

This section describes procedures used to troubleshoot Cisco Fabric Services (CFS) problems in the Cisco MDS 9000 Family multilayer directors and fabric switches.

Contents

- [1 Overview](#)
- [2 Initial Troubleshooting Checklist](#)
 - ◆ [2.1 Verifying CFS Using Fabric Manager](#)
 - ◆ [2.2 Verifying CFS Using the CLI](#)
- [3 Merge Failure Troubleshooting](#)
 - ◆ [3.1 Recovering from a Merge Failure with Fabric Manager](#)
 - ◆ [3.2 Recovering from a Merge Failure with the CLI](#)
- [4 Lock Failure Troubleshooting](#)
 - ◆ [4.1 Resolving Lock Failure Issues Using Fabric Manager](#)
 - ◆ [4.2 Resolving Lock Failure Issues Using the CLI](#)
 - ◆ [4.3 System State Inconsistent and Locks Being Held](#)
 - ◇ [4.3.1 Clearing Locks Using Fabric Manager](#)
 - ◇ [4.3.2 Clearing Locks Using the CLI](#)
- [5 Distribution Status Verification](#)
 - ◆ [5.1 Verifying Distribution Using Fabric Manager](#)
 - ◆ [5.2 Verifying Distribution Using the CLI](#)
- [6 CFS Regions Troubleshooting](#)
 - ◆ [6.1 Distribution Failure](#)
 - ◆ [6.2 Regions for Conditional Service](#)
 - ◆ [6.3 Changing Regions](#)

Overview

Many features in the Cisco MDS 9000 Family switches require configuration synchronization in all switches in the fabric. It is important to maintain configuration synchronization across a fabric for consistency.

As of Cisco MDS SAN-OS Release 2.0(1b), Cisco Fabric Services (CFS) provides a common infrastructure for automatic configuration synchronization in the fabric. The feature provides the transport function as well as extensive set of common services to the applications. CFS can discover CFS-capable switches in the fabric as well as their application capabilities.

Some of the applications that can be synchronized using CFS include:

- IVR
- NTP
- DPVM
- User roles
- AAA server addresses, Radius and TACACS daemons
- SFM
- SDV
- syslog

- port security
- Call Home

As of Cisco MDS SAN-OS Release 3.2(1), the scope of configuration synchronization can be restricted to a limited set of switches within the physical scope of an application. CFS regions are designed to:

- Fine-tune the distribution of configuration for an application.
- Restrict synchronization or merging of configuration information from a switch to a region, rather than distributing information across the entire physical scope of the application.
- Span across some or all of the switches in the topology within the physical scope of the application.

All switches in the fabric must be CFS capable. A Cisco MDS 9000 Family switch is CFS capable if it is running Cisco SAN-OS Release 2.0(1b) or later. Switches that are not CFS capable do not receive distributions, which results in part of the fabric not receiving the intended distribution.

CFS has the following features:

- Implicit CFS usage ? The first time you issue a CFS task for a CFS-enabled application, the configuration modification process begins and the application locks the fabric.
- Pending database ? The pending database is a temporary buffer to hold uncommitted information. The uncommitted changes are not applied immediately to ensure that the database is synchronized with the database in the other switches in the fabric. When you commit the changes, the pending database overwrites the configuration database (also know as active database or the effective database).
- CFS distribution enabled or disabled on a per-application basis ? The default (enable or disable) for CFS distribution state differs among applications. If CFS distribution is disabled for an application, that application does not distribute any configurations nor does it accept a distribution from other switches in the fabric.
- Explicit CFS commit ? Most applications require an explicit commit operation to copy the changes in the temporary buffer to the application database, and distribute the new database to the fabric, and release the fabric lock. The changes in the temporary buffer are not applied if you do not perform the commit operation.
- Globally disable CFS distribution ? Use the **no cfs enable** command, in config mode, to isolate the switch from the rest of the fabric. The switch operates like a single switch fabric. All other functions by the CFS and CFS enabled applications are unaffected.
- Enable IPV4 and IPV6 distribution from Fabric Manager ? Select **Physical Attributes> Switches > CFS**. GLOBAL indicates CFS distribution and IP MULTICAST indicates IPv4 and IPv6 distributions.

As of Cisco SAN-OS Release 3.1(2), some applications, such as Inter-VSAN Routing (IVR), require configuration distribution over some specific VSANs. These applications specify to CFS the set of VSANs over which to restrict the distribution.

