



# Troubleshooting CPS

---

- [General Troubleshooting](#), on page 1
- [Recovery using Remove/Add Members Option](#), on page 11
- [Maintenance Window Procedures](#), on page 14
- [Non-maintenance Window Procedures](#), on page 15
- [Common Troubleshooting Tasks](#), on page 15
- [LDAP Error Codes](#), on page 16
- [Diameter Issues and Errors](#), on page 27
- [Frequently Encountered Scenarios](#), on page 48
- [Troubleshoot REDIS](#), on page 121
- [Troubleshooting Graphite Database](#), on page 124
- [SNMP Traps and Key Performance Indicators \(KPIs\)](#), on page 130
- [Troubleshooting Scenarios in OpenStack Environment](#), on page 155
- [FAQs](#), on page 156
- [Reference Document](#), on page 157

## General Troubleshooting

- Find out if your problem is related to CPS or another part of your network.
- Gather information that facilitate the support call.
- Are their specific SNMP traps being reported that can help you isolate the issue?

## Gathering Information

Determine the Impact of the Issue

- Is the issue affecting subscriber experience?
- Is the issue affecting billing?
- Is the issue affecting all subscribers?
- Is the issue affecting all subscribers on a specific service?
- Is there anything else common to the issue?

- Have there been any changes performed on the CPS system or any other systems?
- Has there been an increase in subscribers?
- Initially, categorize the issue to determine the level of support needed.




---

**Important** Use the `dump_utility.py` script to collect useful troubleshooting information for Cisco technical support. Information is printed on the terminal and stored in a log file:  
`/var/tmp/dumputility-<date_time_when_executed>.log`

For more information about this utility, refer to the list of CPS Commands in the *CPS Operations Guide*.

---

## Collecting MongoDB Information for Troubleshooting

This sections describes steps on how to collect information regarding mongo if a customer has issues with MongoDB:

---

**Step 1** Collect the information from `/etc/broadhop/mongoConfig.cfg` file from `pcrfclient01` VM.

**Step 2** Collect `diagnostics.sh --get_replica_status` output.

**Note** If a member is shown in an unknown state, it is likely that the member is not accessible from one of other members, mostly an arbiter or the member itself is down. In that case, you must go to that member VM and check its connectivity with other members as well the mongo process. Also, you can login to mongo on that member and check its actual status.

- If a mongo member is up and running, then check the network connectivity with other members. In case, there is a connectivity issue, report the issue to network administrator.
- If a member is up and running and network connectivity is also good, check the status of member using mongo CLI. In case, status is "OTHER", restart the mongo process again. After restart, the replica-set will come up as Secondary (based on its priority).
- If a member is down, start the mongo process.

**Step 3** Collect the information from `/var/log/broadhop/mongodb-<dbportnum>.log` file from the sessionmgr VMs where database is hosted (primary/secondary/arbiter for all hosts in the configured replica set. If multiple replica sets experience issues collect from 1 replica set).

**Step 4** Connect to the primary sessionmgr VM hosting the database and collect the data (for example, for 10 minutes) by executing the following commands:

```
/usr/bin/mongotop --port <dbportnum> | awk '{ print strftime("%Y-%m-%d %H:%M:%S"), $0; fflush(); }'
>
```

```
/var/tmp/mongotop-dbportnum.log &
```

where, `<dbportnum>` is the mongoDB port for the given database (session/SPR/balance/admin), such as 27717 for balance database.

```
vmstat 1 | awk '{ print strftime("%Y-%m-%d %H:%M:%S"), $0; fflush(); }' > /var/tmp/vmstat.log &
```

**Note** The above mentioned three commands must not be left running on the system, otherwise there will be performance degradation. After 10 min (or so), kill the above mentioned three processes using the 'kill -9' command on each of the three processes.

**Step 5** Connect to the primary sessionmgr VM hosting the balance and collect all the database dumps by executing the following command:

```
mongodump --host <ipaddress> --port <dbport>
```

**Note** The mongo dump is a disk space intensive operation based on your database size, so run it from a VM which has enough disk space. It is also recommended to remove the collected dump/logs once diagnosis is complete.

**Step 6** Use the following command to check mongoDB statistics on queries/inserts/updates/deletes for all CPS databases (and on all primary and secondary databases) and verify if there are any abnormalities (for example, high number of insert/update/delete considering TPS, large number of queries going to other site). Here considering the session database as an example:

```
mongostat --host <sessionmgr VM name> --port <dBportnumber>
```

For example,

```
mongostat --host sessionmgr01 --port 27717
```

## High CPU Usage Issue

- Thread details and jstack output. It could be captured as:

- From top output see if java process is taking high CPU.

- Capture output of the following command:

```
ps -C java -L -o pcpu,cpu,nice,state,cputime,pid,tid | sort > tid.log
```

- Capture output of the following command where <process pid> is the pid of process causing high CPU (as per top output):

If java process is running as a root user:

```
jstack <process pid> > jstack.log
```

If java process is running as policy server (qns) user :

```
sudo -u qns "jstack <process pid>" > jstack.log
```

If running above commands report error for process hung/not responding then use `-F` option after jstack.

Capture another jstack output as above but with an additional `-l` option

## JVM Crash

JVM generates a fatal error log file that contains the state of process at the time of the fatal error. By default, the name of file has format `hs_err_pid<pid>.log` and it is generated in the working directory from where the corresponding java processes were started (that is the working directory of the user when user started the policy server (qns) process). If the working directory is not known then one could search system for file with name `hs_err_pid*.log` and look into file which has timestamp same as time of error.

## High Memory Usage/Out of Memory Error

- JVM could generate heap dump in case of out of memory error. By default, CPS is not configured to generate heap dump. For generating heap dump the following parameters need to be added to `/etc/broadhop/jvm.conf` file for different CPS instances present.

```
-XX+HeapDumpOnOutOfMemoryError
-XXHeapDumpPath=/tmp
```

Note that the heap dump generation may fail if limit for core is not set correctly. Limit could be set in file `/etc/security/limits.conf` for root and policy server (qns) user.

- If no dump is generated but memory usage is high and is growing for sometime followed by reduction in usage (may be due to garbage collection) then the heap dump can be explicitly generated by running the following command:

- If java process is running as user root:

```
jmap -dumpformat=bfile=<filename> <process_id>
```

- If java process is running as policy server (qns) user:

```
jmap -J-d64 -dump:format=b,file=<filename> <process id>
```

Example: `jmap -J-d64 -dump:format=b,file=/var/tmp/jmapheapdump_18643.map 13382`



### Note

- Capture this during off-peak hour. In addition to that, nice utility could be used to reduce priority of the process so that it does not impact other running processes.
- Create archive of dump for transfer and make sure to delete dump/archive after transfer.

- Use the following procedure to log Garbage Collection:

- Login to VM instance where GC (Garbage Collection) logging needs to be enabled.
- Run the following commands:

```
cd /opt/broadhop/qns-1/bin/
chmod +x jmxterm.sh
./jmxterm.sh
> open <host>:<port>
> bean com.sun.management:type=HotSpotDiagnostic
> run setVMOption PrintGC true
> run setVMOption PrintGCDateStamps true
> run setVMOption PrintGCDetails true
> run setVMOption PrintGCDetails true
> exit
```

- Revert the changes once the required GC logs are collected.

