Troubleshooting APIC Crash Scenarios

This chapter contains information about various failure or crash scenarios and possible recovery solutions.

This chapter contains the following sections:

Cisco APIC Cluster Failure Scenarios

Cluster Troubleshooting Scenarios

The following table summarizes common cluster troubleshooting scenarios for the Cisco APIC.

Problem	Solution
An APIC node fails within the cluster. For example, node 2 of a cluster of 5 APICs fails.	There are two available solutions: • Leave the target size and replace the APIC. • Reduce the cluster size to 4, decommission controller 5, and recommission it as APIC 2.The target size remains 4, and the operational size is 4 when the reconfigured APIC becomes active. **Note** You can add a replacement APIC to the cluster and expand the target and operational size. For instructions on how to add a new APIC, refer to the *Cisco APIC Management, Installation, Upgrade, and Downgrade Guide*..

Problem	Solution
A new APIC connects to the fabric and loses connection to a leaf switch.	Use the following commands to check for an infra (infrastructure) VLAN mismatch: • cat /mit/sys/lldp/inst/if-\[eth1--1\]/ctrlradj/summary—Displays the VLAN configured on the leaf switch. • cat /mit/sys/lldp/inst/if-\[eth1--1\]/ctrlradj/summary—Displays the infra (infrastructure) VLANs advertised by connected APICs. If the output of these commands shows different VLANs, the new APIC is not configured with the correct infra (infrastructure) VLAN. To correct this issue, follow these steps: • Log in to the APIC using rescue-user. **Note** Admin credentials do not work because the APIC is not part of the fabric. • Erase the configuration and reboot the APIC using the **acidiag touch setup** command. • Reconfigure the APIC. Verify that the fabric name, TEP addresses, and infra (infrastructure) VLAN match the APICs in the cluster. • Reload the leaf node.
Two APICs cannot communicate after a reboot.	The issue can occur after the following sequence of events: • APIC1 and APIC2 discover each other. • APIC1 reboots and becomes active with a new ChassisID (APIC1a) • The two APICs no longer communicate. In this scenario, APIC1a discovers APIC2, but APIC2 is unavailable because it is in a cluster with APIC1, which appears to be offline. As a result, APIC1a does not accept messages from APIC2. To resolve the issue, decommission APIC1 on APIC2, and commission APIC1 again.
A decommissioned APIC joins a cluster.	The issue can occur after the following sequence of events: • A member of the cluster becomes unavailable or the cluster splits. • An APIC is decommissioned. • After the cluster recovers, the decommissioned APIC is automatically commissioned. To resolve the issue, decommission the APIC after the cluster recovers.

Problem	Solution
Mismatched ChassisID following reboot.	The issue occurs when an APIC boots with a ChassisID different from the ChassisID registered in the cluster. As a result, messages from this APIC are discarded. To resolve the issue, ensure that you decommission the APIC before rebooting.
The APIC displays faults during changes to cluster size.	A variety of conditions can prevent a cluster from extending the OperationalClusterSize to meet the AdminstrativeClusterSize. For more information, inspect the fault and review the "Cluster Faults" section in the *Cisco APIC Basic Configuration Guide*.
An APIC is unable to join a cluster.	The issue occurs when two APICs are configured with the same ClusterID when a cluster expands. As a result, one of the two APICs cannot join the cluster and displays an expansion-contender-chassis-id-mismatch fault. To resolve the issue, configure the APIC outside the cluster with a new cluster ID.
APIC unreachable in cluster.	Check the following settings to diagnose the issue: • Verify that fabric discovery is complete. • Identify the switch that is missing from the fabric. • Check whether the switch has requested and received an IP address from an APIC. • Verify that the switch has loaded a software image. • Verify how long the switch has been active. • Verify that all processes are running on the switch. For more information, see the "acidiag Command" section in the *Cisco APIC Basic Configuration Guide*. • Confirm that the missing switch has the correct date and time. • Confirm that the switch can communicate with other APICs.

