seamless as possible, select a file within the same directory as the existing trusted host list file. Create the file and set the file permissions to 444, so that the remaining steps will work properly. For example:

   ```
   touch /var/ct/cfg/ct_has.thl.new chmod 444
          /var/ct/cfg/ct_has.thl.new
   ```

9. Edit the file created in Step 7, converting it to a shell script. For each entry, create a new **ctsthl** command to add an entry to a brand new trusted host list file. Specify the new trusted host list file selected in Step 8 as the argument to the **-f** option. Use the "`Host Name:`" listed in each entry as the argument to the **-n** option, the "`Identifer Generation Method:`" listed as the argument to the **-m** option, and the string after the "`Identifier Value:`" as the argument to the **-p** option. Ensure that all new **ctsthl** commands are part of a single script command line. Continuing the example from Step 7, the new contents of the **/tmp/cleanthl** will create a new trusted host list file **/var/ct/cfg/ct_has.thl.new**; the new **/tmp/cleanthl** file contents would be:

   ```
       /usr/sbinrsct/bin/ctsthl
       -f/var/ct/cfg/ct_has.thl.new -a \
       -n avenger.pok.ibm.com \
       -m rsa1024 \
       -p \
   120400a25e168a7eafcbe44fde48799cc3a88cc177019100
   09587ea7d9af5db90f29415db7892c7ec018640eaae9c6bd
   ```

```
a64098efaf6d4680ea3bb83bac663cf340b5419623be80ce
977e153576d9a707bcb8e8969ed338fd2c1df4855b233ee6
533199d40a7267dcfb01e923c5693c4230a5f8c60c7b8e67
9eb313d926beed115464cb0103
      /usr/sbinrsct/bin/ctsthl
      -f/var/ct/cfg/ct_has.thl.new -a \
      -n ppsclnt16.pok.ibm.com \
      -m rsa1024 \
      -p \
120400a25e168a7eafcbe44fde48799cc3a88cc177019100
09587ea7d9af5db90f29415db7892c7ec018640eaae9c6bd
a64098efaf6d4680ea3bb83bac663cf340b5419623be80ce
977e153576d9a707bcb8e8969ed338fd2c1df4855b233ee6
533199d40a7267dcfb01e923c5693c4230a5f8c60c7b8e67
9eb313d926beed115464cb0103
      /usr/sbinrsct/bin/ctsthl
      -f/var/ct/cfg/ct_has.thl.new -a \
      -n sh2n04.pok.ibm.com \
      -m rsa1024 \
      -p \
120400a25e168a7eafcbe44fde48799cc3a88cc177019100
09587ea7d9af5db90f29415db7892c7ec018640eaae9c6bd
a64098efaf6d4680ea3bb83bac663cf340b5419623be80ce
977e153576d9a707bcb8e8969ed338fd2c1df4855b233ee6
533199d40a7267dcfb01e923c5693c4230a5f8c60c7b8e67
9eb313d926beed115464cb0103
```

10. Execute this shell script to create a new trusted host list file. Note that the new trusted host list file will not be used yet, since it is known by a new name. For example:

    ```
    sh /tmp/cleanthl
    ```

11. Verify that Step 10 executed correctly by listing the contents of the new trusted host list file, capturing the output in a file, and comparing those results to the original output captured in Step 6. For example:

    ```
    /usr/sbin/rsct/bin/ctsthl -l
        -f \   /var/ct/cfg/ct_has.thl.new > /tmp/thlnew.out
    diff /tmp/thlnew.out /tmp/thlorig.out
    ```

    There should be no differences detected.

12. Overlay the new trusted host list file over the old. For example:

    ```
     mv /var/ct/cfg/ct_has.thl.new /var/ct/cfg/ct_has.thl
    ```

13. Clean up any temporary files that were made to accomplish this (in our example, the temporary files are `/tmp/thlnew.out`, `/tmp/thlorig.out` , and `/tmp/cleanthl`).
14. Log off the system and resume normal operations.

Repair test:

Repair is tested using Step 11 in the sequence above.

Recovery action:

Read this paragraph in its entirety. A recovery action exists that can help avoid triggering failures related to private and public key mismatches. This recovery action will *disable* the Host Based Authentication (HBA) mechanism and the Enhanced Host Based Authentication (HBA2) mechanism on the local node. Applications on the local node will not be able to authenticate with other applications using either the HBA or HBA2 mechanisms. If no other mechanism is

available, then *all applications on the local node will be unauthenticated* if this recovery action is taken. Do not use this recovery action if this solution is not acceptable.

1. Log on to the node as **root**.
2. Shut down all trusted services on the node.
3. If an override for the cluster security services configuration file does not exist in the file **/var/ct/cfg/ctsec.cfg**, create this file using the following command:

   ```
   cp /usr/sbin/rsct/cfg/ctsec.cfg  /var/ct/cfg/ctsec.cfg
   ```

4. Using a text editor, insert a comment character (#) at the start of the entries for the Host Based Authentication and Enhanced Host Based Authentication mechanisms, as follows:

   ```
   #Prior Mnemonic Code    Path               Flags
   #--------------------------------------------------
   # 1    unix     0x00001 /usr/lib/unix.mpm  i
   # 2    hba2     0x00002 /usr/lib/hba2.mpm  iz[unix]
   ```

5. Restart the trusted services on this node
6. Log off the node.

Recovery removal:

To remove the above recovery action:

1. Log on to the node as **root**.
2. Shut down all trusted services on the node.
3. Using a text editor, edit the override cluster security services configuration file **/var/ct/cfg/ctsec.cfg**. Delete the comment character (#) from the start of the entries for the Host Based Authentication and Enhanced Host Based Authentication mechanisms:

   ```
   #Prior Mnemonic Code    Path               Flags
   #--------------------------------------------------
   1     unix     0x00001 /usr/lib/unix.mpm  i
   2     hba2     0x00002 /usr/lib/hba2.mpm  iz[unix]
   ```

4. Compare the override configuration file to the default configuration file using the **diff** command:

   ```
   diff /var/ct/cfg/ctsec.cfg  /usr/sbin/rsct/cfg/ctsec.cfg
   ```

5. If the files are not different, remove the override file **/var/ct/cfg/ctsec.cfg** from this system; it is no longer required.
6. Restart the trusted services on this node.
7. Log off the node.

Action 4: identify the cause of authentication-related failures

Authentication failures can be specific to the underlying security mechanism, or they can be the result of configuration problems with the cluster security services library. Take this action to identify the cause of authentication-related failures.

Description:

Used to identify the cause of authentication related failures.

Repair action:

Perform the troubleshooting procedures outlined in "Authentication troubleshooting procedures" on page 86. Perform any recommended actions indicated by these procedures. If conditions persist, contact the IBM Support Center for additional assistance.

Action 5: set consistent host name resolution

It may become necessary to set consistent host name resolution. Take this action to set consistent host name resolution and for short host name support.

Description:

Setting consistent host name resolution.

Repair action:

Before performing this action, understand the desired cluster configuration in regards to:

- **Domain name servers**. Does the cluster make use of domain name servers? If so, decide on the name resolution order between the domain name server and the local **/etc/hosts** file. The default setting can vary between AIX, Linux, and Solaris operating systems. It is recommended that the search order be explicitly stated in either the **/etc/netsvc.conf** or the **/etc/irc.conf** files. If the search order will use the **/etc/hosts** file before contacting the domain name server, then updates to the **/etc/hosts** file on each node will be required as follows:
  - **Management Domains:** The host name and address of the Management Server will need to be added to the **/etc/hosts** file for each node within the Management Domain. The name and address of each managed node will need to be added to the **/etc/hosts** file on the Management Server.
  - **Peer Domains:** The host names and addresses of each node within the cluster will need to be added to the **/etc/hosts** file on each node within the cluster.
- **Host name format**. Does the cluster span multiple domains? If so, fully qualified host names should be in use. If the cluster is contained within a single domain, then short host names can be used, although it is recommended that fully qualified host names be used to support future growth.

Perform the following tasks on each node within the cluster:

1. Log onto the node as **root**.
2. If the cluster uses domain name servers, modify the **/etc/netsvc.conf** or the **/etc/irc.conf** files to specify the desired search order. Go to Step 6.
3. If a name server is in use and short host names only are to be used by the cluster nodes, edit the **/etc/hosts** file on this node to specify the address and short host name for this node. Also add any other nodes required for the type of cluster as indicated above, using the address and short host names for the required nodes. Go to Step 6.
4. If a name server is not in use and fully qualified host names only are to be used by the cluster nodes, edit the **/etc/hosts** file on this node to specify the address and fully qualified host name for this node. Also add any other nodes required for the type of cluster as indicated above, using the address and short host names for the required nodes. Go to Step 6.
5. If a name server is not in use and short host names only are to be used by the cluster nodes, edit the **/etc/hosts** file on this node to specify the address and fully qualified host name for this node. Also add any other nodes required for the type of cluster as indicated above, using the address and short host names for the required nodes. Go to Step 6.
6. Issue **Action 7**. Return to this repair action at Step 7, when **Action 7** is completed.
7. Recycle the **ctcasd** daemon using the **stopsrc -s ctcasd** and **startsrc -s ctcasd** commands.

Repair test:

Perform the diagnostic procedures in "Troubleshooting procedures for host-based authentication mechanisms" on page 90.

Action 6: recover from a security breach

Recovery following a security breach may become necessary. Take this action if a private key becomes compromised.

Description:

Recovering from a security breach, when a node's private key has become public knowledge or has otherwise been compromised.

Repair action:

It is impossible to tell for how long a private key may have been public knowledge or have been compromised. Once it is learned that such an incident has occurred, the system administrator must assume that unwarranted access has been granted to critical system information for an unknown amount of time, and the worst must be feared in this case. Such an incident can only be corrected by a disassembly of the cluster, a reinstall of all cluster nodes, and a reformation of the cluster. When reforming the cluster, consider the following when configuring cluster security services in the new cluster:

1. Choose a new password for **root**. It is possible that the security breach may have started with the **root** password being compromised, because the private key file is only accessible to **root** users.
2. Consider using a stronger security protection within the private and public key. Use a more extensive key type such as **rsa1024** over smaller key types.
3. Ensure that only the **root** user is capable of accessing the private key file. No other system users should have any form of access to this file.
4. Ensure that the Host Based Authentication mechanism's configuration file **ctcasd.cfg** can only be modified by the **root** user.
5. Verify that the **ctcasd** binary file, located in **/usr/sbin/rsct/bin/ctcasd**, is the same as the binary file shipped in the RSCT installation media.
6. Monitor the private key file to ensure that the permissions on the file do not change.
7. Monitor the **ctcasd.cfg** configuration file to ensure that the permissions on the file do not change.
8. Monitor the **ctcasd** binary file for any changes in size or modification date.
9. Monitor the system more closely for security breaches.

Action 7: create an initial trusted host list

It may become necessary to create an initial trusted host list on a specific cluster node if no trusted host list exists. Take this action to reset a missing or incorrectly populated trusted host list on a local node.

Description:

This action is used to create an initial trusted host list on a specific cluster node if no trusted host list exists.

This action is also used to reset the information for the local node in its own trusted host list. This may be necessary when a change in host name resolution changes the name used by this local node in authentication requests, as described in **Action 5**. This action may also be necessary when the host name for the local node is changed, or when network addresses for the local node are added, removed, or changed.

Repair action:

Perform the following steps:

1. Locate the trusted host list used by the Cluster Security Subsystem's Host Based Authentication mechanism. This file is specified in the HBA_THLFILE entry of the **/var/ct/cfg/ctcasd.cfg** file (or the **/usr/sbin/rsct/bin/ctcasd.cfg** file, if the other file does not exist). By default, the trusted host list file used by the UNIX Host Based Authentication mechanism is **/var/ct/cfg/ct_has.thl**. Make a note of the trusted host list file in use; this will be required in Step 2.
2. Issue the command **ctsthl -s -f**, using the file name determined in Step 1 as the argument to the **-f** option. For example, if the default trusted host list file is in use, the command is:

   ```
   /usr/sbin/rsct/bin/ctsthl -s -f /var/ct/cfg/ct_has.thl
   ```

Repair test:

Perform the following steps:

1. Locate the trusted host list used by the Cluster Security Subsystem's Host Based Authentication mechanism. This file is specified in the HBA_THLFILE entry of the **/var/ct/cfg/ctcasd.cfg** file (or the **/usr/sbin/rsct/bin/ctcasd.cfg** file, if the other file does not exist). By default, the trusted host list file used by the Host Based Authentication mechanism is **/var/ct/cfg/ct_has.thl**. Make a note of the trusted host list file in use; this will be required in Step 2.
2. Display the contents of the trusted host list file with the command **ctsthl -l -f**, using the file name determined in Step 1 as the argument to the **-f** option. For example, if the default trusted host list file is in use, the command is:

   ```
   /usr/sbin/rsct/bin/ctsthl -l -f /var/ct/cfg/ct_has.thl
   ```

   The output format will be similar to the following example:

   ```
   --------------------
   Host name: avenger.pok.ibm.com
   Identifier Generation Method: rsa1024
   Identifier Value:
   120400a25e168a7eafcbe44fde48799cc3a88cc17701910009587ea7d9af
   5db90f2941 5db7892c7ec018640eaae9c6bda64098efaf6d4680ea3bb83
   bac663cf340b5419623be 80ce977e153576d9a707bcb8e8969ed338fd2c
   1df4855b233ee6533199d40a7267dcfb 01e923c5693c4230a5f8c60c7b8
   e679eb313d926beed115464cb0103
   --------------------
   Host name: 9.117.101.43
   Identifier Generation Method: rsa1024
   Identifier Value:
   120400a25e168a7eafcbe44fde48799cc3a88cc17701910009587ea7d9af5
   db90f2941 5db7892c7ec018640eaae9c6bda64098efaf6d4680ea3bb83ba
   c663cf340b5419623be 80ce977e153576d9a707bcb8e8969ed338fd2c1df
   4855b233ee6533199d40a7267dcfb 01e923c5693c4230a5f8c60c7b8e679
   eb313d926beed115464cb0103
   ------------------
   ```
3. Verify that the trusted host list output from Step 2 contains entries for the known host names and network addresses supported by the local node.

Diagnosing problems with Topology Services

This topic discusses diagnostic procedures and failure responses for the Topology Services component of RSCT. The list of known error symptoms and the associated responses are in the section.

The list of known error symptoms and the associated responses are located in “Error symptoms, responses, and recoveries” on page 178.

Terminology essential for understanding this topic

This topic describes terminology and concepts essential for using the remainder of this section.

Cluster-dependent Topology Services terms

A node can be running in more than one cluster at a time, which means there will be more than one instance of Topology Services running on that node.

Topology Services is used in all of the cluster types listed in “Other cluster types” on page 11. As mentioned in that section, a node can be running in more than one of those clusters at a time – which means there will be more than one instance of Topology Services running on that node.

The primary daemon responsible for most of the work in Topology Services is:

```
/usr/sbin/rsct/bin/hatsd
```

On any node with more than one instance of Topology Services running, there will be more than one active **hatsd** process. However, the multiple instances will be running under different subsystem names and with separate control scripts, configuration data, and log files, so they will never interfere with each other.

This means that while the basics of how the subsystems work will be the same in each cluster, the specifics of how to query each subsystem and where to look for data about it will differ depending on which cluster type it is supporting.

Table 21 presents terms that will be used in the remainder of this topic to represent the actual values of various aspects of Topology Services in each cluster type.

Table 21. Cluster-dependent Topology Services terms used in this topic

This term...	Represents this value in a cluster...
subsystem_name	Name of the Topology Services subsystem, as defined to SRC.
ctrl_script	Name of the control script for manipulation of the subsystem. **Attention:** Direct manipulation of Topology Services by using this script is not advisable in all cases. Only use this script if you are certain about what you need to do or under the direction of the IBM Support Center.
startup_script	Name of the startup script used by SRC to invoke the subsystem. **Attention:** The startup script is for reference purposes only. Do *not* directly invoke the startup script.
log_dir	Path name of the log directory in which all logs generated by the subsystem are located.
startup_log	Name of the log file for the *startup_script*. For more information, see "Topology Services startup log" on page 157.
usr_log	Name of the user log file. For more information, see "Topology Services user log" on page 158.
svc_log	Name of the service log file. For more information, see "Topology Services service log" on page 158.
nim_log	Name of the network interface module (NIM) log file. For more information, see "Network interface modules" on page 136 and "Network interface module (NIM) log" on page 161.
run_dir	Path name of the run directory, where certain configuration files are located.
machines.lst	Name of the machines list configuration file. For more information, see "Machines list" on page 136.

The following topics show the values that each of these terms resolve to for each of the cluster types where Topology Services can run. As you encounter these terms in the remainder of this topic, use this information to determine the actual values that apply to the cluster type being discussed.

The following syntax conventions apply:

DD Day of the month when the subsystem was started

hhmmss
Timestamp when the subsystem was started

lang Language setting in use by the subsystem (such as `en_US`, `ja_JP`, or `fr_FR`)

RPD cluster:

This topic lists the values for the cluster-dependent Topology Services terms in an RPD cluster.

Table 22 lists the values for the cluster-dependent Topology Services terms in an RPD cluster.

Table 22. Cluster-dependent Topology Services terms for an RPD cluster

This Topology Services term...	Resolves to this value for an RPD cluster...
subsystem_name	**cthats**
ctrl_script	**/usr/sbin/rsct/bin/cthatsctrl**
startup_script	**/usr/sbin/rsct/bin/cthats**
log_dir	**/var/ct/***cluster_name***/log/cthats**
startup_log	*log_dir***/cthats.***cluster_name*
usr_log	*log_dir***/cthats.***DD***.***hhmmss***.***lang*
svc_log	*log_dir***/cthats.***DD***.***hhmmss*
nim_log	*log_dir***/nim.cthats.***interface*
run_dir	**/var/ct/***cluster_name***/run/cthats**
machines.lst	*run_dir***/machines.lst**

The value of *cluster_name* can be found by running:

```
/usr/es/sbin/cluster/utilities/cltopinfo -c
```

PowerHA SystemMirror cluster:

This topic lists the values for the cluster-dependent Topology Services terms in an PowerHA SystemMirror cluster.

Table 23 lists the values for the cluster-dependent Topology Services terms in an PowerHA SystemMirror cluster.

Table 23. Cluster-dependent Topology Services terms for an PowerHA SystemMirror cluster

This Topology Services term...	Resolves to this value for an PowerHA SystemMirror cluster...
subsystem_name	**topsvcs**
ctrl_script	**/usr/sbin/rsct/bin/topsvcsctrl**
startup_script	**/usr/sbin/rsct/bin/topsvcs**
log_dir	**/var/ha/log**
startup_log	*log_dir***/topsvcs.default**
usr_log	*log_dir***/topsvcs.***DD***.***hhmmss***.***cluster_name***.***lang*
svc_log	*log_dir***/topsvcs.***DD***.***hhmmss***.***cluster_name*
nim_log	*log_dir***/nim.topsvcs.***interface***.***cluster_name*
run_dir	**/var/ha/run/topsvcs.***cluster_name*
machines.lst	*run_dir***/machines.***cluster_id***.lst**

The value of *cluster_name* can be found by running:

```
/usr/es/sbin/cluster/utilities/cltopinfo -c
```

The value of *cluster_id* can be found by running:

```
/usr/es/sbin/cluster/utilities/clrsctinfo -p cllsclstr
```

PSSP cluster:

This topic lists the values for the cluster-dependent Topology Services terms in a PSSP cluster.

Table 24 lists the values for the cluster-dependent Topology Services terms in a PSSP cluster.

Table 24. Cluster-dependent Topology Services terms for a PSSP cluster

This Topology Services term...	Resolves to this value for a PSSP cluster...
subsystem_name	On the CWS: **/hats.***partition_name* On the nodes: **/hats.**
ctrl_script	**/usr/sbin/rsct/bin/hatsctrl**
startup_script	**/usr/sbin/rsct/bin/hats**
log_dir	**/var/ha/log**
startup_log	*log_dir***/hats.***partition_name*
usr_log	*log_dir***/hats.***DD.hhmmss.partition_name.lang*
svc_log	*log_dir***/hats.***DD.hhmmss.partition_name*
nim_log	*log_dir***/nim.hats.***interface.partition_name*
run_dir	**/var/ha/run/hats.***partition_name*
/machines.lst	*run_dir***/machines.lst**

The value of *partition_name* can be found by running:

```
/usr/lpp/ssp/bin/spget_syspar -n
```

Network interface modules

When Topology Services is started, the main daemon will start a number of child processes – one for each adapter that is being locally monitored. These child processes are called *network interface modules*, or NIMs.

Each NIM is responsible for only one interface and has its own *nim_log*. For details, see:

- Table 22 on page 135: Cluster-dependent Topology Services terms for an RPD cluster
- Table 23 on page 135: Cluster-dependent Topology Services terms for an PowerHA SystemMirror cluster
- Table 24: Cluster-dependent Topology Services terms for a PSSP cluster

A NIM handles the heartbeating work when told to do so and reports any adapter status changes to the **hatsd** daemon.

In this topic, the term "NIM" applies to this aspect of the Topology Services subsystem. Any alternative definitions (such as, for the AIX Network Installation Manager) will be explicitly stated.

Machines list

The machines list is a configuration file for Topology Services that lists all the adapter and tuning information pertinent to the cluster.

The machines list (see *machines.lst* in "Cluster-dependent Topology Services terms" on page 133) is a configuration file for Topology Services that lists all the adapter and tuning information pertinent to the cluster. This file is built from scratch each time Topology Services is started or when a refresh is done and a configuration change is detected. This file should never be edited by hand – doing so will usually have no effect but results could be unpredictable. Inaccuracies in the machines list should be tracked to the source data from which it was built, depending on the cluster type, as described below.

When the subsystem is active, the current machines list should reflect the configuration reported by the **lssrc -ls** *subsystem_name* command. When the subsystem is inactive, the current machines list should show what the configuration was the last time it was active or the last time a verification was run.

The *machines.lst* file is built by the Topology Services *startup_script*, so a copy of it is recorded in the *startup_log*. A limited number of instances (currently, seven) of this log file are kept, so information from a particular instance will be lost after many startup/refresh attempts. This is one reason why it is prudent, when possible, to take a snapshot immediately before attempting any recovery actions. Even if you think a problem might be solvable, you may need a clear picture of what the cluster looked like when the problem occurred, in case you later need to seek help from the IBM Support Center.

The data source for the machines list depends on the cluster type, as follows:

- In RPD, it is built using information propagated by the configuration resource manager (the IBM.ConfigRM subsystem).
- In PowerHA SystemMirror, it is built from the local PowerHA SystemMirror ODM data obtained from PowerHA SystemMirror-provided tools (such as **clrsctinfo**). (The PowerHA SystemMirror code is responsible for populating the ODMs and ensuring that they are synchronized among nodes; however, Topology Services also plays a role in verifying local data during synchronization.)
- In PSSP, the control workstation is responsible for building a master copy from the adapter information in the SDR; the master is then stored in the SDR where the nodes can get a copy of it by an SDRRetrieveFile call.

Requisite function

The Topology Services component of RSCT directly uses required software components that may manifest problems as error symptoms in the Topology Services component.

If you perform all the diagnostic routines and error responses listed in this section and still have problems with the Topology Services component of RSCT, you should consider these components as possible sources of the error. The following list presents components in the order that they are most likely to introduce an error, from the most likely to the least likely.

- UDP/IP communication
- Cluster adapter configuration
- UNIX Domain sockets
- Security libraries
- System Resource Controller (SRC)
- First Failure Data Capture (FFDC) library
- **/var/ct/***cluster_name* directory (for RPD)
- **/var/ha** directory (for PowerHA SystemMirror and PSSP)

Error information

On AIX system platforms, the RSCT component subsystems write this information to the AIX error log. On Linux, Windows, and Solaris system platforms, it writes the information to the respective system log.

Unless otherwise noted, each entry refers to a particular instance of the Topology Services daemon on the local node. Unless otherwise noted, entries are created on each occurrence of the condition. For more information on the AIX error log or the Linux, Windows, or Solaris system logs, see “Accessing logged errors” on page 2.

Error logs and templates

This topic lists Topology Services error log labels and error log types with their associated explanations.

Table 25 lists the error log templates used by Topology Services, sorted by **Error Label**. An **Explanation** and **Details** are given for each error.

Table 25. Error log templates for Topology Services

Label	Type	Description
TS_ASSERT_EM	PEND	**Explanation:** Topology Services daemon exited abnormally. **Details:** This entry indicates that the Topology Services daemon exited with an **assert** statement, resulting in a core dump being generated. Standard fields indicate that the Topology Services daemon exited abnormally. Detail Data fields contain the location of the **core** file. This is an internal error. Data needed by the IBM Service Center to diagnose the problem is stored in the **core** file (whose location is given in the error log) and in the Topology Services daemon service log. See "Topology Services service log" on page 158. Since only six instances of the Topology Services daemon service log are kept, it should be copied to a safe place. Also, only three instances of the **core** file are kept. See "Information to collect before contacting the IBM Support Center" on page 16 and contact the IBM Support Center.
TS_AUTHMETH_ER	PERM	**Explanation:** The Topology Services startup script cannot retrieve active authentication methods using command **/usr/sbin/rsct/bin/lsauthpts**. This entry applies to AIX nodes only. **Details:** This entry indicates that command **/usr/lpp/ssp/bin/lsauthpts**, run by the Topology Service startup script on the control workstation, was unable to retrieve the active authentication methods in a system partition. This error occurs when the startup script is running on the control workstation during initial startup or refresh. When this error occurs, all Topology Services daemons in the system partition will terminate their operations and exit. Diagnosing this problem requires collecting data only on the control workstation. Standard fields indicate that the startup script cannot retrieve active authentication methods in a system partition using command **lsauthpts**. The problem may be one of the following: • The system partition has an incorrect set of active partition methods. • The current system partition cannot be identified. Detail Data fields contain the return code of command **lsauthpts** and the location of the startup script log. The error message returned by command **lsauthpts** can be found in the startup script log.
TS_CMDFLAG_ER	PERM	**Explanation:** Topology Services cannot be started due to incorrect flags. **Details:** This entry indicates that the Topology Services daemon was unable to start because incorrect command line arguments were passed to it. This entry refers to a particular instance of Topology Services on the local node. Other nodes may have been affected by the same problem. Standard fields indicate that the daemon was unable to start because incorrect flags were passed to it. Detail Data fields show the path name to the daemon user log, which contains more detail about the problem. This problem may be one of the following: • Topology Services was started manually in an incorrect way. • Incompatible versions of the daemon and startup script are being used. • The SRC definition for the subsystem was manually set to an incorrect value. Information about the cause of the problem may not be available once the problem is cleared.
TS_CTIPDUP_ER	PERM	**Explanation:** See **TS_HAIPDUP_ER**.
TS_CTNODEDUP_ER	PERM	**Explanation:** See **TS_HANODEDUP_ER**.
TS_CTLOCAL_ER	PERM	**Explanation:** See **TS_HALOCAL_ER**.

Table 25. Error log templates for Topology Services (continued)

Label	Type	Description
TS_CPU_USE_ER	PERM	**Explanation:** The Topology Services daemon is using too much CPU. The daemon will exit. **Details:** This entry indicates that the Topology Services daemon will exit because it has been using almost 100% of the CPU. Since Topology Services runs in a real time fixed priority, exiting in this case is necessary. Otherwise, all other applications in the node will be prevented from running, Also, it is likely that the daemon is not working properly if it is using all the CPU. A **core** dump is created to allow debugging the cause of the problem. This entry refers to a particular instance of Topology Services running on a node. The standard fields indicate that the Topology Services daemon is exiting because it is using too much of the CPU, and explains some of the possible causes. The detailed fields show the amount of CPU used by the daemon (in milliseconds) and the interval (in milliseconds) where the CPU usage occurred. Collect the information recommended in "Information to collect before contacting the IBM Support Center" on page 16 and contact the IBM Support Center. In particular, the daemon log file and the most recent core files should be collected.
TS_DEATH_TR	UNKN	**Explanation:** Lost contact with a neighboring adapter. **Details:** This entry indicates that heartbeat messages are no longer being received from the neighboring adapter. This entry refers to a particular instance of the Topology Services daemon on the local node. The source of the problem could be either the local or remote node. Data from the remote node should also be obtained. Standard fields indicate that a local adapter is no longer receiving packets from the remote adapter. Detail Data fields contain the node number and IP address of the remote adapter. Data about the loss of connectivity may not be available after the problem is cleared. The local or remote adapter may have malfunctioned. Network connectivity to the remote adapter may have been lost. A remote node may have gone down. The Topology Services daemon on the remote node may have been blocked. If the problem is with the local adapter, an error log entry of type **TS_LOC_DOWN_ST** should follow in a few seconds. Information on the remote node should be collected to obtain a better picture of what failure has occurred.
TS_DMS_EXPIRING_EM	PEND	**Explanation:** The Topology Services daemon has decided it must allow the deadman switch to expire. **Details:** This entry indicates that the Topology Services daemon believes too many of its NIM processes are partially or fully blocked to remain in contact with the rest of the cluster. In order to avoid risk of data corruption, it will allow the deadman switch to expire once it has decided that the blockage has persisted for too long on too many of the NIMs. Specifics on which processes are blocked and for how long will be visible in either the daemon's internal log or the NIM logs. If the situation does not improve, a KERNEL_PANIC caused by the deadman switch will be seen next. If the situation improves enough for the daemon to continue updating the deadman switch before it triggers, then a TS_DMS_RESTORED_TE will be seen next instead.

Table 25. Error log templates for Topology Services (continued)

Label	Type	Description
TS_DMS_RESTORED_TE	TEMP	**Explanation:** Topology Services has recovered from the condition which led to the previous TS_DMS_EXPIRING_EM entry. The deadman switch is once again being updated. **Details:** This entry indicates that whatever conditions were putting Topology Services in danger of falling out of contact with the rest of the cluster have been alleviated. It should not be possible to see this entry without a prior TS_DMS_RESTORED_TE event. Depending on how long the DMS was allowed to expire before recovery occurred, it is very possible to see a TS_DMS_WARNING_ST at the same time as this entry.
TS_DMS_WARNING_ST	INFO	**Explanation:** The deadman switch timer is close to triggering. This entry applies to AIX nodes only. **Details:** This entry indicates that the deadman switch has been reset with a small time-to-trigger value left on the timer. This means that the system is in a state where the deadman switch timer is close to triggering. This condition affects the node where the error log entry appears. If steps are not taken to correct the problem, the node may be brought down by the deadman switch timer. This entry is logged on each occurrence of the condition. Some possible causes are outlined. Detailed fields contain the amount of time remaining in the deadman switch timer and also the interval to which the deadman switch timer is being reset. Program **/usr/sbin/rsct/bin/hatsdmsinfo** displays the latest time-to-trigger values and the values of time-to-trigger that are smaller than a given threshold. Small time-to-trigger values indicate that the deadman switch timer is close to triggering.
TS_DUPNETNAME_ER	PERM	**Explanation:** Duplicated network name in **machines.lst** file. **Details:** This entry indicates that a duplicate network name was found by the Topology Services daemon while reading the **machines.lst** configuration file. This entry refers to a particular instance of Topology Services on the local node. Other nodes may be affected by the same problem, since the **machines.lst** file is the same on all nodes. If this problem occurs at startup time, the daemon exits. Standard fields indicate that a duplicate network name was found in the **machines.lst** file. Detail Data fields show the name that was duplicated.
TS_FD_INVAL_ADDR_ST	PERM	**Explanation:** An adapter is not configured or has an address outside the cluster configuration. **Details:** This entry indicates that a given adapter in the cluster configuration is either not configured, or has an address which is outside the cluster configuration. This entry affects the local node, and causes the corresponding adapter to be considered down. Detailed data fields show the interface name, current address of the interface, and expected boot-time address. Probable causes for the problem are: • There is a mismatch among the cluster adapter configuration and the actual addresses configured on the local adapters. • The adapter is not correctly configured. If this is an AIX node, save the output of the command **netstat -in** . If this is a Linux or Solaris node, save the output of the command **ifconfig -a**.) If this is a Windows node, save the output of the command **ipconfig /all**. See "Information to collect before contacting the IBM Support Center" on page 16 and contact the IBM Support Center if the source of the problem cannot be found.