Initial Troubleshooting Checklist

Begin troubleshooting CFS by checking for the following issues:

Checklist	Checkoff
Verify that CFS is enabled for the same applications on all affected switches.	

Verify that CFS distribution is enabled for the same applications on all affected switches.	
If the CFS Regions feature is in use, verify that the application is in the same region on all the affected switches.	
Verify that there are no pending changes for an application and that a CFS commit was issued for any configuration changes in a CFS-enabled application.	
Verify that there are no unexpected CFS locked sessions. Clear any unexpected locked sessions.	

Verifying CFS Using Fabric Manager

To verify CFS using Fabric Manager or Device Manager, follow these steps:

1. Choose **Admin > CFS** on Device Manager to verify that an application is listed and enabled. Repeat this on all switches.
2. To list the set of switches in which an application is registered with CFS, choose the application configuration menu on Fabric Manager and select the CFS tab. For example, to verify that DPVM is enabled and global distribution is enabled on all switches, choose **Fabricxx > All VSANs > DPVM** and select the **CFS** tab. Verify that the Oper field is enabled and the Global field is enabled for all switches in the fabric.
3. To determine if all the switches in the fabric constitute one CFS fabric, or a multitude of partitioned CFS fabrics using Device Manager, follow these steps:
 - a. Choose **Admin > CFS** and highlight the application that you want to verify CFS on.
 - b. Click **Details** and select the **Merge** tab in the Details dialog box.
 - c. If you see multiple rows in the Merge status table, then the fabric is partitioned into multiple CFS fabrics. Some features enable CFS per VSAN and this is expected. If the selected feature should be fabric wide but you see multiple rows in the Merge status table, then the fabric may be partitioned, and the merge status may show that the merge has failed, is pending, or is waiting.

Verifying CFS Using the CLI

To verify CFS using the CLI, follow these steps:

1. To verify that an application is listed and enabled, issue the **show cfs application** command to all switches. An example of the **show cfs application** command follows:

```
Switch# show cfs application
-----
Application      Enabled      Scope
-----
ivr              Yes         Physical
ntp              No          Physical
dpvm             Yes         Physical
```


```
fscm           Yes      Physical
role           Yes      Physical
radius         Yes      Physical
fctimer        No       Physical
syslogd        No       Physical
callhome       No       Physical
device-alias   Yes      Physical
port-security  Yes      Logical
Total number of entries = 11
```

The Physical scope means that CFS applies the configuration for that application to the entire switch. The Logical scope means that CFS applies the configuration for that application to a specific VSAN.

2. Verify the set of switches in which an application is registered with CFS, using the **show cfs peers name application-name** for physical scope applications, and the **show cfs peers name application-name vsan vsan-id** for logical scope applications.

An example command output for a physical scope application follows:

```
Switch# show cfs peers name dpvm
Scope      : Physical
-----
Switch WWN          IP Address
-----
20:00:00:0e:d7:0e:bf:c0 10.76.100.51 [Local]
20:00:00:0e:d7:00:3c:9e 10.76.100.52
Total number of entries = 2
```

 **Note:** The **show cfs peers name application-name** command displays the peers for all VSANs when applied to a logical application.

An example command output for a logical scope application follows:

```
Switch# show cfs peers name port-security
Scope      :Logical [VSAN 1]
-----
Domain     Switch WWN          IP Address
-----
236        20:00:00:0e:d7:00:3c:9e 10.76.100.52 [Local]
239        20:00:00:05:30:00:6b:9e 10.76.100.167
101        20:00:00:0d:ec:06:55:c0 10.76.100.205
Total number of entries = 3
Scope      :Logical [VSAN 2]
-----
Domain     Switch WWN          IP Address
-----
239        20:00:00:0e:d7:00:3c:9e 10.76.100.52 [Local]
211        20:00:00:05:30:00:6b:9e 10.76.100.167
110        20:00:00:0d:ec:06:55:c0 10.76.100.205
Total number of entries = 3
Scope      :Logical [VSAN 3]
-----
Domain     Switch WWN          IP Address
-----
103        20:00:00:0e:d7:00:3c:9e 10.76.100.52 [Local]
221        20:00:00:05:30:00:6b:9e 10.76.100.167
11         20:00:00:0d:ec:06:55:c0 10.76.100.205
Total number of entries = 3
```