Problem	Solution
Cluster does not expand.	The issue occurs under the following circumstances: • The OperationalClusterSize is smaller than the number of APICs. • No expansion contender (for example, the admin size is 5 and there is not an APIC with a clusterID of 4. • There is no connectivity between the cluster and a new APIC • Heartbeat messages are rejected by the new APIC • System is not healthy. • An unavailable appliance is carrying a data subset that is related to relocation. • Service is down on an appliance with a data subset that is related to relocation. • Unhealthy data subset related to relocation.
An APIC is down.	Check the following: • Connectivity issue—Verify connectivity using ping. • Interface type mismatch—Confirm that all APICs are set to in-band communication. • Fabric connectivity—Confirm that fabric connectivity is normal and that fabric discovery is complete. • Heartbeat rejected—Check the fltInfraIICIMsgSrcOutsider fault. Common errors include operational cluster size, mismatched ChassisID, source ID outside of the operational cluster size, source not commissioned, and fabric domain mismatch.

Cluster Faults

The APIC supports a variety of faults to help diagnose cluster problems. The following sections describe the two major cluster fault types.

Discard Faults

The APIC discards cluster messages that are not from a current cluster peer or cluster expansion candidate. If the APIC discards a message, it raises a fault that contains the originating APIC's serial number, cluster ID, and a timestamp. The following table summarizes the faults for discarded messages:

Fault	Meaning
expansion-contender-chassis-id-mismatch	The ChassisID of the transmitting APIC does not match the ChassisID learned by the cluster for expansion.
expansion-contender-fabric-domain-mismatch	The FabricID of the transmitting APIC does not match the FabricID learned by the cluster for expansion.

Fault	Meaning
expansion-contender-id-is-not-next-to-oper-cluster-size	The transmitting APIC has an inappropriate cluster ID for expansion. The value should be one greater than the current OperationalClusterSize.
expansion-contender-message-is-not-heartbeat	The transmitting APIC does not transmit continuous heartbeat messages.
fabric-domain-mismatch	The FabricID of the transmitting APIC does not match the FabricID of the cluster.
operational-cluster-size-distance-cannot-be-bridged	The transmitting APIC has an OperationalClusterSize that is different from that of the receiving APIC by more than 1. The receiving APIC rejects the request.
source-chassis-id-mismatch	The ChassisID of the transmitting APIC does not match the ChassisID registered with the cluster.
source-cluster-id-illegal	The transmitting APIC has a clusterID value that is not permitted.
source-has-mismatched-target-chassis-id	The target ChassisID of the transmitting APIC does not match the Chassis ID of the receiving APIC.
source-id-is-outside-operational-cluster-size	The transmitting APIC has a cluster ID that is outside of the OperationalClusterSize for the cluster.
source-is-not-commissioned	The transmitting APIC has a cluster ID that is currently decommissioned in the cluster.

Cluster Change Faults

The following faults apply when there is an error during a change to the APIC cluster size.

Fault	Meaning
cluster-is-stuck-at-size-2	This fault is issued if the OperationalClusterSize remains at 2 for an extended period. To resolve the issue, restore the cluster target size.
most-right-appliance-remains-commissioned	The last APIC within a cluster is still in service, which prevents the cluster from shrinking.
no-expansion-contender	The cluster cannot detect an APIC with a higher cluster ID, preventing the cluster from expanding.
service-down-on-appliance-carrying-replica-related-to-relocation	The data subset to be relocated has a copy on a service that is experiencing a failure. Indicates that there are multiple such failures on the APIC.
unavailable-appliance-carrying-replica-related-to-relocation	The data subset to be relocated has a copy on an unavailable APIC. To resolve the fault, restore the unavailable APIC.
unhealthy-replica-related-to-relocation	The data subset to be relocated has a copy on an APIC that is not healthy. To resolve the fault, determine the root cause of the failure.