## Issues with Output displayed on Grafana

In case of Grafana issue, whisper database output is required.

```
whisper-fetch.py --pretty /var/lib/carbon/whisper/cisco/quantum/qps/hosts/*
```

For example,

```
whisper-fetch.py --pretty
/var/lib/carbon/whisper/cisco/quantum/qps/dcl-pcrfclient02/load/midterm.wsp
```

## Basic Troubleshooting

Capture the following details in most error cases:

**Step 1** Output of the following commands:

```
diagnostics.sh
```

For more information on `diagnostics.sh`, refer to **diagnostics.sh** section in *CPS Operations Guide*.

```
about.sh
```

**Step 2** Collect all the logs:

- Archive created at `/var/log/broadhop` on `pcrfclient01` and `pcrfclient02` includes consolidated policy server (qns) logs. Make sure that consolidated logs cover logs of time when issue happened.

- SSH to all available policy server (qns) and load balancer (lb) VMs and capture the following logs:

```
/var/log/broadhop/qns-*.log
```

```
/var/log/broadhop/qns-*.log.gz
```

```
/var/log/broadhop/service-qns-*.log
```

```
/var/log/broadhop/service-qns-*.log.gz
```

- SSH to all the available sessionmgr VMs and capture the following mongoDB logs:

```
/var/log/mongodb-*.log
```

```
/var/log/mongodb-*.log.gz
```

- SSH to all available VMs and capture the following logs:

```
/var/log/messages*
```

**Step 3** CPS configuration details present at `/etc/broadhop`.

**Step 4** SVN repository

To export SVN repository, go to `/etc/broadhop/qns.conf` and copy the URL specified against `com.broadhop.config.url`.

For example,

```
-Dcom.broadhop.config.url=http://pcrfclient01/repos/run
```

Run the following command to export SVN repository:

```
svn export <url of run repo copied from qns.conf> <folder name where data is to be exported>
```

**Note** Instead of performing [Step 2, on page 5](#) to [Step 4, on page 5](#), you can use `dump_utility.py` to collect all the logs, configuration and SVN repository details.

**Step 5** Top command on all available VMs to display the top CPU processes on the system:

```
top -b -n 30
```

**Step 6** Output of the following command from perfcclient01 VM top\_qps.sh with output period of 10-15 min and interval of 5 sec:

```
top_qps.sh 5
```

**Step 7** Output of the following command on load balancer (lb) VMs having issue.

```
netstat -plan
```

**Step 8** Output of the following command on all VMs.

```
service iptables status
```

**Step 9** Details mentioned in [Periodic Monitoring](#).

---

## Trace Support Commands

This section covers the following two commands:

- trace.sh
- trace\_id.sh

For more information on trace commands, refer to *Policy Tracing and Execution Analyzer* section in *CPS Operations Guide*.

### trace.sh

trace.sh usage:

```
/var/qps/bin/control/trace.sh -i <specific id> -d sessionmgr01:27719/policy_trace
/var/qps/bin/control/trace.sh -x <specific id> -d sessionmgr01:27719/policy_trace
/var/qps/bin/control/trace.sh -a -d sessionmgr01:27719/policy_trace
/var/qps/bin/control/trace.sh -e -d sessionmgr01:27719/policy_trace
```

This script starts a selective trace and outputs it to standard out.

- Specific Audit Id Tracing

```
/var/qps/bin/control/trace.sh -i <specific id>
```

- Dump All Traces for Specific Audit Id

```
/var/qps/bin/control/trace.sh -x <specific id>
```

- Trace All.

```
/var/qps/bin/control/trace.sh -a
```

- Trace All Errors.

```
/var/qps/bin/control/trace.sh -e
```

## trace\_id.sh

trace\_id.sh usage:

```
/var/qps/bin/control/trace_ids.sh -i <specific id> -d sessionmgr01:27719/policy_trace
/var/qps/bin/control/trace_ids.sh -r <specific id> -d sessionmgr01:27719/policy_trace
/var/qps/bin/control/trace_ids.sh -x -d sessionmgr01:27719/policy_trace
/var/qps/bin/control/trace_ids.sh -l -d sessionmgr01:27719/policy_trace
```

This script starts a selective trace and outputs it to standard out.

- Add Specific Audit Id Tracing

```
/var/qps/bin/control/trace_ids.sh -i <specific id>
```

- Remove Trace for Specific Audit Id

```
/var/qps/bin/control/trace_ids.sh -r <specific id>
```

- Remove Trace for All Ids

```
/var/qps/bin/control/trace_ids.sh -x
```

- List All Ids under Trace

```
/var/qps/bin/control/trace_ids.sh -l
```

## Periodic Monitoring

- Run the following command on perfcient01 and verify that all the processes are reported as Running.

For CPS 7.0.0 and higher releases:

```
/var/qps/bin/control/statusall.sh

Program 'cpu_load_trap'
  status                               Waiting
  monitoring status                    Waiting
Process 'collectd'
  status                               Running
  monitoring status                    Monitored
  uptime                               42d 17h 23m
Process 'auditrpmsh.sh'
  status                               Running
  monitoring status                    Monitored
  uptime                               28d 20h 26m
System 'qns01'
  status                               Running
  monitoring status                    Monitored
The Monit daemon 5.5 uptime: 21d 10h 26m
Process 'snmpd'
  status                               Running
  monitoring status                    Monitored
  uptime                               21d 10h 26m
Process 'qns-1'
  status                               Running
  monitoring status                    Monitored
  uptime                               6d 17h 9m
```

- Run `/var/qps/bin/diag/diagnostics.sh` command on perfcient01 and verify that no errors/failures are reported in output.

```

/var/qps/bin/diag/diagnostics.sh
CPS Diagnostics HA Multi-Node Environment
-----
Ping check for all VMs...
Hosts that are not 'pingable' are added to the IGNORED_HOSTS variable...[PASS
]

Checking basic ports for all VMs...[PASS]
Checking qns passwordless logins for all VMs...[PASS]
Checking disk space for all VMs...[PASS]
Checking swap space for all VMs...[PASS]
Checking for clock skew for all VMs...[PASS]
Checking CPS diagnostics...
  Retrieving diagnostics from qns01:9045...[PASS]
  Retrieving diagnostics from qns02:9045...[PASS]
  Retrieving diagnostics from qns03:9045...[PASS]
  Retrieving diagnostics from qns04:9045...[PASS]
  Retrieving diagnostics from pcrfclient01:9045...[PASS]
  Retrieving diagnostics from pcrfclient02:9045...[PASS]
Checking svn sync status between pcrfclient01 & 02...
svn is not sync between pcrfclient01 & pcrfclient02...[FAIL]
Corrective Action(s): Run ssh pcrfclient01 /var/qps/bin/support/recover_svn_sync.sh
Checking HAProxy statistics and ports...

```

For more information on `diagnostics.sh`, refer to **diagnostics.sh** section in *CPS Operations Guide*.

- Perform the following actions to verify VMs status is reported as UP and healthy and no alarms are generated for any VMs.
  - Login to the VMware console
  - Verify the VM statistics, graphs and alarms through the console.
- Verify if any trap is generated by CPS.




---

**Note** `/var/log/snmp/trap` file is updated on active load balancer (LB) only whenever the trap is generated.

---

```

cd /var/log/snmp
tailf trap

```

- Verify if any error is reported in CPS logs.

```

cd /var/log/broadhop
grep -i error consolidated-qns.log
grep -i error consolidated-engine.log

```

- Monitor the following KPIs on Grafana for any abnormal behavior:
  - CPU usage of all instances on all the VMs
  - Memory usage of all instances on all VMs
  - Free disk space on all instances on all VMs
  - Diameter messages load: CCR-I, CCR-U, CCR-T, AAR, RAR, STR, ASR, SDR
  - Diameter messages response time: CCR-I, CCR-U, CCR-T, AAR, RAR, STR, ASR, SDA



- Errors for diameter messages.