Table 25. Error log templates for Topology Services (continued)

Label	Type	Description
TS_FD_INTFC_NAME_ST	PERM	**Explanation:** An interface name is missing from the adapter configuration. **Details:** The Topology Services startup script reads information from the cluster configuration, containing for each adapter its address, boot-time interface name, and node number. This error entry is created when the interface name information is missing. This usually points to a problem when generating the adapter configuration. The detailed data fields contain the address in the Topology Services configuration and the interface name which has been "assigned" to the adapter by the Topology Services daemon. See “Information to collect before contacting the IBM Support Center” on page 16 and contact the IBM Support Center. This problem, in most of the cases, will not prevent Topology Services from correctly monitoring the adapter. However, internal problems may occur if a subsequent Topology Services refresh.
TS_HAIPDUP_ER	PERM	**Explanation:** IP address duplication in Topology Services configuration file. **Details:** This entry indicates that Topology Services was not able to start or refresh because the same IP address appeared twice in the configuration. This entry refers to a particular instance of Topology Services on the local node, but the problem may affect all the nodes. If this problem occurs at startup time, the daemon exits. Standard fields indicate that the same IP address appeared twice in the Topology Services **machines.lst** configuration file. Detail Data fields show the node number of one of the nodes hosting the duplicated address and the duplicated IP address. Information about the cause of the problem may not be available once the problem is cleared.
TS_HALOCAL_ER	PERM	**Explanation:** Local node missing in Topology Services configuration file. **Details:** Standard fields indicate that the local node was not present in the **machines.lst** file. This is a problem with the cluster configuration.
TS_HANODEDUP_ER	PERM	**Explanation:** Node number duplicated in Topology Services configuration file. **Details:** This entry indicates that Topology Services was not able to start or refresh because the same node appeared twice on the same network. This entry refers to a particular instance of Topology Services on the local node, but the problem should affect all the nodes. If this problem occurs at startup time, the daemon exits. Standard fields indicate that the same node appeared twice in the same network in the Topology Services **machines.lst** configuration file. Detail Data fields show the interface name of one of the adapters and the node number that appears twice. Information about the cause of the problem may not be available once the problem is cleared.
TS_ILLEGAL_ALIAS_ER	PERM	**Explanation:** Topology Services has detected an illegal alias on one of the interfaces and must exit. **Details:** In PowerHA SystemMirror, in an IPAT with IP Takeover configuration, service addresses cannot be aliased to existing boot-time addresses. When PowerHA SystemMirror changes a boot (or standby) address to a service address, the old address is removed. The presence of a service address aliased by mistake to one of its boot or standby addresses can cause Topology Services to misinterpret the adapter configuration during startup, so the daemon must exit.

Table 25. Error log templates for Topology Services (continued)

Label	Type	Description
TS_IOCTL_ER	PERM	**Explanation:** An **ioctl** call failed. **Details:** This entry indicates that an **ioctl()** call used by the Topology Services daemon to obtain local adapter information failed. This is a possible operating system-related problem. The Topology Services daemon issued an **ioctl()** call to obtain information about the network adapters currently installed on the node. If this calls fails, there is a potential problem in the operating system. The Topology Services daemon exits. See "Information to collect before contacting the IBM Support Center" on page 16 and contact the IBM Support Center.
TS_IPADDR_ER	PERM	**Explanation:** Cannot convert IP address in dotted decimal notation to a number. **Details:** This entry indicates that an IP address listed in the **machines.lst**configuration file was incorrectly formatted and could not be converted by the Topology Services daemon. If this problem occurs at startup time, the daemon exits. Standard fields indicate that the daemon was unable to interpret an IP address listed in the **machines.lst** file. The Detail Data fields contain the given IP address in dotted decimal notation and the node number where the address was found. The problem may be that the file system where the **run** directory is located is corrupted, or information in the cluster configuration is not correct. The **machines.lst** file is kept in the daemon "run" directory (**/var/ct/***cluster_name***/run/cthats**). The file is overwritten each time the subsystem is restarted. A copy of the file is kept in the startup script's log file, **/var/ct/***cluster_namecluster_name***/log/cthats/cthats.** *cluster_name*. A number of instances (currently, seven instances) of this log file are kept, but the information is lost if many attempts are made to start the subsystem.

TS_KEYS_ER	PERM	**Explanation:** Topology Services startup script cannot obtain security key information using the **/usr/sbin/rsct/bin/ctmsskf** command.
		
		Details:: This entry indicates that command **/usr/sbin/rsct/bin/ctmsskf**, run by the Topology Services startup script on the control workstation, was unable to retrieve the Topology Services key file. This error occurs when the startup script is running on the control workstation during initial startup or refresh. When this error occurs, all Topology Services daemons in the system partition will terminate their operations and exit.
		
		Diagnosing this problem requires collecting data only on the control workstation. In PSSP, the pathname of Topology Services DCE key file is **/spdata/sys1/keyfiles/rsct/***syspar_name***/hats**, where *syspar_name* is the name of the SP system partition. (the **hats** portion of the pathname can be redefined if file **/spdata/sys1/spsec/spsec_overrides** was used to override default DCE file names). The converted key file is located at **/var/ha/run/hats.** *syspar_name***/hats_cts**.
		
		Standard fields indicate that the **ctmsskf** command, invoked by the startup script, was unable to retrieve the Topology Services key file, and present possible causes. Detail Data fields contain the return code of command **ctmsskf** and the location of the startup script log. The error message returned by command **ctmsskf** is in the startup script log.
		
		In PSSP, this error typically indicates problems in DCE. For DCE configuration problems, see the configuration log file:
		
		`/opt/dcelocal/etc/cfgdce.log`
		
		For other DCE problems, see log files in the following directory:
		
		`/opt/dcelocal/var/svc`
		
		The problem may also occur in a RSCT peer domain, if security is enabled.

Table 25. Error log templates for Topology Services (continued)

Label	Type	Description
TS_LATEHB_PE	PERF	**Explanation:** Late in sending heartbeat to neighbors. **Details:** This entry indicates that the Topology Services daemon was unable to run for a period of time. This entry refers to a particular instance of the Topology Services daemon on the local node. The node that is the Downstream Neighbor may perceive the local adapter as dead and issue a **TS_DEATH_TR** error log entry. A node's Downstream Neighbor is the node whose IP address is immediately lower than the address of the node where the problem was seen. The node with the lowest IP address has a Downstream Neighbor of the node with the highest IP address. Standard fields indicate that the Topology Services daemon was unable to send messages for a period of time. Detail Data fields show how many seconds late the daemon was in sending messages. This entry is created when the amount of time that the daemon was late in sending heartbeats is equal to or greater than the amount of time needed for the remote adapter to consider the local adapter as down. Data about the reason for the Topology Services daemon being blocked is not usually kept, unless system tracing is being run on the node. The service log file keeps information about Topology Services events happening on the node at the time the daemon was blocked. See "Topology Services service log" on page 158. See the symptom labeled Node appears to go down and then up a few/several seconds later in "Error symptoms, responses, and recoveries" on page 178 for recovery information.
TS_LIBERR_EM	PEND	**Explanation:** Topology Services client library error. **Details:** This entry indicates that the Topology Services library had an error. It refers to a particular instance of the Topology Services library on the local node. This problem will affect the client associated with the library (RSCT Event Manager or more likely RSCT Group Services). Standard fields indicate that the Topology Services library had an error. Detail Data fields contain the error code returned by the Topology Services API. Data needed for IBM Service to diagnose the problem is stored in the Topology Services daemon service log, located at **/var/ct/***cluster_name* **/log/cthats/cthats.***DD.hhmmss* The Group Services daemon (the probable client connected to the library) is likely to have exited with an assert and to have produced an error log entry with template **GS_TS_RETCODE_ER**. See "Diagnosing problems with Group Services" on page 192 for a list of the information to save. See "Information to collect before contacting the IBM Support Center" on page 16 and contact the IBM Support Center.

Table 25. Error log templates for Topology Services (continued)

Label	Type	Description
TS_LOC_DOWN_ST	INFO	**Explanation:** Local adapter down. **Details:** This entry indicates that one of the local adapters is down. This entry refers to a particular instance of the Topology Services daemon on the local node. Standard fields indicate that a local adapter is down. Detail Data fields show the interface name, adapter offset (index of the network in the **machines.lst** file), and the adapter address according to Topology Services. This address may differ from the adapter's actual address if the adapter is incorrectly configured. Information about the source of the problem may be lost after the condition is cleared. Possible problems are: • The adapter may have malfunctioned. • The adapter may be incorrectly configured. See the **TS_UNS_SIN_TR** entry. • There is no other adapter functioning in the network. • Connectivity has been lost in the network. • A problem in Topology Services' adapter health logic. Perform these steps: 1. Verify that the address of the adapter listed in the output of **ifconfig** *interface_name* is the same as the one shown in this error log entry. If they are different, the adapter has been configured with an incorrect address. 2. If the output of the **ifconfig** command does not show the **UP** flag, this means that the adapter has been forced down by the command: **ifconfig** *interface_name***down** 3. Issue the command **netstat -in** to verify whether the `receive` and `send` counters are being incremented for the given adapter. On AIX and Solaris, the counters are the numbers below the `Ipkts` (receive) and **Opkts** (send) columns. On Linux, the counters are the numbers below the RX-OK (receive) and TX-OK (send) columns. If both counters are increasing, the adapter is likely to be working and the problem may be in Topology Services. 4. Issue the **ping** command to determine whether there is connectivity to any other adapter in the same network. If **ping** receives responses, the adapter is likely to be working and the problem may be in Topology Services. 5. See “Operational test 4: check the address of the local adapter” on page 168.
TS_LOGFILE_ER	PERM	**Explanation:** The daemon failed to open the log file. **Details:** This entry indicates that the Topology Services daemon was unable to open its log file. Standard fields indicate that the daemon was unable to open its log file. Detail Data fields show the name of the log file. The situation that caused the problem may clear when the file system problem is corrected. The Topology Services daemon exits. See “Information to collect before contacting the IBM Support Center” on page 16 and contact the IBM Support Center.

Table 25. Error log templates for Topology Services (continued)

Label	Type	Description
TS_LONGLINE_ER	PERM	**Explanation:** The Topology Services daemon cannot start because the **machines.lst** file has a line that is too long. **Details:** This entry indicates that the Topology Services daemon was unable to start because there is a line which is too long in the **machines.lst** configuration file. This entry refers to a particular instance of Topology Services on the local node. If this problem occurs at startup time, the daemon exits. The problem is likely to affect other nodes, since the **machines.lst** file should be the same at all nodes. Standard fields indicate that the daemon was unable to start because the **machines.lst** configuration file has a line longer than 80 characters. Detail Data fields show the path name of the **machines.lst** configuration file. It is possible that the network name is too long, or there is a problem in the **/var/ct** file system.
TS_LSOCK_ER	PERM	**Explanation:** The daemon failed to open a listening socket for connection requests. **Details:** This entry indicates that the Topology Services daemon was unable to open a socket connection to communicate with its clients. Standard fields indicate that the daemon was unable to open the socket. Detail Data fields show the operation being attempted at the socket (in English) and the system error value returned by the system call. The situation that caused the problem may clear with a reboot. The **netstat** command shows the sockets in use in the node. The Topology Services daemon exits. See "Information to collect before contacting the IBM Support Center" on page 16 and contact the IBM Support Center.
TS_MACHLIST_ER	PERM	**Explanation:** The Topology Services configuration file cannot be opened. **Details:** This entry indicates that the Topology Services daemon was unable to read its **machines.lst** configuration file. Standard fields indicate that the daemon was unable to read the **machines.lst** file. Detail Data fields show the path name of the file. Information about the cause of the problem is not available after the condition is cleared. If this problem occurs at startup time, the daemon exits. See "Information to collect before contacting the IBM Support Center" on page 16 and contact the IBM Support Center.
TS_MIGRATE_ER	PERM	**Explanation:** Migration-refresh error. This entry applies to AIX nodes only. **Details:** This entry indicates that the Topology Services daemon has found a problem during a migration-refresh. The migration-refresh is a refresh operation issued at the end of an PowerHA SystemMirror node by node migration, when the last node is moved to the newer release. The problem may be caused by the information placed on the Global ODM when the migration protocol is complete. This entry refers to a particular instance of the Topology Services daemon on the local node. It is likely that some of the other nodes have a similar problem. Standard fields indicate that the Topology Services daemon encountered problems during a migration-refresh. PowerHA SystemMirror may have loaded incorrect information into the Global ODM. Data read by the Topology Services startup script is left on the Topology Services run directory and will be overwritten in the next refresh or startup operation. The data in the "run" directory should be saved. The Topology Services "Service" log file has a partial view of what was in the Global ODM at the time of the refresh operation.

Table 25. Error log templates for Topology Services (continued)

Label	Type	Description
TS_MISCFG_ER	PEND	**Explanation:** Local adapter incorrectly configured. **Details:** This entry indicates that one local adapter is either missing or has an address that is different from the address that Topology Services expects. Standard fields indicate that a local adapter is incorrectly configured. Detail Data fields contain information about the adapter, such as the interface name, adapter offset (network index in the **machines.lst** file), and expected address. Possible sources of the problem are: • The adapter may have been configured with a different IP address. • The adapter is not configured. • Topology service was started after a *Force Down* in PowerHA SystemMirror. This entry is created on the **first occurrence** of the condition. No data is stored about the condition after the problem is cleared. Use the interface name in the error report to find the adapter that is incorrectly configured. Command: **ifconfig***interface_name* displays information about the adapter.
TS_NIM_DIED_ER	PERM	**Explanation:** One of the NIM processes terminated abnormally. **Details:** This entry is created when one of the NIM (Network Interface Modules) processes used by Topology Services to monitor the state of each adapter, terminates abnormally. When a NIM terminates, the Topology Services daemon will restart another. If the replacement NIM also terminates quickly, no other NIM will be started, and the adapter will be flagged as down. Detailed data fields show: • Process exit value, if not terminated with a signal (A value from 1 to 99), will be an *errno* value from invoking the NIM process. • Signal number (0: no signal). • Whether a core file was created (1: core file; 0: no core file). • Process id (PID). • Interface name being monitored by the NIM. • Path name of NIM executable file. See “Information to collect before contacting the IBM Support Center” on page 16 and contact the IBM Support Center.
TS_NIM_ERROR_INTERNAL_ER	PERM	**Explanation:** An internal error occurred at the NIM process. **Details:** This entry indicates that there was an error in the execution of the NIM. This could be a serious enough error that will cause the NIM process to exit. It could also be a less severe error. In case the NIM exits, a new NIM will be respawned in its place. The standard fields describe the most likely causes for the problem: an internal "assert" or some internal limit was exceeded. The detailed fields show the error level (serious, error, information), an error description, some error data, and the interface name to which the NIM is associated.

Table 25. Error log templates for Topology Services (continued)

Label	Type	Description
TS_NIM_ERROR_MSG_ER	PERM	**Explanation:** Too many incorrect messages exchanged between the Topology Services daemon and the NIM. **Details:** This entry indicates that the daemon was unable to interpret messages sent to it by the NIM via the Unix-domain socket. The probable causes for this are: • The NIM and the daemon lost the "frame synchronization" on the packets flowing through the UNIX-domain socket. This causes the daemon to interpret packets incorrectly. • The daemon and the NIM are using different versions of the protocol, resulting in the daemon being unable to interpret messages sent by the NIM. • The NIM has an internal problem that causes it to send invalid packets to the daemon. After the daemon has received a number of messages from the NIM that it cannot handle, the daemon will issue this error log entry and then terminate the connection with the NIM. As soon as the NIM terminates, the daemon will start a new one. The standard fields describe the problem and offers some possible causes. The detailed fields show the last kind of error received, the last packet type received, the error count, the message's protocol version and the daemon's protocol version, and finally the interface name to which the NIM is associated.
TS_NIM_ERROR_RDWR_ER	PERM	**Explanation:** The NIM encountered a read or write error when sending data to or receiving data from the network adapter or non-IP device. **Details:** This entry indicates that there were I/O errors when trying to send data to the adapter or device, or when trying to receive data from it. The most likely causes are that the adapter is down (in the *ifconfig* sense) or has been unconfigured. For non-IP devices, it is possible that the remote side of the connection is no longer active. The standard fields present the possible causes for the problem. The detailed fields indicate whether the problem was a write or read error, and also some details about the error. For example, for errors when sending data, the detailed fields show the *errno* value and the number of times the error occurred. For RS232 links, an error entry will be issued if there are too many checksum errors. In this case the error count will be shown. The interface name to which the NIM is associated is also shown.
TS_NIM_ERROR_STUCK_ER	PERM	**Explanation:** One of the threads in a NIM process was blocked. **Details:** This entry indicates that a thread in one of the NIM processes did not make progress and was possibly blocked for a period of time. Depending on which of the threads was blocked and for how long, the adapter corresponding to the NIM process may be erroneously considered down. The standard fields indicate that the NIM was blocked and present possible causes and actions to prevent the problem from reoccurring. The problem may have been caused by resource starvation at the node, or possibly excessive I/O activity. The detailed fields show the name of the thread which was blocked, the interval in seconds during which the thread was blocked, and the interface name which is associated with this instance of the NIM. If there is no false adapter down event caused by the blockage then no action is needed. If there is then the cause for the blockage needs to be understood. To investigate the problem, follow the same steps as those taken to investigate the error entry TS_LATEHB_PE.

Table 25. Error log templates for Topology Services (continued)

Label	Type	Description
TS_NIM_ERROR_TRAF_ER	PERM	**Explanation:** The NIM has detected too much traffic being received from the adapter or being sent to the adapter. **Details:** This entry indicates either too much data has been received from the adapter or (more likely) the NIM detected that more data is being sent by the Topology Services daemon than what can be pumped into the adapter. This is more likely to happen with slow non-IP connections. Usually any device can support the "normal traffic" sent for heartbeating. However, in situations where Group Services protocols need to be run over these slow links then it is possible for this error to occur. If this error occurs repeatedly and a "slow" device is being used for heartbeating then a faster device should be pursued. The standard fields describe the problem and possible causes. The detailed fields indicate whether the problem occurred when sending or receiving data. For send errors, the size of the packet queue length at the NIM is shown. The interface name to which the NIM is associated is also shown.
TS_NIM_NETMON_ERROR_ER	PERM	**Explanation:** An error occurred in the network monitoring (**netmon**) library, which the Network Interface Module (NIM) uses. Topology Services uses NIM processes to monitor the state of each adapter, to determine whether the local adapter is "up". **Details:** This entry is created when there is an internal error in the **netmon** library. As a result, the local adapter will be flagged as "down", even though the adapter might still be working properly. Other than a problem in the library, another possible cause for the problem is the presence of a non-supported adapter in the cluster configuration. Detailed data fields show: • The *errno* value. • The error code from the **netmon** library. • The function name in the library that presented a problem. • The name of the interface that is being monitored. See “Information to collect before contacting the IBM Support Center” on page 16 and contact the IBM Support Center. **Note:** It is important to collect the information as soon as possible because logging information for the **netmon** library is kept in log files in which the information might be overwritten within 30 minutes.

Table 25. Error log templates for Topology Services (continued)

Label	Type	Description
TS_NIM_OPEN_ERROR_ER	PERM	**Explanation:** NIM (Network Interface Module) - processes used by Topology Services to monitor the state of each adapter, failed to connect to the local adapter that it is supposed to monitor. **Details:** This entry is created when the NIM is unable to connect to the local adapter that needs to be monitored. As a result, the adapter will be flagged as down, even though the adapter might still be working properly. Detailed data fields show: • Interface name. • Description 1: description of the problem. • Description 2: description of the problem. • Value 1 - used by the IBM Support Center. • Value 2 - used by the IBM Support Center. Some possible causes for the problem are: • NIM process was blocked while responding to NIM open command. • NIM failed to open non-IP device. • NIM received an unexpected error code from a system call. See "Information to collect before contacting the IBM Support Center" on page 16 and contact the IBM Support Center.
TS_NODENUM_ER	PERM	**Explanation:** The local node number is not known to Topology Services. **Details:** This entry indicates that Topology Services was not able to find the local node number. Standard fields indicate that the daemon was unable to find its local node number. The Topology Services daemon exits. See "Information to collect before contacting the IBM Support Center" on page 16 and contact the IBM Support Center.
TS_NODEUP_ST	INFO	**Explanation:** Remote nodes that were previously down were seen as up by Topology Services. This is an indication that the Topology Services daemon detected one or more previously down nodes as being up. It refers to a particular instance of the Topology Services daemon. **Details:** In case the same nodes were seen as dead a short time before, data should be collected on the remote nodes. Standard fields indicate that remote nodes were seen as up and present possible causes. Detailed fields contain, in the section, a reference to the entry where the same nodes were seen as dead. If these nodes were seen as down before at different times, the reference code will be for one of these instances. The Detail Data also contains the path name of a file which stores the numbers of the nodes that were seen as up, along with the error id for the error log entry where each node was seen as dead previously. The file with the node numbers may eventually be deleted by the system. The file is located in: **/var/adm/ffdc/dumps/sh.*** . If the same nodes were recently seen as dead (follow the REFERENCE CODE), examine the remote nodes for the reason why the nodes were temporarily seen as dead. This entry is logged when a remote node is seen as alive. The same node may have been seen as dead some time ago. If so, the **TS_NODEUP_ST** will have, as part of the Detail Data, a location of a file whose contents are similar to: `.ZOWYB/Z5Kzr.zBI14tVQ7.....  1`

Table 25. Error log templates for Topology Services (continued)

Label	Type	Description
TS_OFF_LIMIT_ER	PERM	**Explanation:** Number of network offsets exceeds Topology Services limit. **Details:** This entry is created whenever the number of adapters and networks in the cluster configuration exceeds the Topology Services daemon's internal limit for maximum number of "heartbeat rings" of 48. Notice that a single cluster network may map to multiple "heartbeat rings". This will happen when a node has multiple adapters in the same network, since a heartbeat ring is limited to a single adapter per node. If this error occurs, a number of adapters and networks in the configuration may remain unmonitored by Topology Services. The detailed data fields contain the first network in the configuration to be ignored and the maximum number of networks allowed. When attempting to eliminate the problem, initially focus on the nodes that have the most adapters in the configuration, and proceed to remove some adapters from the configuration.
TS_REFRESH_ER	PERM	**Explanation:** Topology Services refresh error. **Details:** This entry indicates that a problem occurred during a Topology Services refresh operation. A refresh operation can be a result of a configuration change, such as adding or deleting a node in the cluster, or changing characteristics of a communication group. It can also be the result of the **cthatstune -r** command. In PowerHA SystemMirror/ES, a refresh occurs as a result of synchronizing topology changes in a cluster. This entry refers to a particular instance of the Topology Services daemon on the local node. On PowerHA SystemMirror, or in an RSCT peer domain, the problem may have occurred in other nodes as well. Standard fields indicate that a refresh error occurred. The **machines.lst** file has some incorrect information. The problem is probably created during a migration-refresh on an PowerHA SystemMirror node by node migration. Data used to build the **machines.lst** file is stored in the daemon's "run" directory and may be lost if Topology Services is restarted or a new refresh is attempted. More details about the problem are found in the User log file. See “Topology Services user log” on page 158. Additional details are stored in the Service log. See “Topology Services service log” on page 158. See “Information to collect before contacting the IBM Support Center” on page 16 and contact the IBM Support Center.
TS_RSOCK_ER	PERM	**Explanation:** The daemon failed to open socket for peer daemon communication. **Details:** This entry indicates that the Topology Services daemon was unable to open a UDP socket for communication with peer daemons in other nodes. Standard fields indicate that the daemon was unable to open the socket. Detail Data fields describe the operation being attempted at the socket (in English), the reason for the error, the system error value, and the port number. The port number may be in use by either another subsystem or by another instance of the Topology Services daemon. If the SRC subsystem loses its connection to the Topology Services daemon, the SRC may erroneously allow a second instance of the daemon to be started, leading to this error. The situation that caused the problem may clear with a node reboot. Follow the procedures described for the "Nodes or adapters leave membership after refresh" symptom in “Error symptoms, responses, and recoveries” on page 178 to find a possible Topology Services daemon running at the node and stop it. If no process is found that is using the peer socket, see “Information to collect before contacting the IBM Support Center” on page 16 and contact the IBM Support Center. Include also a System Dump.

Table 25. Error log templates for Topology Services (continued)

Label	Type	Description
TS_SECURITY_ST	INFO	**Explanation:** Authentication failure in Topology Services. **Details:** This entry indicates that the Topology Services daemon cannot authenticate a message from one of the peer daemons running in a remote node. This entry refers to a particular instance of the Topology Services daemon on the local node. The node which is sending these messages must also be examined. Standard fields indicate that a message cannot be authenticated. Detail Data fields show the source of the message. The possible problems are: • There is an attempt at a security breach. • The Time-Of-Day clocks in the nodes are not synchronized. • There are stale packets flowing through the network. • IP packets are being corrupted. • The security key file is not in sync across all nodes in the domain. An entry is created the first time a message cannot be authenticated. After that, entries are created less frequently. Information about the network must be collected while the messages are still being received. The command **tcpdump** should be used to examine the packets arriving at the node. Perform the following steps: 1. Examine the output of the **lssrc -ls hats** command (PSSP) or **lssrc -ls cthats** (RSCT peer domain) on the local node and on the node sending the message. Look for field "Key version" in the output and check whether the numbers are the same on both nodes. 2. Check that the key file is the same in all the nodes in the domain.
TS_SECURITY2_ST	INFO	**Explanation:** More authentication failures in Topology Services. **Details:** This entry indicates that there have been additional incoming messages that could not be authenticated. For the first such message, error log entry **TS_SECURITY_ST** is created. If additional messages cannot be authenticated, error log entries with label **TS_SECURITY2_ST** are created less and less frequently. The standard fields indicate that incoming messages cannot be authenticated. The detailed fields show an interval in seconds and the number of messages in that interval that could not be authenticated. For more details and diagnosis steps, see the **TS_SECURITY_ST** label entry.
TS_SEMGET_ER	PERM	**Explanation:** Cannot get shared memory or semaphore segment. This indicates that the Topology Services daemon was unable to start because it could not obtain a shared memory or semaphore segment. This entry refers to a particular instance of the Topology Services daemon on the local node. The daemon exits **Details:** Standard fields indicate that the daemon could not start because it was unable to get a shared memory or a semaphore segment. The Detail Data fields contain the key value and the number of bytes requested for shared memory, or the system call error value for a semaphore. The reason why this error has occurred may not be determined if the subsystem is restarted and this error no longer occurs.

Table 25. Error log templates for Topology Services (continued)

Label	Type	Description
TS_SERVICE_ER	PERM	**Explanation:** Unable to obtain port number from the **/etc/services** file. **Details:** This entry indicates that the Topology Services daemon was unable to obtain the port number for daemon peer communication from **/etc/services** . This entry refers to a particular instance of the Topology Services daemon on the local node. The daemon exits. Other nodes may be affected if their **/etc/services** have similar contents as that on the local node. Standard fields indicate that the daemon was unable to obtain the port number from **/etc/services** . Detail Data fields show the service name used as search key to query **/etc/services** .
TS_SHMAT_ER	PERM	**Explanation:** Cannot attach to shared memory segment. **Details:** This entry indicates that the Topology Services daemon was unable to start because it could not attach to a shared memory segment. Standard fields indicate that the daemon could not start because it was unable to attach to a shared memory segment. The daemon exits. The Detail Data fields contain the shared memory identifier and number of bytes requested. The reason why the error occurred may not be found if the subsystem is restarted and the same error does not occur.
TS_SHMEMKEY_ER	PERM	**Explanation:** Cannot get IPC key. **Details:** This indicates that the Topology Services daemon was unable to start because it could not obtain an IPC key. This refers to a particular instance of the Topology Services daemon on the local node. The daemon exits. Standard fields indicate that the daemon could not start because it was unable to obtain an IPC key. The Detail Data fields contain the path name of the UNIX-domain socket used for daemon-client communication. This path name is given to the **ftok()** subroutine in order to obtain an IPC key. This entry is created when the UNIX-domain socket file has been removed. The reason why this error has occurred may not be determined if the subsystem is restarted and this error no longer occurs.
TS_SHMGET_ER	PERM	**Explanation:** Cannot create directory. **Details:** This entry indicates that the Topology Services startup script **cthats** was unable to create one of the directories it needs for processing. Standard fields indicate that a directory could not be created by the startup script **cthats**. Detail Data fields show the directory that could not be created. Information about the cause of the problem may not be available once the problem is cleared.
TS_SPIPDUP_ER	PERM	See **TS_HAIPDUP_ER**.
TS_SPLOCAL_ER	PERM	See **TS_HALOCAL_ER**.
TS_SPNODEDUP_ER	PERM	See **TS_HANODEDUP_ER**.
TS_START_ST	INFO	**Explanation:** The Topology Services daemon has started. This is an indication that the Topology Services daemon has started. This entry refers to a particular instance of the Topology Services daemon on the local node. **Details:** Standard fields indicate that the daemon started. The Topology Services subsystem was started by a user or during system boot. Detail Data will be in the language where the **errpt** (or **fcslogrpt**) command is run. The Detail Data contains the location of the log and run directories and also which user or process started the daemon.

Table 25. Error log templates for Topology Services (continued)

Label	Type	Description
TS_STOP_ST	INFO	**Explanation:** The Topology Services daemon has stopped. This is an indication that the Topology Services daemon has stopped. This entry refers to a particular instance of the Topology Services daemon on the local node. **Details:** The Topology Services subsystem shutdown was caused by a signal sent by a user or process. Standard fields indicate that the daemon stopped. The standard fields are self-explanatory. If stopping the daemon is not desired, you must quickly understand what caused this condition. If the daemon was stopped by the SRC, the word "SRC" is present in the Detail Data . The REFERENCE CODE field in the Detail Data section refers to the error log entry for the start of Topology Services. Detail Data is in English. Detail Data fields point to the process (SRC) or signal that requested the daemon to stop.
TS_THATTR_ER	PERM	**Explanation:** Cannot create or destroy a thread attributes object. **Details:** This entry indicates that Topology Services was unable to create or destroy a thread attributes object. Standard fields indicate that the daemon was unable to create or destroy a thread attributes object. Detail Data fields show which of the Topology Services threads was being handled. The Topology Services daemon exits. See “Information to collect before contacting the IBM Support Center” on page 16 and contact the IBM Support Center.
TS_THCREATE_ER	PERM	**Explanation:** Cannot create a thread. **Details:** This entry indicates that Topology Services was unable to create one of its threads. Standard fields indicate that the daemon was unable to create a thread. Detail Data fields show which of the Topology Services threads was being created.
TS_THREAD_STUCK_ER	PERM	**Explanation:** Main thread is blocked. Daemon will exit. **Details:** This entry indicates that the Topology Services daemon will exit because its main thread was blocked for longer than a pre-established time threshold. If the main thread remains blocked for too long, it is possible that the node is considered dead by the other nodes. The main thread needs to have timely access to the CPU, otherwise it would fail to send "heartbeat" messages, run adapter membership protocols, and notify Group Services about adapter and node events. If the main thread is blocked for too long, the daemon exits with a core dump, to allow debugging of the cause of the problem. This entry refers to a particular instance of Topology Services running on a node. The standard fields indicate that the Topology Services daemon will exit because the main thread was blocked for too long, and explains some of the possible causes. The detailed fields show the number of seconds that the main thread appeared to be blocked, the number of recent page faults involving I/O operations, and the interval in milliseconds where these page faults occurred. If the number of page faults is non-zero, the problem could be related to memory contention. For information about diagnosing and working around the problem in case its root cause is a resource shortage, see “Action 5: investigate a hatsd problem” on page 181. If a resource shortage does not seem to be a factor, the cause could be a problem in the daemon or in a service invoked by it. Contact the IBM Support Center.

Table 25. Error log templates for Topology Services (continued)

Label	Type	Description
TS_UNS_SIN_TR	UNKN	**Explanation:** Local adapter in unstable singleton state. **Details:** This entry indicates that a local adapter is staying too long in a singleton unstable state. Though the adapter is able to receive some messages, there could be a problem with it, which may prevent outgoing messages from reaching their destinations. This entry refers to a particular instance of the Topology Services daemon on the local node. Examine the Service log on other nodes to determine if other nodes are receiving messages from this adapter. See "Topology Services service log" on page 158. Standard fields indicate that a local adapter is in an unstable singleton state. Detail Data fields show the interface name, adapter offset (index of the network in the **machines.lst** file), and the adapter address according to Topology Services, which may differ from the adapter's actual address if the adapter is incorrectly configured. The adapter may be unable to send messages. The adapter may be receiving broadcast messages but not unicast messages. Information about the adapter must be collected while the adapter is still in this condition. Issue the commands: **ifconfig** *interface_name* and **netstat -in** and record the output. Perform these steps: 1. Check if the address displayed in the error report entry is the same as the actual adapter address, which can be obtained by issuing this command: **ifconfig** *interface_name*. If they are not the same, the adapter has been configured with the wrong address. 2. Issue command **ping***address* from the local node for all the other addresses in the same network. If **ping** indicates that there is no reply (for example: `10 packets transmitted, 0 packets received, 100% packet loss`) for all the destinations, the adapter may be incorrectly configured. 3. See "Operational test 6: check whether the adapter can communicate with other adapters in the network" on page 170.

Dump and snapshot information

This topic describes the core dump and snapshot information pertinent to Topology Services.

Core dump

The Topology Services daemon generates a core dump. The dump contains information normally saved in a core dump: user-space data segments for the Topology Services daemon.

The core dump refers to a particular instance of the Topology Services daemon on the local node. Other nodes may have a similar core dump. The core dump file will be located in the *run_dir* directory. An approximate size for the core dump file is between 7MB and 10MB.

When the Topology Services daemon invokes an **assert()** statement, or when it receives a segmentation violation signal for accessing its data incorrectly, it creates the dump automatically. Force Topology Services to generate a dump only under the direction of the IBM Support Center, as the daemon has an internal check to protect against getting hung. (See the TS_THREAD_STUCK_ER error entry in Table 25 on page 138. When directed to do so, you can create the dump manually by issuing the following command:

```
kill -6 pid_of_daemon
```

You can obtain the *pid_of_daemon* by issuing the following command:

```
lssrc -s subsystem_name
```

The dump remains valid as long as the executable file **/usr/sbin/rsct/bin/hatsd** is not replaced. The system keeps only the last three core file instances. Copy the core dumps and the executable to a safe place.