APIC Unavailable

The following cluster faults can apply when an APIC is unavailable:

Fault	Meaning
fltInfraReplicaReplicaState	The cluster is unable to bring up a data subset.
fltInfraReplicaDatabaseState	Indicates a corruption in the data store service.
fltInfraServiceHealth	Indicates that a data subset is not fully functional.
fltInfraWiNodeHealth	Indicates that an APIC is not fully functional.

Troubleshooting Application Centric Infrastructure Crash Scenarios

Troubleshooting Fabric Node and Process Crash

The ACI switch node has numerous processes which control various functional aspects on the system. If the system has a software failure in a particular process, a core file will be generated and the process will be reloaded.

If the process is a Data Management Engine (DME) process, the DME process will restart automatically. If the process is a non-DME process, it will not restart automatically and the switch will reboot to recover.

This section presents an overview of the various processes, how to detect that a process has cored, and what actions should be taken when this occurs

DME Processes

The essential processes running on an APIC can be found through the CLI. Unlike the APIC, the processes that can be seen via the GUI in **FABRIC** > **INVENTORY** > **Pod 1** > ***node*** shows all processes running on the leaf.

Through the **ps-ef | grep svc_ifc**:

```
rtp_leaf1# ps -ef |grep svc_ifc
root 3990 3087 1 Oct13 ? 00:43:36 /isan/bin/svc_ifc_policyelem --x
root 4039 3087 1 Oct13 ? 00:42:00 /isan/bin/svc_ifc_eventmgr --x
root 4261 3087 1 Oct13 ? 00:40:05 /isan/bin/svc_ifc_opflexelem --x -v
dptcp:8000
root 4271 3087 1 Oct13 ? 00:44:21 /isan/bin/svc_ifc_observerelem --x
root 4277 3087 1 Oct13 ? 00:40:42 /isan/bin/svc_ifc_dbgrelem --x
root 4279 3087 1 Oct13 ? 00:41:02 /isan/bin/svc_ifc_confelem --x
rtp_leaf1#
```

Each of the processes running on the switch writes activity to a log file on the system. These log files are bundled as part of the techsupport file but can be found via CLI access in /tmp/logs/ directory. For example, the Policy Element process log output is written into /tmp/logs/svc_ifc_policyelem.log.

The following is a brief description of the DME processes running on the system. This can help in understanding which log files to reference when troubleshooting a particular process or understand the impact to the system if a process crashed:

Process	Function
policyelem	Policy Element: Process logical MO from APIC and push concrete model to the switch
eventmgr	Event Manager: Processes local faults, events, health score
opflexelem	Opflex Element: Opflex server on switch
observerelem	Observer Element: Process local stats sent to APIC
dbgrelem	Debugger Element: Core handler
nginx	Web server handling traffic between the switch and APIC

Identify When a Process Crashes

When a process crashes and a core file is generated, a fault as well as an event is generated. The fault for the particular process is shown as a "process-crash" as shown in this syslog output from the APIC:

```
Oct 16 03:54:35 apic3 %LOG_LOCAL7-3-SYSTEM_MSG [E4208395][process-crash][major]
[subj-[dbgs/cores/node-102-card-1-svc-policyelem-ts-2014-10-16T03:54:55.000+00:00]/
rec-12884905092]Process policyelem cored
```

When the process on the switch crashes, the core file is compressed and copied to the APIC. The syslog message notification comes from the APIC.

The fault that is generated when the process crashes is cleared when the process is Troubleshooting Cisco Application Centric Infrastructure 275 restarted. The fault can be viewed via the GUI in the fabric history tab at **FABRIC** > **INVENTORY** > **Pod 1**.

Collecting the Core Files

The APIC GUI provides a central location to collect the core files for the fabric nodes.

An export policy can be created from **ADMIN** > **IMPORT/EXPORT** > **Export Policies** > **Core**. However, there is a default core policy where files can be downloaded directly.