Run the following command on pcrfclient01:

```
tailcons | grep diameter | grep -i error
```

- Response time for sessionmgr insert/update/delete/query.

- Average read, write, and total time per sec:

```
mongotop --host sessionmgr* --port port_number
```

- For requests taking more than 100ms:

SSH to sessionmgr VMs:

```
tailf /var/log/mongodb-<portnumber>.log
```




---

**Note** Above commands will by default display requests taking more than 100 ms, until and unless the following parameter has been configured on mongod process --slows XYZms. XYZ represents the value in milliseconds desired by user.

---

- Garbage collection.

Check the `service-qns-*.log` from all policy server (QNS), load balancer (lb) and PCRF VMs. In the logs look for “GC” or “FULL GC”.

- Session count.

Run the following command on pcrfclient01:

```
session_cache_ops.sh --count
```

- Run the following command on pcrfclient01 and verify that the response time is under expected value and there are no errors reported.

```
/opt/broadhop/qns-1/control/top_qps.sh
```

- Use the following command to check mongoDB statistics on queries/inserts/updates/deletes for all CPS databases (and on all primary and secondary databases) and verify if there are any abnormalities (for example, high number of insert/update/delete considering TPS, large number of queries going to other site).

```
mongostat --host <sessionmgr VM name> --port <dBportnumber>
```

For example,

```
mongostat --host sessionmgr01 --port 27717
```

- Use the following command for all CPS databases and verify if there is any high usage reported in output. Here considering session database as an example:

```
mongotop --host <sessionmgr VM name> --port <dBportnumber>
```

For example,

```
mongotop --host sessionmgr01 --port 27717
```

- Verify EDRs are getting generated by checking count of entries in CDR database.
- Verify EDRs are getting replicated by checking count of entries in the databases.
- Determine most recently inserted CDR record in MySQL database and compare the insert time with the time the CDR was generated. Time difference should be within 2 min or otherwise signifies lag in replication.
- Count of CCR-I/CCR-U/CCR-T/RAR messages from/to GW.
- Count of failed CCR-I/CCR-U/CCR-T/RAR messages from/to GW. If GW has capability, capture details at error code level.

Run the following command on pcrfclient01:

```
cd /var/broadhop/stats
grep "Gx_CCR-" bulk-*.csv
```

- Response time of CCR-I/CCR-U/CCR-T messages at GW.
- Count of session in PCRF and count of session in GW. There could be some mismatch between the count due to time gap between determining session count from CPS and GW. If the count difference is high then it could indicate stale sessions on PCRF or GW.
- Count of AAR/RAR/STR/ASR messages from/to Application Function.
- Count of failed AAR/RAR/STR/ASR messages from/to Application Function. If Application Function has capability, capture details at error code level.

Run the following command on pcrfclient01:

```
cd /var/broadhop/stats
grep "Gx_CCR-" bulk-*.csv
```

- Response time of AAR/RAR/STR/ASR messages at Application Function.
- Count of session in PCRF and count of session in Application Function. There could be some mismatch between the count due to time gap between determining session count from CPS and Application Function. If the count difference is high then it could indicate stale sessions on PCRF or Application Function.

Count of session in PCRF:

```
session_cache_ops.sh -count
```

## E2E Call Flow Troubleshooting

- On an All-in-One deployment, run the following commands:

```
tcpdump -i <any port 80 or 8080 or 1812 or 1700 or 1813 or 3868> -s 0 -vv
```

- Append a `-w /tmp/callflow.pcap` to capture output to Wireshark file
- Open the file in WireShark and filter on HTTP to assist debugging the call flow.
- In a distributed model, you need to tcpdump on individual VMs:
  - Load balancers on port 1812, 1813, 1700, 8080 and 3868

Correct call flows are shown [Call Flows](#).

## Recovery using Remove/Add Members Option

When Arbiter blade and a sessionmgr blade goes down there is not any primary sessionmgr node to cater requests coming from CPS VMs (Classic HA setup-1 arbiter 2 sessionmgrs). As a result the system becomes unstable.

A safe way to recover from the issue is to bring UP the down blades to working state. If bringing blades back to working state is not possible then only way to keep setup working is removing failed members of replica-set from mongo-config. In doing so UP and running sessionmgr node becomes primary. It is must to add failed members back to replica-set once they come online.

The following sections describe how to remove failed members from mongo-replica set and how to add them back in replica-set once they are online.




---

**Note** The steps mentioned in the following sections should be executed properly.

---




---

**Note** The following steps are done only when only one sessionmgr is UP but is in secondary mode and cannot become primary on its own and bringing back down blades (holding arbiter and primary sessionmgr VMs) to operational mode is not possible.

---

### Remove Failed Members

This option is usually used when member/s are not running and treated as failed member. The script removes all such failed members from replica-set.

**Step 1** Login to pcrfclient01/02.

**Step 2** Execute the diagnostics script to know which replica-set or respective component is failed and you want to remove.

```
diagnostics.sh --get_replica_status
```

**Note** If a member is shown in an unknown state, it is likely that the member is not accessible from one of other members, mostly an arbiter. In that case, you must go to that member and check its connectivity with other members.

Also, you can login to mongo on that member and check its actual status.

**Step 3** Execute `build_set.sh` with below options to remove failed member/s from replica set. This operation removes the all failed members across the site.

```
cd /var/qps/bin/support/mongo/
```

For session database:

```
./build_set.sh --session --remove-failed-members
```

Remove Failed Members

For SPR database:

```
./build_set.sh --spr --remove-failed-members
```

For balance database:

```
./build_set.sh --balance --remove-failed-members
```

For report database:

```
./build_set.sh --report --remove-failed-members
```

**Step 4** Execute the diagnostics script again to verify if that particular member is removed.

```
diagnostics.sh --get_replica_status
```

**Note** If status is not seen properly by above command, login to mongo port on sessionmgr and check replica status.

Figure 1: Replica Status

```

ONS Diagnostics
Mongo:2.4.6                                MONGODB REPLICASET STATUS INFORMATION                                Date : 2014-04-03 00:47:30
-----
SET NAME - PORT : IP ADDRESS - REPLICASET STATE - HOST NAME - HEALTH - LAG TIME - PRIORITY
-----
BALANCE:set02
Member-1 - 27718 : 192.168.94.123 - UNKNOWN - pcrfclient01 - OFF-LINE - ----- - 0
Member-2 - 27718 : 192.168.94.226 - UNKNOWN - sessionmgr01 - OFF-LINE - No Primary - 1
Member-3 - 27718 : 192.168.94.227 - SECONDARY - sessionmgr02 - ON-LINE - No Primary - 1
-----
REPORTING:set03
Member-1 - 27719 : 192.168.94.123 - UNKNOWN - pcrfclient01 - OFF-LINE - ----- - 0
Member-2 - 27719 : 192.168.94.226 - UNKNOWN - sessionmgr01 - OFF-LINE - No Primary - 1
Member-3 - 27719 : 192.168.94.227 - SECONDARY - sessionmgr02 - ON-LINE - No Primary - 1
-----
NONE - NONE : NONE - NONE - NONE - NONE - NONE - NONE
Current setup have problem while connecting to the server on port : 27717
-----
SPR:set04
Member-1 - 27720 : 192.168.94.123 - UNKNOWN - pcrfclient01 - OFF-LINE - ----- - 0
Member-2 - 27720 : 192.168.94.226 - UNKNOWN - sessionmgr01 - OFF-LINE - No Primary - 1
Member-3 - 27720 : 192.168.94.227 - SECONDARY - sessionmgr02 - ON-LINE - No Primary - 1
-----

[root@pcrfclient02 mongo]# mongo sessionmgr02:27717
MongoDB shell version: 2.4.6
connecting to: sessionmgr02:27717/test
set01:PRIMARY> rs.status()
{
  "set" : "set01",
  "date" : ISODate("2014-04-03T06:48:15Z"),
  "myState" : 1,
  "members" : [
    {
      "_id" : 2,
      "name" : "sessionmgr02:27717",
      "health" : 1,
      "state" : 1,
      "stateStr" : "PRIMARY",
      "uptime" : 540,
      "optime" : Timestamp(1396507695, 20),
      "optimeDate" : ISODate("2014-04-03T06:48:15Z"),
      "self" : true
    }
  ],
  "ok" : 1
}
set01:PRIMARY>
    
```

**Note** If a member is shown in an unknown state, it is likely that the member is not accessible from one of other members, mostly an arbiter. In that case, you must go to that member and check its connectivity with other members.