Table 26 on page 156 describes how to analyze core dumps. These procedures vary by node type, as follows:

Table 26. Dump analysis of Linux, AIX, Windows, and Solaris nodes

On Linux nodes:	On AIX nodes:	On Windows nodes:	On Solaris nodes:
To analyze the core dump, issue the command: `gdb /usr /sbin/rsct/bin /hatsd core_file`	To analyze the core dump, issue the command: `dbx /usr /sbin/rsct/bin /hatsd core_file` **Good results** are similar to the following: `Type 'help' for help.` `reading symbolic information ...` `[using memory image in core]` `IOT/Abort trap in evt._pthread_ksleep [/usr/lib/libpthreads.a] at 0xd02323e0 ($t6)` `0xd02323e0 (_pthread_ksleep+0x9c) 80410014 lwz r2,0x14(r1)` Some of the error results are: 1. This means that the current executable file was not the one that created the core dump. `Type 'help' for help.` `Core file program (hatsd) does not match current program (core ignored)` `reading symbolic information ... (dbx)` 2. This means that the core file is incomplete due to lack of disk space. `Type 'help' for help.` `warning: The core file is truncated. You may need to increase the ulimit for file and coredump, or free some space on the filesystem.` `reading symbolic information ...` `[using memory image in core]` `IOT/Abort trap in evt._pthread_ksleep [/usr/lib/libpthreads.a] at 0xd02323e0` `0xd02323e0 (_pthread_ksleep +0x9c) 80410014 lwz r2,0x14(r1)` `(dbx)`	To analyze the core dump, issue the command: `$ gdb /usr /sbin/ rsct/bin/hatsd core_file` A possible error result is: 1. This warning means that the current executable file was not the one that created the core dump. `Rerun gdb using /usr /sbin/rsct /bin/hats_nim instead of /usr /sbin/rsct/bin/hatsd ...` `warning: core file may not match specified executable file.` `Core was generated by `/usr/sbin/rsct /bin/hats_nim'.` `Cannot access memory at address 0x0` `#0 0x7c82ed54 in ?? ()` `(gdb)`	To analyze the core dump, issue the command: `dbx /usr/sbin /rsct/bin/hatsd core_file` **Good results** are similar to the following: `Type 'help' for help.` `reading symbolic information ...` `[using memory image in core]` `IOT/Abort trap in evt._pthread_ksleep [/usr/lib/libpthr eads.a] at 0xd02323e0 ($t6)` `0xd02323e0 (_pthread_ksleep +0x9c) 80410014 lwz r2,0x14(r1)` Some of the error results are: 1. This means that the current executable file was not the one that created the core dump. `Type 'help' for help.` `dbx: core object name "hatsd" doesn't match object name " xxx"` `core file ignored.` `Use -f to force loading of corefile` `dbx: warning: core file header read failed` 2. This means that the core file is incomplete due to lack of disk space. `Type 'help' for help.` `warning: The core file is truncated. You may need to increase the ulimit for file and coredump, or free some space on the filesystem.` `reading symbolic information ...`

Snapshot
A snapshot is a collection of configuration data, log and trace files, and other diagnostic data for the RSCT components used for problem determination.

Follow the directions in "Taking a snapshot" on page 11 to manually produce a snapshot. When run on a node that is using Topology Services, a snapshot will automatically gather any existing core dumps as part of its data collection.

Related tasks:

"Information to collect before contacting the IBM Support Center" on page 16
There are things for you to do before you contact the IBM Support Center.

Trace information
Consult these logs for debugging purposes. Each refers to a particular instance of the Topology Services daemon running on the local node.

Do *not* activate this trace facility until you have read this section completely, and understand this material. If you are not certain how to properly use this facility, or if you are not under the guidance of IBM Service, do **not** activate this facility.

Activating this facility may result in degraded performance of your system. Activating this facility may also result in longer response times, higher processor loads, and the consumption of system disk resources. Activating this facility may also obscure or modify the symptoms of timing-related problems.

Topology Services startup log
The Topology Services startup log is located in *startup_log*.

It contains the output from the *startup_script*, including a copy of the configuration data used to build the *machines.lst*.

It also contains error messages if the script was unable to produce a valid machines.lst and start the daemon. The startup script is run at subsystem startup time and at refresh time. This log refers to a particular instance of the startup script running on the local node.

The size of the file varies according to the size of the machine. It is about 500 bytes in size for a three-node system, and is larger for systems with more nodes. A new instance of the startup log is created each time the startup script is run. A copy of the log is made just before the script exits. Only the last seven instances of the log file are kept and they are named *startup_log*.**1** through *startup_log*.**7**. Therefore, the contents of the log must be saved before the subsystem is restarted or refreshed many times. The **.1** instance is an identical copy of the current startup log. At each startup, **.1** is renamed to **.2**; **.2** is renamed to **.3**, and so on. Therefore, the previous **.7** instance is lost.

Entries in the startup script log are kept both in English and in the node's language (if different). Trace records are created for these conditions:

- The *machines.lst* file is built or retrieved from whichever source is used for the cluster type.
- An error is encountered that prevents the *startup_script* from making progress.

There is no fixed format for the records of the log. The following information is in the file:

- The date and time when the *startup_script* started running
- A copy of *machines.lst* file generated
- The date and time when the *startup_script* finished running
- If the script was called for a refresh operation, the output of the **refresh** command is included in the log file.

The main source for diagnostics is the error log. The startup log should be used when the error log shows that the startup script was unable to complete its tasks and start the daemon.

Related tasks:

"Action 2: investigate refresh failure" on page 179
A number of possible conditions suggest this action. Take this action when the refresh operation fails or has no effect.

Topology Services user log

The Topology Services user log is located in *usr_log*.

It contains error and informational messages produced by the daemon.

This trace is always running. It has negligible impact on the performance of the system, under normal circumstances.

Data in the user log is written in the language specified where the daemon is run, which is the node's administrative language. Messages in the user log have a catalog message number, which can be used to obtain a translation of the message in the desired language.

The size of the log file is changed using the same commands that change the size of the service log. Truncation of the log, saving of log files, and other considerations are the same as for the service log.

Each user log entry has this format:

```
date     daemon_name     message
```

Adapters are identified by a pair:

```
(IP address:incarnation number)
```

Groups are identified by a pair:

```
(IP address of Group Leader:incarnation number of group)
```

The main source for diagnostics is the error log. Some of the error messages produced in the user log occur under normal circumstances. If they occur repeatedly, however, they indicate an error. Some error messages give additional detail for an entry in the error log. Therefore, this log file should be examined when an entry is created in the system error log.

Topology Services service log

The Topology Services service log is located in *svc_log*.

It contains trace information about the activities performed by the daemon.

When a problem occurs, logs from multiple nodes will often be needed. These log files must be collected before they wrap or are removed.

If obtaining logs from all nodes is not feasible, the following is a list of nodes from which logs should be collected:

1. The node where the problem was seen
2. The group leader node on each network

   The Group Leader is the node which has the highest IP address on a network.
3. The downstream neighbor on each network

This is the node whose IP address is immediately lower than the address of the node where the problem was seen. The node with the lowest IP address has a downstream neighbor of the node with the highest IP address.

4. The upstream neighbor on each network

   This is the node whose IP address is immediately higher than the address of the node where the problem was seen. The node with the highest IP address has an upstream neighbor of the node with the lowest IP address.

Data in the service log is in English. Each service log entry has this format:

```
date     daemon_name     message
```

Adapters are identified by a pair:

```
(IP address:hexadecimal incarnation number)
```

Groups are identified by a pair:

```
(IP address of group leader:hexadecimal incarnation number of group )
```

When the log file reaches the maximum line number, the current log is saved in a file with a suffix of **.bak** and the original file is truncated. When the daemon is restarted, a new log file is created. Only the last five log files are retained.

Service log normal tracing:

Service log normal tracing is the default and is always running. There is negligible impact if no node or adapter events occur on the system.

An adapter death event may result in approximately 50 lines of log information for the group leader and "mayor" nodes, or up to 250 lines for the Group Leader and "mayor" nodes on systems of approximately 400 nodes. All other nodes will produce less than 20 lines. You can increase log file sizes as described in “Changing the service log size” on page 160.

Normal tracing generates trace records for the following conditions:

- Each adapter that is disabled or re-enabled
- Some protocol messages sent or received
- Refresh
- Client requests and notifications
- Groups formed, members added and removed

Normal tracing generates no entries when no adapter or node events occur on the system.

With normal tracing, the log trimming rate depends heavily on the frequency of adapter or node events on the system. If the service log file, using normal tracing, continues to grow, even when no events appear to be happening on the system, a problem may be indicated. Search for possible entries in the syslog or in the user log. See “Topology Services user log” on page 158.

Service log long tracing:

The most detailed level of tracing is service log long tracing.

It is started with the command:

```
ctrl_script -t
```

The long trace is stopped with the command:

```
ctrl_script -o
```

which causes normal tracing to be in effect.

With service log long tracing, trace records are generated for the following conditions:

- Each message sent or received
- Each adapter that is disabled or re-enabled
- Details of protocols being run
- Details of node reachability information
- Refresh
- Client requests and notifications
- Groups formed, elements added and removed

Activate long tracing only on request from the IBM Support Center. It can be activated just for a few minutes (to avoid overwriting other data in the log file) when the error condition is still present.

Changing the service log size:

The long trace generates approximately 10KB of data per minute of trace activity. By default, log files have a maximum of 5000 lines, which will be filled in 30 minutes or less if long tracing is requested.

The method for changing the log file size depends on the cluster type.

To change the log file size on an RPD cluster, issue the following command on any node:

```
/usr/sbin/rsct/bin/cthatstune -l new_max_lines -r
```

Example: The command **cthatstune -l 10000 -r** changes the maximum number of lines in a log file to 10•000. The **-r** flag causes the Topology Services subsystem to be refreshed on all of the nodes.

To change the log file size on an PowerHA SystemMirror cluster, use the PowerHA SystemMirror **smit** panels on any node:

1. Enter: `smit hacmp`
2. In SMIT, select **Extended Configuration > Extended Topology Configuration > Configure Topology Services and Group Services > Change/Show Topology and Group Services configuration** and press the *Enter* key.

   Result: SMIT displays the **Change/Show Topology and Group Services Configuration** panel.
3. Enter the new log size, in lines, in the **Topology Services log length (lines)** field.

After making the change, synchronize the cluster from that node.

Note: As with most cluster changes, this can be done with a dynamic sync while the cluster is active but it is preferable to have the cluster down, if possible.

To change the log file size on a PSSP cluster, issue the following command on the control workstation:

```
/usr/sbin/rsct/bin/hatstune -l new_max_lines -r
```

Example: The command **hatstune -l 10000 -r** changes the maximum number of lines in a log file to 10•000. The **-r** flag causes the Topology Services subsystem to be refreshed on all of the nodes.

Network interface module (NIM) log

The network interface module log is located in *nim_log*.

It contains trace information about the activities of the network interface modules (NIMs), which are processes used by the Topology Services daemon to monitor each network interface. These logs need to be collected before they wrap or are removed.

There will be a separate log for each NIM that is running on the system, which equates to one for each adapter being monitored locally by Topology Services. Each NIM will keep four instances of its log – the current and three previous (*nim_log***.001**, *nim_log***.002**, and *nim_log***.003**). When the current log file is full, log file **.003** is overwritten by **.002**, **.002** is overwritten by **.001**, and **.001** is overwritten by the current log file to make room for a new one.

Trace records are generated for the following conditions:

- A connection with a given adapter is established.
- A connection with a given adapter is closed.
- A daemon has sent a command to start or stop heartbeating.
- A daemon has sent a command to start or stop monitoring heartbeats.
- A local adapter goes up or down.
- A message is sent or received.
- A heartbeat from the remote adapter has been missed

Data in the NIM log is in English only. The format of each message is:

```
time-of-day    message
```

At default logging levels, an instance of the NIM log file will wrap when the file reaches approximately 200 KB. Normally, it takes about 10 minutes to fill an instance of the log file. Since three instances are kept, the NIM log files need to be saved within 30 minutes of when the adapter-related problem occurred. If a higher level of NIM tracing has been enabled under the direction of the IBM Support Center, the wrapping size of the NIM logs will automatically be increased to accommodate the extra logging and could grow as large as 800 KB, depending on the debugging level.

Diagnostic procedures

These tests verify the configuration and operation of Topology Services.

To verify that RSCT has been installed, see the chapter entitled *Verifying RSCT installation* in the *Administering RSCT* guide.

Configuration verification test

This test verifies that Topology Services has the configuration data it needs to build the *machines.lst* file.

The configuration data is propagated by the configuration resource manager and can be retrieved with the commands:

- **/usr/sbin/rsct/bin/ct_clusterinfo**
- **/usr/sbin/rsct/bin/ct_hats_info**
- **/usr/sbin/rsct/bin/ct_topology_info**

The output of **ct_clusterinfo** is similar to the following:

```
CLUSTER_NAME  gpfs
 CLUSTER_ID    b181ecec-7055-4374-a998-ccd3f71db16a
 NODE_NUMBER   2
```

The node number information is probably the most important.

The output of **ct_hats_info** is similar to the following:

```
REALM CLUSTER
LOGFILELEN 5000
FIXED_PRI -1
PORT 12347
PIN NONE
```

This command displays overall options for Topology Services. Any "-1" or "DEFAULT" values will prompt the Topology Services scripts to use appropriate default values.

- REALM: execution environment. Should be always "CLUSTER".
- LOGFILELEN: maximum number of lines in the Topology Services daemon log file.
- FIXED_PRI: fixed priority value.
- PORT: UDP port number for peer-to-peer communication.
- PIN: whether to pin the Topology Services daemon in memory.

The output of **ct_topology_info** is similar to the following:

```
NETWORK_NAME gpfs
NETWORK_SENS -1
NETWORK_NIM_PAR
NETWORK_BCAST 0
NETWORK_NIM_EXEC
NETWORK_SRC_ROUTING 0
NETWORK_FREQ -1
NETWORK_TYPE myrinet
ADAPTER 192.168.1.43 myri0 1 gpfs
ADAPTER 192.168.1.44 myri0 2 gpfs
```

The output has a section for each of the configured networks. For each network, tunable information is given, along with a list of all the adapters in the network. For each adapter, its IP address, interface name, node number, and network to which it belongs are given. Note that the node number for each node is given by the output of the **ct_clusterinfo** command.

The tunable values for each network are:

- NETWORK_FREQ: "frequency" value: how often to send heartbeat messages in seconds.
- NETWORK_SENS: "sensitivity" value: how many missed heartbeats before declaring the adapter dead.
- NETWORK_NIM_EXEC: Path name for NIM executable file.
- NETWORK_NIM_PAR: command-line argument to NIM.
- NETWORK_BCAST: 1 if network supports broadcast; 0 otherwise.
- NETWORK_SRC_ROUTING: 1 if network supports IP loose source routing, 0 otherwise.

Good results are indicated by the configuration, in terms of tunable values and network configuration, matching the user expectation for the cluster topology.

Error results are indicated if there is any inconsistency between the displayed configuration data and the desired configuration data. Issue the **cthatstune** command with the desired values.

Operational verification tests

The following names apply to the operational verification tests in this section.

In a configuration resource manager environment (RSCT peer domain):

- Subsystem name: **cthats**
- User log file: **/var/ct/***cluster_name***/log/cthats/cthats.** *DD.hhmmss.lang*
- Service log file: **/var/ct/***cluster_name* **/log/cthats/cthats.***DD.hhmmss*
- **run** directory: **/var/ct/***cluster_name***/run/cthats**
- **machines.lst** file: **/var/ct/***cluster_name***/run/cthats/machines.lst**

On AIX nodes, in an PowerHA SystemMirror environment:

- Subsystem name: **topsvcs**
- User log file: **/var/ha/log/topsvcs**.*DD.hhmmss.cluster_name.lang*
- Service log file: **/var/ha/log/topsvcs**.*DD.hhmmss.cluster_name*
- **run** directory: **/var/ha/run/topsvcs**.*cluster_name/*
- **machines.lst** file: **/var/ha/run/topsvcs**.*cluster_name/***machines**.*cluster_id* .**lst**

Operational test 1: verify status and adapters:

This test verifies whether Topology Services are working and that all of the adapters are up. Use this test when adapter membership groups do not include all of the nodes in the configuration.

Issue the **lssrc** command:

```
lssrc -ls cthats
```

Good results are indicated by output that looks like this:

```
Subsystem        Group            PID     Status
cthats           cthats           20494    active

Network Name   Indx Defd Mbrs St Adapter ID      Group ID
ethernet1      [ 0]   15   15  S 9.114.61.195    9.114.61.195
ethernet1      [ 0] eth0         0x3740dd5c      0x3740dd62
HB Interval = 1 secs. Sensitivity = 4 missed beats
SPswitch       [ 1]   14   14  S 9.114.61.139    9.114.61.139
SPswitch       [ 1] css0         0x3740dd5d      0x3740dd62
HB Interval = 1 secs. Sensitivity = 4 missed beats
  Configuration Instance = 926566126
  Default: HB Interval = 1 secs. Sensitivity = 4 missed beats
  Daemon employs no security
  Data segment size: 6358 KB. Number of outstanding malloc: 588
  Number of nodes up: 15. Number of nodes down: 0.
```

If the number under the Members heading (`Mbrs`) is the same as the number under the Defined heading (`Defd`), all adapters defined in the configuration are part of the adapter membership group. The numbers under the `Group ID` heading should remain the same over subsequent invocations of **lssrc** several seconds apart. This is the expected behavior of the subsystem.

Error results are indicated by outputs that are similar to the following:

1. `0513-036 The request could not be passed to the cthats subsystem. Start the subsystem and try your command again.`

   In this case, the subsystem is down. Issue the **errpt -a** command and look for an entry for the subsystem name. See "Accessing logged errors" on page 2 to check the system log and look for an entry for the subsystem name. Proceed to "Operational test 2: determine why the Topology Services subsystem is inactive" on page 166.
2. `0513-085 The cthats Subsystem is not on file.`

The subsystem is not defined to the SRC.

3. This output requires investigation because the number under `Mbrs` is smaller than the number under `Defd`.

```
Subsystem         Group            PID     Status
cthats            cthats           20494   active

Network Name   Indx Defd Mbrs St Adapter ID      Group ID
ethernet1      [ 0]   15    8  S 9.114.61.195    9.114.61.195
ethernet1      [ 0] eth0          0x3740dd5c      0x3740dd62
HB Interval = 1 secs. Sensitivity = 4 missed beats
SPswitch       [ 1]   14    7  S 9.114.61.139    9.114.61.139
SPswitch       [ 1] css0          0x3740dd5d      0x3740dd62
HB Interval = 1 secs. Sensitivity = 4 missed beats
  Configuration Instance = 926566126
  Default: HB Interval = 1 secs. Sensitivity = 4 missed beats
  Daemon employs no security
  Data segment size: 6358 KB. Number of outstanding malloc: 588
  Number of nodes up: 8. Number of nodes down: 7.
  Nodes down: 17-29(2)
```

Some remote adapters are not part of the local adapter's group. Proceed to "Operational test 3: determine why remote adapters are not in the local adapter's membership group" on page 167.

4. This output requires investigation because a local adapter is disabled.

```
Subsystem         Group            PID     Status
cthats            cthats           20494   active

Network Name   Indx Defd Mbrs St Adapter ID      Group ID
ethernet1      [ 0]   15   15  S 9.114.61.195    9.114.61.195
ethernet1      [ 0] eth0          0x3740dd5c      0x3740dd62
HB Interval = 1 secs. Sensitivity = 4 missed beats
SPswitch       [ 1]   14    0  D 9.114.61.139
SPswitch       [ 1] css0          adapter_state_information
HB Interval = 1 secs. Sensitivity = 4 missed beats
  Configuration Instance = 926566126
  Default: HB Interval = 1 secs. Sensitivity = 4 missed beats
  Daemon employs no security
  Data segment size: 6358 KB. Number of outstanding malloc: 588
  Number of nodes up: 15. Number of nodes down: 0.
```

When a network adapter is in the disabled state, **lssrc** provides additional state information to identify the reason why the adapter is down. This state information appears after the adapter interface name in the **lssrc -ls cthats** command output. The following are the possible values for *adapter_state_information* and their explanations.

Adapter state	Explanation
Adapter state unknown	This is the initial value for the adapter state before any determination has been done.
No traffic on adapter	The adapter has no incoming traffic.
Adapter's interface flags set to down	The adapter's interface flags have been set to down.
Adapter is misconfigured	There is a problem with the adapter's configuration, such as a missing or incorrect adapter address.
Broadcast address is misconfigured	The configured broadcast address is inconsistent with the adapter's IP address and subnet mask. Broadcast addresses apply to IPv4 networks only.
Adapter is not monitored	The adapter is intentionally not being monitored.
Adapter has no NIM running	The adapter has no living network interface module (NIM) associated with it.
The adapter has no living network interface module (NIM) associated with it	Indicates an error from the netmon library, which is used to monitor adapter status.
NIM could not bind UDP socket	The NIM was unable to bind to the UDP socket, possibly because the port is already in use.

Adapter state	Explanation
NIM could not open device	A non-IP NIM was unable to open the device.

A local adapter is disabled. Proceed to "Operational test 4: check the address of the local adapter" on page 168.

5. This output requires investigation because there is a **U** below the `St` heading.

```
Subsystem       Group           PID     Status
ctchats         cthats          20494   active

Network Name   Indx Defd Mbrs St Adapter ID      Group ID
ethernet1      [ 0]   15    8  S 9.114.61.195    9.114.61.195
ethernet1      [ 0] eth0          0x3740dd5c      0x3740dd62
HB Interval = 1 secs. Sensitivity = 4 missed beats
SPswitch       [ 1]   14    1  U 9.114.61.139    9.114.61.139
SPswitch       [ 1] css0         0x3740dd5d      0x3740dd5d
HB Interval = 1 secs. Sensitivity = 4 missed beats
  Configuration Instance = 926566126
  Default: HB Interval = 1 secs. Sensitivity = 4 missed beats
  Daemon employs no security
  Data segment size: 6358 KB. Number of outstanding malloc: 588
  Number of nodes up: 8. Number of nodes down: 7.
  Nodes down: 17-29(2)
```

The last line of the output shows a list of nodes that are either up or down, whichever is smaller. The list of nodes that are down includes only the nodes that are configured and have at least one adapter that Topology Services monitors. Nodes are specified by a list of node ranges, as follows:

N1`-`*N2*`(`*I1*`)  `*N3*`-`*N4*`(`*I2*`) ...`

Here, there are two ranges, *N1*-*N2*(*I1*) and *N3*-*N4*(*I2*). They are interpreted as follows:

- *N1* is the first node in the first range
- *N2* is the last node in the first range
- *I1* is the increment for the first range
- *N3* is the first node in the second range
- *N4* is the last node in the second range
- *I2* is the increment for the second range

If the increment is 1, it is omitted. If the range has only one node, only that node's number is displayed. Examples are:

a. `Nodes down: 17-29(2)` means that nodes 17 through 29 are down. In other words, nodes 17, 19, 21, 23, 25, 27, and 29 are down.

b. `Nodes up: 5-9(2) 13` means that nodes 5, 7, 9, and 13 are up.

c. `Nodes up: 5-9 13-21(4)` means that nodes 5, 6, 7, 8, 9, 13, 17, and 21 are up.

An adapter stays in a singleton unstable membership group. This normally occurs for a few seconds after the daemon starts or after the adapter is re-enabled. If the situation persists for more than one minute, this may indicate a problem. This usually indicates that the local adapter is receiving some messages, but it is unable to obtain responses for its outgoing messages. Proceed to "Operational test 7: check for partial connectivity" on page 172.

6. An output similar to the expected output, or similar to output 3, but where the numbers under the `Group ID` heading (either the address of the Group Leader adapter or the "incarnation number" of the group) change every few seconds without ever becoming stable.

   This kind of output indicates that there is some partial connectivity on the network. Some adapters may be able to communicate only with a subset of adapters. Some adapters may be able to send messages only or receive messages only. This output indicates that the adapter membership groups are constantly reforming, causing a substantial increase in the CPU and network resources used by the subsystem.

A partial connectivity situation is preventing the adapter membership group from holding together. Proceed to "Operational test 10: check neighboring adapter connectivity" on page 175.

If this test is successful, proceed to "Operational test 11: verify node reachability information" on page 176.

Operational test 2: determine why the Topology Services subsystem is inactive:

Use this test is to determine why the Topology Services subsystem might be inactive.

Table 27 details those tests useful in determining why the Topology Services subsystem may be inactive. These tests vary by node type, as follows:

Table 27. Test Topology Services subsystem inactivity on AIX, Linux, Solaris, and Windows nodes

On AIX nodes:	On Linux nodes:	On Solaris nodes:	On Windows nodes:

For PowerHA SystemMirror/ES, enter:
`errpt -N topsvcs -a`
 	
For an RSCT peer domain, enter:
`errpt -N cthats -a`
	Issue the command:
	`fcslogrpt /var/log/messages`
	and look for entries for subsystem **cthats**.
		For an RSCT peer domain, issue the command:
		`grep cthats /var/adm/messages > cthats.out`
			For an RSCT peer domain, issue the command:
			`grep cthats /var/adm/log/messages > cthats.out`

The AIX error log entries produced by this command, together with their descriptions in Table 25 on page 138, explain why the subsystem is inactive. If no entry that explains why the subsystem went down or could not start exists, the daemon might have exited abnormally.	The syslog entries produced by this command, together with their descriptions in Table 25 on page 138, explain why the subsystem is inactive. If no entry exists that explains why the subsystem went down or could not start, it is possible that the daemon may have exited abnormally.	The Solaris system log entries produced by this command, together with their descriptions in Table 25 on page 138 explain why the subsystem is inactive. If no entry that explains why the subsystem went down or could not start exists, it is possible that the daemon may have exited abnormally.	The Windows system log entries produced by this command, together with their descriptions in Table 25 on page 138 explain why the subsystem is inactive. If no entry that explains why the subsystem went down or could not start exists, it is possible that the daemon may have exited abnormally.
In this case, issue the **errpt -a** command and look for an error. Look for an error entry with a LABEL: of CORE_DUMP and PROGRAM NAME of **hatsd**. (Issue the command: **errpt -J CORE_DUMP -a.**) If you find such an entry, see "Information to collect before contacting the IBM Support Center" on page 16 and contact the IBM Support Center.	In this case, issue the **fcslogrpt /var/log/messages** command and look for an error. Look for an error entry with a LABEL: of CORE_DUMP and PROGRAM NAME of **hatsd**. If such an entry is found, see "Information to collect before contacting the IBM Support Center" on page 16 and contact the IBM Support Center.	In this case, issue the **cat /var/adm/messages** command and look for an error. Look for an error entry with a LABEL: of CORE_DUMP and PROGRAM NAME of **hatsd**. If you find such an entry, see "Information to collect before contacting the IBM Support Center" on page 16 and contact the IBM Support Center.	In this case, issue the **cat /var/adm/log/messages** command and look for an error. Look for an error entry with a LABEL: of CORE_DUMP and PROGRAM NAME of **hatsd**. If you find such an entry, see "Information to collect before contacting the IBM Support Center" on page 16 and contact the IBM Support Center.

Table 27. Test Topology Services subsystem inactivity on AIX, Linux, Solaris, and Windows nodes (continued)

On AIX nodes:	On Linux nodes:	On Solaris nodes:	On Windows nodes:

Another possibility, when there is no **TS_** error log entry, is that the Topology Services daemon could not be loaded. In this case, a message similar to the following may be present in the Topology Services user startup log:
`0509-036 Cannot load program hatsd because of the following errors:`
`0509-023 Symbol dms_debug_tag in hatsd is not defined.`
`0509-026 System error: Cannot run a file that does not have a valid format.`
	Another possibility, when there is no **TS_** error log entry, is that the Topology Services daemon could not be loaded. In this case, a message similar to the following may be present in the Topology Services startup script log:
	`0509-036 Cannot load program hatsd because of the following errors:`
	`0509-023 Symbol dms_debug_tag in hatsd is not defined.`
	`0509-026 System error: Cannot run a file that does not have a valid format.`
		Another possibility, when there is no **TS_** error log entry, is that the Topology Services daemon could not be loaded. In this case, a message similar to the following may be present in the Topology Services user startup log:
		`0509-036 Cannot load program hatsd because of the following errors:`
		`0509-023 Symbol dms_debug_tag in hatsd is not defined.`
		`0509-026 System error: Cannot run a file that does not have a valid format.`
			Another possibility, when there is no **TS_** error log entry, is that the Topology Services daemon could not be loaded. In this case, a message similar to the following may be present in the Topology Services user startup log:
			`0509-036 Cannot load program hatsd because of the following errors:`
			`0509-023 Symbol dms_debug_tag in hatsd is not defined.`
			`0509-026 System error: Cannot run a file that does not have a valid format.`

The message may refer to the Topology Services daemon, or to some other program invoked by the startup script. If you find such an error, contact the IBM Support Center. For errors where the daemon started up, but exited during initialization, see information about the problem in the Topology Services user error log.	The message may refer to the Topology Services daemon, or to some other program invoked by the startup script **cthats**. If you find such an error, contact the IBM Support Center. For errors where the daemon did start up but exited during initialization, see detailed information about the problem in the Topology Services user error log.	The message may refer to the Topology Services daemon, or to some other program invoked by the startup script. If you find such an error, contact the IBM Support Center. For errors where the daemon did start up but exited during initialization, see detailed information about the problem in the Topology Services user error log.	The message may refer to the Topology Services daemon, or to some other program invoked by the startup script. If you find such an error, contact the IBM Support Center. For errors where the daemon did start up but exited during initialization, see detailed information about the problem in the Topology Services User error log.

Operational test 3: determine why remote adapters are not in the local adapter's membership group:

Use this test is to determine why the Topology Services subsystem may be inactive.

Issue the following **lssrc** command on all the nodes:

`lssrc -ls` *subsystem*

Issue the **lssrc** command on all the nodes.

If this test follows output 3 of “Operational test 1: verify status and adapters” on page 163, at least one node will not have the same output as the node from which output 3 was taken.

Some of the possibilities are:

1. The node is down or unreachable. Diagnose that node by using “Operational test 1: verify status and adapters” on page 163.
2. The output is similar to output of 3 but with a different group ID, such as in this output:

```
Subsystem       Group            PID     Status
cthats          cthats           20494   active

Network Name   Indx Defd Mbrs St Adapter ID      Group ID
ethernet1      [ 0]   15    7  S 9.114.61.199    9.114.61.201
ethernet1      [ 0] eth0          0x3740dd5c      0x3740dd72
HB Interval = 1 secs. Sensitivity = 4 missed beats
SPswitch       [ 1]   14    7  S 9.114.61.141    9.114.61.141
SPswitch       [ 1] css0          0x3740dd5d      0x3740dd72
```

```
HB Interval = 1 secs. Sensitivity = 4 missed beats
  Configuration Instance = 926566126
  Default: HB Interval = 1 secs. Sensitivity = 4 missed beats
  Daemon employs no security
  Data segment size: 6358 KB. Number of outstanding malloc: 588
  Number of nodes up: 7. Number of nodes down: 8.
  Nodes up: 17-29(2)
```

Compare this with the output from 3 Proceed to "Operational test 8: check if configuration instance and security status are the same across all nodes" on page 173.

3. The output is similar to the outputs of 1, 2, 4, or 5 of "Operational test 1: verify status and adapters" on page 163. Return to "Operational test 1: verify status and adapters" on page 163, but this time, focus on this new node.

Operational test 4: check the address of the local adapter:

Use this test to verify that a local adapter is configured with the correct address.

Assuming that this test is being run because the output of the **lssrc** command indicates that the adapter is disabled, there should be an entry in the error log that points to the problem.