The core files can be accessed via SSH/SCP through the APIC at /data/techsupport on the APIC where the core file is located. Note that the core file will be available at /data/ techsupport on one APIC in the cluster, the exact APIC that the core file resides can be found by the Export Location path as shown in the GUI. For example, if the Export Location begins with “files/3/”, the file is located on node 3 (APIC3).

APIC Process Crash Verification and Restart

Symptom 1

Process on switch fabric crashes. Either the process restarts automatically or the switch reloads to recover.

- **Verification**:

  As indicated in the overview section, if a DME process crashes, it should restart automatically without the switch restarting. If a non-DME process crashes, the process will not automatically restart and the switch will reboot to recover.

Depending on which process crashes, the impact of the process core will vary.

When a non-DME process crashes, this will typical lead to a HAP reset as seen on the console:

```
[ 1130.593388] nvram_klm wrote rr=16 rr_str=ntp hap reset to nvram
[ 1130.599990] obfl_klm writing reset reason 16, ntp hap reset
[ 1130.612558] Collected 8 ext4 filesystems
```

- **Check Process Log**:

  The process which crashes should have at some level of log output prior to the crash. The output of the logs on the switch are written into the /tmp/logs directory. The process name will be part of the file name. For example, for the Policy Element process, the file is svc_ifc_policyelem.log

```
rtp_leaf2# ls -l |grep policyelem
-rw-r--r-- 2 root root 13767569 Oct 16 00:37 svc_ifc_policyelem.log
-rw-r--r-- 1 root root 1413246 Oct 14 22:10 svc_ifc_policyelem.log.1.gz
-rw-r--r-- 1 root root 1276434 Oct 14 22:15 svc_ifc_policyelem.log.2.gz
-rw-r--r-- 1 root root 1588816 Oct 14 23:12 svc_ifc_policyelem.log.3.gz
-rw-r--r-- 1 root root 2124876 Oct 15 14:34 svc_ifc_policyelem.log.4.gz
-rw-r--r-- 1 root root 1354160 Oct 15 22:30 svc_ifc_policyelem.log.5.gz
-rw-r--r-- 2 root root 13767569 Oct 16 00:37 svc_ifc_policyelem.log.6
-rw-rw-rw- 1 root root 2 Oct 14 22:06 svc_ifc_policyelem.log.PRESERVED
-rw-rw-rw- 1 root root 209 Oct 14 22:06 svc_ifc_policyelem.log.stderr
rtp_leaf2#
```

  There will be several files for each process located at /tmp/logs. As the log file increases in size, it will be compressed and older log files will be rotated off. Check the core file creation time (as shown in the GUI and the core file name) to understand where to look in the file. Also, when the process first attempts to come up, there be an entry in the log file that indicates “Process is restarting after a crash” that can be used to search backwards as to what might have happened prior to the crash.

- **Check Activity**:

  A process which has been running has had some change which then caused it to crash. In many cases the changes may have been some configuration activity on the system. What activity occurred on the system can be found in the audit log history of the system.

- **Contact TAC**:

  A process crashing should not normally occur. In order to understand better why beyond the above steps it will be necessary to decode the core file. At this point, the file will need to be collected and provided to the TAC for further processing.

  Collect the core file (as indicated above how to do this) and open up a case with the TAC.

Symptom 2

Fabric switch continuously reloads or is stuck at the BIOS loader prompt.

- **Verification**:

  If a DME process crashes, it should restart automatically without the switch restarting. If a non-DME process crashes, the process will not automatically restart and the switch will reboot to recover. However in either case if the process continuously crashes, the switch may get into a continuous reload loop or end up in the BIOS loader prompt.

```
[ 1130.593388] nvram_klm wrote rr=16 rr_str=policyelem hap reset to nvram
[ 1130.599990] obfl_klm writing reset reason 16, policyelem hap reset
[ 1130.612558] Collected 8 ext4 filesystems
```

- **Break the HAP Reset Loop**:

  First step is to attempt to get the switch back into a state where further information can be collected.