3. To determine if all the switches in the fabric compose one CFS fabric, or a multitude of partitioned CFS fabrics, issue the **show cfs merge status name application-name** command and the **show cfs peers name application-name** command and compare the outputs. If the outputs contain the same list of switches, the entire set of switches composes one CFS fabric. When this is the case, the merge status should always show success at all switches. Example command outputs follow:


```
Switch# show cfs merge status name dpvm
Physical Merge Status: Success [ Sat Nov 20 11:59:36 2004 ]
Local Fabric
-----
Switch WWN                IP Address
-----
20:00:00:05:30:00:4a:de  10.76.100.51    [Merge Master]
20:00:00:0d:ec:0c:f1:40  10.76.100.204
Switch# show cfs peers name dpvm
Scope      : Physical
-----
Switch WWN                IP Address
-----
20:00:00:0d:ec:0c:f1:40  10.76.100.204  [Local]
20:00:00:05:30:00:4a:de  10.76.100.51
Total number of entries = 2
```

If the list of switches in the **show cfs merge status name** command output is shorter than the **show cfs peers name** command output, the fabric is partitioned into multiple CFS fabrics and the merge status may show that the merge has failed, is pending, or is waiting.

Merge Failure Troubleshooting

During a merge, the merge managers in the merging fabrics exchange their configuration databases with each other. The application on one of them merges the information, decides if the merge is successful, and informs all switches in the combined fabric of the status of the merge. When a merge is successful, the merged database is distributed to all switches in the combined fabric and the entire new fabric remains in a consistent state. A merge failure indicates that the merged fabrics contain inconsistent data that could not be merged.

If a new switch is added to the fabric and the merge status for any application shows In Progress for a prolonged period of time, there might be an active session for that application in some switch. Check the lock status for that application on all the switches using the **show cfs lock** CLI command. If there are any locks, the merge will not proceed. Commit the changes or clear the session lock so that the merge can proceed.

 **Note:** Ensure that merge failures are analyzed correctly. Exercise caution when choosing a switch for blank commit because small configurations can delete the large configurations.

Recovering from a Merge Failure with Fabric Manager

To recover from a merge failure using Fabric Manager, follow these steps:

1. Click the **CFS** tab for the application that you are configuring and check the merge field to identify a switch that shows a merge failure. For example, choose **Fabricxx > All VSANS > DPVM** and select the **CFS** tab to determine if there is a merge failure for DPVM.
 2. In the Config Action drop-down menu choose **commit** and click **Apply Changes** to restore all peers in the fabric to the same configuration database.
-

Recovering from a Merge Failure with the CLI

To recover from a merge failure using the CLI, follow these steps:

1. To identify a switch that shows a merge failure, enter the **show cfs merge status name *application-name*** command. Example command output follows:

```
Switch# show cfs merge status name ntp
Physical Merge Status:Failure [ Mon Nov 22 06:49:52 2004 ]
Failure Reason: Conflicting entries in the compared databases
Local Fabric
-----
Switch WWN                IP Address
-----
20:00:00:05:30:00:6b:9e   10.76.100.167   [Merge Master]
20:00:00:0e:d7:00:3c:9e   10.76.100.52
Remote Fabric
-----
Switch WWN                IP Address
-----
20:00:00:0d:ec:06:55:c0   10.76.100.205   [Merge Master]
```

2. For a more detailed description of the merge failure, enter the **show cfs internal session-history name *application name* detail** command. Example command output follows:

```
switch# show cfs internal session-history name ntp detail
-----
Time Stamp                Source WWN                Event
User Name                 Session ID
-----
Fri Aug 24 04:30:19 2007 20:00:00:0d:ec:04:99:c0 LOCK_REQUEST
admin                    3848
Fri Aug 24 04:30:19 2007 20:00:00:0d:ec:04:99:c0 LOCK_ACQUIRED
admin                    3848
Fri Aug 24 04:30:19 2007 20:00:00:0d:ec:04:99:c0 COMMIT
admin                    3849
Fri Aug 24 04:30:19 2007 20:00:00:0d:ec:04:99:c0 LOCK_RELEASE_REQUEST
admin                    3848
Fri Aug 24 04:30:19 2007 20:00:00:0d:ec:04:99:c0 LOCK_RELEASED
admin                    3848
Fri Aug 24 04:33:07 2007 20:00:00:0d:ec:04:99:c0 LOCK_REQUEST
admin                    3868
Fri Aug 24 04:33:07 2007 20:00:00:0d:ec:04:99:c0 LOCK_ACQUIRED
admin                    3868
-----
```