Also, you can login to mongo on that member and check its actual status.

# Add Failed Members

- Step 1** Login to pcrfclient01/02.
- Step 2** Once the failed members are back online, they can be added back in replica-set.
- Step 3** Execute the diagnostics script to know which replica-set member is not in configuration or failed member.

```
diagnostics.sh --get_replica_status
```

If status is not seen properly by above command, login to mongo port on sessionmgr and check replica status.

**Note** If a member is shown in an unknown state, it is likely that the member is not accessible from one of other members, mostly an arbiter. In that case, you must go to that member and check its connectivity with other members.

Also, you can login to mongo on that member and check its actual status.

Figure 2: Replica Status

```

ONS Diagnostics
Mongo:2.4.6 MONGODB REPLICA-SETS STATUS INFORMATION Date : 2014-04-03 00:47:30
-----
SET NAME - PORT : IP ADDRESS - REPLICA STATE - HOST NAME - HEALTH - LAG TIME - PRIORITY
-----
BALANCE: set02
Member-1 - 27718 : 192.168.94.123 - UNKNOWN - pcrfclient01 - OFF-LINE - ----- - 0
Member-2 - 27718 : 192.168.94.226 - UNKNOWN - sessionmgr01 - OFF-LINE - No Primary - 1
Member-3 - 27718 : 192.168.94.227 - SECONDARY - sessionmgr02 - ON-LINE - No Primary - 1
-----
REPORTING: set03
Member-1 - 27719 : 192.168.94.123 - UNKNOWN - pcrfclient01 - OFF-LINE - ----- - 0
Member-2 - 27719 : 192.168.94.226 - UNKNOWN - sessionmgr01 - OFF-LINE - No Primary - 1
Member-3 - 27719 : 192.168.94.227 - SECONDARY - sessionmgr02 - ON-LINE - No Primary - 1
-----
NONE - NONE : NONE - NONE - NONE - NONE - NONE - NONE
Current setup have problem while connecting to the server on port : 27717
-----
SPR: set04
Member-1 - 27720 : 192.168.94.123 - UNKNOWN - pcrfclient01 - OFF-LINE - ----- - 0
Member-2 - 27720 : 192.168.94.226 - UNKNOWN - sessionmgr01 - OFF-LINE - No Primary - 1
Member-3 - 27720 : 192.168.94.227 - SECONDARY - sessionmgr02 - ON-LINE - No Primary - 1
-----
[root@pcrfclient02 mongo]# mongo sessionmgr02:27717
MongoDB shell version: 2.4.6
connecting to: sessionmgr02:27717/test
set01:PRIMARY> rs.status()
{
  "set" : "set01",
  "date" : ISODate("2014-04-03T06:48:15Z"),
  "myState" : 1,
  "members" : [
    {
      "_id" : 2,
      "name" : "sessionmgr02:27717",
      "health" : 1,
      "state" : 1,
      "stateStr" : "PRIMARY",
      "uptime" : 540,
      "optime" : Timestamp(1396507695, 20),
      "optimeDate" : ISODate("2014-04-03T06:48:15Z"),
      "self" : true
    }
  ],
  "ok" : 1
}
set01:PRIMARY>
    
```

```
cd /var/qps/bin/support/mongo
```

For session database:

```
./build_set.sh --session --add-members
```

For SPR database:

```
./build_set.sh --spr --add-members
```

For balance database:

```
./build_set.sh --balance --add-members
```

For report database:

```
./build_set.sh --report --add-members
```

---

## Maintenance Window Procedures

The usual tasks for a maintenance window might include these:

### Prior to Any Maintenance

Backup all relevant information to an offline resource. For more information on backup see Cisco Policy Suite Backup and Restore Guide.

- Data - Backup all database information. This includes Cisco MsBM Cisco Unified SuM.



---

**Note** Sessions can be backed up as well.

---

- Configurations - Backup all configuration information. This includes SVN (from PCRF Client) the /etc/broadhop directory from all PCRFs
- Logs - Backup all logs for comparison to the upgrade. This is not required but will be helpful if there are any issues.

### Change Request Procedure

- Have proper sign off for any change request. Cisco and all customer teams must sign off.
- Make sure the proposed procedures are well defined.
- Make sure the rollback procedures are correct and available.

### Software Upgrades

- Determine if the software upgrade will cause an outage and requires a maintenance window to perform the upgrade.
- Typically software upgrades can be done on one node a time and so minimize or eliminate any outage.
- Most of the time an upgrade requires a restart of the application. Most applications can be started in less than 1 minute.

### VM Restarts

- LINUX must be shutdown normally for VM restarts.

- All VMs are Linux.
- The preferred methods are `init 0` or `shutdown -h`
- Failure to use the Linux OS shutdown can result in VM corruption and problems restarting the VM and applications.
- VM restart is typically done to increase resources to the VM (disk memory CPU).

## Hardware Restarts

- Hardware restarts should be rare.
- When a hardware restart is needed VMs must be shutdown first.
- When all VMs are stopped shutdown the hardware with either the ESXi console or as a power off.

## Planned Outages

- Planned outages are similar to hardware restarts.
- VMs need to be shutdown hardware can then be stopped.
- When hardware is started the typical hardware starting order is:
  - Start the servers with PCRFCClient01 LB01 and SessionMgr01 first.
  - Start all other servers in any order after that.

## Non-maintenance Window Procedures

Tasks you can perform as non-maintenance that is at any time are these

- Data archiving or warehousing
- Log removal

## Common Troubleshooting Tasks

This section describes frequently used troubleshooting tasks you might use before calling support or as directed by support.

### Low or Out of Disk Space

To determine the disk space used use these Linux disk usage and disk free commands

- `du`
- `df`

#### df Command

**df**

For example:

```
home# df -h
[root@lab home]# df -h
Filesystem Size Used Avail Use% Mounted on
/dev/cciss/c0d0p5 56G 27G 26G 51% /
/dev/cciss/c0d0p1 99M 12M 83M 12% /boot
tmpfs 2.0G 0 2.0G 0% /dev/shm
none 2.0G 0 2.0G 0% /dev/shm
/dev/cciss/c0d0p2 5.8G 4.0G 1.6G 73% /home
```

As shown above the /home directory is using the most of it's allocated space (73%).

## du Command

The /home directory is typically for /home/admin but in some cases there is also /home/qns or /home/remote. You can check both

### du

For example:

```
home# du -hs
[root@lab home]# du -hs
160M .
[root@lab home]# du -hs *
1.3M qns
158M remote
36K testuser
```

The **du** command shows where the space is being used. By default the du command by itself gives a summary of quota usage for the directory specified and all subdirectories below it.