On Linux nodes, enter this command:	On AIX nodes, enter this command:	On Windows nodes, enter this command:	On Solaris nodes, enter this command:
`fcslogrpt /var/log /messages`	`errpt -J TS_LOC_DOWN_ST, TS_MISCFG_ER -a | more`	`cat /var/adm/log /messages`	`cat /var/adm/log /messages`

Examples of the error log entries that appear in the output are:

-
```
LABEL:          TS_LOC_DOWN_ST
IDENTIFIER:     D17E7B06
Date/Time:       Mon May 17 23:29:34
Sequence Number: 227
Machine Id:      000032054C00
Node Id:         c47n11
Class:           S
Type:            INFO
Resource Name:   cthats.c47s

Description       Possible malfunction on local adapter
```
-
```
LABEL:          TS_MISCFG_ER
IDENTIFIER:     6EA7FC9E

Date/Time:       Mon May 17 16:28:45
Sequence Number: 222
Machine Id:      000032054C00
Node Id:         c47n11
Class:           U
Type:            PEND
Resource Name:   cthats.c47s
Resource Class:  NONE
Resource Type:   NONE
Location:        NONE
VPD:
Description
Local adapter misconfiguration detected
```

Good results are indicated by the absence of the **TS_MISCFG_ER** error entry. To verify that the local adapter has the expected address, issue the command:

```
ifconfig interface_name
```

where *interface_name* is the interface name listed on the output of **lssrc**, such as:

```
SPswitch       [ 1]   14    0  D 9.114.61.139
SPswitch       [ 1] css0
```

Table 28 details those tests useful in verifying that a local adapter is configured with the correct address. These tests vary by node type, as follows:

Table 28. Verify that a local adapter is configured with the correct address on Linux, AIX, Windows, and Solaris nodes

On Linux nodes:	On AIX nodes:	On Windows nodes:	On Solaris nodes:
For the **lssrc** command output, the output of **ifconfig eth0** is similar to: eth0 Link encap:Ethernet HWaddr 00:10:5A:61:74:42 inet addr:9.114.67.71 Bcast:9.114.67.127 Mask:25 5.255.255.192 UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1 RX packets:24403521 errors:0 dropped:0 overruns:0 frame:0 TX packets:8830412 errors:0 dropped:0 overruns:0 c arrier:82 collisions:4089 txqueuelen:100 Interrupt:9 Base address:0x2000	For the **lssrc** command output, the output of **ifconfig css0** is similar to: css0: flags=800847 <UP,BROADCAST, DEBUG, RUNNING,SIMPLEX> inet 9.114.61.139 netmask 0xffffffc0 broadcast 9.114.61.191	For the **lssrc** command output, the output of **ipconfig /all** is similar to: $ ipconfig /all Ethernet adapter Local Area Connection: Connection-specific DNS Suffix . : Description . . . . : VMware Accelerated AMD PCNet Adapter Physical Address. . . . . : 00-0C-29-7A-92- FE DHCP Enabled. . . . . : No IP Address. . . . . : 9.114.42.95 Subnet Mask . . . . : 255.255.255.0 Default Gateway . . . . : 9.114.42.254 DNS Servers . . . . : 9.0.3.1 9.0.2.1	For the **lssrc** command output, the output of **ifconfig bge0** is similar to: bge0: flags=1000843 <UP,BROADCAST, RUNNING,MULTICAST, IPv4> mtu 1500 index 2 inet 9.114.42.46 netmask ffffff00 broadcast 9.114.42.255 ether 0:9:6b:63:39:46

Error results are indicated by the **TS_MISCFG_ER** error entry and by the output of the **ifconfig** command not containing the address displayed in the **lssrc** command output. Diagnose the reason why the adapter is configured with an incorrect address.

If this test is a success, proceed to "Operational test 5: check to see if the adapter is enabled for IP."

Operational test 5: check to see if the adapter is enabled for IP:

Use this test to verify whether an adapter is enabled for IP.

Issue the command:

```
ifconfig interface_name
```

Table 29 on page 170 details those tests useful when checking if an adapter is enabled for IP. These tests vary by node type, as follows:

Table 29. Validate that an adapter is enabled for IP on Linux, AIX, and Windows nodes

On Linux Nodes:	On AIX Nodes:	On Windows Nodes:
The output is similar to the following: eth0 Link encap:Ethernet HWaddr 00:10:5A:61:74:42 inet addr: 9.114.67.71 Bcast:9.114.67.127 Mask: 25 5.255.255.192 UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1 RX packets: 24403521 errors:0 dropped:0 overruns:0 frame:0 TX packets:8830412 errors:0 dropped:0 overruns:0 carrier:82 collisions:4089 txqueuelen:100 Interrupt:9 Base address:0x2000	The output is similar to the following: css0: flags=800847 <UP,BROADCAST,DEBUG, RUNNING,SIMPLEX> inet 9.114.61.139 netmask 0xffffffc0 broadcast 9.114.61.191 An IPv6 example follows the table.	The output is similar to the following: $ ipconfig /all Windows IP Configuration Host Name . . . . : rsctvm2 Primary Dns Suffix . . . . . : Node Type . . . . : Unknown IP Routing Enabled. . . . . . : No WINS Proxy Enabled. . . . . . : No

```
[c47n01][/]> ifconfig -a
en0: flags=e080863<UP,BROADCAST,NOTRAILERS,RUNNING,SIMPLEX,MULTICAST,GROUPRT,64BIT>
        inet 9.114.61.65 netmask 0xfffffc0 broadcast 9.114.61.127
        inet6 fe80::204:acff:fe49:7c1a/64
sit0: flags=8100041<UP,RUNNING,LINK0>
        inet6 ::9.114.61.65/96
lo0: flags=e08084b<UP,BROADCAST,LOOPBACK,RUNNING,SIMPLEX,MULTICAST,GROUPRT,64BIT>
        inet 127.0.0.1 netmask 0xff000000 broadcast 127.255.255.255
        inet6 ::1/128
         tcp_sendspace 131072 tcp_recvspace 131072 rfc1323 1
```

Good results are indicated by the presence of the `UP` string in the third line of the output. In this case, proceed to "Operational test 6: check whether the adapter can communicate with other adapters in the network."

Error results are indicated by the absence of the `UP` string in the third line of the output.

Issue the command:

`ifconfig` *`interface_name`* `up`

...to re-enable the adapter to IP.

Operational test 6: check whether the adapter can communicate with other adapters in the network:

Use this test to verify whether an adapter can communicate with other adapters in the network.

Root authority is needed to access the contents of the **machines.lst** file. Display the contents of the **machines.lst** file. The output is similar to the following:

```
*InstanceNumber=925928580
*configId=1244520230
```

```
*!HaTsSeCStatus=off
*FileVersion=1
*!TS_realm=CLUSTER
TS_Frequency=1
TS_Sensitivity=4
TS_FixedPriority=38
TS_LogLength=5000
*!TS_PinText
Network Name ethernet1
Network Type ether
*
*Node Type Address
    0 en0 9.114.61.125
    1 en0  9.114.61.65
    3 en0  9.114.61.67
    11 en0  9.114.61.195
...
Network Name SPswitch
Network Type hps
*
*Node Type Address
    1 css0 9.114.61.129
    3 css0 9.114.61.131
    11 css0 9.114.61.139
```

Locate the network to which the adapter under investigation belongs. For example, the css0 adapter on node 11 belongs to network `SPswitch`. Depending on your operating system platform, issue either of the following commands:

- For Linux, AIX, or Windows system platforms:

  ping -c 5 *address*

  for the addresses listed in the **machines.lst** file.

  Good results are indicated by outputs similar to the following.

  ```
  PING 9.114.61.129: (9.114.61.129): 56 data bytes
  64 bytes from 9.114.61.129: icmp_seq=0 ttl=255 time=0 ms
  64 bytes from 9.114.61.129: icmp_seq=1 ttl=255 time=0 ms
  64 bytes from 9.114.61.129: icmp_seq=2 ttl=255 time=0 ms
  64 bytes from 9.114.61.129: icmp_seq=3 ttl=255 time=0 ms
  64 bytes from 9.114.61.129: icmp_seq=4 ttl=255 time=0 ms
   ----9.114.61.129 PING Statistics----
  5 packets transmitted, 5 packets received, 0% packet loss
  round-trip min/avg/max = 0/0/0 ms
  ```

  The number before `packets received` should be greater than 0.

  Error results are indicated by outputs similar to the following:

  ```
   PING 9.114.61.129: (9.114.61.129): 56 data bytes
   ----9.114.61.129 PING Statistics----
  5 packets transmitted, 0 packets received, 100% packet loss
  ```

  The command should be repeated with different addresses until it succeeds or until several different attempts are made. After that, pursue the problem as an adapter or IP-related problem.

- For Solaris system platforms:

```
ping address
```

for the addresses listed in the **machines.lst** file.

Good results are indicated by outputs similar to the following.

```
             ping 9.114.42.253
9.114.42.253 is alive
```

The number before `packets received` should be greater than 0.

Error results are indicated by outputs similar to the following:

```
              ping 9.114.42.253
no answer from 9.114.42.253
```

The command should be repeated with different addresses until it succeeds or until several different attempts are made. After that, pursue the problem as an adapter or IP-related problem.

If this test succeeds, but the adapter is still listed as disabled in the **lssrc** command output, collect the data listed in "Information to collect before contacting the IBM Support Center" on page 16 and contact the IBM Support Center.

Operational test 7: check for partial connectivity:

Adapters stay in a singleton unstable state when there is partial connectivity between two adapters. One reason for an adapter to stay in this state is that it keeps receiving PROCLAIM messages, to which it responds with a JOIN message, but no PTC message comes in response to the JOIN message.

Check in the Topology Services User log file to see if a message similar to the following appears repeatedly:

```
2523-097 JOIN time has expired. PROCLAIM message was sent
                    by (10.50.190.98:0x473c6669)
```

If this message appears repeatedly in the Topology Services User log, investigate IP connectivity between the local adapter and the adapter whose address is listed in the User log entry (10.50.190.98 in the example here). Issue command:

```
ping -c 5 address
```

address is 10.50.190.98 in this example.

See "Operational test 5: check to see if the adapter is enabled for IP" on page 169 for a description of **good results** for this command.

The local adapter cannot communicate with a Group Leader that is attempting to attract the local adapter into the adapter membership group. The problem may be with either the local adapter or the Group Leader adapter (*proclaimer* adapter). Pursue this as an IP connectivity problem. Focus on both the local adapter and the Group Leader adapter.

If the **ping** command succeeds, but the local adapter still stays in the singleton unstable state, contact the IBM Support Center.

On AIX nodes, in a PowerHA SystemMirror/ESenvironment, it is possible that there are two adapters in different nodes both having the same service address. This can be verified by issuing:

```
lssrc -ls subsystem_name
```

...and looking for two different nodes that have the same IP address portion of Adapter ID. In this case, this problem should be pursued as a PowerHA SystemMirror/ES problem. Contact the IBM Support Center.

With IPv6, the Adapter ID will look like this:
`([2]:0x48ef80ff)`

where **[2]** indicates the node number.

If this test fails, proceed to “Operational test 4: check the address of the local adapter” on page 168, concentrating on the local and Group Leader adapters.

Operational test 8: check if configuration instance and security status are the same across all nodes:

Use this test to verify whether all nodes are using the same configuration instance number and same security setting.

This test is used when there seem to be multiple partitioned adapter membership groups across the nodes, as in “Operational test 3: determine why remote adapters are not in the local adapter's membership group” on page 167 number 2.

This test verifies whether all nodes are using the same configuration instance number and same security setting. The instance number changes each time the **machines.lst** file is generated by the startup script. In an RSCT peer domain, the configuration instance always increases.

Issue the **lssrc** command:

`lssrc -ls` *`subsystem_name`*

on all nodes. If this is not feasible, issue the command at least on nodes that produce an output that shows a different `Group ID`.

Compare the line `Configuration Instance = (number)` in the **lssrc** outputs. Also, compare the line `Daemon employs` in the **lssrc** command outputs.

Good results are indicated by the number after the `Configuration Instance` phrase being the same in all the **lssrc** outputs. This means that all nodes are working with the same version of the **machines.lst** file.

Error results are indicated by the configuration instance being different in the two "node partitions". In this case, the adapters in the two partitions cannot merge into a single group because the configuration instances are different across the node partitions. This situation is likely to be caused by a refresh-related problem. One of the node groups, probably that with the lower configuration instance, was unable to run a refresh. If a refresh operation was indeed attempted, consult the description of the "Nodes or adapters leave membership after refresh" problem in “Error symptoms, responses, and recoveries” on page 178

The situation may be caused by a problem in the SRC subsystem, which fails to notify the Topology Services daemon about the refresh. The description of the "Nodes or adapters leave membership after refresh" problem in “Error symptoms, responses, and recoveries” on page 178 explains how to detect the situation where the Topology Services daemon has lost its connection with the SRC subsystem. In this case, contact the IBM Support Center.

If this test is successful, proceed to “Operational test 9: check connectivity among multiple node partitions” on page 174.

Operational test 9: check connectivity among multiple node partitions:

Use this test when adapters in the same Topology Services network form multiple adapter membership groups, rather than a single group encompassing all the adapters in the network.

Follow the instructions in "Operational test 8: check if configuration instance and security status are the same across all nodes" on page 173 to obtain **lssrc** outputs for each of the node partitions.

The IP address listed in the **lssrc** command output under the `Group ID` heading is the IP address of the Group Leader. If two node partitions are unable to merge in to one, this is caused by the two Group Leaders being unable to communicate with each other. Note that even if some adapters in different partitions can communicate, the group merge will not occur unless the Group Leaders are able to exchange point-to-point messages. Use **ping** (as described in "Operational test 6: check whether the adapter can communicate with other adapters in the network" on page 170.

For example, assume on one node the output of the **lssrc -ls cthats** command is:

```
Subsystem     Group            PID     Status
cthats        cthats           15750   active

Network Name   Indx Defd Mbrs St Adapter ID      Group ID
ethernet1      [0]   15    9   S 9.114.61.65     9.114.61.195
ethernet1      [0]              0x373897d2      0x3745968b
HB Interval = 1 secs. Sensitivity = 4 missed beats
SPswitch       [1]   14    14  S 9.114.61.129    9.114.61.153
SPswitch       [1]              0x37430634      0x374305f1
HB Interval = 1 secs. Sensitivity = 4 missed beats
```

and on another node it is:

```
Subsystem         Group            PID     Status
cthats          cthats           13694    active

Network Name   Indx Defd Mbrs St Adapter ID      Group ID
ethernet1      [0]  15    6  S 9.114.30.69     9.114.61.71
ethernet1      [0]              0x37441f24      0x37459754
HB Interval = 1 secs. Sensitivity = 4 missed beats
SPswitch       [1]   14   14  S 9.114.61.149    9.114.61.153
SPswitch       [1]             0x374306a4      0x374305f1
```

In this example, the partition is occurring on network ethernet1. The two Group Leaders are IP addresses 9.114.61.195 and 9.114.61.71. Login to the node that hosts one of the IP addresses and issue the **ping** test to the other address. In case the two adapters in question are in the same subnet, verify whether they have the same subnet mask and the same valid broadcast address (based on the IP address and the subnet mask).

Good results and **error results** for the **ping** test are described in "Operational test 6: check whether the adapter can communicate with other adapters in the network" on page 170. If the **ping** test is not successful, a network connectivity problem between the two Group Leader nodes is preventing the groups from merging. Diagnose the network connectivity problem.

Good results for the subnet mask test are indicated by the adapters that have the same subnet id also having the same subnet mask. The binary representation of the subnet mask must contain a sequence of 1s, followed by a sequence of 0s. If the subnet mask test fails, the subnet mask at one or more nodes must be corrected by issuing the command:

```
ifconfig interface_name addressnetmask netmask
```

All the adapters that belong to the same subnet must have the same subnet mask.

Good results for the broadcast address test are indicated by the adapters that have the same subnet ID also having the same broadcast address, which must be in the valid range, based on the subnet mask and IPv4 addresses of each adapter. Broadcast addresses apply to IPv4 networks only.

The broadcast address must be:

IP Address <logical or> (one's complement of subnet mask)

For example:

```
IP Address = 1.2.3.4;
subnet mask = 255.255.255.0
one's complement of subnet mask = 0.0.0.255
So broadcast address must be: 1.2.3.255
```

Broadcast addresses apply to IPv4 networks only.

If the broadcast address test fails, the broadcast address at one or more nodes must be corrected by issuing the command:

ifconfig *interface_name address* **broadcast** *broadcast_address*

If the **ping** test is successful (the number of `packets received` is greater than 0), and the subnet masks match, there is some factor other than network connectivity preventing the two Group Leaders from contacting each other. The cause of the problem may be identified by entries in the Topology Services User log. If the problem persists, collect the data listed in "Information to collect before contacting the IBM Support Center" on page 16 and contact the IBM Support Center. Include information about the two Group Leader nodes.

Operational test 10: check neighboring adapter connectivity:

Use this test to check neighboring adapter connectivity, in order to investigate partial connectivity situations.

Table 30 details those tests useful for checking neighboring adapter connectivity. These tests vary by node type, as follows:

Table 30. Validate neighboring adapter connectivity on Linux, AIX, Windows, and Solaris nodes

On Linux nodes, enter this command:	On AIX nodes, enter this command:	On Windows nodes, enter this command:	On Solaris nodes, enter this command:
`fcslogrpt /var/log /messages`	`errpt -J TS_DEATH_TR | more`	`cat /var/adm/log /messages`	`cat /var/adm /messages`

Look for recent entries with label **TS_DEATH_TR**. This is the entry created by the subsystem when the local adapter stops receiving heartbeat messages from the neighboring adapter. For the adapter membership groups to be constantly reforming, such entries should be found in the error log.

Issue the **ping** test on the node where the **TS_DEATH_TR** entry exists. The target of the **ping** should be the adapter whose address is listed in the Detail Data of the error log entry. "Operational test 6: check whether the adapter can communicate with other adapters in the network" on page 170 describes how to perform the **ping** test and interpret the results.

If the **ping** test fails, this means that the two neighboring adapters have connectivity problems, and the problem should be pursued as an IP connectivity problem.

If the **ping** test is successful, the problem is probably not due to lack of connectivity between the two neighboring adapters. The problem may be due to one of the two adapters not receiving the COMMIT

message from the "mayor adapter" when the group is formed. The **ping** test should be used to probe the connectivity between the two adapters and all other adapters in the local subnet.

Operational test 11: verify node reachability information:

Use this test to check node reachability information.

Issue the **lssrc** command:

```
lssrc -ls subsystem_name
```

and examine lines:

1. `Number of nodes up: # . Number of nodes down: #.`
2. `Nodes down: [...]` or `Nodes up: [...]`

in the command output.

Good results are indicated by the line `Number of Nodes down: 0`. For example,

```
Number of nodes up: 15    Number of nodes down: 0
```

However, such output can only be considered correct if indeed all nodes in the system are known to be up. If a given node is indicated as being up, but the node seems unresponsive, perform problem determination on the node. Proceed to "Operational test 12: verify the status of an unresponsive node that is shown to be up by Topology Services."

Error results are indicated by `Number of Nodes down:` being nonzero. The list of nodes that are flagged as being up or down is given in the next output line. An output such as `Nodes down: 17-23(2)` indicates that nodes 17, 19, 21, and 23 are considered down by Topology Services. If the nodes in the list are known to be down, this is the expected output. If, however, some of the nodes are thought to be up, it is possible that a problem exists with the Topology Services subsystem on these nodes. Proceed to "Operational test 1: verify status and adapters" on page 163, focusing on each of these nodes.

Operational test 12: verify the status of an unresponsive node that is shown to be up by Topology Services:

Use this test to check the status of an unresponsive node that is shown to be up by Topology Services.

Examine the **machines.lst** configuration file and obtain the IP addresses for all the adapters in the given node that are in the Topology Services configuration. For example, for node 9, entries similar to the following may be found in the file:

```
 9 eth0  9.114.61.193  9 css0 9.114.61.137
```

Issue this command.

```
ping -c5 IP_address
```

If there is no response to the **ping** packets (the output of the command shows `100% packet loss`) for all the node's adapters, the node is either down or unreachable. Pursue this as a node health problem. If Topology Services still indicates the node as being up, contact the IBM Support Center because this is probably a Topology Services problem. Collect long tracing information from the Topology Services logs. See "Topology Services service log" on page 158. Run the **tcpdump** command as described in"Information to collect before contacting the IBM Support Center" on page 16.

If the output of the **ping** command shows some response (for example, `0% packet loss`), the node is still up and able to send and receive IP packets. The Topology Services daemon is likely to be running and able to send and receive heartbeat packets. This is why the node is still seen as being up. Pursue this as a Linux-related problem.

If there is a response from the **ping** command, and remote Topology Services daemons consider the node up, but the node is unresponsive and no user application is apparently able to run, obtain a system dump to identify the cause of the problem.

Operational test 13: verify that disk heartbeating is occurring:

Use this test to verify that disk heartbeating is occurring.

First, determine the disk heartbeating resource name and its communication group. Run the following command to determine the name of the disk heartbeating resource and the name of the communication group:

```
lsrsrc IBM.HeartbeatInterface
```

In this sample command output, **dhb-1-2** is the name of the disk heartbeating resource and **dhbTEST_CG** is the name of the related disk heartbeating communication group:

```
Resource Persistent Attributes for IBM.HeartbeatInterface
resource 1:
    Name       = "dhb-1-2"
    DeviceInfo    = "/dev/hdisk20"
    HeartbeatActive  = 1
    CommGroup        = "dhbTEST_CG"
    Qualifier    = ""
    MediaType    = 2
    ActivePeerDomain  = "RPD_DHB_Task2a-1_dm"
    NodeNameList      = {"c48f1rp03.ppd.pok.ibm.com"}
```

Run the following command to determine if disk heartbeating is occurring:

```
lssrc -ls cthats
```

The output will look like this:

```
Subsystem         Group            PID     Status
 cthats           cthats           372762  active
Network Name   Indx Defd  Mbrs  St   Adapter ID      Group ID
dhbTEST_CG     [ 1]    2     2  S    255.255.10.1    255.255.10.1
dhbTEST_CG     [ 1] dhb-1-2          0x01c8f4fe      0x01c8f511
```

In this sample command output, find the disk heartbeating stanza by locating the name of the disk heartbeating communication group, which you obtained from the **lsrsrc IBM.HeartbeatInterface** command output. The name of the **IBM.HeartbeatInterface** resource is also displayed.

Good Results: Disk heartbeating is occurring if **Defd** and **Mbrs** are the same number:

```
 Defd  Mbrs
    2     2
```

Error Results: Disk heartbeating is not occurring if **Defd** and **Mbrs** are different:

```
Defd  Mbrs
   2     1
```

If the output shows that disk heartbeating is not occurring, test disk heartbeating connectivity. To do so, use the **dhb_read** utility on this device on both nodes. Set the first node in receive mode (**-r**) and the second node in transmit mode (**-t**). For example:

Run this command on the first node:

```
/usr/sbin/rsct/bin/dhb_read -p device_name -r
```

The output will look like this:

```
#  dhb_read -p /dev/sdr -r
DHB CLASSIC MODE
 First node byte offset: 61440
Second node byte offset: 62976
Handshaking byte offset: 65024
       Test byte offset: 64512

Receive Mode:
Waiting for response . . .
Magic number = 0x87654321
Magic number = 0x87654321
Magic number = 0x87654321
Magic number = 0x87654321
Link operating normally
#
```

Then, run this command on the second node:

```
/usr/sbin/rsct/bin/dhb_read -p device_name -t
```

The output will look like this:

```
# dhb_read -p /dev/sdr -t
DHB CLASSIC MODE
 First node byte offset: 61440
Second node byte offset: 62976
Handshaking byte offset: 65024
       Test byte offset: 64512

Transmit Mode:
Magic number = 0x87654321
Detected remote utility in receive mode.  Waiting for response . . .
Magic number = 0x87654321
Magic number = 0x87654321
Link operating normally
#
```

Good Results: Disk heartbeating connectivity is working if both **dhb_read** commands return **Link operating normally**.

Error Results: If both **dhb_read** commands return **Link operating normally**, there is a problem with this device.

Error symptoms, responses, and recoveries

Use this information to diagnose problems with the Topology Services component of RSCT.

Use the information in Table 31 on page 179 to diagnose problems with the Topology Services component of RSCT. Locate the symptom and perform the specified recovery action.

Table 31. Topology Services symptoms and recovery actions

Symptom	Recovery
Adapter membership groups do not include all of the nodes in the configuration.	"Operational test 1: verify status and adapters" on page 163
Topology Services subsystem fails to start.	"Action 1: investigate startup failure"
The refresh operation fails or has no effect.	"Action 2: investigate refresh failure"
A local adapter is reported as being down by Topology Services.	"Action 3: investigate local adapter problems" on page 180
Adapters appear to be going up and down continuously.	"Action 4: investigate partial connectivity problem" on page 180
A node appears to go down and then up a few seconds later.	"Action 5: investigate a hatsd problem" on page 181
Adapter appears to go down and then up a few seconds later.	"Action 6: investigate an IP communication problem" on page 186
Group Services exits abnormally because of a Topology Services Library error. Error log entry with template **GS_TS_RETCODE_ER** is present.	"Action 7: investigate a Group Services failure" on page 187
Nodes or adapters leave membership after a refresh.	"Action 8: investigate problems after a refresh" on page 187
An AIX node has crashed.	"Action 9: investigate an AIX node crash" on page 190

Action 1: investigate startup failure

A number of possible conditions suggest this action. Take this action when the Topology Services subsystem fails to start.

Some of the possible conditions for which this action is suggested include:

- Adapter configuration problems, such as duplicated IP addresses in the configuration.
- Operating system-related problems, such as a shortage of space in the **/var** directory or a port number already in use.
- Security services problems that prevent Topology Services from obtaining credentials, determining the active authentication method, or determining the authentication keys to use.

See "Operational test 2: determine why the Topology Services subsystem is inactive" on page 166. To verify the correction, see "Operational test 1: verify status and adapters" on page 163.

Action 2: investigate refresh failure

A number of possible conditions suggest this action. Take this action when the refresh operation fails or has no effect.

The most probable condition suggesting this action is that in which an incorrect adapter or network configuration was passed to Topology Services. Refresh errors are listed in the **/var/ct/***cluster_name***/log/cthats/refreshOutput** file, and the startup script log.

Also, configuration errors result in error entries being created. On AIX nodes, the entries are added to the AIX Error Log. On Linux, Windows, and Solaris nodes, these entries are added to the respective system log. Some of the template labels that may appear are:

- TS_CTNODEUP_ER
- TS_CTIPDUP_ER
- TS_CL_FATAL_GEN_ER
- TS_HANODEDUP_ER
- TS_HAIPDUP_ER

The error entries should provide enough information to determine the cause of the problem. Detailed information about the configuration and the error or can be found in the startup script log and the Topology Services user log.

For the problems that result in the error entries listed here, the solution involves changing the IP address of one or more adapters.

A Topology Services refresh will occur whenever changes are made to the topology, such as when a communication group is modified by the **chcomg** command.

Incorrect or conflicting adapter information will result in the refresh having no effect, as well as the creation of AIX error log entries (on AIX nodes), or system log entries (on the respective Linux, Windows, or Solaris nodes).

Related reference:

"Topology Services startup log" on page 157
The Topology Services startup log is located in *startup_log*.

Action 3: investigate local adapter problems

A number of possible conditions suggest this action. Take this action when Topology Services sends a notification that a local adapter is down.

The most common local adapter problems that suggest the need for this action include:

1. The adapter is not working.
2. The network might be down.
3. The adapter might have been configured with an incorrect IP address.
4. Topology Services is unable to get response packets back to the adapter.
5. There is a problem in the subsystem's "adapter self-death" procedures.

See "Operational test 4: check the address of the local adapter" on page 168 to analyze the problem. The repair action depends on the nature of the problem. For problems 1 through 3, the underlying cause for the adapter being unable to communicate must be found and corrected.

For problem 4, Topology Services requires that at least one other adapter in the network exist, so that packets can be exchanged between the local and remote adapters. Without such an adapter, a local adapter would be unable to receive any packets. Therefore, there would be no way to confirm that the local adapter is working.

To verify the repair, issue the **lssrc** command as described in "Operational test 1: verify status and adapters" on page 163. If the problem is because Topology Services cannot get response packets back to the adapter (problem 4), the problem can be circumvented by adding traditional entries to the **netmon.cf** file.

For detailed information about the **netmon.cf** file, see the *RSCT: Administration Guide*.

To remove this recovery action, remove the entries added to the file, delete the file, or rename the file.

Action 4: investigate partial connectivity problem

A number of possible conditions suggest this action. Take this action when adapters appear to be going up and down continuously.

The most probable condition suggesting this action is a partial connectivity scenario. This means that one adapter or a group of adapters can communicate with some, but not all, remote adapters. Stable groups in Topology Services require that all adapters in a group be able to communicate with each other.

Some possible sources of partial connectivity are:

1. Physical connectivity
2. Incorrect routing at one or more nodes
3. Adapter or network problems which result in packets larger than a certain size being lost

4. Incorrect ARP setting in large machine configurations
5. High network traffic, which causes a significant portion of the packets to be lost
6. Proxy ARP is set on an intermediate switch but is not working properly

To check whether there is partial connectivity on the network, run “Operational test 10: check neighboring adapter connectivity” on page 175. You must isolate and correct the underlying connectivity problem. To verify the correction, issue the **lssrc** command from “Operational test 1: verify status and adapters” on page 163.

You can bypass this problem if the connectivity test revealed that one or more nodes have only partial connectivity to the others. In this case, you can stop Topology Services on these partial connectivity nodes. If the remaining adapters in the network have complete connectivity to each other, they should form a stable group.

Topology Services subsystem can be stopped on a node by issuing the **cthatsctrl** command:

```
/usr/sbin/rsct/bin/cthatsctrl -k
```

Note: The nodes on which the subsystem was stopped will be marked as "down" by the others. Applications such as IBM Virtual Shared Disk will be unable to use these nodes.

To test and verify this recovery, issue the **lssrc** command as described in “Operational test 1: verify status and adapters” on page 163. The `Group ID` information in the output should not change across two invocations approximately one minute apart.

Once you no longer need this recovery action, restart Topology Services by issuing the **cthatsctrl** command:

```
/usr/sbin/rsct/bin/cthatsctrl -s
```

Proxy ARP is a network setting that often behaves incorrectly in PowerHA SystemMirror environments during IP Takeover, although it can be a problem in any environment. If problems related to Topology Services are accompanied by ARP table entries (as displayed by the **arp -a** command) that do not reflect the actual owner of an IP address but, instead, reflect the IP address of the intermediate network hardware, then disable proxy ARP on the switch that immediately connects the affected nodes.

Action 5: investigate a hatsd problem

A number of possible conditions suggest this action. Take this action when a node appears to go down and then up a few seconds later.

Probable conditions which suggest this action include:

1. The Topology Services daemon is temporarily blocked.
2. The Topology Services daemon exited on the node.
3. IP communication problem, such as mbuf shortage or excessive adapter traffic.

You can determine probable cause 1 by the presence of an error log entry with the TS_LATEHB_PE template on the affected node. This entry indicates that the daemon was blocked and for how long. When the daemon is blocked, it cannot send messages to other adapters, and as a result other adapters may consider the adapter dead in each adapter group. This results in the node being considered dead.

Some of the reasons for the daemon to be blocked include:

1. A memory shortage, which causes excessive paging and thrashing behavior; the daemon stays blocked, awaiting a page-in operation.

2. A memory shortage combined with excessive disk I/O traffic, which results in slow paging operations.
3. The presence of a fixed-priority process with higher priority than the Topology Services daemon, which prevents the daemon from running.
4. Excessive interrupt traffic, which prevents any process in the system from being run in a timely manner.

Command:

```
/usr/samples/kernel/vmtune -f 256 -F 264 -p 1 -P 2
```

...can be used to increase **minfree** to 256 and give more preference to computational pages. For more information about the **minfree** parameter, see the **Summary of Tunable AIX Parameters** appendix in the *AIX Performance Tuning Guide*.

When a system that appears to have enough memory continues to perform a very high level of I/O operations, it is possible that the virtual memory manager may be "stealing" pages from processes ("computational pages") and assigning them to file I/O ("permanent pages").

You must understand and resolve the underlying problem that is causing the Topology Services daemon to be blocked.

For problems related to memory thrashing, observations suggest that an inability of the Topology Services daemon to run in a timely manner indicates that the amount of paging is causing little useful activity to be accomplished on the node.

If the problem is related to a process running with a fixed priority which is higher (that is, a larger number) than that of the Topology Services daemon, you may correct the problem by changing the daemon's priority. To do so, issue the **cthatstune** command:

```
/usr/sbin/rsct/bin/cthatstune -p new_value -r
```

You can determine probable cause 2 by the presence of a syslog entry that indicates that the daemon exited. See “Error logs and templates” on page 137 for the list of possible error templates used. Look also for an error entry with a LABEL of CORE_DUMP and PROGRAM NAME of **hatsd**. This indicates that the daemon exited abnormally, and a **core** file should exist in the daemon's **run** directory.

If the daemon produced one of the error log entries before exiting, the error log entry itself, together with the information from “Error logs and templates” on page 137 should provide you with enough information to diagnose the problem. If the CORE_DUMP entry was created, follow instructions in “Information to collect before contacting the IBM Support Center” on page 16 and contact the IBM Support Center.

Probable cause 3 is the most difficult to analyze, since there may be multiple causes for packets to be lost. Some commands are useful in determining whether packets are being lost or discarded at the node, as follows:

1. `netstat -D`

   The `Idrops` and `Odrops` headings are the number of packets dropped in each interface or device.
2. `netstat -m`

   The `failed` heading is the number of *mbuf* allocation failures.
3. `netstat -s`

   The `socket buffer overflows` text is the number of packets discarded due to lack of socket space.

   The `ipintrq overflows` text is the number of input packets discarded because of lack of space in the packet interrupt queue.

4. `netstat -v`

   This command shows several adapter statistics, including packets lost due to lack of space in the adapter transmit queue, and packets lost probably due to physical connectivity problems ("CRC Errors").
5. `vmstat -i`

   This command shows the number of device interrupts for each device, and gives an idea of the incoming traffic.

There can be many causes for packets to be discarded or lost, and the problem needs to be pursued as an IP-related problem. Usually the problem is caused by one or more of the following:

1. Excessive IP traffic on the network or the node itself.
2. Inadequate IP or UDP tuning.
3. Physical problems in the adapter or network.

If causes 1 and 2 do not seem to be present, and you could not determine cause 3, issue some of the commands listed previously in loop, so that enough IP-related information is kept in case the problem happens again.

You must understand and solve the underlying problem that is causing the loss of packets. Your repair is effective when the node is no longer considered temporarily down under a similar workload.

In some environments (probable causes 1 and 3) you may bypass the problem by relaxing the Topology Services tunable parameters, to allow a node not to be considered down when it cannot temporarily send network packets. Changing the tunable parameters, however, also means that it will take longer to detect a node or adapter as down.

Note: Before you change the tunable parameters, record the current values, so that they can be restored if needed.