  If the switch is continuously rebooting, when the switch is booting up, break into the BIOS loader prompt through the console by typing CTRL C when the switch is first part of the boot cycle.

  Once the switch is at the loader prompt, enter in the following commands:

  - cmdline no_hap_reset
  - boot

  The cmdline command will prevent the switch from reloading with a hap reset is called. The second command will boot the system. Note that the boot command is needed instead of a reload at the loader as a reload will remove the cmdline option entered.

  Though the system should now remain up to allow better access to collect data, whatever process is crashing will impact the functionality of the switch.

As in the previous table, check the process log, activity, and contact TAC steps.

Troubleshooting an APIC Process Crash

The APIC has a series of Data Management Engine (DME) processes which control various functional aspects on the system. When the system has a software failure in a particular process, a core file will be generated and the process will be reloaded.

The following sections cover potential issues involving system processes crashes or software failures, beginning with an overview of the various system processes, how to detect that a process has cored, and what actions should be taken when this occurs. The displays taken on a working healthy system can then be used to identify processes that may have terminated abruptly.

DME Processes

The essential processes running on an APIC can be found either through the GUI or the CLI. Using the GUI, the processes and the process ID running is found in **System** > **Controllers** > **Processes**.

Using the CLI, the processes and the process ID are found in the summary file at /aci/system/controllers/1/processes (for APIC1):

```
admin@RTP_Apic1:processes> cat summary
processes:
process-id process-name max-memory-allocated state
---------- ---------------- ------------------- -------------------
0 KERNEL 0 interruptible-sleep
331 dhcpd 108920832 interruptible-sleep
336 vmmmgr 334442496 interruptible-sleep
554 neo 398274560 interruptible-sleep
1034 ae 153690112 interruptible-sleep
1214 eventmgr 514793472 interruptible-sleep
2541 bootmgr 292020224 interruptible-sleep
4390 snoopy 28499968 interruptible-sleep
5832 scripthandler 254308352 interruptible-sleep
19204 dbgr 648941568 interruptible-sleep
21863 nginx 4312199168 interruptible-sleep
32192 appliancedirector 136732672 interruptible-sleep
32197 sshd 1228800 interruptible-sleep
32202 perfwatch 19345408 interruptible-sleep
```

```
32203 observer 724484096 interruptible-sleep
32205 lldpad 1200128 interruptible-sleep
32209 topomgr 280576000 interruptible-sleep
32210 xinetd 99258368 interruptible-sleep
32213 policymgr 673251328 interruptible-sleep
32215 reader 258940928 interruptible-sleep
32216 logwatch 266596352 interruptible-sleep
32218 idmgr 246824960 interruptible-sleep
32416 keyhole 15233024 interruptible-sleep
admin@apic1:processes>
```

Each of the processes running on the APIC writes to a log file on the system. These log files can be bundled as part of the APIC techsupport file but can also be observed through SSH shell access in /var/log/dme/log. For example, the Policy Manager process log output is written into /var/log/dme/log/svc_ifc_policymgr.bin.log.

The following is a brief description of the processes running on the system. This can help in understanding which log files to reference when troubleshooting a particular process or understand the impact to the system if a process crashed:

Process	Function
KERNEL	Linux kernel
dhcpd	DHCP process running for APIC to assign infra addresses
vmmmgr	Handles process between APIC and Hypervisors
neo	Shell CLI Interpreter
ae	Handles the state and inventory of local APIC appliance
eventmgr	Handles all events and faults on the system
bootmgr	Controls boot and firmware updates on fabric nodes
snoopy	Shell CLI help, tab command completion
scripthandler	Handles the L4-L7 device scripts and communication
dbgr	Generates core files when process crashes
nginx	Web service handling GUI and REST API access
appliancedirector	Handles formation and control of APIC cluster
sshd	Enabled SSH access into the APIC
perfwatch	Monitors Linux cgroup resource usage
observer	Monitors the fabric system and data handling of state, stats, health
lldpad	LLDP Agent
topomgr	Maintains fabric topology and inventory