---

**Note** By deleting any directories you remove the ability to roll back if for some reason an update is not working correctly. Only delete those updates to which you would probably never roll back perhaps those 6 months old and older.

---

## LDAP Error Codes

The following table describes LDAP error codes:

**Table 1: LDAP Error Codes**

	Name	Definition	Counts as Timeout	Triggers Retry	Sent To Policy Server	Terminate Connection	Not Applicable to Search
0	SUCCESS	The result code (0) that will be used to indicate a successful operation			Y		



	Name	Definition	Counts as Timeout	Triggers Retry	Sent To Policy Server	Terminate Connection	Not Applicable to Search
1	OPERATIONS_ERROR	The result code (1) that will be used to indicate that an operation was requested out of sequence.		Y		Y	
2	PROTOCOL_ERROR	The result code (2) that will be used to indicate that the client sent a malformed request.		Y		Y	
3	TIME_LIMIT_EXCEEDED	The result code (3) that will be used to indicate that the server was unable to complete processing on the request in the allotted time limit.	Y	Y			
4	SIZE_LIMIT_EXCEEDED	The result code (4) that will be used to indicate that the server found more matching entries than the configured request size limit.					Y
5	COMPARE_FALSE	The result code (5) that will be used if a requested compare assertion does not match the target entry.					Y
6	COMPARE_TRUE	The result code (6) that will be used if a requested compare assertion matched the target entry.					Y

	Name	Definition	Counts as Timeout	Triggers Retry	Sent To Policy Server	Terminate Connection	Not Applicable to Search
7	AUTH_METHOD_NOT_SUPPORTED	The result code (7) that will be used if the client requested a form of authentication that is not supported by the server.					Y
8	STRONG_AUTH_REQUIRED	The result code (8) that will be used if the client requested an operation that requires a strong authentication mechanism.					Y
10	REFERRAL	The result code (10) that will be used if the server sends a referral to the client to refer to data in another location.					Y
11	ADMIN_LIMIT_EXCEEDED	The result code (11) that will be used if a server administrative limit has been exceeded.					Y
12	UNAVAILABLE_CRITICAL_EXTENSION	The integer value (12) for the "UNAVAILABLE_CRITICAL_EXTENSION" result code.					Y
13	CONFIDENTIALITY_REQUIRED	The result code (13) that will be used if the server requires a secure communication mechanism for the requested operation.					Y

	Name	Definition	Counts as Timeout	Triggers Retry	Sent To Policy Server	Terminate Connection	Not Applicable to Search
14	SASL_BIND_IN_PROGRESS	The result code (14) that will be returned from the server after SASL bind stages in which more processing is required.					Y
16	NO_SUCH_ATTRIBUTE	The result code (16) that will be used if the client referenced an attribute that does not exist in the target entry.					Y
17	UNDEFINED_ATTRIBUTE_TYPE	The result code (17) that will be used if the client referenced an attribute that is not defined in the server schema.					Y
18	INAPPROPRIATE_MATCHING	The result code (18) that will be used if the client attempted to use an attribute in a search filter in a manner not supported by the matching rules associated with that attribute.					Y
19	CONSTRAINT_VIOLATION	The result code (19) that will be used if the requested operation would violate some constraint defined in the server.					Y

	<b>Name</b>	<b>Definition</b>	<b>Counts as Timeout</b>	<b>Triggers Retry</b>	<b>Sent To Policy Server</b>	<b>Terminate Connection</b>	<b>Not Applicable to Search</b>
20	ATTRIBUTE_OR_VALUE_EXISTS	The result code (20) that will be used if the client attempts to modify an entry in a way that would create a duplicate value, or create multiple values for a single-valued attribute.					Y
21	INVALID_ATTRIBUTE_SYNTAX	The result code (21) that will be used if the client attempts to perform an operation that would create an attribute value that violates the syntax for that attribute.					Y
32	NO_SUCH_OBJECT	The result code (32) that will be used if the client targeted an entry that does not exist.					Y
33	ALIAS_PROBLEM	The result code (33) that will be used if the client targeted an entry that as an alias.					Y
34	INVALID_DN_SYNTAX	The result code (34) that will be used if the client provided an invalid DN.					Y
36	ALIAS_DEREFERENCING_PROBLEM	The result code (36) that will be used if a problem is encountered while the server is attempting to dereference an alias.					Y

	Name	Definition	Counts as Timeout	Triggers Retry	Sent To Policy Server	Terminate Connection	Not Applicable to Search
48	INAPPROPRIATE_AUTHENTICATION	The result code (48) that will be used if the client attempts to perform a type of authentication that is not supported for the target user.					Y
49	INVALID_CREDENTIALS	The result code (49) that will be used if the client provided invalid credentials while trying to authenticate.					Y
50	INSUFFICIENT_ACCESS_RIGHTS	The result code (50) that will be used if the client does not have permission to perform the requested operation.					Y
51	BUSY	The result code (51) that will be used if the server is too busy to process the requested operation.		Y		Y	
52	UNAVAILABLE	The result code (52) that will be used if the server is unavailable.		Y		Y	
53	UNWILLING_TO_PERFORM	The result code (53) that will be used if the server is not willing to perform the requested operation.		Y		Y	
54	LOOP-DETECT	The result code (54) that will be used if the server detects a chaining or alias loop.					Y

	Name	Definition	Counts as Timeout	Triggers Retry	Sent To Policy Server	Terminate Connection	Not Applicable to Search
60	SORT_CONTROL_MISSING	The result code (60) that will be used if the client sends a virtual list view control without a server-side sort control.					Y
61	OFFSET_RANGE_ERROR	The result code (61) that will be used if the client provides a virtual list view control with a target offset that is out of range for the available data set.					Y
64	NAMING_VIOLATION	The result code (64) that will be used if the client request violates a naming constraint (e.g., a name form or DIT structure rule) defined in the server.					Y
65	OBJECT_CLASS_VIOLATION	The result code (65) that will be used if the client request violates an object class constraint (e.g., an undefined object class, a disallowed attribute, or a missing required attribute) defined in the server.					Y
66	NOT_ALLOWED_ON_NONLEAF	The result code (66) that will be used if the requested operation is not allowed to be performed on non-leaf entries.					Y

	Name	Definition	Counts as Timeout	Triggers Retry	Sent To Policy Server	Terminate Connection	Not Applicable to Search
67	NOT_ALLOWED_ON_RDN	The result code (67) that will be used if the requested operation would alter the RDN of the entry but the operation was not a modify DN request.					Y
68	ENTRY_ALREADY_EXISTS	The result code (68) that will be used if the requested operation would create a conflict with an entry that already exists in the server.					Y
69	OBJECT_CLASS_MODS_PROHIBITED	The result code (69) that will be used if the requested operation would alter the set of object classes defined in the entry in a disallowed manner.					Y
71	AFFECTS_MULTIPLE_DSAS	The result code (71) that will be used if the requested operation would impact entries in multiple data sources.					Y
76	VIRTUAL_LIST_VIEW_ERROR	The result code (76) that will be used if an error occurred while performing processing associated with the virtual list view control.					Y
80	OTHER	The result code (80) that will be used if none of the other result codes are appropriate.		Y		Y	