You can apply this solution only when:

1. There seems to be an upper bound on the amount of "outage" the daemon is experiencing.
2. The applications running on the system can withstand the longer adapter or node down detection time.

The **cthatstune** command:

```
cthatstune -f VIEW -s VIEW
```

...can be used to display the current *Frequency* and *Sensitivity* values for all the networks being monitored.

The adapter and node detection time is given by the formula:

```
2 * Sensitivity * Frequency
```

(two multiplied by the value of *Sensitivity* multiplied by the value of *Frequency*)

These values can be changed with:

```
cthatstune [-f [network:]Frequency] [-s [network:]Sensitivity] -r
```

...where:

- The **-f** flag represents the *Frequency* tunable value.
- The **-s** flag represents the *Sensitivity* tunable value.

You can do this tuning on a network-basis by specifying the **network** operand. If you omit the **network** operand, the changes apply to all the networks.

To verify that the tuning changes have taken effect, issue the **lssrc** command:

```
lssrc -ls subsystem_name
```

... approximately one minute after making the changes. The tunable parameters in use are shown in the output in a line similar to the following:

```
HB Interval = 1 secs. Sensitivity = 4 missed beats
```

For each network, `HB Interval` is the *Frequency* parameter, and `Sensitivity` is the *Sensitivity* parameter.

For examples of tuning parameters that can be used in different environments, see the *Administering RSCT* guide and the **cthatstune** command.

Good results are indicated by the tunable parameters being set to the desired values.

Error results are indicated by the parameters having their original values or incorrect values.

To verify whether the tuning changes were effective in masking the daemon outage, the system has to undergo a workload similar to that which caused the outage.

To remove the tuning changes, follow the same tuning changes outlined previously, but this time restore the previous values of the tunable parameters.

Reducing I/O rate on AIX nodes:

For problems related to excessive disk I/O, take these steps in AIX to reduce the I/O rate.

1. Set I/O pacing.

   I/O pacing limits the number of pending write operations to file systems, thus reducing the disk I/O rate. AIX is installed with I/O pacing disabled. You can enable I/O pacing with the command:

   ```
   chdev -l sys0 -a maxpout='33' -a minpout='24'
   ```

   This command sets the high- and low-water marks for pending write-behind I/Os per file. The values can be tuned if needed.
2. Change the frequency of **syncd**.

   If you run this daemon more frequently, you will need to flush a fewer number of pending I/O operations to disk. Therefore, the invocation of **syncd** will cause a reduced peak in I/O operations.

   To change the frequency of **syncd**, edit (as **root**) the **/sbin/rc.boot** file. Search for the following two lines:

   ```
   echo "Starting the sync daemon" | alog -t boot nohup
         /usr/sbin/syncd 60 > /dev/null 2>&1 &
   ```

   The period is set in seconds in the second line, immediately following the invocation of **/usr/sbin/syncd**. In this example, the interval is set to 60 seconds. A recommended value for the period is 10 seconds. Reboot for the change to take effect.

Preventing memory contention problems with the AIX Workload Manager:

Use the AIX Workload Manager on AIX nodes to prevent memory contention problems.

Memory contention often causes the Topology Services daemon to be blocked for significant periods of time. It results in "false node downs", and in the triggering of the deadman switch timer in PowerHA SystemMirror/ES. An AIX error log entry with label TS_LATEHB_PE might appear when you run RSCT 1.2 or higher. The message "Late in sending Heartbeat by ..." appears in the daemon log file in any release of RSCT, indicating that the Topology Services daemon was blocked. Another error log entry that might be created is TS_DMS_EXPIRING_EM or TS_DMS_WARNING_ST.

In many cases, such as when the system is performing a high level of disk I/O, it is possible for the Topology Services daemon to be blocked in paging operations, even though it looks like the system has enough memory. Three possible causes for this phenomenon follow:

- In steady state, when there are no node and adapter events on the system, the Topology Services daemon uses a working set of pages that is substantially smaller than its entire addressing space. When node or adapter events happen, the daemon faces the situation where additional pages it must process the events are not present in memory.
- When a high level of file I/O is taking place, the operating system might reserve a larger percentage of memory pages to files, making fewer pages available to processes.
- When a high level of file I/O is taking place, paging I/O operations might be slowed down by contention for the disk.

The probability that the Topology Services daemon gets blocked for paging I/O might be reduced by using the AIX Workload Manager (WLM). WLM is an operating system feature that is designed to give the system administrator greater control over how the scheduler and Virtual Memory Manager (VMM) allocate CPU and physical memory resources to processes. WLM gives the system administrator the ability to create different classes of service, and specify attributes for those classes.

The following explains how WLM can be used to allow the Topology Services daemon to obtain favorable treatment from the VMM. There is no need to involve WLM in controlling the daemon's CPU use because the daemon is already configured to run at a real time fixed scheduling priority. WLM does not assign priority values smaller than 40 to any thread.

These instructions are given by using SMIT, but it is also possible to use WLM or AIX commands to achieve the same goals.

Initially, use the sequence:

```
smit wlm
   Add a Class
```

...to add a `TopologyServices` class to WLM. Ensure that the class is at `Tier 0` and has `Minimum Memory` of 20%. These values cause processes in this class to receive favorable treatment from the VMM. Tier 0 means that the requirement from this class is satisfied before the requirements from other classes with higher tiers. Minimum memory prevents the process's pages from being taken by other processes, while the process in this class is using less than 20% of the machine's memory.

Use the sequence:

```
smit wlm
   Class Assignment Rules
     Create a new Rule
```

to create a rule for classifying the Topology Services daemon into the new class. In this screen, specify 1 as *Order of the Rule*, **TopologyServices** as *Class*, and `/usr/sbin/rsct/bin/hatsd` as *Application*.

To verify the rules that are defined, use the sequence:

```
smit wlm
   Class Assignment Rules
      List all Rules
```

To start WLM after the new class and rule are already in place, use the sequence:

```
smit wlm
   Start/Stop/Update WLM
      Start Workload Management
```

To verify that the Topology Services daemon is indeed classified in the new class, use the following command:

```
ps -ef -o pid,class,args | grep hatsd | grep -v grep
```

One sample output of this command follows:

```
15200 TopologyServices /usr/sbin/rsct/bin/hatsd -n 5
```

The `TopologyServices` text in this output indicates that the Topology Services daemon is a member of the `TopologyServices` class.

If WLM is already being used, the system administrator must ensure that the new class created for the Topology Services daemon does not conflict with other already defined classes. For example, the sum of all "minimum values" in a tier must be less than 100%. However, if WLM is already in use, the administrator must ensure that other applications in the system do not cause the Topology Services daemon to be deprived of memory. One way to prevent other applications from being more privileged than the Topology Services daemon in regard to memory allocation is to place other applications in tiers other than tier 0.

If WLM is already active on the system when you add the new classes and rules, you must restart WLM to recognize the new classes and rules.

Action 6: investigate an IP communication problem

A number of possible conditions suggest this action. Take this action when an adapter appears to go down and then up a few seconds later.

Probable conditions for which this action is suggested include:

1. The Topology Services daemon was temporarily blocked.
2. The Topology Services daemon exited on the node.
3. IP communication problem, such as mbuf shortage or excessive adapter traffic.

Probable cause 1 and probable cause 2 are usually only possible when all the monitored adapters in the node are affected. It is because these are conditions that affect the daemon as a whole, and not just one of the adapters in a node.

Probable cause 3 on the other hand, might result in a single adapter in a node being considered as down. Follow the procedures described to diagnose symptom "Node appears to go down and then up", “Action 5: investigate a hatsd problem” on page 181. If probable cause 1 or probable cause 2 is identified as the source of the problem, follow the repair procedures described under the same symptom.

If these causes are ruled out, the problem is likely related to IP communication. The instructions in "Node appears to go down and then up", “Action 5: investigate a hatsd problem” on page 181 describe what communication parameters to monitor in order to pinpoint the problem.

Table 32 lists those commands that are used for identifying the network that affects an IP communication problem. These identification procedures vary by node type, as follows:

Table 32. Commands to identify the network that affects an IP communication problem on Linux, AIX, Windows, and Solaris nodes

On Linux nodes:	On AIX nodes:	On Windows nodes:	On Solaris nodes:
`fcslogrpt /var/log /messages`	`errpt -J TS_DEATH_TR | more`	`cat /var/adm/log /messages`	`cat /var/adm /messages`

Once you have entered the appropriate command shown in the preceding table, look for the entry **TS_DEATH_TR**. This is the error entry created when the local adapter stopped receiving heartbeat messages from its neighbor adapter. The neighbor's address, which is listed in the error log entry, indicates which network is affected.

Action 7: investigate a Group Services failure

A number of possible conditions suggest this action. Take this action when Group Services exits abnormally because of a Topology Services Library error. Error log entry with template **GS_TS_RETCODE_ER** is present.

This action is most likely suggested to resolve a problem in the Topology Services daemon, or a problem related to the communication between the daemon and the Topology Services library, which is used by the Group Services daemon. This problem may happen during Topology Services refresh in Linux.

When this problem occurs, the Group Services daemon exits and produces an error log entry with a `LABEL` of GS_TS_RETCODE_ER. This entry will have the Topology Services return code in the Detail Data field. Topology Services will produce an error log entry with a `LABEL` of TS_LIBERR_EM. Follow the instructions in "Information to collect before contacting the IBM Support Center" on page 16 and contact the IBM Support Center.

Action 8: investigate problems after a refresh

Two possible conditions suggest this action. Take this action when nodes or adapters leave membership after a refresh.

Probable conditions suggesting this action include:

- A refresh operation fails on the node.
- Adapters are configured with an incorrect address in the cluster configuration.

Verify whether all nodes were able to complete the refresh operation, by running "Operational test 8: check if configuration instance and security status are the same across all nodes" on page 173. If this test reveals that nodes are running with different Configuration Instances (from the **lssrc** command output), at least one node was unable to complete the refresh operation successfully.

Table 33 details how to begin investigating problems that mat occur following refresh. To do so, issue one of the following commands on the correct node type:

Table 33. Initiate an investigation of problems after a refresh on Linux, AIX, Windows, and Solaris nodes

Enter this command on all Linux nodes:	Enter this command on all AIX nodes:	Enter this command on all Windows nodes:	Enter this command on all Solaris nodes:
`fcslogrpt /var/log /messages`	`errpt -J TS_* | more`	`cat /var/adm/log /messages`	`cat /var/adm /messages`

Once you have entered the appropriate command shown in the preceding table, look for **TS_** error labels. The startup script log provides more details about this problem.

Other error log entries that may be present are:

- TS_REFRESH_ER
- TS_MACHLIST_ER
- TS_LONGLINE_ER
- TS_SPNODEDUP_ER, TS_HANODEDUP_ER, or TS_CTNODEDUP_ER
- TS_SPIPDUP_ER, TS_HAIPDUP_ER, or TS_CTIPDUP_ER
- TS_IPADDR_ER
- TS_KEY_ER

For information about each error log entry and how to correct the problem, see "Error information" on page 137.

If a node does not respond to the command: **lssrc -ls** *subsystem*, (the command hangs), this indicates a problem in the connection between Topology Services and the SRC subsystem. Such problems will also cause in the Topology Services daemon to be unable to receive the refresh request.

If no **TS_** error log entry is present, and all nodes are responding to the **lssrc** command, and **lssrc** is returning different Configuration Instances for different nodes, contact the IBM Support Center.

If all nodes respond to the **lssrc** command, and the Configuration Instances are the same across all nodes, follow "Configuration verification test" on page 161 to find a possible configuration problem. Error log entry **TS_MISCFG_ER** is present if the adapter configuration collected by the configuration resource manager does not match the actual address configured in the adapter.

Table 34 details procedures for investigating various problems that may occur after refresh. These procedures vary by node type, as follows:

Table 34. Investigate problems by type after refresh on Linux, AIX, Windows, and Solaris nodes

Node	Problem investigation
On Linux nodes:	For problems caused by loss of connection with the SRC, the Topology Services subsystem may be restarted. Issuing the command: **/usr/sbin/rsct/bin/cthatsctrl -k** *will not work* because the connection with the SRC subsystem is lost. To recover, issue the **killall -q hatsd** and the **killall -q default_ip_nim** commands. If the SRC subsystem does not restart the Topology Services subsystem automatically, issue the **cthatsctrl** command: `/usr/sbin/rsct/bin/cthatsctrl -s`

Table 34. Investigate problems by type after refresh on Linux, AIX, Windows, and Solaris nodes (continued)

Node	Problem investigation

On AIX nodes:	For problems caused by loss of connection with the AIX SRC, the Topology Services subsystem may be restarted. Be aware that issuing the **/usr/sbin/rsct/bin/cthatsctrl -k** command **will not work** because the connection with the AIX SRC subsystem was lost. To recover, perform these steps:
	1. Issue the command:
	`ps -ef | grep hats | grep -v grep`
	
	to find the daemon's *process_ID*:
	The output of the command is similar to the following:
	`root 13446 8006 0 May 27 - 26:47 /usr/sbin/rsct /bin/hatsd -n 3`
	In this example, the *process_ID* is 13446.
	2. Issue the command:
	`kill process_ID`
	
	This stops the Topology Services daemon.
	3. If the AIX SRC subsystem does not restart the Topology Services subsystem automatically, issue this command:
	`/usr/sbin/rsct/bin/cthatsctrl -s`
	
	For PowerHA SystemMirror, restarting the Topology Services daemon requires shutting down the PowerHA SystemMirror cluster on the node, which can be done with the sequence:
	`smit hacmp`
	`Cluster Services`
	`Stop Cluster Services`
	
	After PowerHA SystemMirror is stopped, find the process id of the Topology Services daemon and stop it, using the command:
	`/usr/sbin/rsct/bin/topsvcsctrl`
	
	instead of the command:
	`/usr/sbin/rsct/bin/hatsctrl`
	
	Now, restart PowerHA SystemMirror on the node using this sequence:
	`smit hacmp`
	`Cluster Services`
	`Start Cluster Services`
	
	Follow the procedures in "Operational verification tests" on page 163 to ensure that the subsystem is behaving as expected across all nodes.
	Note: In the PowerHA SystemMirror/ES environment, **DO NOT STOP** the Topology Services daemon by issuing any of these commands.
	• **kill**
	• **stopsrc**
	• **topsvcsctrl -k**
	This is because stopping the Topology Services daemon while the cluster is up on the node results in the node being stopped by the PowerHA SystemMirror cluster manager.

On Windows nodes:	For problems caused by loss of connection with the SRC, the Topology Services subsystem may be restarted. Issuing the command: **/usr/sbin/rsct/bin/cthatsctrl -k** *will not work* because the connection with the SRC subsystem is lost. To recover, issue the **pkill hatsd** command. If the SRC subsystem does not restart the Topology Services subsystem automatically, issue the **cthatsctrl** command: `/usr/sbin/rsct/bin/cthatsctrl –s`

Table 34. Investigate problems by type after refresh on Linux, AIX, Windows, and Solaris nodes (continued)

Node	Problem investigation
On Solaris nodes:	For problems caused by loss of connection with the SRC, the Topology Services subsystem may be restarted. Issuing the command: **/usr/sbin/rsct/bin/cthatsctrl -k** *will not work* because the connection with the SRC subsystem is lost. To recover, issue the **pkill hatsd** command. If the SRC subsystem does not restart the Topology Services subsystem automatically, issue the **cthatsctrl** command: `/usr/sbin/rsct/bin/cthatsctrl –s`

Action 9: investigate an AIX node crash

Two possible conditions suggest this action. Take this action when an AIX node has crashed.

If an AIX node crashes, perform AIX system dump analysis.

Probable conditions suggesting this action include:

1. The deadman switch timer was triggered, probably because the Topology Services daemon was blocked.
2. An AIX-related problem.

When the node restarts, issue the command:

```
errpt -J KERNEL_PANIC
```

to look for any AIX error log entries that were created when the node crashed. If this command produces an output like:

```
IDENTIFIER TIMESTAMP  T C RESOURCE_NAME  DESCRIPTION
225E3B63   0821085101 T S PANIC          SOFTWARE
                                        PROGRAM ABNORMALLY
                                        TERMINATED
```

... then run:

```
errpt -a
```

...to get details for the event. The output of the command may be similar to the following:

```
LABEL:          KERNEL_PANIC
IDENTIFIER:     225E3B63

Date/Time:       Tue Aug 21 08:51:29
Sequence Number: 23413
Machine Id:      000086084C00
Node Id:         c47n16
Class:           S
Type:            TEMP
Resource Name:   PANIC

Description  SOFTWARE PROGRAM ABNORMALLY TERMINATED
           Recommended Actions
           PERFORM PROBLEM DETERMINATION PROCEDURES
```

```
Detail Data
ASSERT STRING

PANIC STRING
RSCT Dead Man Switch Timeout for PSSP; halting non-responsive node
```

If the `RSCT Dead Man Switch Timeout for PSSP` string appears in the output above, the crash was caused by the deadman switch timer trigger. Otherwise, there is another source for the problem. For problems unrelated to the deadman switch timer, contact the IBM Support Center.

If the dump was produced by the deadman switch timer, it is likely that the problem was caused by the Topology Services daemon being blocked. PowerHA SystemMirror/ES uses this mechanism to protect data in multitailed disks. When the timer is triggered, other nodes are already in the process of taking over this node's resources, since Topology Services is blocked in the node. If the node was allowed to continue functioning, both this node and the node taking over this node's disk would be concurrently accessing the disk, possibly causing data corruption.

The dead man switch timer is periodically stopped and reset by the Topology Services daemon. If the daemon gets blocked and does not have a chance to reset the timer, the timer-handling function runs, causing the node to crash. Each time the daemon resets the timer, the remaining amount left in the previous timer is stored. The smaller the remaining time, the closer the system is to triggering the timer. These "time-to-trigger" values can be retrieved with command:

```
/usr/sbin/rsct/bin/hatsdmsinfo
```

The output of this command is similar to:

```
Information for Topology Services -- PowerHA SystemMirror/ES
DMS Trigger time: 8.000 seconds.
Last DMS Resets                          Time to Trigger (seconds)
11/11/08 09:21:28.272                     7.500
11/11/08 09:21:28.772                     7.500
11/11/08 09:21:29.272                     7.500
11/11/08 09:21:29.772                     7.500
11/11/08 09:21:30.272                     7.500
11/11/08 09:21:30.782                     7.490

DMS Resets with small time-to-trigger     Time to Trigger (seconds)
Threshold value: 6.000 seconds.
11/11/08 09:18:44.316                     5.540
```

If you see small "time-to-trigger" values, the PowerHA SystemMirror tunables described in “Action 5: investigate a hatsd problem” on page 181 need to be changed, and the root cause for the daemon being blocked needs to be investigated. Small "time-to-trigger" values also result in an AIX error log entry with template **TS_DMS_WARNING_ST**. Therefore, when this error log entry appears, it indicates that the system is getting close to triggering the deadman switch timer. Take actions to correct the system condition that leads to the timer trigger.

Another situation which can lead to a deadman switch timeout is when the Topology Services daemon detects that too many of its NIM processes are suffering from blockage, and it is in danger of loss of contact with other nodes due to its inability to participate in heartbeating or other cluster activities. This will cause it to deliberately allow the deadman switch to expire, in order to avoid the cluster becoming sundered and multiple nodes trying to access disks or other shared resources without coordination to prevent data corruption. To determine whether this is the case, look for the TS_DMS_EXPIRING_EM label to occur (without an accompanying TS_DMS_RESTORED_TE) prior to the KERNEL_PANIC.

Diagnosing problems with Group Services

This topic addresses diagnostic procedures and failure responses for the Group Services subsystem of RSCT.

Starting with RSCT version 3.1.0.0 on AIX, the Group Services subsystem can operate in a Cluster-Aware AIX (CAA) environment. In this environment, Group Services relies on CAA's cluster infrastructure to provide node/adapter liveness and node-to-node communication, thus removing its dependency on RSCT's Topology Services subsystem. See "Diagnosing problems with Group Services in a CAA environment" on page 223 for diagnosis guidance that is specific to RSCT in a CAA environment.

Related information:

"Diagnosing problems with Group Services in a CAA environment" on page 223
Starting with RSCT version 3.1.0.0 on AIX, the Group Services subsystem can operate in a Cluster Aware AIX (CAA) environment. In this environment, Group Services rely on the CAA cluster infrastructure to provide node and adapter liveness and node-to-node communication, thus removing its dependency on RSCT Topology Services subsystem.

Requisite function

The Group Services component of RSCT directly uses required software components that may manifest problems as error symptoms in Group Services.

If you perform all the diagnostic routines and error responses listed in this topic, and still have problems with the GS component of RSCT, you should consider these components as possible sources of the error. The following list presents components in the order that they are most likely to introduce an error, from the most likely to the least likely.

- Topology Services subsystem of RSCT
- System Resource Controller (SRC)
- **/var/ct** directory
- FFDC library
- UDP communication
- UNIX Domain Sockets

Error information

On AIX system platforms, the RSCT component subsystems write this information to the AIX error log. On Linux, Windows, and Solaris, system platforms, it writes the information to the respective system log.

For more information on the AIX error log, or the Linux, Windows, or Solaris system logs, see "Accessing logged errors" on page 2.

Error logs and templates

This topic includes the error log labels, error log types, and associated explanations for group servicdes.

Table 35 on page 193 shows the error log templates that Group Services uses.

Each entry refers to a particular instance of the Group Services daemon on the local node. One entry is logged for each occurrence of the condition, unless otherwise noted in the entry's **Detail Data** section in the AIX error log or **Details File** section in the Linux, Solaris, or Windows system log. The condition is logged on every node where the event occurred.

The **Detail Data** or **Details File** section of these entries is not translated and appears only in English.

The valid error types are:

A Alert, which indicates a failure in a Group Services client

E Error, which indicates a failure in Group Services

I Informational, which indicates status information

Table 35. Error log templates for Group Services

Label	Type	Diagnostic explanation and details
GS_ASSERT_EM	E	**Explanation:** The Group Services daemon produced a core dump. **Details:** The Group Services daemon encountered an irrecoverable assertion failure. This occurs only if the daemon core dumps due to a specific Group Services assertion failure. Group services will be restarted automatically and the situation will be cleared. However, its state is not cleared and the system administrator must determine the cause of the failure. In AIX error logs, the **REFERENCE CODE** field in the **Detail Data** section might refer to the error log entry that caused this event. See "Information to collect before contacting the IBM Support Center" on page 16 and contact the IBM Support Center.
GS_AUTH_DENIED_ST	A	**Explanation:** An unauthorized user tried to access Group Services. **Details:** An unauthorized user tried to connect to the Group Services daemon. Standard fields indicate that Group Services daemon detected an attempt to connect from an unauthorized user. Detailed fields explain the detail information. Possibilities are: the user is not a **root** user, the user is not a member of the **hagsuser** group, or the user is not a supplemental member of the **hagsuser** group.
GS_CLNT_SOCK_ER	E	**Explanation:** A warning or error occurred on the Group Services client socket. **Details:** Group Services has an error on the client socket or the **hagsuser** group is not defined. Standard fields indicate that Group Services received an error or warning condition on the client socket. Detailed fields explain which error or warning caused this problem.
GS_DAEMON_UNRESP_WA	E	**Explanation:** The RSCT daemon (**rmcd**) is not responding, so the Group Services daemon will exit. **Details:** The RSCT daemon is not working correctly or might be blocked or the RSCT subsystem might be overloaded. Check the RSCT daemons.
GS_DEACT_FAIL_ST	I	**Explanation:** Failure of the deactivation script. **Details:**The Group Services daemon is unable to run the deactivation script. Standard fields indicate that the Group Services daemon is unable to run the script. Detailed fields give more information. The deactivation script might not exist or system resources are not sufficient to run the deactivation script.
GS_DOM_MERGE_ER	A, E	**Explanation:** Two Group Services domains were merged. **Details:** Two disjoint Group Services domains are merged because Topology Services has merged two disjoint node groups into a single node group. There may be several nodes with the same entries. Detailed fields contains the merging node numbers. At the time of domain merge, Group Services daemons on the nodes that generate **GS_DOM_MERGE_ER** entries will exit and be restarted. After the restart, (by **GS_START_ST**) Group Services will clear this situation. See "Action 2: verify the status of the Group Services subsystem" on page 218. In AIX error logs, the **REFERENCE CODE** field in the **Detail Data** section might refer to the error log entry that caused this event. See "Information to collect before contacting the IBM Support Center" on page 16 and contact the IBM Support Center.

Table 35. Error log templates for Group Services (continued)

Label	Type	Diagnostic explanation and details
GS_DOM_NOT_FORM_WA	I	**Explanation:** A Group Services domain was not formed. **Details:** The Group Services daemon writes this entry periodically until the Group Services domain is formed. There may be several nodes in the same situation at the same time. The Group Services domain cannot be formed because: • On some nodes, Topology Services might be running, but Group Services is not running. • Name server recovery protocol is not complete. This entry is written periodically until the domain is established. The entry is written as follows: every 5, 30, 60, and 90 minutes, and then once every two hours as long as the domain is not established. The domain establishment is recorded by a **GS_MESSAGE_ST** template label. In AIX error logs, the **REFERENCE CODE** field in the **Detail Data** section might refer to the error log entry that caused this event.
GS_ERROR_ER	A, E	**Explanation:** A Group Services logic failure occurred. **Details:** The Group Services daemon encountered an irrecoverable logic failure. Detailed fields describes what kind of error is encountered. The Group Services daemon exits due to the Group Services logic failure. Group Services will be restarted automatically and the situation will be cleared. However, if the state is not cleared, the administrator must determine what caused the group services daemon to terminate. In AIX error logs, the **REFERENCE CODE** field in the **Detail Data** section might refer to the error log entry that caused this event. See "Information to collect before contacting the IBM Support Center" on page 16 and contact the IBM Support Center.
GS_GLSM_ERROR_ER	A, E	**Explanation:** A Group Services globalized switch membership (GLSM) daemon logic failure occurred. This entry applies to AIX only. **Details:** The Group Services GLSM daemon encountered an irrecoverable logic failure. Standard fields indicate that the daemon stopped. Detailed fields point to the error log entry created when the daemon started. The Group Services GLSM daemon exited due to the logic failure. The Group Services GLSM daemon will be restarted automatically and the situation will be cleared. However, if the state is not cleared, the administrator must determine what caused the problem. The standard fields are self-explanatory. The **REFERENCE CODE** field in the **Detail Data** section might refer to the error log entry that caused this event. See "Information to collect before contacting the IBM Support Center" on page 16 and contact the IBM Support Center.
GS_GLSM_START_ST	I	**Explanation:** The Group Services GLSM daemon started. This entry applies to AIX only. **Details:** The Group Services GLSM daemon has started. Standard fields indicate that the daemon started. Detailed fields contain the path name of the log file. The Group Services GLSM subsystem was started by a user or by a process. Issue this command: **lssrc -l -s** *glsm_subsystem* If the daemon is started, the output will contain a status of "active" for **cthagsglsm**. Otherwise, the output will contain a status of "inoperative" for **cthagsglsm**.

Table 35. Error log templates for Group Services (continued)

Label	Type	Diagnostic explanation and details
GS_GLSM_STARTERR_ER	A, E	**Explanation:** The Group Services GLSM daemon cannot be started. This entry applies to AIX only. **Details:** The Group Services GLSM daemon encountered a problem during startup. Standard fields indicate that the daemon is stopped. Detailed fields point to the error log entry created when the daemon started. The Group Services daemon cannot be started because **exec** to **hagsglsmd** has failed. The AIX log entry may be the only remaining information about the cause of the problem after it is cleared.
GS_GLSM_STOP_ST	I	**Explanation:** The HA Group Services globalized switch membership (HAGSGLSM) daemon stopped. This entry applies to AIX only. **Details:** The Group Services GLSM daemon was stopped by a user or by a process. Standard fields indicate that the daemon stopped. Detailed fields point to the error log entry created when the daemon started. If the daemon was stopped by the system resource controller (SRC), the phrase **SRC** will be present in the **Detail Data** section. The **REFERENCE CODE** field in the **Detail Data** section might refer to the error log entry that caused this event. Issue this command: **lssrc -l -s** *glsm_subsystem* If the daemon is stopped, the output will contain a status of "inoperative" for **cthagsglsm**. Otherwise, the output will contain a status of "active" for **cthagsglsm**.
GS_HATS_BLOCKED_ER	E	**Explanation:** The Topology Services subsystem is not responding. **Details:** The Topology Services daemon is blocked due to resource contention.
GS_INVALID_MSG_ER	A, E	**Explanation:** The Group Services daemon received an unknown message. **Details:** The Group Services daemon received an incorrect or unknown message from another daemon. The transmitted messages may be corrupted on the wire, or a daemon sent a corrupted message. The Group Services daemon will restart and clear the problem. See “Information to collect before contacting the IBM Support Center” on page 16 and contact the IBM Support Center.
GS_MESSAGE_ST	I	**Explanation:**This is an informational message about Group Services. **Details:** The Group Services daemon has an informational message about the Group Services activity, or condition. Detailed fields describes the information. It is one of the following: 1. The Group Services daemon is not connected to Topology Services. 2. The Group Services domain has not recovered or been established after a long time. 3. Any other message, which will be in the detailed field. In AIX error logs, the **REFERENCE CODE** field in the **Detail Data** section might refer to the error log entry that caused this event.
GS_START_ST	I	**Explanation:** The Group Services daemon started. **Details:** The Group Services subsystem is started by a user or by a process. Detailed fields contain the log file name.

Table 35. Error log templates for Group Services (continued)

Label	Type	Diagnostic explanation and details
GS_STARTERR_ER	A, E	**Explanation:** Group Services cannot be started. **Details:** The Group Services daemon encountered a problem during startup. **Information about the cause of this problem may not be available once the problem is cleared.** The group services daemon cannot start because one of the following conditions occurred: 1. **exec** to **hagsd** failed. 2. The environment variables used by the startup scripts are not set properly. 3. Daemon initialization failed.
GS_STOP_ST	I	**Explanation:** The Group Services daemon stopped. **Details:** The Group Services daemon was stopped by a user or by a process. Detailed fields indicate how the daemon stops. If this was not intended, the system administrator must determine what caused the Group Services daemon to terminate. If the daemon was stopped by the system resource controller (SRC), the phrase **SRC** will be present in the **Detail Data** or **Details File** section.
Group Services_TS_RETCODE_ER	A, E	**Explanation:** The Topology Services library detected an error condition. **Details:** The Group Services daemon received an incorrect or unknown message from another daemon. This entry refers to a particular instance of the Topology Services library on the local node. Standard fields indicate that Group Services received an error condition from Topology Services. Detailed fields contain the explanation and Topology Services library error number. The Group Services daemon will restart and clear the problem.
GS_XSTALE_PRCLM_ER	A, E	**Explanation:** A "non-stale" proclaim message was received. This means that inconsistent domain join request messages were received. **Details:** The local node received a valid domain join request (proclaim) message from its name server twice. This should not happen in a normal situation. Detailed fields point to the error log entry of a **NodeUp** event. Topology Services reports inconsistent node down and up events among nodes. The Group Services daemon will restart and clear the problem. For more information, see the symptom "Non-stale proclaim message received" in “Error symptoms, responses, and recoveries” on page 178. In AIX error logs, the **REFERENCE CODE** field in the **Detail Data** section might refer to the error log entry that caused this event. See “Information to collect before contacting the IBM Support Center” on page 16 and contact the IBM Support Center.

Dump information

Group Services creates a core dump automatically when certain errors occur, and also provides service information that can be obtained automatically by the **ctsnap** command.

Core dump

A core dump is generated by the Group Services daemon if it encounters an undefined condition. It contains normal information saved in a core dump. The dump is specific to a particular instance of the GS daemon on the local node. Other nodes may have a similar core dump. Each core dump file is approximately 10MB in size.

The core dumps are located in: **/var/ct/***cluster_name***/run/cthags/core***. For an AIX PowerHA SystemMirror node, the core dumps are located in: **/var/ha/run/grpsvcs.***cluster***/core*** and **/var/ha/run/grpglsm.** *cluster***/core***.

Core dumps are created automatically when:

- One of the GS daemons invokes an **assert()** statement if the daemon state is undefined or encounters an undefined condition by design.
- The daemon attempts an incorrect operation, such as division by zero.
- The daemon receives a segmentation violation signal for accessing its data incorrectly.

A core dump is created manually by issuing the command:

```
kill -6 pid_of_daemon
```

where *pid_of_daemon* is obtained by issuing the command:

```
lssrc -s cthags
```

The core dump is valid as long as the executable file **/usr/sbin/rsct/bin/hagsd** is not replaced. Copy the core dumps and the executable file to a safe place. Table 36 details procedures for verifying a core dump. These procedures vary by node type, as follows:

Table 36. Verify a core dump on Linux, AIX, Windows, and Solaris nodes

On Linux nodes:	On AIX nodes:	On Windows nodes:	On Solaris nodes:
Issue this command: `gdb /usr/sbin/rsct /bin/hagsd core_file` ...where *core_file* is one of the **core*** files described previously.	Issue this command: `dbx /usr/sbin/rsct /bin/hagsd core_file` ...where *core_file* is one of the **core*** files described previously.	Issue this command: `gdb /usr/sbin/rsct /bin/hagsd core_file` ...where *core_file* is one of the **core*** files described previously.	Issue this command: `dbx /usr/sbin /rsct /bin/hagsd core_file` ...where *core_file* is one of the **core*** files described previously.