3. Enter configuration mode and enter the ***application-name* commit** command to restore all peers in the fabric to the same configuration database. Example command output follows:

```
Switch# config terminal
Switch(config)# ntp commit
Switch(config)#
```

Lock Failure Troubleshooting

In order to distribute a configuration in the fabric, a lock must first be acquired on all switches in the fabric. Once this action is accomplished, a commit can be issued that distributes the data to all switches in the fabric before releasing the lock.

When a lock has been acquired by another application peer, you cannot commit new configuration changes. This is normal operation and you should postpone any changes to an application until the lock is released. Use the troubleshooting steps in this section only if you believe the lock has not been properly released.

A lock occurs when an administrator configures a change for a CFS-enabled application. If two administrators on the same switch attempt to configure the same application, only one administrator is given the lock. The other administrator is prevented from making changes to that application until the first administrator commits a change or discards any changes. Use the **show cfs lock name** CLI command to determine the name of the administrator who holds the lock for an application. You should check with that administrator before clearing the lock.

A CFS lock can also be held by another switch in your fabric. Use the **show cfs peers name** CLI command to determine all of the switches that participate in the CFS distribution for this application. Use the **show cfs lock name** CLI command on each switch to determine who owns the CFS lock for that application. Check with the administrator before clearing the lock.

Use the CFS **abort** option to release the lock without distributing the data to the fabric.

Resolving Lock Failure Issues Using Fabric Manager

To resolve a lock failure using Fabric Manager, follow these steps:

1. Click the **CFS** tab for the application that you are configuring and check the **Master** check box to identify the master switch for that CFS application. For example, choose **Fabricxx > All VSANS > DPVM** and check the **CFS** tab.
2. In the **Config Action** drop-down menu on the master switch choose **commit** or **abort** and click **Apply Changes** to restore all peers in the fabric to the same configuration database and free the CFS lock.

Resolving Lock Failure Issues Using the CLI

To resolve a lock failure using the CLI, follow these steps:

1. Enter the **show cfs lock name** command to determine the lock holder. An example of the **show cfs lock name** command follows:

```
Switch# show cfs lock ntp
Application:ntp
Scope      :Physical
-----
Switch WWN          IP Address      User Name      User Type
-----
20:00:00:05:30:00:6b:9e  10.76.100.167  admin         CLI/SNMP v3
Total number of entries = 1
```

2. For a detailed description of the lock failure, enter the **show cfs internal session-history name application name detail** command. Example command output follows:

```
switch# show cfs internal session-history name ntp detail
-----
Time Stamp          Source WWN          Event
User Name           Session ID
-----
Fri Aug 24 04:30:19 2007 20:00:00:0d:ec:04:99:c0  LOCK_REQUEST
admin              3848
```

```
Fri Aug 24 04:30:19 2007 20:00:00:0d:ec:04:99:c0 LOCK_ACQUIRED
admin 3848
Fri Aug 24 04:30:19 2007 20:00:00:0d:ec:04:99:c0 COMMIT
admin 3849
Fri Aug 24 04:30:19 2007 20:00:00:0d:ec:04:99:c0 LOCK_RELEASE_REQUEST
admin 3848
Fri Aug 24 04:30:19 2007 20:00:00:0d:ec:04:99:c0 LOCK_RELEASED
admin 3848
Fri Aug 24 04:33:07 2007 20:00:00:0d:ec:04:99:c0 LOCK_REQUEST
admin 3868
Fri Aug 24 04:33:07 2007 20:00:00:0d:ec:04:99:c0 LOCK_ACQUIRED
admin 3868
```

3. If the lock is being held by a remote peer, an *application-name* **commit** command or an *application-name* **abort** command must be executed at that switch. An example of the *application-name* **commit** command follows:

```
Switch# config terminal
Switch(config)# ntp commit
Switch(config)#
```

An example of the *application-name* **abort** command follows:

```
Switch# config terminal
Switch(config)# ntp abort
Switch(config)#
```

System State Inconsistent and Locks Being Held

An inconsistent system state occurs when locks are not held on all of the switches in the fabric, or when locks are held on all switches in the fabric, but a session does not exist with the lock-holding switch. In either case, it may be necessary to use the **clear** option to release the locks.