	<b>Name</b>	<b>Definition</b>	<b>Counts as Timeout</b>	<b>Triggers Retry</b>	<b>Sent To Policy Server</b>	<b>Terminate Connection</b>	<b>Not Applicable to Search</b>
81	SERVER_DOWN	The client-side result code (81) that will be used if an established connection to the server is lost.		Y		Y	
82	LOCAL_ERROR	The client-side result code (82) that will be used if a generic client-side error occurs during processing.		Y		Y	
83	ENCODING_ERROR	The client-side result code (83) that will be used if an error occurs while encoding a request.		Y		Y	
84	DECODING_ERROR	The client-side result code (84) that will be used if an error occurs while decoding a response.		Y		Y	
85	TIMEOUT	The client-side result code (85) that will be used if a client timeout occurs while waiting for a response from the server.	Y	Y		Y	
86	AUTH_UNKNOWN	The client-side result code (86) that will be used if the client attempts to use an unknown authentication type.					Y
87	FILTER_ERROR	The client-side result code (87) that will be used if an error occurs while attempting to encode a search filter.			Y		



	Name	Definition	Counts as Timeout	Triggers Retry	Sent To Policy Server	Terminate Connection	Not Applicable to Search
88	USER_CANCELED	The client-side result code (88) that will be used if the end user canceled the operation in progress.					Y
89	PARAM_ERROR	The client-side result code (89) that will be used if there is a problem with the parameters provided for a request.			Y		
90	NO_MEMORY	The client-side result code (90) that will be used if the client does not have sufficient memory to perform the requested operation.		Y		Y	
91	CONNECT_ERROR	The client-side result code (91) that will be used if an error occurs while attempting to connect to a target server.		Y		Y	
92	NOT_SUPPORTED	The client-side result code (92) that will be used if the requested operation is not supported.					Y
93	CONTROL_NOT_FOUND	The client-side result code (93) that will be used if the response from the server did not include an expected control.					Y
94	NO_RESULTS_RETURNED	The client-side result code (94) that will be used if the server did not send any results.			Y		

	Name	Definition	Counts as Timeout	Triggers Retry	Sent To Policy Server	Terminate Connection	Not Applicable to Search
95	MORE_RESULTS_TO_RETURN	The client-side result code (95) that will be used if there are still more results to return.					Y
96	CLIENT_LOOP	The client-side result code (96) that will be used if the client detects a loop while attempting to follow referrals.					Y
97	REFERRAL_LIMIT_EXCEEDED	The client-side result code (97) that will be used if the client encountered too many referrals in the course of processing an operation.					Y
118	CANCELED	The result code (118) that will be used if the operation was canceled					Y
119	NO_SUCH_OPERATION	The result code (119) that will be used if the client attempts to cancel an operation that the client doesn't exist in the server.					Y
120	TOO_LATE	The result code (120) that will be used if the client attempts to cancel an operation too late in the processing for that operation.					Y
121	CANNOT_CANCEL	The result code (121) that will be used if the client attempts to cancel an operation that cannot be canceled.					Y

	Name	Definition	Counts as Timeout	Triggers Retry	Sent To Policy Server	Terminate Connection	Not Applicable to Search
122	ASSERTION_FAILED	The result code (122) that will be used if the requested operation included the LDAP assertion control but the assertion did not match the target entry.					Y
123	AUTHORIZATION_DENIED	The result code (123) that will be used if the client is denied the ability to use the proxied authorization control.					Y

## Diameter Issues and Errors

### Diameter Issues

The following details need to be captured for diameter issues:

- Details of service associated with subscribers in failure case.
- Pcaps capturing calls having issue.
- If the issue is with no response pcap should be captured both at CPS and the peer.
- Subscriber trace information can be captured using the following process

- To add the subscriber that needs to be traced

```
/var/qps/bin/control/trace_ids.sh -i <msisdn/imsi> -d sessionmgr01:<port no>/policy_trace
```

```
cd /var/qps/bin/control
```

- Run the following command to obtain subscriber information

```
/var/qps/bin/control/trace.sh -i <msisdn/imsi> -d sessionmgr01<port no>/policy_trace
```

If CPS receives the request message for the same subscriber the trace result will be displayed.



**Note** Port no. can be found in “Trace DB Database” configuration in Cluster-1. If Trace Database is not configured then by default “Admin Db Configuration” will pick up the trace database.

## Diameter Proxy Error in diagnostics.sh Output

When you execute `diagnostics.sh` script on `pcrfclient01` VM and it shows the following errors related to diameter proxy

For more information on `diagnostics.sh`, refer to **diagnostics.sh** section in *CPS Operations Guide*.

```
diameter_proxy-lb01_A DOWN L4CON
Sessions (current,max,limit): 0,0, Rate (sessions,max,limit): 0,0, Last Status change
(seconds): 2513094
diameter_proxy-lb01_B DOWN L4CON
Sessions (current,max,limit): 0,0, Rate (sessions,max,limit): 0,0, Last Status change
(seconds): 2513093
diameter_proxy-lb01_C DOWN L4CON
Sessions (current,max,limit): 0,0, Rate (sessions,max,limit): 0,0, Last Status change
(seconds): 2513092
diameter_proxy-BACKEND DOWN
Sessions (current,max,limit): 0,0,2000 Rate (sessions,max,limit): 0,0, Last Status change
(seconds): 2513092
```

The error L4CON message indicates that there is connection problem (e.g. “Connection refused” or “No route to host”) at layer 1-4. And the error message `diameter_proxy-BACKEND DOWN` signifies that all the service specified in `diameter_proxy` section in `haproxy.cfg` file are down.

1. Check whether HAProxy is running on load balancer VM. Specifically for this error message we should check in `lb01`.
2. Check the HAProxy configuration:

```
vi /etc/haproxy/haproxy.cfg
```

It should show similar entries as shown below. Try to telnet to corresponding load balancer VM with corresponding ports:

```
diameter_proxy-lb01_A DOWN L4CON
Sessions (current,max,limit): 0,0, Rate (sessions,max,limit): 0,0, Last Status change
(seconds): 2513094
diameter_proxy-lb01_B DOWN L4CON
Sessions (current,max,limit): 0,0, Rate (sessions,max,limit): 0,0, Last Status change
(seconds): 2513093
diameter_proxy-lb01_C DOWN L4CON
Sessions (current,max,limit): 0,0, Rate (sessions,max,limit): 0,0, Last Status change
(seconds): 2513092
diameter_proxy-BACKEND DOWN
Sessions (current,max,limit): 0,0,2000 Rate (sessions,max,limit): 0,0, Last Status change
(seconds): 2513092
```

## Diameter Peer Connectivity is Down

If your Diameter Peer connectivity is down check the following:

1. Check the TCP connection on the diameter port (i.e.,) “`netstat -pant | grep 3868`”. It should be in established state.
2. If the TCP connection is not getting established disable the firewall service `iptables stop` and check the port status `/opt/broadhop/installer/support/add_open_port.sh pcrf 3868`.
3. Open the Internet browser and go to your repository and check the published policies in runtime environment. You should notice the following configuration. If the following configuration is not there, then most probably it is a bad publish.

```
DiameterConfiguration-_4davIF2KEeOXe-MDH-2FEQ.xml
```

```
DiameterStack-default-_A5cgQF2LEeOXe-MDH-2FEQ.xml
```

4. If the problem is not in CPS and something is mis-configured in PCEF then you may notice the following messages in CPS

```
tail -f /var/log/broadhop/service-qns-1.log  
  
Sending Alert Notification for host pcef realm lab.realm is down  
Sending Alert Notification for host pcef realm lab.realm is back up  
Sending Alert Notification for host pcef realm lab.realm is down  
Sending Alert Notification for host pcef realm lab.realm is back up
```

## No Response to Diameter Request

### Using TCPDUMP

- Collect tcpdump packet capture from the primary policy director (IOmanager).

```
tcpdump -i any -port 3868 -s0 -w filename test.pcap
```