Good results are indicated by output similar to:

Table 37 details good results of a core dump. These results vary by node type, as follows:

Table 37. Sample good results of a core dump on Linux, AIX, and Solaris nodes

Node	Sample output
On Linux nodes:	GNU gdb 19991004 Copyright 1998 Free Software Foundation, Inc. GDB is free software, covered by the GNU General Public License, and you are welcome to change it and/or distribute copies of it under certain conditions. Type "show copying" to see the conditions. There is absolutely no warranty for GDB. Type "show warranty" for details. This GDB was configured as "i386-redhat-linux"... Core was generated by `hagsd cthags'. Program terminated with signal 6, Aborted. Reading symbols from /usr/lib/libsrc.so...done. Reading symbols from /usr/lib/libhb_client.so...done. Reading symbols from /usr/lib/libprm.so...done. Reading symbols from /usr/lib/libct_ffdc.so...done. Reading symbols from /usr/lib/libct_cu.so...done. Reading symbols from /usr/lib/libstdc++.so.2.9...done. Reading symbols from /lib/libm.so.6...done. Reading symbols from /lib/libc.so.6...done. Reading symbols from /usr/lib/libodm.so...done. Reading symbols from /lib/libpthread.so.0...done. Reading symbols from /usr/lib/libstdc++-libc6.1-1.so.2... done. Reading symbols from /lib/ld-linux.so.2...done. Reading symbols from /lib/libnss_files.so.2...done. Reading symbols from /lib/libnss_nisplus.so.2...done. Reading symbols from /lib/libnsl.so.1...done. Reading symbols from /lib/libnss_nis.so.2...done. #0 0x402b5d41 in __kill () from /lib/libc.so.6

Table 37. Sample good results of a core dump on Linux, AIX, and Solaris nodes (continued)

Node	Sample output
On AIX nodes:	Type 'help' for help. reading symbolic information ... [using memory image in core] IOT/Abort trap in evt._pthread _ksleep [/usr/lib/libpthrea ds.a] at 0xd02323e0 ($t6) 0xd02323e0 (_pthread_ksleep+0x9c) 804 10014 lwz r2,0x14(r1)
On Solaris nodes:	dbx /usr/sbin/rsct/bin/hagsd /var/ct /1xjOiSGFzAC930sZUTB7s0/run /cthags/core_1.5 For information about new features see `help changes' To remove this message, put `dbxenv suppress_startup_message 7.5' in your .dbxrc Reading hagsd core file header read successfully ... t@null (l@1) terminated by signal ABRT (Fin anormale)

Error results may look like output shown in the following table.

Table 38 details error results of a core dump. These results vary by node type, as follows:

Table 38. Sample error results of a core dump on Linux, AIX, and Solaris nodes

Node	Sample output
On Linux nodes:	This means that the current executable file was not the one that created the core dump. GNU gdb 19991004 Copyright 1998 Free Software Foundation, Inc. GDB is free software, covered by the GNU General Public License, and you are welcome to change it and/or distribute copies of it under certain conditions. Type "show copying" to see the conditions. There is absolutely no warranty for GDB. Type "show warranty" for details. This GDB was configured as "i386-redhat-linux"... warning: core file may not match specified executable file. Core was generated by 'hagsd cthags'. Program terminated with signal 6, Aborted. #0 0x402b5d41 in ?? ()

Table 38. Sample error results of a core dump on Linux, AIX, and Solaris nodes (continued)

Node	Sample output
On AIX nodes:	1. This means that the current executable file was not the one that created the core dump. Type 'help' for help. Core file program (hagsd) does not match current program (core ignored) reading symbolic information ... (dbx) 2. This means that the dump is incomplete due to lack of disk space. Type 'help' for help. warning: The core file is truncated.You may need to increase the ulimit for file and coredump, or free some space on the file system. reading symbolic information ... [using memory image in core] IOT/Abort trap in evt._pthread_ksleep [/usr/lib/libp threads.a] at 0xd02323e0 0xd02323e0 (_pthread_ksleep+0x9c) 80410014 lwz r2,0x14(r1) (dbx)
On Solaris nodes:	Some of the error results are: 1. This means that the current executable file was not the one that created the core dump. Type 'help' for help. dbx: core object name "hagsd" doesn't match object name "xxx" core file ignored. Use -f to force loading of corefile dbx: warning: core file header read failed 2. This means that the core file is incomplete due to insufficient disk space. Type 'help' for help. warning: The core file is truncated. You may need to increase the ulimit for file and coredump, or free some space on the filesystem. reading symbolic information ...

ctsnap dump

This dump contains diagnostic data used for RSCT problem determination. It is a collection of configuration data, log files, and other trace information for the RSCT components.

Related tasks:

"Information to collect before contacting the IBM Support Center" on page 16
There are things for you to do before you contact the IBM Support Center.

Trace information

Do *not* activate this trace facility until you have read this section completely, and understand this material. If you are not certain how to properly use this facility, or if you are not under the guidance of IBM Service, do *not* activate this facility.

Activating this facility may result in degraded performance of your system. Activating this facility may also result in longer response times, higher processor loads, and the consumption of system disk resources. Activating this facility may also obscure or modify the symptoms of timing-related problems.

The log files, including the Group Services trace logs and startup logs, are preserved as long as their total size does not exceed the default value of 5MB. If the total size is greater than 5MB, the oldest log file is removed at Group Services startup time. The total log size can be changed by issuing the **cthagstune** command.

GS service log trace

The GS service log contains a trace of the GS daemon. It is intended for IBM Support Center use only. It refers to a particular instance of the GS daemon running on the local node. When a problem occurs, logs from multiple nodes are often needed.

If obtaining logs from all nodes is not feasible, collect logs from these nodes:

- The node where the problem was detected
- The Group Services name server (NS) node. To find the NS node, see "Finding the GS name server (NS) node" on page 202.
- If the problem is related to a particular GS group, the Group Leader node of the group that is experiencing the problem. To find a Group Leader node for a specific group, see "Finding the group leader (GL) node for a specific group" on page 205.

The GS service log uses the RSCT common trace facility. The log files contain binary data and must be converted to text using the **/usr/sbin/rsct/bin/rpttr** command. The log files can only be read or written by users with **root** authority.

Service log short tracing is always in effect. Service log long tracing is activated by this command:

```
traceson -l -s cthags
```

The trace is deactivated, (reverts to short tracing) by issuing this command:

```
tracesoff -s cthags
```

The trace may produce 20MB or more of data, depending on GS activity level and length of time that the trace is running. Make sure there is adequate space in the **/var/ct** directory.

The trace is located in:

- **/var/ct/***cluster_name***/log/cthags/trace** and **/var/ct/***cluster_name***/log/cthags/trace.summary** on nodes in RSCT peer domains
- **/var/ct/***cluster_name***/log/cthags/grpsvcs_trace** and **/var/ct/***cluster_name***/log/cthags/grpsvcs_trace.summary** on nodes in PowerHA SystemMirror domains

Each time the Group Services daemon (**hagsd** is restarted, the previous trace files will be copied as:

- **/var/ct/***cluster_name***/log/cthags/trace.last** and **/var/ct/***cluster_name***/log/cthags/trace.summary.last** on nodes in RSCT peer domains
- **/var/ct/***cluster_name***/log/cthags/grpsvcs_trace.last** and **/var/ct/***cluster_name***/log/cthags/grpsvcs_trace.summary.last** on nodes in PowerHA SystemMirror domains

If there is a core file, the trace files will be copied as:

- **/var/ct/***cluster_name***/log/cthags/trace.0** and **/var/ct/***cluster_name***/log/cthags/trace.summary.0** on nodes in RSCT peer domains
- **/var/ct/***cluster_name***/log/cthags/grpsvcs_trace.0** and **/var/ct/***cluster_name***/log/cthags/grpsvcs_trace.summary.0** on nodes in PowerHA SystemMirror domains

The long trace contains this information:

1. Each Group Services protocol message sent or received
2. Each significant processing action as it is started or finished
3. Details of protocols being run

For many of the cases, log files from multiple nodes must be collected. The other nodes' log files must be collected before they wrap or are removed. By default, during the long tracing, log files will expand to a maximum of 5 times the configured log size value.

To change the configured value of the log size on a node, issue this command:

```
cthagstune -l new_length
```

where *new_length* is the number of lines in the trace log file. Then, restart the GS daemon.

To change the configured value on an AIX PowerHA SystemMirror node, perform these steps:

1. Issue this command: **smit hacmp**.
2. Select **Cluster Configuration.**
3. Select **Cluster Topology**.
4. Select **Configure Topology Services and Group Services**.
5. Select **Change/Show Topology and Group Services Configuration**.
6. Select **Group Services log length** (number of lines).
7. Enter the number of lines for each Group Services log file.

The trace file will be self-wrapped when it reaches a size limit. A file with a suffix of **.bak** is created only when **traceson** is issued. This will save the original trace file. Since **traceson** will change the trace file size while the current common trace facility cannot change size dynamically, **hagsd** will save the original file as a file with the suffix **.bak** .

Externally, the size of the log file is specified by the number of lines. However, for the trace facility, this number is converted to a file size expressed in bytes. The default file size is 1M bytes and can grow in increments of 256K bytes per 32 nodes.

Each time the daemon is restarted, a new log file is created. Only the last 3 log files are kept.

Long tracing should be activated on request from IBM Service. It can be activated (for about one minute, to avoid overwriting other data in the log file) when the error condition is still present.

Each entry is in the format: *date message*.

The short form of the service log trace is always running. It contains this information:

1. Each Group Services protocol message sent or received.
2. Brief information for significant protocols being run.
3. Significant information for possible debugging.

GS service log trace - summary log

The GS service log is a summary log that contains a trace of the GS daemon, but records only important highlights of daemon activity.

This log does not record as much information as the GS service log, and therefore it will not wrap as quickly as the GS service log. This log is more useful in diagnosing problems whose origin occurred a while ago. All information in this log is also recorded in the GS service log, provided that the log has not yet wrapped. The GS service log - summary log is intended for IBM Support Center use only. It refers to a particular instance of the GS daemon running on the local node. When a problem occurs, both logs from multiple nodes are often needed.

The GS service log uses the RSCT common trace facility. The log files contain binary data and must be converted to text using the **/usr/sbin/rsct/bin/rpttr** command. The log files can only be read or written by users with **root** authority.

The trace is located in:

- **/var/ct/***cluster_name***/log/cthags/trace** and **/var/ct/***cluster_name***/log/cthags/trace.summary** on nodes in RSCT peer domains
- **/var/ct/***cluster_name***/log/cthags/grpsvcs_trace** and **/var/ct/***cluster_name***/log/cthags/grpsvcs_trace.summary** on nodes in PowerHA SystemMirror domains

Each time the Group Services daemon (**hagsd** is restarted, the previous trace files will be copied as:

- **/var/ct/***cluster_name***/log/cthags/trace.last** and **/var/ct/***cluster_name***/log/cthags/trace.summary.last** on nodes in RSCT peer domains
- **/var/ct/***cluster_name***/log/cthags/grpsvcs_trace.last** and **/var/ct/***cluster_name***/log/cthags/grpsvcs_trace.summary.last** on nodes in PowerHA SystemMirror domains

If there is a core file, the trace files will be copied as:

- **/var/ct/***cluster_name***/log/cthags/trace.0** and **/var/ct/***cluster_name***/log/cthags/trace.summary.0** on nodes in RSCT peer domains
- **/var/ct/***cluster_name***/log/cthags/grpsvcs_trace.0** and **/var/ct/***cluster_name***/log/cthags/grpsvcs_trace.summary.0** on nodes in PowerHA SystemMirror domains

The size of the summary log file is proportional to the size of the normal log file. It is set to one-fourth the size of the normal log file, with a minimum size of 256K bytes.

Group Services startup script log

This log contains the GS daemon's environment variables and error messages where the startup script cannot start the daemon. The trace refers to a particular instance of the GS startup script running on the local node. This trace is always running. One file is created each time the startup script runs. The size of the file varies from 5KB to 10KB.

The GS service log uses the RSCT common trace facility. Because the log files do not contain binary data, they do not require conversion to text using the **/usr/sbin/rsct/bin/rpttr** command. The log files can only be read or written by users with root authority.

The trace is located in:

- **/var/ct/***cluster_name***/log/cthags/cthags.default.** *nodenum_incarnation*
- **/var/ha/log/grpsvcs.default.***nodenum_incarnation* on PowerHA SystemMirror nodes

...where *incarnation* is an increasing integer set by the GS daemon. This value can be obtained from the **NodeId** field returned by the following command:

```
hagsns -l -s gssubsys
```

The format of the data is the same as that of the GS service log trace with the long option. This information is for use by the IBM Support Center.

Finding the GS name server (NS) node

This topic details how to find the GS name server (NS) node.

Perform these steps to find out which node is the GS name server node.

1. Issue the **lssrc** command:

   ```
   lssrc -ls cthags
   ```

   If the output is similar to:

   ```
     Subsystem         Group            PID     Status
      cthags           cthags           14460   active
   0 locally-connected clients.
   HA Group Services domain information:
   ```

```
Domain established by node 6
Number of groups known locally: 1
                    Number of   Number of local
Group name          providers   providers/subscribers
cssMembership             9           1           0
```

you can obtain the node number of the name server. In this case, it is node 6, from the line `Domain established by node 6`. Do not perform any of the remaining steps.

2. If the output indicates `Domain not established`, wait to see if the problem is resolved in a few minutes, and if not, proceed to "Operational test 3: determine why the Group Services domain is not established" on page 210.
3. There is another command that is designed for the NS status display. Issue the **hagsns** command:

   ```
   /usr/sbin/rsct/bin/hagsns -s cthags
   ```

   Output is similar to:

   ```
     HA GS NameServer Status
   NodeId=1.16, pid=14460, domainId=6.14, NS established,
   CodeLevel=GSlevel(DRL=8)
   NS state=kCertain, protocolInProgress=kNoProtocol,
   outstandingBroadcast=kNoBcast
   Process started on Jun 19 18:34:20, (10d 20:19:22) ago,
   HB connection took (19:14:9).
   Initial NS certainty on Jun 20 13:48:45, (10d 1:4:57) ago,
   taking (0:0:15).
   Our current epoch of Jun 23 13:05:19 started
   on (7d 1:48:23), ago.
   Number of UP nodes: 12
   List of UP nodes:  0 1 5 6 7 8 9 11 17 19 23 26
   ```

   In this example, `domainId=6.14` means that node 6 is the NS node. Note that the `domainId` consists of a node number and an incarnation number. The incarnation number is an integer, incremented whenever the GS daemon is started.
4. The **hagsns** command output on the NS also displays the list of groups:

   ```
   We are: 6.14 pid: 10094 domaindId = 6.14 noNS = 0 inRecovery = 0,
   CodeLevel=GSlevel(DRL=8)
   NS state=kBecomeNS, protocolInProgress = kNoProtocol,
   outstandingBroadcast = kNoBcast
   Process started on Jun 19 18:35:55, (10d 20:22:39) ago,
   HB connection took (0:0:0).
   Initial NS certainty on Jun 19 18:36:12,
   (10d 20:22:22) ago, taking (0:0:16).
   Our current epoch of certainty started on Jun 23 13:05:18,
   (7d 1:53:16) ago.
   Number of UP nodes: 12
   List of UP nodes: 0 1 5 6 7 8 9 11 17 19 23 26
   List of known groups: 2.1 ha_gpfs:
   GL: 6 seqNum: 30 theIPS: 6 0 8 7 5 11 lookupQ:
   ```

   In this example, the group is **ha_gpfs**.

Displaying the preferred node list using lsrpnode -P

By default, when Group Services initially establishes the name server in a domain, it selects the lowest-numbered node as the name server candidate node, and the node of the first provider to create a

group as the group leader. During recovery, the default is to choose the next node in the node list as the new name server candidate node, and the next node in the group member list as the new group leader.

Because the Group Services daemons acting in the name server or group leader roles can consume a significant amount of system resources while performing those duties, it is sometimes desirable to define the preferred nodes for Group Services to consider (or non-preferred nodes to avoid) when selecting candidates for these roles.

A *preferred node* for the selection of name server and group leader candidates is indicated by a node having the optional **IsPreferredGSGL** attribute set to **1**. The **IsPreferredGSGL** attribute is in the **IBM.PeerNode** class.

When preferred nodes are defined in a peer domain:

- Group Services selects the lowest-numbered, preferred node to be the initial name server candidate node. Likewise, during name server recovery, Group Services will select the next node in the node list that is also a preferred node to be the new name server candidate node.
- When a provider first attempts to create a group, if the provider resides on a preferred node, that node becomes the group leader. If the provider does not reside on a preferred node but the name server does, then the name server node becomes the group leader. Otherwise, the provider's node becomes the group leader by default. During group leader recovery, Group Services will select the next node in the group member list that is also a preferred node to be the new group leader.

When creating a peer domain, you can use the **mkrpdomain** command's **-f** option with a node definition file to specify which nodes are to be preferred nodes. In the node definition file, node names that are followed by **@P** will be considered as preferred nodes for name server and group leader candidate selection. Node names that are followed by **@P** will be considered as non-preferred nodes.

Similarly, when adding one or more nodes to a peer domain, you can also use the **addrpnode** command's **-f** option with a node definition file to specify which nodes are to be considered preferred nodes in the same manner as for the **mkrpdomain** command.

To display whether the nodes in a peer domain are preferred nodes for name server and group leader candidate selection, use the **lsrponode** command with the **-P** option. For instance:

```
lsrpnode -P
```

The following output is displayed:

```
Name    OpState   RSCTVersion   Preferred
nodeA   Online    2.5.0.0       yes
nodeB   Online    2.5.0.0       yes
nodeC   Online    2.5.0.0       no
```

You can also directly display the value of the **IsPreferredGSGL** attribute of the IBM.PeerNode class for a node by using the **lsrsrc** command. For instance:

```
lsrsrc IBM.PeerNode IsPreferredGSGL
```

You can use the **chrsrc** command to change the value of the **IsPreferredGSGL** attribute for a node. For instance, the following command changes the node to a preferred node:

```
chrsrc IBM.PeerNode IsPreferredGSGL=1 -s "selection_string"
```

Even when preferred nodes are defined, a non-preferred node could still end up being selected as the name server or group leader if no preferred nodes are available. In this case, the default algorithm will be used to select the node for the name server or group leader role. Subsequently, as the topology is refreshed and changes occur to the preferred node list, Group Services will be notified of these changes

but will not take any immediate action to change name server or group leader nodes. The arrival of a preferred node will *not* cause the name server or group leader role to switch to that node from a non-preferred node. Such node changes can only occur during recovery of a failing name server or group leader node.

Finding the group leader (GL) node for a specific group

This topic describes two ways of finding the group leader node of a specific group.

There are two ways of finding the group leader node of a specific group:

1. The **hagsns** command on the NS displays the list of membership for groups, including their group leader nodes. To use this method:
   a. Find the NS node from "Finding the GS name server (NS) node" on page 202.
   b. Issue the following command on the NS node:

      ```
      /usr/sbin/rsct/bin/hagsns -s cthags
      ```

      The output is similar to:

      ```
      HA GS NameServer Status
      NodeId=6.14, pid=10094, domainId=6.14, NS established,
      CodeLevel=GSlevel(DRL=8)
      NS state=kBecomeNS, protocolInProgress=kNoProtocol,
      outstandingBroadcast=kNoBcast
      Process started on Jun 19 18:35:55, (10d 20:22:39) ago,
      HB connection took (0:0:0).
      Initial NS certainty on Jun 19 18:36:12,
      (10d 20:22:22) ago, taking (0:0:16).
      Our current epoch of certainty started on Jun 23 13:05:18,
      (7d 1:53:16) ago.
      Number of UP nodes: 12
      List of UP nodes:  0 1 5 6 7 8 9 11 17 19 23 26
      List of known groups:
      2.1 ha_gpfs: GL: 6
      seqNum: 30 theIPS: 6 0 8 7 5 11 lookupQ:
      ```

      The bottom few lines display the group membership information. For example, the GL node of the group **ha_gpfs** is node 6, and its participating nodes are "6 0 8 7 5 11".
2. If you need only the GL node of a specific group, the **hagsvote** command gives the answer. Issue the command:

   ```
   hagsvote -s cthags
   ```

   The output is similar to:

   ```
   Number of groups: 3
   Group slot #[0] Group name [HostMembership]
   GL node [Unknown] voting data:
   No protocol is currently executing in the group.
   ------------------------------------------------
   Group slot #[1] Group name [enRawMembership]
   GL node [Unknown] voting data:
   No protocol is currently executing in the group.
   ------------------------------------------------
   Group slot #[2] Group name [enMembership]
   ```

```
GL node [6] voting data:
No protocol is currently executing in the group.
```

In this output, node 6 is the GL node of the group **enMembership**. If the GL node is `Unknown`, this indicates that no client applications tried to use the group on this node, or the group is one of the adapter groups.

Changing the trace level for trace files

This topic details trace level change commands.

Issue any of the following commands described in Table 39 to dynamically change the trace level for trace files:

Table 39. Trace level change commands

Trace level	Command
PRM trace at level 2	**traceson –s cthags**
PRM trace at level 4	**ctsettrace –a "_PRM:*=4"–s cthags**
PRM trace at level 6	Use either of the following commands: • **traceson –ls cthags** • **ctsettrace –a "_PRM:*=6" –s cthags**
Enable hagsd DEBUG level traces	**traceson –ls cthags**
Disable PRM traces	Use either of the following commands: • **tracesoff** • **ctsettrace –a "_PRM:*=0" –s cthags**

Diagnostic procedures

These tests verify the configuration and operation of Group Services.

To verify that RSCT has been installed, see the *Verifying RSCT installation* chapter of the *Administering RSCT* guide.

Configuration verification test

This test verifies that Group Services on a node has the configuration data that it needs.

Perform the following steps:

1. Perform the Topology Services Configuration verification diagnosis. See "Diagnosing problems with Topology Services" on page 133.
2. Verify that the **cthats** and **cthags** subsystems are added, by issuing the **lssrc -a** command. If **lssrc -a** does not contain **cthats** or **cthags**, or if **lssrc -s cthats** and **lssrc -s cthags** cause an error, the above setup may not be correct.
3. Verify the cluster status by issuing the command: **/usr/sbin/rsct/bin/lsclcfg** . The output of this command must contain:

   ```
   cluster_name cluster_name  node_number local-node-number
   ```

   If anything is missing or incorrect, the setup procedure may not be correct.

If this test is successful, proceed to "Operational verification tests."

Operational verification tests

Information in this topic applies to diagnostic procedures covered in this section.

The following information applies to the diagnostic procedures that follow:

- Subsystem Name: **cthags**

- Service and User log files: **/var/ct/***cluster_name***/log/cthags/trace***
- Startup Script log: **/var/ct/***cluster_name***/log/cthags/cthags.default***

Operational test 1: verify that Group Services is working properly:

Use this test to verify that Group Services is working properly.

Issue the **lssrc** command:

```
lssrc -ls cthags
```

Good results are indicated by an output similar to:

```
 Subsystem         Group            PID     Status
  cthags          cthags           22962    active
1 locally-connected clients.  Their PIDs:
25028(haemd)
HA Group Services domain information:
Domain established by node 21
Number of groups known locally: 2
                   Number of   Number of local
Group name         providers   providers/subscribers
ha_gpfs             6            1           0
```

Error results are indicated by one of the following:

1. A message similar to:

   ```
   0513-036 The request could not be passed to the cthags subsystem.
        Start the subsystem and try your command again.
   ```

   This means that the Group Services daemon is not running. The Group Services subsystem is down. Proceed to "Operational test 2: determine why the Group Services subsystem is not active" on page 208.
2. A message similar to:

   ```
   0513-085 The cthags Subsystem is not on file.
   ```

   This means that the Group Services subsystem is not defined to the SRC.

   Use the **lsrpnode** command to determine whether or not the node is online in the cluster. For complete syntax information on the **lsrpnode** command, see its man page in the *Technical Reference: RSCT for AIX* or *Technical Reference: RSCT for Multiplatforms* guides.
3. Output similar to:

   ```
   Subsystem         Group            PID     Status
   cthags            cthags           7350     active
    Subsystem cthags trying to connect to Topology Services.
   ```

   This means that Group Services is not connected to Topology Services. Check the Topology Services subsystem. See "Diagnosing problems with Topology Services" on page 133.
4. Output similar to:

   ```
   Subsystem         Group            PID     Status
   cthags            cthags          35746    active
    No locally-connected clients.
    HA Group Services domain information:
    Domain not established.
    Number of groups known locally: 0
   ```

This means that the GS domain is not established. This is normal during the Group Services startup period. Retry this test after about three minutes. If this situation continues, perform "Operational test 3: determine why the Group Services domain is not established" on page 210.

5. Output similar to:

```
Subsystem         Group            PID     Status
 cthags            cthags          35746    active
  No locally-connected clients.
  HA Group Services domain information:
  Domain is recovering.
  Number of groups known locally: 0
```

This means that the GS domain is recovering. It is normal during Group Services domain recovery. Retry this test after waiting three to five minutes. If this situation continues, perform "Operational test 3: determine why the Group Services domain is not established" on page 210.

6. For AIX, an output similar to the **Good results** , but no **cssMembership** group is shown on nodes that have the SP switch. Proceed to "Operational test 7 (AIX only): verify the hagsglsm subsystem" on page 215.

Operational test 2: determine why the Group Services subsystem is not active:

Use this test to determine why the Group Services subsystem has become inactive.

Table 40 describes how to determine why a Group Services subsystem may not be active. These procedures vary by node type, as follows:

Table 40. Determine why the Group Services subsystem is not active on Linux, AIX, Windows, and Solaris nodes

Node	Instructions
On Linux nodes:	Examine at the **/var/log/messages*** files which have system logs that may indicate what the error is. For details about error log entries, look at the entries related to Group Services, which have labels beginning with **GS_**, such as **GS_START_ST**, **GS_STARTERR_ER**, and others. The error log entry, together with its description, is located in "Error logs and templates" on page 192. If there is no **GS_** error log entry explaining why the subsystem went down or could not start, the daemon may have exited abnormally. See "Core dump" on page 196 if RSCT has produced a core file in **/var/ct/***cluster_name***/run/cthags**. If there is a core file, see "Information to collect before contacting the IBM Support Center" on page 16 and contact the IBM Support Center. For errors where the daemon did start up, but then exited during initialization, locate detailed information about the problem in the Group Service start script log. For additional information, see "Group Services startup script log" on page 202.

Table 40. Determine why the Group Services subsystem is not active on Linux, AIX, Windows, and Solaris nodes (continued)

Node	Instructions

On AIX nodes:	Enter the command:
	`errpt -N cthags`
	
	and look for an entry for the *cthags*. It appears under the `RESOURCE_NAME` heading.
	
	If an entry is found, enter the command:
	
	`errpt -a -N cthags`
	
	to get details about error log entries. The entries related to Group Services are those with `LABEL` beginning with **GS_**.
	
	The error log entry, together with its description in "Error logs and templates" on page 192, explains why the subsystem is inactive.
	
	If there is no **GS_** error log entry explaining why the subsystem went down or could not start, it is possible that the daemon may have exited abnormally. Look for an error entry with `LABEL` of CORE_DUMP and `PROGRAM NAME` of **hagsd**, by issuing the command:
	
	`errpt -J CORE_DUMP`
	
	If this entry is found, see "Information to collect before contacting the IBM Support Center" on page 16 and contact the IBM Support Center.
	
	Another possibility when there is no **GS_** error log entry is that the Group Services daemon could not be loaded. In this case, a message similar to the following may be present in the Group Services startup log:
	
	`0509-036 Cannot load program hagsd because of the following errors:`
	`0509-026 System error: Cannot run a file that does not have a valid format.`
	
	The message may refer to the Group Services daemon, or to some other program invoked by the startup script **cthags**. If this error is found, see "Information to collect before contacting the IBM Support Center" on page 16 and contact the IBM Support Center.
	
	For errors where the daemon did start up but then exited during initialization, locate detailed information about the problem in the Group Service start script log. For additional information, see "Group Services startup script log" on page 202

On Windows nodes:	Examine at the **/var/adm/log/messages*** files which have system logs that may indicate what the error is. For details about error log entries, look at the entries related to Group Services, which have labels beginning with **GS_**, such as **GS_START_ST**, **GS_STARTERR_ER**, and others. The error log entry, together with its description, is located in "Error logs and templates" on page 192. If there is no **GS_** error log entry explaining why the subsystem went down or could not start, the daemon may have exited abnormally. See "Core dump" on page 196 if RSCT has produced a core file in **/var/ct/***cluster_name***/run/cthags**. If there is a core file, see "Information to collect before contacting the IBM Support Center" on page 16 and contact the IBM Support Center. For errors where the daemon did start up, but then exited during initialization, locate detailed information about the problem in the Group Service start script log. For additional information, see "Group Services startup script log" on page 202.

Table 40. Determine why the Group Services subsystem is not active on Linux, AIX, Windows, and Solaris nodes (continued)

Node	Instructions
On Solaris nodes:	Examine at the **/var/adm/messages*** files which have system logs that may indicate what the error is. For details about error log entries, look at the entries related to Group Services, which have labels beginning with **GS_**, such as **GS_START_ST**, **GS_STARTERR_ER**, and others. The error log entry, together with its description, is located in "Error logs and templates" on page 192. If there is no **GS_** error log entry explaining why the subsystem went down or could not start, the daemon may have exited abnormally. See "Core dump" on page 196 if RSCT has produced a core file in **/var/ct/***cluster_name***/run/cthags**. If there is a core file, see "Information to collect before contacting the IBM Support Center" on page 16 and contact the IBM Support Center. For errors where the daemon did start up, but then exited during initialization, locate detailed information about the problem in the Group Service start script log. For additional information, see "Group Services startup script log" on page 202.

Operational test 3: determine why the Group Services domain is not established:

Use this procedure to determine why the Group Services domain is not established or why it is not recovered.

The **hagsns** command is used to determine the name server (NS) state and characteristics. Issue the command:

```
hagsns -s cthags
```

The output is similar to:

```
HA GS NameServer Status
NodeId=0.32, pid=18256, domainId=0.Nil, NS not established,
CodeLevel=GSlevel(DRL=8)
The death of the node is being simulated.
NS state=kUncertain, protocolInProgress=kNoProtocol,
outstandingBroadcast=kNoBcast
Process started on Jun 21 10:33:08, (0:0:16) ago,
HB connection took (0:0:0).
Our current epoch of uncertainty started
on Jun 21 10:33:08, (0:0:16) ago.
Number of UP nodes: 1
List of UP nodes:  0
```

Error results are indicated by output of `NS state is kUncertain`, with the following considerations:

1. `kUncertain` is normal for a while after Group Services startup.
2. Group Services may have instructed Topology Services to simulate a node death. This is so that every other node will see the node down event for this local node. This simulating node death state will last approximately two or three minutes.

If this state does not change or takes longer than two or three minutes, proceed to check Topology Services. See "Diagnosing problems with Topology Services" on page 133.

If the Group Services daemon is not in `kCertain` or `kBecomeNS` state, and is waiting for the other nodes, the **hagsns** command output is similar to:

```
HA GS NameServer Status
NodeId=11.42, pid=21088, domainId=0.Nil, NS not established,
CodeLevel=GSlevel(DRL=8)
```

```
NS state=kGrovel, protocolInProgress=kNoProtocol,
outstandingBroadcast=kNoBcast
Process started on Jun 21 10:52:13, (0:0:22) ago,
HB connection took (0:0:0).
Our current epoch of uncertainty started
on Jun 21 10:52:13, (0:0:22) ago.
Number of UP nodes: 2
List of UP nodes:  0 11
Domain not established for (0:0:22).
   Currently waiting for node 0
```

In the preceding output, this node is waiting for an event or message from node 0 or for node 0. The expected event or message differs depending on the NS state which is shown in the second line of the **hagsns** command output.

Analyze the `NSstate` as follows:

1. `kGrovel` means that this node believes that the waiting node (node 0 in this example) will become his NS. This node is waiting for node 0 to acknowledge it (issue a Proclaim message).
2. `kPendingInsert` or `kInserting` means that the last line of the **hagsns** command output is similar to:

   ```
   Domain not established for (0:0:22).  Currently waiting for node 0.1
   ```

   This node received the acknowledge (Proclaim or InsertPhase1 message) and is waiting for the next message (InsertPhase1 or Commit message) from the NS (node 0).

   If this state does not change to `kCertain` in a two or three minutes, proceed to "Operational test 1: verify that Group Services is working properly" on page 207, for Topology Services and Group Services on the waiting node (node 0 in this example).
3. `kAscend`, `kAscending`, `kRecoverAscend`, or `kRecoverAscending` means that the last line of the **hagsns** command output is similar to:

   ```
   Domain not established for (0:0:22).  Waiting for
      3 nodes: 1 7 6
   ```

   If there are many waiting nodes, the output is similar to:

   ```
    Domain not established for(0:0:22).Waiting for
      43 nodes: 1 7 6 9 4 ....
   ```

   This node is trying to become a name server, and the node is waiting for responses from the nodes that are listed in the **hagsns** command output. If this state persists for a period between three and five minutes, proceed to "Operational test 1: verify that Group Services is working properly" on page 207, for Topology Services and Group Services on the nodes that are on the waiting list.
4. `kKowtow` or `kTakeOver` means that the last line of the **hagsns** command output is similar to:

   ```
   Domain not recovered for (0:0:22).  Currently waiting for
         node 0.1
   ```

   After the current NS failure, this node is waiting for a candidate node that is becoming the NS. If this state stays too long, proceed to "Operational test 1: verify that Group Services is working properly" on page 207, for the Topology Services and Group Services on the node that is in the waiting list.