Clearing Locks Using Fabric Manager

To clear a lock using Fabric Manager, follow these steps:

1. Click the **CFS** tab for the application that you are configuring and view the Master check box to identify the master switch for that CFS application. For example, choose **Fabricxx > All VSANS > DPVM** and click the **CFS** tab.
2. In **Config Action** drop-down menu on the master switch choose **clear** and click **Apply Changes** to free the CFS lock.

Clearing Locks Using the CLI

When a lock is being held on a remote peer, entering the *application-name* **commit** command or the *application-name* **abort** command does not clear the lock. Enter the **clear application-name session** command to clear all locks in the fabric. After all locks are cleared, a new distribution must be started to restore all the switches in the fabric to the same state.

Example command output follows:

```
Switch# clear ntp session
Switch# config terminal
```



```
Switch(config)# ntp commit
Switch(config)#
```

Distribution Status Verification

After configuring an application and committing the changes, you may want to verify that CFS is distributing the configuration change throughout the fabric or VSAN.

Verifying Distribution Using Fabric Manager

In Fabric Manager, click the **CFS** tab for the application that you are configuring and choose the **Last Results** field to view the distribution status for your latest commit.

Verifying Distribution Using the CLI

In the CLI, use the **show cfs lock name *application-name*** command to determine if a distribution is in progress on the fabric. If the application does not show in the output, the distribution has completed.

Example command output follows:

```
Switch# show cfs lock name ntp
Scope      :Physical
-----
Switch WWN          IP Address      User Name      User Type
-----
20:00:00:05:30:00:6b:9e 10.76.100.167  admin         CLI/SNMP v3
Total number of entries = 1
```

CFS Regions Troubleshooting

The following rules apply to CFS Regions:

- When using CFS Regions, an application on a given switch can only belong to one region at a time.
- CFS Regions are only applicable to applications within physical scope. You cannot create a CFS Region in the logical scope of an application.
- Assigning a region to an application takes precedence in distribution over its initial physical scope.
- For backwards compatibility, all applications are in Region 0 by default. If you are running switches with previous SAN-OS releases on the same topology as switches running SAN OS 3.2x, only applications in Region 0 are supported when those switches are synchronized. Only Region 0 applications on the switches running SAN-OS 3.2x synchronize or merge with switches running SAN-OS 3.1x. All applications in other regions on the switch running SAN-OS 3.2x are ignored by the switch running SAN-OS 3.1x.
- CFS Regions configuration is not supported for deregistered applications (conditional services) or a physical scope application that is currently locked.
- Regions 1 through 200 are available for user configuration. Regions 201 through 255 are reserved regions and are not available for user configuration.

Distribution Failure

To resolve a configuration distribution failure to all switches for a CFS Region, follow these steps:

1. Verify that application distribution is enabled. See the "Initial Troubleshooting Checklist" section for detailed instructions.

2. Verify that the application is in the same region on all switches. From Fabric Manager, select the application tab (for example, **Physical Attributes > Switches > Clock > NTP**), and then click the **CFS** tab. Under the region-id column, check that the source switch and destination switch are in the same region, and click the **Regions** tab.

Using the CLI from each switch, issue the **show cfs application name *application-name*** command.

For example, for the device-alias application:


```
Switch# show cfs application name device-alias
Enabled          : Yes
Timeout          : 20s
Merge Capable    : Yes          <<<<<< Application is capable of being merged.
Scope            : Physical-fc
Region           : 1           <<<<<< Application is in Region 1
```

Regions for Conditional Service

When a conditional service goes down (for example, when a conditional service deregisters with CFS), it loses its region configuration. When the conditional service is restarted, it is automatically put into the default region. To avoid this situation, reconfigure the appropriate region information for the conditional service before starting it again.

Changing Regions

If you move an application from one region to another, you may encounter a database mismatch when attempting a merge. Follow the steps outlined in the "Merge Failure Troubleshooting" section to identify and resolve the conflicts.

 **Note:** When an application is moved from one region to another (including the default region), it loses all histories.

Back to Main Page: [Cisco MDS SAN-OS Troubleshooting Guide](#)