In the collected trace file,

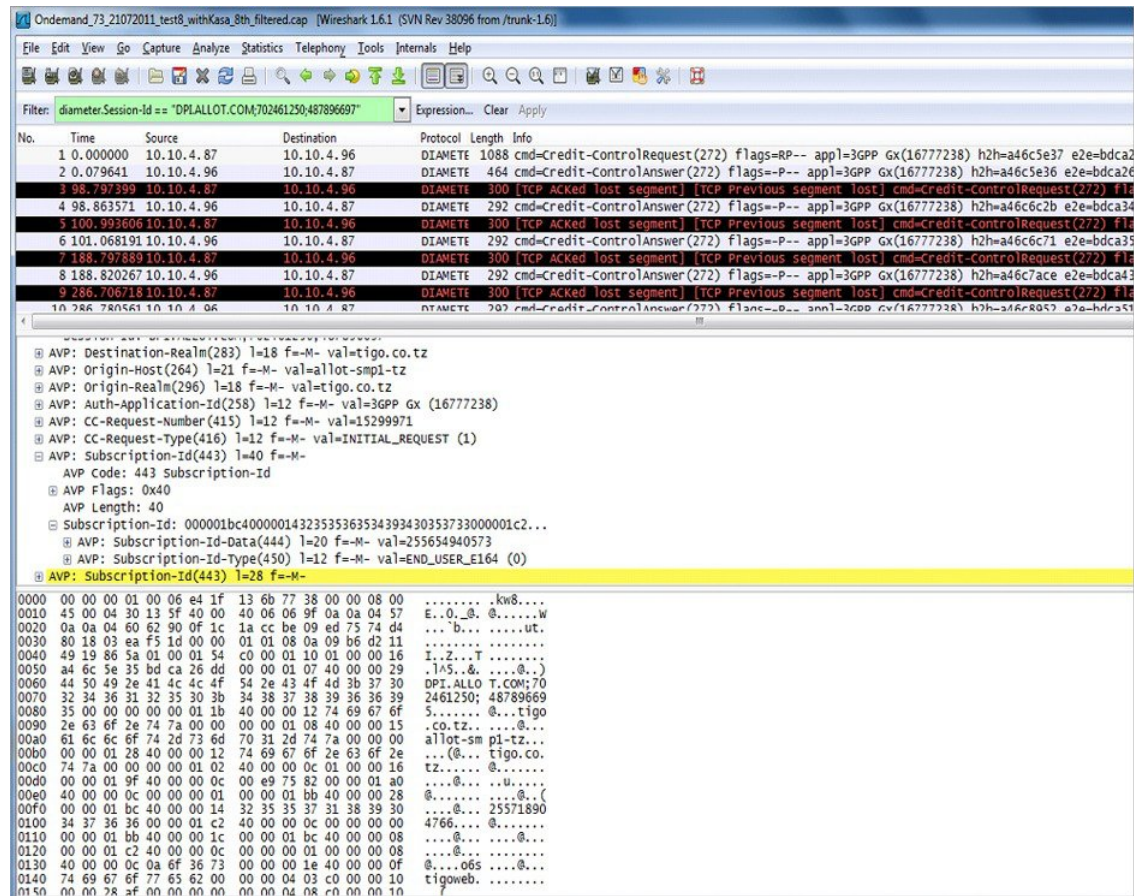
- Verify that the response message is sent back to PCEF.
- Use Session-Id as filter if the Session-Id of the user's session is available.
- If Session-Id for the user is not available use MSISDN as filter to retrieve the Session-Id. Then apply Session-Id filter to view all the messages for the session.
- Match the request to response for Credit Control Request CC-Request-Type attribute (Initial/Update/Terminate).

### CPS Logs

- Verify the consolidated-qns.log on PCRCLIENT01 for any exceptions with policy executions for example Null Pointer Exception.
- Filter using Session-Id

### TCPDUMP – User Id Filter

Figure 3: TCPDUMP – User Id Filter



- Filter using Subscription-Id-Data (MSISDN) to retrieve the CCR initial request.

## Diagnose Diameter No Response for Peer Message

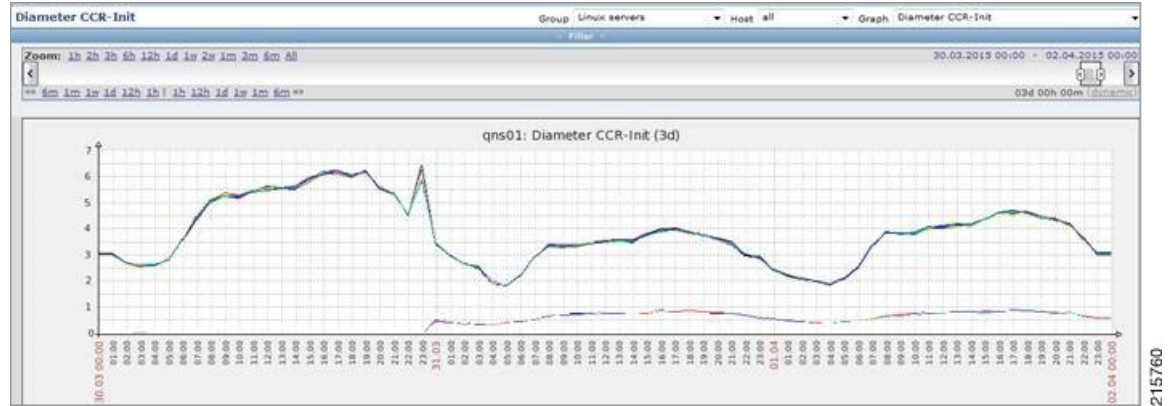


**Note** The port numbers provided in this section are an example and can differ based on the network deployment. For more information on port numbers, refer to [Table 2: Policy Director/Policy Server Listening Ports](#), on page 32.

### Traffic Failover or Similar

In a Geo-Redundant deployment when there are issues in message processing on primary-site A policy director (LB) VMs then there is an increase in diameter traffic sent to secondary-site. This is an indication that there is a failure in responding to messages sent on primary-site A due to message response timeouts. For example, the following zabbix graphs shows diameter meter traffic failing over to secondary from 30th Mar 2300 onwards.

Figure 4: Zabbix Graph



**Note** Here Zabbix graph is an example and similar graph in Grafana (6.x.x) or client traffic graphs reports CPS dropping response.

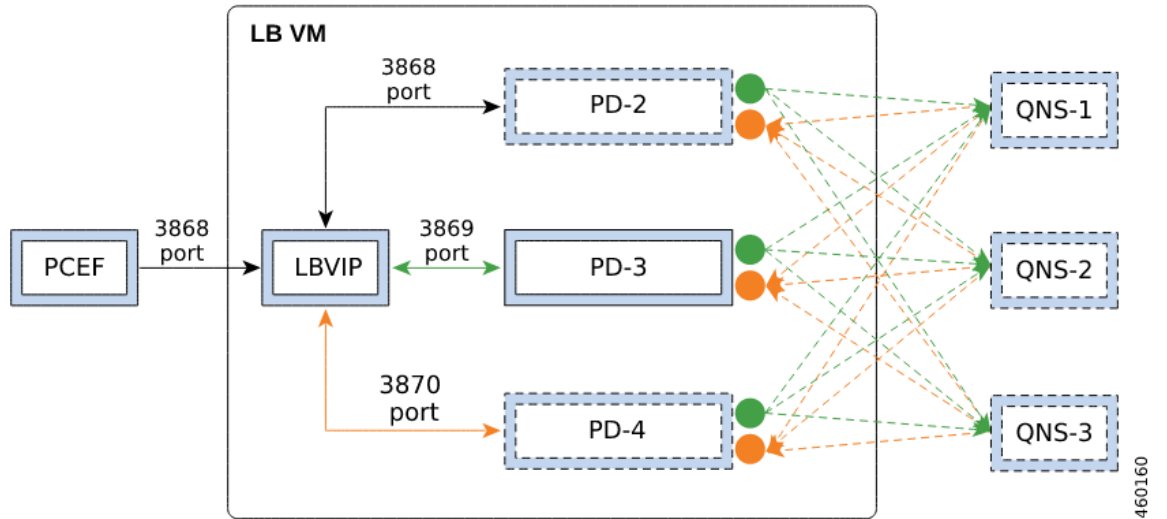
**Policy Director (LB)<->Policy Server (QNS) Messaging**

The following diagram describes processing of diameter messages sent from PCEF on EBW secondary policy director (lb).