   In this output, the value `0.1` means the following:

   - The first number ("0") indicates the node number that this local node is waiting for.
   - The second number("1") is called the incarnation number, which is increased by one whenever the GS daemon starts.

   Therefore, this local node is waiting for a response from the GS daemon of node 0, and the incarnation is 1.

Related information:

"Diagnosing problems with Group Services in a CAA environment" on page 223
Starting with RSCT version 3.1.0.0 on AIX, the Group Services subsystem can operate in a Cluster Aware AIX (CAA) environment. In this environment, Group Services rely on the CAA cluster infrastructure to provide node and adapter liveness and node-to-node communication, thus removing its dependency on RSCT Topology Services subsystem.

Operational test 4: verify whether a specific group is found on a node:

Use this procedure to verify whether a specific group is found on a node.

Issue the **lssrc** command:

```
lssrc -ls  cthags
```

Error results are indicated by outputs similar to the **error results** of "Operational test 1: verify that Group Services is working properly" on page 207 through "Operational test 3: determine why the Group Services domain is not established" on page 210

Good results are indicated by an output similar to:

```
 Subsystem         Group             PID     Status
  cthags           cthags           22962    active
1 locally-connected clients.  Their PIDs:
25028(haemd)
HA Group Services domain information:
Domain established by node 2
Number of groups known locally: 1
                   Number of   Number of local
Group name         providers   providers/subscribers
ha_gpfs             6            1           0
```

In this output, examine the `Group name` field to see whether the requested group name exists. For example, the group **ha_gpfs** has 1 local provider, 0 local subscribers, and 6 total providers.

For more information about the given group, issue the **hagsns** command:

```
hagsns -s cthags
```

on the NS node. The output is similar to:

```
HA GS NameServer Status
NodeId=6.14, pid=10094, domainId=6.14, NS established,
CodeLevel=GSlevel(DRL=8)
NS state=kBecomeNS, protocolInProgress=kNoProtocol,
outstandingBroadcast=kNoBcast
Process started on Jun 19 18:35:55, (10d 20:22:39) ago,
HB connection took (0:0:0).
Initial NS certainty on Jun 19 18:36:12,
(10d 20:22:22) ago, taking (0:0:16).
Our current epoch of certainty started
on Jun 23 13:05:18, (7d 1:53:16) ago.
Number of UP nodes: 12 List of UP nodes:  0 1 5 6 7 8 9 11 17 19 23 26
List of known groups: 2.1 ha_gpfs:
GL: 6 seqNum: 30 theIPS: 6 0 8 7 5 11 lookupQ:
```

In the last line, the nodes that have the providers of the group **ha_gpfs** are 6 0 8 7 5 11.

Operational test 5: verify whether Group Services is running a protocol for a group:

Use this test to verify whether Group Services is running a protocol for a group.

Issue the **hagsvote** command:

```
hagsvote -ls cthags
```

Compare the output to this list of choices.

1. If no protocol is running, the output is similar to:

```
Number of groups: 2
Group slot #[0] Group name [HostMembership] GL node [Unknown]
 voting data: No protocol is currently executing in the group.
------------------------------------------------------------
Group slot #[1] Group name [theSourceGroup] GL  node [1]
 voting data: No protocol is currently executing in the group.
------------------------------------------------------------
```

   In this output, no protocol is running for "theSourceGroup".

2. A protocol is running and waiting for a vote. For the group `theSourceGroup` , this node is soliciting votes and waiting for the local providers to vote. The output is similar to:

```
 Group slot #[1] Group name [theSourceGroup] GL node [1]
 voting data: Not GL in phase [1] of n-phase protocol of type [Join].
Local voting data:
Number of providers: 1
Number of providers not yet voted: 1 (vote not submitted).
Given vote:[No vote value] Default vote:[No vote value]
------------------------------------------------------
```

   The number of local providers is 1, and no voting is submitted. Its Group Leader (GL) node is 1. The output of the same command on the GL node (node 1) is similar to:

```
 Group slot #[3] Group name [theSourceGroup] GL node [1]
voting data: GL in phase [1] of n-phase protocol of type [Join].
 Local voting data:
Number of providers: 1
Number of providers not yet voted: 0 (vote submitted).
Given vote:[Approve vote] Default vote:[No vote value]
Global voting data:
Number of providers not yet voted: 1
Given vote:[Approve vote] Default vote:[No vote value]
------------------------------------------------
```

   This indicates that a total of one provider has not voted.

Operational test 6 (AIX only): verify whether the cssMembership or css1Membership groups are found on a node:

Perform this test to verify whether the cssMembership or css1Membership groups are found on a node.

If “Operational test 1: verify that Group Services is working properly” on page 207 through “Operational test 3: determine why the Group Services domain is not established” on page 210 succeeded, issue the following command:

```
lssrc -ls subsystem_name
```

The output is similar to:

```
Subsystem         Group            PID     Status
cthags            cthags           22962   active
2 locally-connected clients.  Their PIDs:
20898(hagsglsmd) 25028(haemd)
HA Group Services domain information:
Domain established by node 21
Number of groups known locally: 2
                  Number of   Number of local
Group name        providers   providers/subscribers
cssMembership          10           1           0
ha_em_peers            6            1           0
```

In the preceding output, the **cssMembership** group has 1 local provider. Otherwise, the following conditions apply:

1. No **cssMembership** or **css1Membership** exists in the output.

   There are several possible causes:

   a. **/dev/css0** or **/dev/css1** devices are down.

      Perform switch diagnosis.

   b. Topology Services reports that the switch is not stable.

      Issue the following command:

      ```
      lssrc -ls hats_subsystem
      ```

      ...where *hats_subsystem* is **cthats**, or, on PowerHA SystemMirror nodes, **topsvcs**.

      The output is similar to:

      ```
      Subsystem       Group           PID     Status
      cthats          cthats          17058   active
      Network Name   Indx Defd Mbrs St Adapter ID      Group ID
      SPether        [0]   15    2  S 9.114.61.65     9.114.61.125
      SPether        [0] en0          0x37821d69      0x3784f3a9
      HB Interval = 1 secs. Sensitivity = 4 missed beats
      SPswitch       [1]   14    0  D 9.114.61.129
      SPswitch       [1] css0
      HB Interval = 1 secs. Sensitivity = 4 missed beats
        1 locally connected Client with PID:
      hagsd( 26366)
        Configuration Instance = 926456205
        Default: HB Interval = 1 secs. Sensitivity = 4 missed beats
        Control Workstation IP address = 9.114.61.125
        Daemon employs no security
        Data segment size 7044 KB
      ```

      Find the first SPswitch row in the Network Name column. Find the St (state) column in the output. At the intersection of the first SPswitch row and state column is a letter. If it is not **S**, wait for few minutes longer since the Topology Services SPswitch group is not stable. If the state stays too long as **D** or **U**, proceed to Topology Services diagnosis. See "Diagnosing problems with Topology Services" on page 133. If the state is **S**, proceed to Step c. In this example, the state is **D**.

      The state has the following values:

      - **S** - stable or working correctly

- **D** - dead, or not working
- **U** - unstable (not yet incorporated)

c. **HAGSGLSM** is not running or waiting for Group Services protocols.
Proceed to "Operational test 7 (AIX only): verify the hagsglsm subsystem."

2. **cssMembership** or **css1Membership** exist in the output, but the number of local providers is zero.
Proceed to "Operational test 7 (AIX only): verify the hagsglsm subsystem."

Operational test 7 (AIX only): verify the hagsglsm subsystem:

Perform this test to verify the Group Services globalized switch membership (**hagsglsm**) subsystem.

Issue the following command:

```
lssrc -ls glsm_subsystem
```

...where *glsm_subsystem* is **cthagsglsm**, or, on PowerHA SystemMirror nodes, **grpglsm**.

Good results are indicated by output similar to:

- On the control workstation,

```
Subsystem    Group          PID     Status
cthagsglsm   cthags         22192   active
Status information for subsystem hagsglsm.c47s:
Connected to Group Services.
 Adapter  Group                  Mbrs   Joined  Subs'd  Aliases
 css0     (device does not exist)
          cssMembership         0       No      Yes     -
 css1     (device does not exist)
          css1Membership        0       No      Yes     -
 ml0      ml0Membership         -       No      -
Aggregate Adapter Configuration
 The current configuration id is 0x1482933.
 ml0[css0] Nodes: 1,5,9,13,17,21,25,29,33,37,41,45,49,53,57,61
 ml0[css1] Nodes: 1,5,9,13,17,21,25,29,33,37,41,45,49,53,57,61
```

- On other nodes,

```
Subsystem         Group            PID     Status
cthagsglsm        cthags           16788   active
Status information for subsystem cthagsglsm:
Connected to Group Services.
 Adapter  Group                  Mbrs   Joined  Subs'd  Aliases
 css0     cssRawMembership      16      -       Yes     1
          cssMembership         16      Yes     Yes     -
 css1     css1RawMembership     16      -       Yes     1
          css1Membership        16      Yes     Yes     -
 ml0      ml0Membership         16      Yes     -       cssMembership
Aggregate Adapter Configuration
 The current configuration id is 0x23784582.
 ml0[css0] Nodes: 1,5,9,13,17,21,25,29,33,37,41,45,49,53,57,61
 ml0[css1] Nodes: 1,5,9,13,17,21,25,29,33,37,41,45,49,53,57,61
```

Error results are indicated by one of the following outputs:

1. A message similar to:

```
0513-036 The request could not be passed to the cthags subsystem.
         Start the subsystem and try your command again.
```

This means that The **hagsglsm** daemon is not running. The subsystem is down. Issue the **errpt** command and look for an entry for the subsystem name. Proceed to "Operational test 2: determine why the Group Services subsystem is not active" on page 208.

2. A message similar to:

```
0513-085 The cthagsglsm Subsystem is not on file.
```

This means that the **hagsglsm** subsystem is not defined to the AIX SRC.

In PowerHA SystemMirror/ES, PowerHA SystemMirror may have not been installed on the node. Check the PowerHA SystemMirror subsystem.

3. Output similar to:

```
Subsystem      Group            PID     Status
cthagsglsm     cthags           26578   active
Status information for subsystem cthagsglsm:
Not yet connected to Group Services after 4 connect tries
```

The **hagsglsm** subsystem is not connected to group services. The Group Services daemon is not running. If the state is **S**, proceed to "Operational test 1: verify that Group Services is working properly" on page 207 for Group Services subsystem verification.

4. Output similar to:

```
Subsystem         Group            PID     Status
cthagsglsm        cthags            16048   active
Status information for subsystem bhagsglsm:
Waiting for Group Services response.
```

The **hagsglsm** subsystem is being connected to Group Services. Wait for a few seconds. If this condition does not change after several seconds, proceed to "Operational test 3: determine why the Group Services domain is not established" on page 210.

5. Output similar to:

```
Subsystem         Group            PID     Status
cthagsglsm        cthags             26788   active
Status information for subsystem hagsglsm:
Connected to Group Services.
 Adapter  Group                  Mbrs   Joined  Subs'd  Aliases
 css0     cssRawMembership      -       -       No      -
          cssMembership         16       No     No      -
 css1     css1RawMembership     15      -       Yes     1
          css1Membership        15      Yes     Yes     -
 ml0      ml0Membership          -      -      -        -
Aggregate Adapter Configuration
 The current configuration id is 0x23784582.
 ml0[css0] Nodes: 1,5,9,13,17,21,25,29,33,37,41,45,49,53,57,61
 ml0[css1] Nodes: 1,5,9,13,17,21,25,29,33,37,41,45,49,53,57,61
```

On nodes that have the switch, the line "cssRawMembership" has No in the Subs'd column.

Check Topology Services to see whether the switch is working. Issue the command:

```
lssrc -ls hats_subsystem
```

The output is similar to:

```
Subsystem         Group            PID     Status
 cthats            cthats             25074   active
```

```
Network Name   Indx Defd Mbrs St Adapter ID      Group ID
SPether        [0]   15   11  S 9.114.61.65     9.114.61.193
SPether        [0] en0          0x376d296c      0x3784fdc5
HB Interval = 1 secs. Sensitivity = 4 missed beats
SPswitch       [1]   14    8  S 9.114.61.129    9.114.61.154
SPswitch       [1] css0         0x376d296d      0x3784fc48
HB Interval = 1 secs. Sensitivity = 4 missed beats
 1 locally connected Client with PID:
hagsd( 14460)
 Configuration Instance = 925928580
 Default: HB Interval = 1 secs. Sensitivity = 4 missed beats
 Control Workstation IP address = 9.114.61.125
 Daemon employs no security
 Data segment size 7052 KB
```

Find the first row under `Network Name` with `SPswitch`. Find the column with heading `St` (state). Intersect this row and column. If the value at the intersection is not **S**, see **TS_LOC_DOWN_ST** (the label of a Topology Services error log template covered in "Error logs and templates" on page 137), and proceed to "Action 3: investigate local adapter problems" on page 180.

If the state is **S**, proceed to "Operational test 1: verify that Group Services is working properly" on page 207 to see whether the Group Services domain is established or not.

Error symptoms, responses, and recoveries

Use this information to diagnose problems with Group Services. Locate the symptom and perform the specified recovery action.

Use the information in Table 41 to diagnose problems with Group Services. Locate the symptom and perform the specified action.

Table 41. Group Services symptoms and recovery actions

Symptom	Error Label	Recovery
GS daemon cannot start.	GS_STARTERR_ER	"Action 1: start the Group Services daemon" on page 218
GS domains merged.	GS_DOM_MERGE_ER	"Action 2: verify the status of the Group Services subsystem" on page 218
GS clients cannot connect or join the GS daemon.	The following errors may be present: GS_AUTH_DENIED_ST GS_CLNT_SOCK_ER GS_DOM_NOT_FORM_WA	"Action 3: correct a Group Services access problem" on page 219
GS daemon died unexpectedly.	The following errors may be present: GS_ERROR_ER GS_DOM_MERGE_ER GS_TS_RETCODE_ER GS_STOP_ST GS_XSTALE_PRCLM_ER	"Action 4: correct a Group Services daemon problem" on page 220
GS domain cannot be established or recovered.	The following errors may be present: GS_STARTERR_ER GS_DOM_NOT_FORM_WA	"Action 5: correct a domain problem" on page 221
GS protocol has not been completed for a long time.	None	"Action 6: correct a protocol problem" on page 221

Table 41. Group Services symptoms and recovery actions (continued)

Symptom	Error Label	Recovery
Non-stale proclaim message received.	GS_XSTALE_PRCLM_ER	"Action 7: investigate a non-stale proclaim message" on page 222
HAGSGLSM cannot start. (AIX only.)	GS_GLSM_STARTERR_ER	"Action 8 (AIX only): correct a **hagsglsm** startup problem" on page 223
HAGSGLSM has stopped. (AIX only.)	GS_GLSM_ERROR_ER or None	"Action 9 (AIX only): the hagsglsm daemon has stopped" on page 223

Action 1: start the Group Services daemon

The symptom that suggests this action, the Group Services daemon cannot start, may be the result of several possible conditions.

Some of the possible conditions suggesting this action include:

- Configuration-related problems that prevent the startup script from obtaining configuration data from the configuration resource manager.
- Operating system-related problems such as a shortage of space in the **/var** directory or a port number already in use.
- SRC-related problems that prevent the daemon from setting the appropriate SRC environment.
- **cthags.***default-node-number_incarnation-number* is still a text file.

Run the diagnostics in "Operational test 2: determine why the Group Services subsystem is not active" on page 208 to determine the cause of the problem.

Action 2: verify the status of the Group Services subsystem

On AIX nodes, the AIX error log entry **GS_DOM_MERGE_ER** indicates that the Group Services daemon has restarted. On Linux nodes, the same entry, **GS_DOM_MERGE_ER**, in the file **/var/log/messages*** also indicates that the Group Services daemon has restarted. On Windows nodes, the same entry, **GS_DOM_MERGE_ER**, in the file **/var/adm/log/messages*** indicates that the Group Services daemon has restarted. On Solaris nodes, the **GS_DOM_MERGE_ER** entry in the file **/var/adm/messages*** also indicates that the Group Services daemon has restarted.

The most common condition suggesting this action is the Group Services daemon receipt of a **NODE_UP** event from Topology Services after the Group Services daemon formed more than one domain.

If the Group Services daemon has been restarted and a domain has been formed, no action is needed. However, if the Group Services daemon is not restarted, perform "Operational test 1: verify that Group Services is working properly" on page 207 to verify the status of the Group Services subsystem.

Perform these steps:

1. Find a node with the **GS_DOM_MERGE_ER** in the AIX error log (on AIX nodes), or in the file **/var/log/messages*** (on Linux nodes), in the file **/var/adm/log/messages*** (on Windows nodes) or in the file **/var/adm/messages*** (on Solaris nodes).
2. Find the **GS_START_ST** entry before the **GS_DOM_MERGE_ER** in the log.
3. If there is a **GS_START_ST** entry, issue the **lssrc** command:

   ```
   lssrc -l -s subsystem_name
   ```

   Where *subsystem_name* is **cthags**.
4. The **lssrc** output contains the node number that established the Group Services domain.
5. Otherwise, proceed to "Operational test 3: determine why the Group Services domain is not established" on page 210.

After the merge, the Group Services daemon must be restarted. See **TS_NODEUP_ST** (the label of a Topology Services error log template) located in "Error logs and templates" on page 137. Check it with "Operational test 2: determine why the Group Services subsystem is not active" on page 208.

Action 3: correct a Group Services access problem

This action is suggested by a number of possible conditions. Take this action when Group Services clients cannot connect to or join the Group Services daemon.

For the nodes that cannot join, some of the possible conditions suggesting this action are:

1. Group Services might not be running.
2. The Group Services domain might not be established.
3. The clients might not have permission to connect to the Group Services daemon.
4. Group Services is currently doing a protocol for the group that is trying to join or subscribe.

Analyze and correct this problem as follows:

1. Issue the **lssrc** command:

   ```
   lssrc -s cthags
   ```

   The output is similar to:

   ```
    Subsystem         Group            PID     Status
   cthags           cthags          23482    active
   ```

   If `Status` is not `active`, this indicates that the node cannot join the Group Services daemon. Perform "Operational test 2: determine why the Group Services subsystem is not active" on page 208. Start the Group Services subsystem by issuing this command:

   ```
   /usr/sbin/rsct/bin/cthagsctrl -s
   ```

   If `Status` is `active`, proceed to Step 2.
2. Perform "Operational test 1: verify that Group Services is working properly" on page 207 to check whether the Group Services domain is established or not.
3. On Linux nodes, check the file **/var/log/messages*** for an entry containing the string "GS_AUTH_DENIED_ST". On Windows nodes, check the file **/var/adm/log/messages*** for an entry containing the string "GS_AUTH_DENIED_ST". On Solaris nodes, check the file **/var/adm/messages*** for an entry containing the string "GS_AUTH_DENIED_ST". This string indicates that the user of the client program does not have correct permission to use Group Services.

   On AIX nodes, Issue the command:

   ```
   errpt -a -N subsystem_name | more
   ```

   where *subsystem_name* is **cthags**, or, on PowerHA SystemMirror nodes, **grpsvsc**.

   Check the AIX error log for this entry:

   ```
   Resource Name:   hags
   ---------------------------------------------------
   LABEL:          GS_AUTH_DENIED_ST
   IDENTIFIER:     23628CC2

   Date/Time:       Tue Jul 13 13:29:52
   Sequence Number: 213946
   Machine Id:      000032124C00
   Node Id:         c47n09
   Class:           O
   ```

```
Type:            INFO
Description
User is not allowed to use Group Services daemon

Probable Causes
The user is not the root user
The user is not a member of hagsuser group

Failure Causes Group
Services does not allow the user

          Recommended Actions
          Check whether the user is the root
Check whether the user is a member of hagsuser group

Detail Data
DETECTING MODULE
RSCT,SSuppConnSocket.C,           1.17, 421
ERROR ID
.0ncMX.ESrWr.0in//rXQ7.....................
REFERENCE CODE

DIAGNOSTIC EXPLANATION
User myuser1 is not a supplementary user of group 111.
Connection refused.
```

This explains that the user of the client program does not have correct permission to use Group Services.

The following users can access Group Services on Linux, AIX, and Solaris system platforms:

- The **root** user.
- A user who is a primary or supplementary member of the **hagsuser** group, which is defined in the **/etc/group** file.

The following user can access Group Services on Windows system platforms:

- The **Administrator** user specified during the RSCT installation.

Note: The RSCT Administrator cannot be the same as the local system Administrator user.

Change the ownership of the client program to a user who can access Group Services.

4. Issue the **hagsvote** command:

```
hagsvote -ls cthags
```

to determine whether the group is busy, and to find the Group Leader node for the specific group.

5. Issue the same command on the Group Leader Node to determine the global status of the group. Resolve the problem by the client programs.

Action 4: correct a Group Services daemon problem

This action is suggested by a number of possible conditions. Take this action when the Group Services daemon dies unexpectedly.

Some of the possible conditions suggesting this action include:

1. Domain merged.
2. The Group Services daemon received a non-stale proclaim message from its name server (NS).

If the Topology Services daemon is alive when the current NS restarts and tries to become a NS, the newly-started NS sends a proclaim message to the other nodes. These nodes consider the newly started node as their NS. The receiver nodes consider the proclaim message current (that is, "non-stale") but undefined by design. Therefore, the received Group Services daemon will be core dumped.

3. The Topology Services daemon has died.
4. The Group Services daemon has stopped.
5. Group Services has an internal error that caused a core dump.

Table 42 describes how to correct Group Services daemon problems by node type.

Table 42. Correct a Group Services daemon problem on Linux, AIX, Windows, and Solaris nodes

On Linux nodes:	On AIX nodes:	On Windows nodes:	On Solaris nodes:
Examine the error log in **/var/log/messages*** and search for **GS**_ labels or a RESOURCE NAME of any of the Group Services subsystems. If you find such an entry, the cause is explained in the DIAGNOSTIC EXPLANATION field.	Examine the AIX error log by issuing the command: `errpt -J GS_DOM_ MERGE_ER, GS_XSTALE_PRCLM_ER,GS _ERROR_ER,\ GS_STOP_ST, GS_TS_RETCODE_ER | more` ...and search for**GS**_ labels or a RESOURCE NAME of any of the Group Services subsystems. If you find such an entry, the cause is explained in the DIAGNOSTIC EXPLANATION field.	Examine the error log in **/var/adm/log/messages*** and search for **GS**_ labels or a RESOURCE NAME of any of the Group Services subsystems. If you find such an entry, the cause is explained in the DIAGNOSTIC EXPLANATION field.	Examine the error log in **/var/adm/messages*** and search for **GS**_ labels or a RESOURCE NAME of any of the Group Services subsystems. If you find such an entry, the cause is explained in the DIAGNOSTIC EXPLANATION field.

If there has been a Group Services core dump, the core file is in: **/var/ct/***cluster_name***/run/cthags**. Save this file for error analysis.

Action 5: correct a domain problem

This action is suggested by a number of possible conditions. Take this action when the Group Services domain cannot be established or recovered.

Some of the possible conditions suggesting this action include:

1. Topology Services are running, but the Group Services daemon is not running on some of the nodes.
2. Group Services internal NS protocol is currently running.

Proceed to "Operational test 3: determine why the Group Services domain is not established" on page 210.

Action 6: correct a protocol problem

This action is suggested when the related client failed to vote for a specific protocol. Take this action when the Group Services protocol has not been completed for a long time.

Issue the **hagsvote** command on any node that has target groups:

```
hagsvote -ls cthags
```

If this node did not vote for the protocol, the output is similar to:

```
 Number of groups: 1
Group slot #[3] Group name [theSourceGroup] GL node [0] voting data:
Not GL in phase [1] of n-phase protocol of type [Join].
Local voting data: Number of providers: 1
```

```
Number of providers not yet voted: 1 (vote not submitted).
Given vote:[No vote value] Default vote:[No vote value]
ProviderId      Voted?  Failed? Conditional?
[101/11]        No      No      Yes
```

As the preceding text explains, one of local providers did not submit a vote. If this node has already voted but the overall protocol is still running, the output is similar to:

```
 Number of groups: 1
Group slot #[3] Group name [theSourceGroup] GL node [0] voting data:
Not GL in phase [1] of n-phase protocol of type [Join].
Local voting data: Number of providers: 1
Number of providers not yet voted: 0 (vote submitted).
Given vote:[Approve vote] Default vote:[No vote value]
ProviderId      Voted?  Failed? Conditional?
[101/11]        Yes     No      Yes
```

In this case, issue the same command on the Group Leader node. The output is similar to:

```
 Number of groups: 1
Group slot #[2] Group name [theSourceGroup] GL node [0] voting data:
GL in phase [1] of n-phase protocol of type [Join].
Local voting data: Number of providers: 1
Number of providers not yet voted: 1 (vote not submitted).
Given vote:[Approve vote] Default vote:[No vote value]
ProviderId      Voted?  Failed? Conditional?
[101/0] No      No      No

Global voting data:
Number of providers not yet voted: 1
Given vote:[Approve vote] Default vote:[No vote value]
Nodes that have voted: [11]
Nodes that have not voted: [0]
```

The Group Leader's output contains the information about the nodes that did not vote. Investigate the reason for their failure to do so. Debug the Group Services client application.

Action 7: investigate a non-stale proclaim message

The local Group Services daemon receives a valid domain join request (proclaim) message from its name server more than once. This typically happens when Topology Services notifies Group Services of inconsistent node events.

RSCT should automatically resolve the symptom that suggests this action, receipt of a non-stale proclaim message, if you see a **GS_START_ST** entry after the symptom occurs.

Perform these actions:

1. Find the **GS_START_ST** entry in the AIX error log (on AIX nodes), the **/var/log/messages** file (on Linux nodes), the **/var/adm/log/messages** file (on Windows nodes), or the **/var/adm/messages** file (on Solaris nodes).
2. If there is a **GS_START_ST** entry, issue the **lssrc** command:

   ```
   lssrc -l -s cthags
   ```
3. The **lssrc** output contains the node number that established the Group Services domain.
4. Otherwise, proceed to "Operational test 4: verify whether a specific group is found on a node" on page 212.

If this problem continues, contact the IBM Support Center (see “Information to collect before contacting the IBM Support Center” on page 16).

Action 8 (AIX only): correct a hagsglsm startup problem

This action is suggested by a number of possible conditions. Take this action when **HAGSGLSM** cannot start.

Some of the possible conditions suggesting this action include:

- AIX-related problems such as a shortage of space in the **/var** directory or a port number already in use.
- SRC-related problems that prevent the daemon from setting the appropriate SRC environment.

Proceed to “Operational test 7 (AIX only): verify the hagsglsm subsystem” on page 215.

Action 9 (AIX only): the hagsglsm daemon has stopped

This action is suggested by one condition. Take this action when the **hagsglsm** daemon has stopped.

To resolve that condition, issue the command:

```
lssrc -l -s cthagsglsm
```

If the daemon is stopped, the output will contain a status of "inoperative" for **hagsglsm**. Otherwise, the output will contain a status of "active" for **hagsglsm**. If stopping the daemon was not intended, see “Information to collect before contacting the IBM Support Center” on page 16 and contact the IBM Support Center.

Diagnosing problems with Group Services in a CAA environment

Starting with RSCT version 3.1.0.0 on AIX, the Group Services subsystem can operate in a Cluster Aware AIX (CAA) environment. In this environment, Group Services rely on the CAA cluster infrastructure to provide node and adapter liveness and node-to-node communication, thus removing its dependency on RSCT Topology Services subsystem.

Instead of connecting to the Topology Services daemon, an alternative version of the Topology Services client library obtains node and adapter liveness information directly from the low-level cluster services in the CAA environment.

In a CAA environment, Group Services provides the same published APIs as it does in a non-CAA environment:

- Application-initiated protocols must operate as they do in a non-CAA environment.
- Protocols initiated by Group Services also operate the same, except that node failures are detected by the CAA infrastructure, and not by Topology Services.
- Adapter subscriptions are supported. Using the AIX Event Infrastructure (Autonomic Health Advisor File System) and the cluster query APIs, the CAA infrastructure provides information about the configuration and liveness for all network interfaces in the cluster. The AIX Event Infrastructure is an event-monitoring framework for monitoring predefined and user-defined events.
- Adapter subscriptions include network interface aliases.

In addition, a mechanism is introduced that allows PowerHA SystemMirror clusters or RSCT peer domains to migrate a cluster from a pre-CAA environment to a CAA environment.

Starting with RSCT version 3.1.4.0 on the AIX operating system, RSCT supports the linked cluster over two sites in a CAA environment. The CAA environment provides a cluster view across sites.

Health check for CAA clusters, linked clusters, and RSCT in a CAA environment

The minimum software versions you need to use CAA environment are AIX 6.1 with the 6100-06 Technology Level (or later) or AIX 7.1, PowerHA SystemMirror 7.1, and RSCT 3.1.0.0 (or later).

The minimum software versions that you need to use CAA linked clusters are IBM AIX 6 with Technology Level 8 and IBM AIX 7 with Technology Level 2, PowerHA SystemMirror 7.1.2, and RSCT 3.1.4.0.

The CAA environment provides a cluster view across sites.

1. Check the AIX and RSCT versions.

   To check the AIX version, run this command:

   ```
   cat /proc/version
   ```

   Example

   This output indicates that you are running AIX 6.1 with the 6100-06 Technology Level:

   ```
   > cat /proc/version
   Jul 31 2011
   13:37:58
   1031A_61L
   @(#) _kdb_buildinfo unix_64 Jul 31 2011 13:37:58 1031A_61L
   ```

   To check the RSCT version, run this command:

   ```
   /usr/sbin/rsct/install/bin/ctversion
   ```

   Example

   This output indicates that you are running RSCT version 3.1.4.0:

   ```
   > /usr/sbin/rsct/install/bin/ctversion
   rkod1231a 3.1.4.0
   ```

2. Check the CAA state.

   a. Check to see whether the CAA state is active and the **clcmd** command is working.

      First, run this command:

      ```
      lscluster
      ```

      Example

      This output indicates that a CAA cluster is not created on this node:

      ```
      > lscluster
      Cluster services are not active.
      ```

      Note: The command output might change.

   b. Check to see whether a CAA cluster is created.

      Run this command:

      ```
      lscluster -m
      ```

      Example 1

      A normal cluster is considered as a local site cluster.

      This output indicates that the CAA cluster named **z** has two nodes and shows the cluster ID and the UUIDs of the nodes:

      ```
      > lscluster -m
      Calling node query for all nodes
      Node query number of nodes examined: 2


      Node name: e95n1sq11.ppd.pok.ibm.com
      Cluster shorthand id for node: 1
      uuid for node: 61cbd2ec-9597-11df-a7f1-c291e3facdeb
      State of node: UP NODE_LOCAL
      Smoothed rtt to node: 0
      Mean Deviation in network rtt to node: 0
      Number of clusters node is a member in: 1
      ```

```
CLUSTER NAME   SHID         UUID
z               0         aa9cf1b0-a61b-11df-bbae-c291e3facdeb
SITE NAME      SHID         UUID
LOCAL           1         51735173-5173-5173-5173-517351735173

 Points of contact for node: 0
------------------------------


Node name: e95n1sq12.ppd.pok.ibm.com
Cluster shorthand id for node: 2
uuid for node: 61a9c45e-9597-11df-a25b-c291ef38e2f4
State of node: UP
Smoothed rtt to node: 7
Mean Deviation in network rtt to node: 3
Number of clusters node is a member in: 1
CLUSTER NAME   SHID          UUID
z               0            aa9cf1b0-a61b-11df-bbae-c291e3facdeb
SITE NAME      SHID          UUID
LOCAL           1            51735173-5173-5173-5173-517351735173

Points of contact for node: 2
----------------------------------------
Interface     State  Protocol    Status
----------------------------------------
dpcom        DOWN     none      RESTRICTED
en3          UP       IPv4         none
```

Note: The command output might change.

Example 2

This output is a sample of a linked cluster setup. It indicates that the CAA cluster named **z** has two nodes and shows the cluster ID, the UUIDs of the nodes, and UUIDs of the site:

```
> lscluster -m
Calling node query for all nodes
Node query number of nodes examined: 2
Node name: e95n1sq11.ppd.pok.ibm.com
Cluster shorthand id for node: 1
uuid for node: 61cbd2ec-9597-11df-a7f1-c291e3facdeb
State of node: UP NODE_LOCAL
Smoothed rtt to node: 0
Mean Deviation in network rtt to node: 0
Number of clusters node is a member in: 1
CLUSTER NAME   SHID         UUID
z               0           aa9cf1b0-a61b-11df-bbae-c291e3facdeb
SITE NAME      SHID         UUID
site2           2           ebaf84a4-c60a-11e1-9d8a-36f9d3883002
Points_of_contact for node: 0
------------------------------
Node name: e95n1sq12.ppd.pok.ibm.com
Cluster shorthand id for node: 2
uuid for node: 61a9c45e-9597-11df-a25b-c291ef38e2f4
State of node: UP
Smoothed rtt to node: 7
Mean Deviation in network rtt to node: 3
Number of clusters node is a member in: 1
CLUSTER NAME    SHID       UUID
z                0         aa9cf1b0-a61b-11df-bbae-c291e3facdeb
SITE NAME       SHID       UUID
site1            1         680d04d8-c609-11e1-8bde-36f9d3883002

 Points of contact for node: 1
 ----------------------------------------
 Interface     State  Protocol    Status
 ----------------------------------------
 tcpsock->01   UP     none       none
```

Example 3

This output indicates that a CAA cluster is not created on this node:

```
> clcmd date
Node is not in a cluster.
Node must be in clustered environment to run clcmd.
> lscluster -m
Cluster services are not active.
>
```

Note: The command output might change.

c. Check the CAA cluster configuration.

The **/usr/sbin/rsct/bin/caa_config** command is used to check CAA configuration status. The **/usr/sbin/rsct/bin/caa_config -h** command provides information about how to use this command to check the CAA cluster configuration. The output of this command is used for debugging purposes and for health checking in the diagnosis of problems.

Example 1

Run the **caa_config -h** command to get usage information:

```
/usr/sbin/rsct/bin/caa_config -h
```

The following output is displayed:

```
> /usr/sbin/rsct/bin/caa_config -h
Usage: ./caa_config [-v] |
[-C | -c cluster_name]
[-N | -n cluster_name]
[-I | -i cluster_name]
[-S | -s cluster_name]
[-T trace_file_name]
[-L trace_levels_string] (default is "*:*=8")
[-b net_intf_blob_file_from_trace]
[-h]
.
.
.
```

Note: The command output might change.

Example 2

Run the **caa_config -c** command to display the node number, UUID, cluster ID, and other information for cluster **z**:

```
/usr/sbin/rsct/bin/caa_config -c z
```

The following output is displayed:

```
cluster(z) node(1)
uuid(aa9cf1b0-a61b-11df-bbae-c291e3facdeb)
id(2gdF6mfXiHtxkkmf7Z~ith) cluster_shid(0)
cluster_type(1)

OS supports CAA
cluster is configured on node
cluster is enabled on node
local node's uuid(61cbd2ec-9597-11df-a7f1-c291e3facdeb)
Site bit set: 1
=== /tmp/Cluster/ENVIRONMENT content:
scaffold : lscluster_clcmd : false
supported : simulate : false
configured : simulate : false
enabled : simulate : false
===
```

Note: The command output might change.

Example 3

Run the **caa_config -n** command to display information about the two nodes of the cluster **z**:

```
/usr/sbin/rsct/bin/caa_config -n z
```

The following output is displayed:

```
> ./caa_config -n z
cluster(z)
number(1)
id(61cbd2ec-9597-11df-a7f1-c291e3facdeb)
site_id(680d04d8-c609-11e1-8bde-36f9d3883002)
p_name(e95n1sq11.ppd.pok.ibm.com) status(1)
number(2)
id(61a9c45e-9597-11df-a25b-c291ef38e2f4)
site_id(680d04d8-c609-11e1-8bde-36f9d3883002)
p_name(e95n1sq12.ppd.pok.ibm.com) status(1)

OS supports CAA
cluster is configured on node
cluster is enabled on node
local node's uuid(61cbd2ec-9597-11df-a7f1-c291e3facdeb)
Site bit set: 1
=== /tmp/Cluster/ENVIRONMENT content:
scaffold : lscluster_clcmd : false
supported : simulate : false
configured : simulate : false
enabled : simulate : false
===
```

Note: The command output might change.

Example 4

Run the **caa_config -i** command to display network interface information for the two nodes of the cluster **z**:

```
/usr/sbin/rsct/bin/caa_config -i z
```

The following output is displayed:

```
> ./caa_config -i z
cluster(z)
node_number(1) p_name(en0)
ip_addr(9.114.55.11)
netmask(255.255.255.128)
status(1)
flags(0x1e080863) type(1) hwaddr(0xc291e3facdeb)
node_number(1) p_name(en2)
ip_addr(9.114.122.122)
netmask(255.255.255.0)
status(0)
flags(0x1e080862) type(1) hwaddr(0xc291e3facded)
node_number(1) p_name(en1)
ip_addr(9.114.111.111)
netmask(255.255.255.0)
status(1)
flags(0x1e080863) type(1) hwaddr(0xc291e3facdec)
node_number(2) p_name(en0)
ip_addr(9.114.55.12)
netmask(255.255.255.128)
status(1)
flags(0x1e080863) type(1) hwaddr(0xc291ef38e2f4)
node_number(2) p_name(en1)
ip_addr(::ffff:0.0.0.0)
netmask(::ffff:0.0.0.0)
status(0)
flags(0x1e080862) type(1) hwaddr(0xc291ef38e2f5)
```

```
OS supports CAA
cluster is configured on node
cluster is enabled on node
local node's uuid(61cbd2ec-9597-11df-a7f1-c291e3facdeb)
Site bit set: 1
=== /tmp/Cluster/ENVIRONMENT content:
scaffold : lscluster_clcmd : false
scaffold : discover_all_net_intf_addrs : false
supported : simulate : false
configured : simulate : false
enabled : simulate : false
===
[e95n1sq11][/usr/sbin/rsct/bin]>
```

Note: The command output might change.

Example 5

Run the **caa_config -s** command to display site information for cluster **z**:

```
/usr/sbin/rsct/bin/caa_config –s z
```

The following output is displayed:

```
cluster(z)
site_number(2) site_id(ebaf84a4-c60a-11e1-9d8a-36f9d3883002) site_name(site2) site_policy(1)
site_priority(2) site_merge_flag(2) site_nodes_up(1)
     node_number(2) node_id(faa326b6-c77a-11e1-a38f-36f9d3883002) node_name(e95n1sq11.ppd.pok.ibm.com) node_status(1)
site_number(1) site_id(680d04d8-c609-11e1-8bde-36f9d3883002) site_name(site1) site_policy(1)
site_priority(1) site_merge_flag(2) site_nodes_up(2)
     node_number(1) node_id(68136ada-c609-11e1-8bde-36f9d3883002) node_name(e95n1sq12.ppd.pok.ibm.com) node_status(1)

OS supports CAA R2
Mapped Active RSCT Version is 0x3010400
Mapped Active RSCT Version String is 3.1.4.0
cluster is configured on node
cluster is enabled on node
Not running in a VIOS
local node's uuid(faa326b6-c77a-11e1-a38f-36f9d3883002)
Site bit set: 1
=== /tmp/Cluster/ENVIRONMENT content:
supported : simulate : false
configured : simulate : false
scaffold : lscluster_clcmd : false
enabled : simulate : false
===
```

Note: The command output might change.

d. Check CAA communication status.

The **/usr/sbin/rsct/bin/caa_comm** command is used to check CAA communication status. The **/usr/sbin/rsct/bin/caa_comm -h** command provides information about how to use this command to check CAA communication status. The output of this command is used for debugging purposes and for health checking in the diagnosis of problems.

Example 1

Run the **caa_comm -h** command to get usage information:

```
/usr/sbin/rsct/bin/caa_comm -h
```

The following output is displayed:

```
> /usr/sbin/rsct/bin/caa_comm -h
Usage: ./caa_comm -n node -p port {-d destination_node | -b}
[{-c [-t timeout_seconds] | -s}] [-v] [-h]
[-l number_of_bytes] ...

where:

-n specifies this node's CAA short node number (integer)
-d specifies target node's CAA short node number (integer)
-b broadcast to all nodes in the CAA cluster
-c send messages continuously (instead of just once)
-t specifies timeout seconds (long float)
```

```
-v run in verbose node
-h show usage
-s send only once and quit
-l specifies number of bytes per send
.
.
.
```

Note: The command output might change.

Example 2

This example shows that for a short message (17 bytes), the CAA communication (message transfer between the two nodes through a CAA socket) is working.

On node 1:

```
> ./caa_comm -n 1 -p 3 -d 2
PID: 7012540
RECEIVED message from sending_node(2) bytes(17) content:
iov_len(17) iov_base(message from (2))
>
> ./caa_comm -n 1 -p 3 -d 2
PID: 7012548
```

Note: The command output might change.

On node 2:

```
> ./caa_comm -n 2 -p 3 -d 1
PID: 7995448
RECEIVED message from sending_node(1) bytes(17) content:
iov_len(17) iov_base(message from (1))
```

Note: The command output might change.

Example 3

This example shows that node 2 sent a 3000-byte message, but node 1 did not get the whole message.

On node 2:

```
> ./caa_comm -n 2 -p 3 -d 1 -l 3000
PID: 6357236
```

Note: The command output might change.

On node 1:

```
> ./caa_comm -n 1 -p 3 -d 2
PID: 7012548
RECEIVED message from sending_node(2) bytes(17) content:
iov_len(17) iov_base(12345678901234567)
```

Note: The command output might change.

3. Check to see whether the Group Services subsystem is started and active.

   Run the **lssrc** command:

   ```
   lssrc -ls cthags
   ```

   Example

   This example shows that the domain is established:

   ```
   > lssrc -ls cthags
   Subsystem Group PID Status
   cthags cthags 7077938 active
   4 locally-connected clients. Their PIDs:
   5701794(IBM.ConfigRMd) 7798864(rmcd) 6619282(IBM.GFVTRMd)
   7929884(IBM.StorageRMd)
   HA Group Services domain information:
   ```

```
Domain established by node 1
Number of groups known locally: 5
Number of Number of local
Group name providers providers/subscribers
rmc_peers 2 1 0
IBM.ConfigRM 2 1 0
IBM.StorageRM.v1 2 1 0
IBM.GFVTRM 2 1 0
IBM.GFVTRMCache 2 1 0

Critical clients will be terminated if unresponsive
```

Problem determination in a CAA environment

1. **Where to find Group Services trace files in a CAA environment**
   a. For an RSCT peer domain, before and after migration to CAA mode, the Group Services trace files are in the **/var/ct/*cluster_name*/log/cthags/** directory.
   b. For a PowerHA SystemMirror domain, before migration to CAA mode, the Group Services trace files are in the **/var/ha/log/** directory. After migration to CAA mode, without stopping Group Services, the trace files are still in the **/var/ha/log/** directory. If, after migration to CAA mode, Group Services is restarted, the trace files are in the **/var/ct/*cluster_name*/log/cthags/** directory.
2. **Which Group Services subsystems are active in a CAA environment**
   a. For an RSCT peer domain in a CAA environment, the Group Services subsystem is **cthags**.
   b. For a PowerHA SystemMirror cluster in a CAA environment, the Group Services subsystem is **grpsvcs**, if Group Services are not restarted after migration to the CAA mode, or **cthags**, if Group Services are restarted after migration to CAA mode.
3. **Why Group Services are not active**

   Symptom

   In a CAA environment, PowerHA SystemMirror is started, but Group Services is not active.

   a. Check to see whether Group Services are started ever.

      Example

      This output indicates that the Group Services subsystem (**cthags**) is not started yet:

      ```
      > lssrc -ls cthags
      0513-036 The request could not be passed to the cthags subsystem.
      Start the subsystem and try your command again.
      ```

      It is possible that **ctcaactrl -s** was not called. The **ctcaactrl -s** command must be called automatically when the CAA cluster is created and when a node is rebooted.

      There is a **ctcaactrl** log file called **caa.ctcaactrl.log** in **/var/ct/**. Check the contents of this log file to see whether **ctcaactrl -s** was called.

   b. It is possible that Group Services were started, but CAA socket creation failed. First, run the **errpt -a** command and search for a Group Services error.

      Example

      If **errpt** displays this error, there might a port binding problem:

      ```
      LABEL: GS_ERROR_ER
      IDENTIFIER: 463A893D

      Date/Time: Fri Aug 6 22:19:23 GMT+08:00 2011
      Sequence Number: 51
      Machine Id: 00F610C74C00
      Node Id: prve22
      Class: O
      Type: PERM
      WPAR: Global
      Resource Name: cthags

      Description
      Internal logic error in Group Services daemon
      ```

```
Probable Causes
An internal logic failure occurs in daemon
Unexpected program failure

Failure Causes
Unrecoverable logic failure in daemon

Recommended Actions
Verify that Group Services daemon is still running
Verify that Group Services daemon has been restarted
Call IBM Service if problem persists

Detail Data
DETECTING MODULE
RSCT,pbs_init.C,1.19,165
ERROR ID
6xYcC4/fb/LA/P/y.KIUA8.....................
REFERENCE CODE

DIAGNOSTIC EXPLANATION
PrmInit(cthags) failed with rc = -1, PrmErrno = -9(System call error number -9.
)
```

Next, run this command:

```
/usr/sbin/rsct/bin/caa_comm -n local_node_number -d _node_number -p 1
```

This output indicates that there is a port binding problem with the CAA communication socket and the AF_CLUST port is busy:

```
> /usr/sbin/rsct/bin/caa_comm -n 1 -d 2 -p 1
PID: 12255468
ct_caa_get_socket() fd(-1) rc(-9) errno(67)
```

4. **Why PowerHA SystemMirror cannot connect with Group Services**

   Symptom

   PowerHA SystemMirror is started and it cannot connect with Group Services.

   Problem determination

   If PowerHA SystemMirror is migrated from pre-CAA mode to CAA mode and the Group Services daemon is not restarted, the subsystem is **grpsvcs**. So, when checking the state of Group Services, use **grpsvcs** as the subsystem name.

   If the daemon is restarted or if it is from an RSCT peer domain that is migrated from pre-CAA mode to CAA mode, the subsystem is **cthags**. So, when checking the state of Group Services, use **cthags** as the subsystem name.

   Here are some possible reasons why PowerHA SystemMirror cannot connect with Group Services:

   a. It is possible that Group Services were not started or that CAA port binding failed. First, check to see whether Group Services are active and if the CAA communication socket has a problem.
   b. It is possible that the Group Services name server domain is not established yet. If Group Services are active, check to see whether the domain is established by running one of these **lssrc** commands:

      ```
      lssrc -ls cthags

      lssrc -ls grpsvcs
      ```

      Example 1

      In this example, the Group Services subsystem is **cthags**. This **lssrc -ls cthags** output indicates that the domain is established by node 1:

      ```
      > lssrc -ls cthags
      Subsystem Group PID Status
      cthags cthags 6619250 active
      ```

```
4 locally-connected clients. Their PIDs:
5701794(IBM.ConfigRMd) 7929980(rmcd) 7077944(IBM.GFVTRMd)
7798860(IBM.StorageRMd)
HA Group Services domain information:
Domain established by node 1
Number of groups known locally: 5
Number of Number of local
Group name providers providers/subscribers
rmc_peers 2 1 0
IBM.ConfigRM 2 1 0
IBM.GFVTRM 2 1 0
IBM.StorageRM.v1 2 1 0
IBM.GFVTRMCache 2 1 0

Critical clients will be terminated if unresponsive
```

Example 2

In this example, the Group Services subsystem is **cthags**. This **hagsns -s cthags** output indicates that the domain is established by node 1 (this node is node 1):

```
> ./hagsns -s cthags
HA GS NameServer Status
NodeId=1.2, pid=6619250, domainId=1.2,
NS established, CodeLevel=GSLevel(DRL=23)
NS state=kBecomeNS, protocolInProgress=kNoProtocol,
outstandingBroadcast=kNoBcast
Process started on Fri Aug 12 10:28:11 2011,
(1d 0:15:35.924685) ago.
HB connection took (0:0:3.440651).
Initial NS certainty on Fri Aug 12 10:28:15 2011,
(1d 0:15:31.735882) ago, taking (0:0:0.748152).
Our current epoch of certainty
is started on Fri Aug 12 10:28:15 2011,
(1d 0:15:31.735882) ago
Number of UP nodes: 2
List of UP nodes: 1-2
List of known groups:
1.1 rmc_peers: GL: 2 seqNum: 0 theIPS: 2 1 lookupQ:
2.1 IBM.ConfigRM: GL: 1 seqNum: 0 theIPS: 1-2 lookupQ:
3.1 IBM.GFVTRM: GL: 1 seqNum: 0 theIPS: 1-2 lookupQ:
4.1 IBM.StorageRM.v1: GL: 1 seqNum: 0 theIPS: 1-2 lookupQ:
5.1 IBM.GFVTRMCache: GL: 1 seqNum: 0 theIPS: 1-2 lookupQ:
```

If the nameserver domain is not established yet, it does not accept a client connection.

Example:

In this example, the Group Services subsystem is **cthags**. This **lssrc -ls cthags** output indicates that the nameserver domain is not established yet, so it cannot accept a client connection:

```
> lssrc -ls cthags
Subsystem Group PID Status
cthags cthags 6488094 active
No locally-connected clients.
HA Group Services domain information:
Domain not established.
Number of groups known locally: 0

Critical clients will be terminated if unresponsive
```

If the domain is established and the client still cannot connect with the Group Services, you can enable a Group Service library trace to investigate this problem further. To enable a Group Service library log, restart the client with these variable set such that the clients that use the Group Services pick them up as is:

```
HA_GSDBG_USE_TRACE=1
CT_TR_FILENAME=/tmp/trace.gsapi
CT_TR_TRACE_LEVELS="*.*=255"
CT_TR_SIZE=2000000              # in bytes
```

How and where to set the variables depend on the clients that are involved. For example, if the client runs a subsystem controlled by system resource controller (SRC), the **-e** flag can be used in the **startsrc** command to introduce additional environment variables. The variables can also be set in a startup script or configuration file if the client uses them. The file name that is specified can be of the client's choice, but it must be set as a full path name, otherwise the resulting location of the file can be unpredictable depending on how the client runs the subsystem. By default, the size of the file is in bytes. The size of the created file never changes; it is written internally in a looping fashion as tracing occurs.

5. **A cluster or site partition problem**

   Symptom

   On each node in the cluster, the **lssrc -ls cthags** command shows only that node as up, and its local node is the nameserver node. That is, each node acts as if it is the only node in the domain though in actuality, all of the nodes are in the domain.

   Problem determination

   a. This can be a CAA problem. Each node received an Autonomic Health Advisor File System (AHAFS) event that indicates that only the local node is up. Check the CAA event log file (**/var/ct/***cluster_name***/log/cthags/trace.ahafs_events**) to see which AHAFS events the Group Services daemon (**hagsd**) received and if they match its current state. Use **/usr/sbin/rsct/bin/rpttr** to convert this log file to a text file.

      You can also run the **/usr/sbin/rsct/bin/caa_config -n** *cluster_name* command to see whether the node state shown is consistent with **lscluster -m** output.

      You can also look at the UUID for the node ID and cluster ID in ODM. Check the **lscluster -m** output to see if there is a mismatch or if the UUIDs are different on all of the nodes.

   b. If the AHAFS events, CAA configuration, and command output are correct, it is possible that the cluster or site was partitioned due to a Group Services problem.

      Run **hagsns -s cthags** on all of the nodes to see whether they all have the same list of "up" nodes. If not, collect the data from the Group Services command output and the **hagsd** log files in a **ctsnap** file, as you need to look at this data to determine the problem.

6. **Problem in acquiring resources after site partition occurs or when taking necessary actions during site merge**

   Symptom

   Resources are not acquired according to the site policy set in the CAA environment or according to the resource attributes set in RSCT.

   Problem determination

   a. Verify whether the site merge policy is set in the CAA environment by entering the following command:

      ```
      /usr/sbin/clctrl -tune -o site_merge_policy
      ```

      where, *site_merge_policy* can be **p**, **m**, or **h** (priority, manual, or heuristic).

      Enter the following command to list the site priority if the site policy is set as `priority`:

      ```
      /usr/lib/cluster/clras dumprepos
      ```

   b. Determine whether the problem is related to Group Services or the ConfigRM resource class by entering the following command:

      ```
      export CT_MANAGEMENT_SCOPE=2
      lsrsrc -c IBM.PeerNode
      ```

      The following output is displayed:

      ```
      resource 1:
              CommittedRSCTVersion       = ""
              ActiveVersionChanging      = 0
              OpQuorumOverride           = 0
              CritRsrcProtMethod         = 1
      ```

```
OpQuorumTieBreaker         = "Operator"
QuorumType                 = 4
QuorumGroupName            = ""
Fanout                     = 32
OpFenceGroup               = ""
NodeCleanupCommand         = ""
NodeCleanupCriteria        = ""
QuorumLessStartupTimeout   = 120
CriticalMode               = 1
NotifyQuorumChangedCommand = ""
```

If the site merge policy is set to heuristic, determine whether the **QuorumType** attribute value in the output is expected for the cluster manager that is used on the node. For example, if the **QuorumType** value is 4, the PowerHA SystemMirror or shared storage pool (SSP) environments are used, and if it is 0, the Tivoli System Automation for Multiplatforms (TSAMP) environment is used.

The current active tiebreaker is specified in the **OpQuorumTieBreaker** attribute. The operational quorum is specified in the **OpQuorumState** attribute.

The **CriticalMode** attribute is used to specify whether a critical resource is considered on the node or not.

Table 43. CriticalMode setting

Attribute value	Description
0	Ignores the critical resource even if a node has critical resource.
1	Considers that the node has critical resources.
2	Considers that all nodes have critical resources.

The **CritRsrcProtMethod** attribute is used to take an appropriate action if the quorum, site, or node is lost.

7. **How to examine AHAFS events**

   There is a trace file called **trace.ahafs_events** in the **/var/ct/***cluster_name***/log/cthags/** directory. This AHAFS event log file records the AHAFS events that **hagsd** received from AHAFS. Use **/usr/sbin/rsct/bin/rpttr** to convert this log file to a text file.

   Example

   ```
   /usr/sbin/rsct/bin/rpttr -odtic \
   /var/ct/cluster1/log/cthags/trace.ahafs_events>
   /var/ct/cluster1/log/cthags/trace.ahafs_events.txt
   ```

   The contents of a **trace.ahafs_events.txt** file look like the following sample:

   ```
   Filename: trace.ahafs_events
   Program Name: /usr/sbin/rsct/bin/hagsd
   Properties: Big Endian, 32-bit mode
   Platform: AIX/PowerPC
   FileVersion: 5
   Node Number: 1
   Process Id: 6619250
   Machine Id: 0x0a0d98d6
   Node ID: 0xc63a107d766ddc34
   Trace Library Build Level: rjop1135a-8_9_11_2_1_hol_1
   08/12/11 10:28:12.571194 T( 1) ____ ****************** Trace Started - Pid =
   6619250 **********************
   08/12/11 10:28:34.569978 T( 1) _CAA AHAFS event fd(9) path(/aha/cluster/networ
   kAdapterState.monFactory/networkAdapterStateEvent.mon) content:
   BEGIN_EVENT_INFO
   TIME_tvsec=1281626914
   TIME_tvnsec=561738745
   SEQUENCE_NUM=0
   RC_FROM_EVPROD=0
   BEGIN_EVPROD_INFO
   ```

```
EVENT_TYPE=ADAPTER_DOWN
INTERFACE_NAME=en1
NODE_NUMBER=1
NODE_ID=0x61CBD2EC959711DFA7F1C291E3FACDEB
CLUSTER_ID=0xAA9CF1B0A61B11DFBBAEC291E3FAC
DEB
END_EVPROD_INFO
END_EVENT_INFO
08/12/11 10:28:40.588879 T( 1) _CAA AHAFS event fd(9)
path(/aha/cluster/networkAdapterState.monFactory/
networkAdapterStateEvent.mon) content:
BEGIN_EVENT_INFO
TIME_tvsec=1281626920
TIME_tvnsec=577971410
SEQUENCE_NUM=1
RC_FROM_EVPROD=0
BEGIN_EVPROD_INFO
EVENT_TYPE=ADAPTER_UP
INTERFACE_NAME=en1
NODE_NUMBER=1
NODE_ID=0x61CBD2EC959711DFA7F1C291E3FACDEB
CLUSTER_ID=0xAA9CF1B0A61B11DFBBAEC291E3FAC
DEB
END_EVPROD_INFO
END_EVENT_INFO
08/12/11 10:28:45.626605 T( 1) _CAA AHAFS event fd(9)
path(/aha/cluster/networkAdapterState.monFactory/
networkAdapterStateEvent.mon) content:
BEGIN_EVENT_INFO
TIME_tvsec=1281626925
TIME_tvnsec=617015130
SEQUENCE_NUM=2
RC_FROM_EVPROD=0
BEGIN_EVPROD_INFO
EVENT_TYPE=ADAPTER_UP
INTERFACE_NAME=en2
NODE_NUMBER=1
NODE_ID=0x61CBD2EC959711DFA7F1C291E3FACDEB
CLUSTER_ID=0xAA9CF1B0A61B11DFBBAEC291E3FAC
DEB
END_EVPROD_INFO
END_EVENT_INFO
08/12/11 10:28:54.644746 T( 1) _CAA AHAFS event fd(9)
path(/aha/cluster/networkAdapterState.monFactory/
networkAdapterStateEvent.mon) content:
.
.
.
```

8. **Migrating to a CAA environment: problem determination**

   a. **Migrating a PowerHA SystemMirror cluster to a CAA environment**

   First, ensure that you have the correct software versions. The minimum versions you need to use CAA are AIX 6.1 with the 6100-06 Technology Level (or later) or AIX 7.1, PowerHA SystemMirror 7.1, and RSCT 3.1.0.0 (or later).

   1) **PowerHA SystemMirror does not receive a domain notification**

   Symptom

   PowerHA SystemMirror does not receive a migration to CAA domain notification from Group Services.

   Problem determination

   Check the versions of AIX, PowerHA SystemMirror, and RSCT.

   To check the AIX version, run this command:

   ```
   cat /proc/version
   ```

Example

This output indicates that you are running AIX 6.1 with the 6100-06 Technology Level:

```
> cat /proc/version
Jul 31 2011
13:37:58
1031A_61L
@(#) _kdb_buildinfo unix_64 Jul 31 2011 13:37:58 1031A_61L
```

To check the RSCT version, run this command:

```
/usr/sbin/rsct/install/bin/ctversion
```

Example

This output indicates that you are running RSCT version 3.1.4.0:

```
> /usr/sbin/rsct/install/bin/ctversion
rkod1231a 3.1.4.0
```

hagsd sends a domain notification for migration to CAA mode to a client only if the client enabled the **HA_GS_ENABLE_MIGRATION_CALLBACK** option when connecting with Group Services. A client code must have this option set in the **ha_gs_init()** subroutine. This option is available in RSCT 3.1.0.0, and later.

2) **PowerHA SystemMirror receives a rejection notification**

Symptom

In the PowerHA SystemMirror log, there is a rejection notification for migration to a CAA environment.

Problem determination

Check the return codes from the **hagsd** domain notification for information about why a failure occurred.

Check to see whether the node configuration is the same before migration and after migration. The node number must be same before and after migration.

b. **Migrating an RSCT peer domain to a CAA environment**

In an RSCT peer domain, the configuration resource manager drives the migration to a CAA environment. Migration can fail because of a cluster creation error or because of rejection by Group Services.

1) **Migration to a CAA environment failed with a cluster creation error**

Symptom

During the migration process from an RSCT peer domain, a CAA migration command returns this error:

```
2632-328 command (mkcluster) for domain (z1) failed exit_code(1)

stderr:cluster_repository_init: create_clv failed
cl_mkcluster: cluster_repository_init(hdisk2) failed.
cl_create_cluster: cl_mkcluster: No such process
cl_create_cluster(): No such process

stdout:
e95n1sq10.ppd.pok.ibm.com: 2632-324 Migration to CAA failed
at function (CAA_MIGR_PHASE_2_CAA_CREATE) with result=-1.
Resource Class Action Response for MigrateToCAA
```

Note: The command output might change.

Problem determination

Check the AIX version, which must be AIX 6.1 with the 6100-06 Technology Level (or later) or AIX 7.1.

To check the AIX version, run this command:

```
cat /proc/version
```

Example

This output indicates that you are running AIX 6.1 with the 6100-06 Technology Level:

```
> cat /proc/version
Jul 31 2011
13:37:58
1031A_61L
@(#) _kdb_buildinfo unix_64 Jul 31 2011 13:37:58 1031A_61L
>
```

Check the RSCT version, which must be RSCT 3.1.0.0 (or later).

To check the RSCT version, run this command:

```
/usr/sbin/rsct/install/bin/ctversion -bv
```

Example

To check the RSCT version, run this command:

```
/usr/sbin/rsct/install/bin/ctversion
```

Example

This output indicates that you are running RSCT version 3.1.4.0:

```
> /usr/sbin/rsct/install/bin/ctversion
rkod1231a 3.1.4.0
```

If the AIX and RSCT versions are correct, there might be a CAA problem. Collect RSCT data (by using **ctsnap**) and CAA data for further debugging.

2) Migration to a CAA environment is rejected by Group Services

Symptom

There is a "migration to CAA rejected" entry in the configuration resource manager log file.

Problem determination

Check the error codes returned for the rejection notification, which indicate what the problem is. It can be a creation of CAA socket error, a Topology Services library error in creating AHAFS event handlers, or some other error.

Example

The configuration resource manager failed to migrate to a CAA environment. Its log file might look similar to the following sample:

```
[00] 07/06/11 16:24:25.639917 T(1808) _CFD Debug: Received
HA_GS_MIGRATE_TO_CAA notification, type=3
[01] 07/06/11 16:24:25.639919 T(1808) _CFD PeerDomainRccp::setCAAMigrPrepNotify
entry, notify_type=3, reason=3, detail_reason=
74
[00] 07/06/11 16:24:25.639920 T(1808) _CFD Debug: Detail reason_msg:
hb_migrate_to_caa_prep() returned 3, hb_errno 74
[01] 07/06/11 16:24:25.639924 T(1808) _CFD PeerDomainRccp::setCAAMigrPrepNotify
exit

[00] 07/06/11 16:24:25.639927 T(1808) _GSA DomainControl_callback() Leaving
[00] 07/06/11 16:24:25.640227 T(3602) _CFD Debug: PeerDomainRccp::waitForCAA
MigrPrepApproval returned result=0
[02] 07/06/11 16:24:25.640245 T(3602) _CFD id=0xffffffffError 36 was returned
from "nPhaseCb_MigrateToCAA" on line 2011 in file
"../../../../../src/rsct/rm/ConfigRM/ConfigRMProvider.C(1.60)".
Message=2632-324 Migration to CAA
failed at function (waitForMigrApproval(reason
=0, detail_reason=0, msg=)) with result=0.

[02] 07/06/11 16:24:25.640271 T(3602) _CFD id=0xffffffffError 36 was returned
from "nPhaseCb_MigrateToCAA" on line 2055 in file
"../../../../../src/rsct/rm/ConfigRM/ConfigRMProvider.C(1.60)".
```

```
Message=2632-324 Migration to CAA
failed at function (waitForMigrApproval
(reason=0, detail_reason=0, msg=)) with result=0.
```

This example indicates that the migration was rejected. Reason code 3 indicates that it failed at the Topology Services library layer. The location of the failure was Topology Services library function **hb_migrate_to_caa_prep()**. The error code is 74, which indicates that the creation of the global event handler failed. You need to collect RSCT data (by using **ctsnap**) and CAA data for further investigation.

9. **Investigating an AIX node crash**

   Similar to how Topology Services behave in a non-CAA environment, Group Services can trigger the deadman switch timer if the daemon is suspended in a CAA environment.

   When the node restarts, enter the following command:

   ```
   errpt -J KERNEL_PANIC
   ```

   If this command displays output that indicates an event with that label is found, enter the following command to get the details for the event:

   ```
   errpt -a
   ```

   The output of the command might be similar to the following sample output:

   ```
   ---------------------------------------------------------------------------
   LABEL:          KERNEL_PANIC
   IDENTIFIER:     225E3B63

   Date/Time:       Tue Jun 12 11:31:01 EDT 2012
   Sequence Number: 3010
   Machine Id:      00C17E554C00
   Node Id:         e105n1ec08
   Class:           S
   Type:            TEMP
   WPAR:            Global
   Resource Name:   PANIC

   Description
   SOFTWARE PROGRAM ABNORMALLY ENDED

   Recommended Actions
   PERFORM PROBLEM DETERMINATION PROCEDURES

   Detail Data
   ASSERT STRING

   PANIC STRING
   RSCT Dead Man Switch Timeout for CLUSTER; halting nonresponsive node
   ```

   If the `RSCT Dead Man Switch Timeout for CLUSTER` string is displayed in the output, the crash was caused by the deadman switch timer trigger. Otherwise, something else is causing the problem. For problems that are not related to the deadman switch timer, contact the IBM Support Center.

   If the dump file was created by the deadman switch timer, the problem might be caused by the blockage of the Group Services daemon. When the timer is triggered, other nodes are already in the process of taking over the resources of this node. If the node continues to operate, both this node and the node that takes over its disk are concurrently accessing the disk, which might cause data corruption.

Related tasks:

"Operational test 3: determine why the Group Services domain is not established" on page 210
Use this procedure to determine why the Group Services domain is not established or why it is not recovered.

Related information:

Operating System and Device Management

"Diagnosing problems with Group Services" on page 192
This topic addresses diagnostic procedures and failure responses for the Group Services subsystem of RSCT.

Notices

This information was developed for products and services offered in the U.S.A.

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 give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

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

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: 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 states 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 supply 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 Corporation
Dept. LRAS/Bldg. 903
11501 Burnet Road
Austin, TX 78758-3400
U.S.A.

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.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

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.

All 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 the names and addresses used by an actual business enterprise 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_.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

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.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Windows is a trademark of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Index

Special characters

A

B

C

D

E

F

G

H

I

L

M

N

O

P

Q

R

S

T

U

V

W

Printed in USA