**Note** The port numbers provided in this section are an example and can differ based on the network deployment. For more information on port numbers contact your Cisco Technical Representative.

Figure 5: Messaging between Policy Director (LB) and Policy Server (QNS)



As per the PCRF deployment PCEF sends diameter traffic on the 3868 port of the LBVIP running on the active policy director (LB) VM. These messages are distributed in a round-robin scheduling between three

Policy Director (PD) instances based on the haproxy configuration. All the PDs are connected to all the policy server (QNS) VMs instances using the ZMQ queues. Each PD uses a PUSH queue to send data to policy server (QNS) VM and PULL Queue to process a response from policy server (QNS) VM. The following table describes the various PUSH and PULL queue ports mapping

**Table 2: Policy Director/Policy Server Listening Ports**

VM	Listening Ports	Description
Policy Director (lb)	2800x	All the Policy Servers (QNS) connect to all the Policy Director-2/3/4 (lb) instances local to the site listening on this port.
Policy Server (QNS)	28500	All the Policy Director-2/3/4 (lb) instances (local + remote) connect to all the Policy Servers (local + remote) listening on this port.
Policy Director (lb)	2825x	All the Policy Servers (QNS) connect to all the Policy Director-2/3/4 (lb) instances (local + remote) to the site listening on this port.
Policy Director (lb)	2925x, 2900x, 2875x	All the Policy Director-2/3/4 (lb) instances connect to all the Policy Director (lb) instances (Remote) to the site listening on this port.
<b>Applicable only when LDAP feature is enabled</b>		
Policy Server (QNS)	20500, 30500	PolicyDirector-1 instance (local + remote) connects to all the Policy Servers (local + remote) listening on this port.
Policy Director (lb)	20250, 30250	All the Policy Servers (QNS) connect to all the PolicyDirector-1 instance (local + remote) to the site listening on this port.
Policy Director (lb)	20200, 30000	All the Policy Servers (QNS) connect to all the PolicyDirector-1 instance (local) to the site listening on this port.



**Note** where, x is the Policy Director instance number - 1.

- For example, for PD-2 ports will be 28001 | 29251 | 29001 | 28751 | 28251.
- For example, for PD-3 ports will be 28002 | 29252 | 29002 | 28752 | 28252.
- For example, for PD-4 ports will be 28003 | 29253 | 29003 | 28753 | 28253

Port Details



### 1. HAProxy ports

monit status qnsXX

#### PD-2 port

```
netstat -anp | grep 31654 | grep 3868
tcp 0 0 ::ffff:198.51.100.3:3868 :::* LISTEN 31654/java
tcp 0 0 ::ffff:198.51.100.3:3868 ::ffff:198.51.100.3:52762 ESTABLISHED 31654/java
```

#### PD-3 port

```
netstat -anp | grep 31701 | grep 3869
tcp 0 0 ::ffff:198.51.100.3:3869 :::* LISTEN 31701/java
tcp 0 0 ::ffff:198.51.100.3:3869 ::ffff:198.51.100.3:60936 ESTABLISHED 31701/java
```

#### PD-4 port

```
netstat -anp | grep 31753 | grep 3870
tcp 0 0 ::ffff:198.51.100.3:3870 :::* LISTEN 31753/java
tcp 0 0 ::ffff:198.51.100.3:3870 ::ffff:198.51.100.3:34338 ESTABLISHED 31753/java
```

### 2. Policy Server (QNS):

- Receive Port (ZMQ PULL Queue ports): 28500

```
netstat -apn | grep 3576087 | grep -i 28500
tcp6 0 0 192.0.2.1:28500 :::* LISTEN 3576087/java
tcp6 0 0 192.0.2.1:28500 192.0.2.3:35560 ESTABLISHED 3576087/java
tcp6 0 0 192.0.2.1:28500 192.0.2.4:51442 ESTABLISHED 3576087/java
tcp6 0 0 192.0.2.1:28500 192.0.2.4:51398 ESTABLISHED 3576087/java
tcp6 0 0 192.0.2.1:28500 192.0.2.2:45628 ESTABLISHED 3576087/java
tcp6 0 0 192.0.2.1:28500 192.0.2.5:43130 ESTABLISHED 3576087/java
tcp6 0 0 192.0.2.1:28500 192.0.2.2:45632 ESTABLISHED 3576087/java
tcp6 0 0 192.0.2.1:28500 192.0.2.4:51416 ESTABLISHED 3576087/java
tcp6 0 0 192.0.2.1:28500 192.0.2.5:43158 ESTABLISHED 3576087/java
tcp6 0 0 192.0.2.1:28500 192.0.2.3:35572 ESTABLISHED 3576087/java
tcp6 0 0 192.0.2.1:28500 192.0.2.3:35556 ESTABLISHED 3576087/java
tcp6 0 0 192.0.2.1:28500 192.0.2.5:43116 ESTABLISHED 3576087/java
tcp6 0 0 192.0.2.1:28500 192.0.2.2:45622 ESTABLISHED 3576087/java
```

- For LDAP feature:

Receive Port (ZMQ PULL Queue ports): 20500, 30500

```
netstat -apn | grep 3576087 | egrep "20500|30500"
tcp6 0 0 192.0.2.1:20500 :::* LISTEN 3576087/java
tcp6 0 0 192.0.2.1:30500 :::* LISTEN 3576087/java
tcp6 0 0 192.0.2.1:20500 192.0.2.3:36104 ESTABLISHED 3576087/java
tcp6 0 0 192.0.2.1:20500 192.0.2.4:43238 ESTABLISHED 3576087/java
tcp6 0 0 192.0.2.1:30500 192.0.2.5:37712 ESTABLISHED 3576087/java
tcp6 0 0 192.0.2.1:30500 192.0.2.3:46352 ESTABLISHED 3576087/java
tcp6 0 0 192.0.2.1:30500 192.0.2.4:42880 ESTABLISHED 3576087/java
tcp6 0 0 192.0.2.1:30500 192.0.2.2:38208 ESTABLISHED 3576087/java
tcp6 0 0 192.0.2.1:20500 192.0.2.2:36520 ESTABLISHED 3576087/java
tcp6 0 0 192.0.2.1:20500 192.0.2.5:41146 ESTABLISHED 3576087/java
```

### 3. Policy Director (PD):

- Receive port (ZMQ PULL Queue ports): 2825x, 2875x, 2925x

```
netstat -apn | grep 1783577 | egrep "28251|28751|29251"
tcp6 0 0 192.0.2.2:29251 :::* LISTEN 1783577/java
tcp6 0 0 192.0.2.2:28751 :::* LISTEN 1783577/java
tcp6 0 0 192.0.2.2:28251 :::* LISTEN 1783577/java
tcp6 0 0 192.0.2.2:28251 192.0.2.18:36248 ESTABLISHED 1783577/java
tcp6 0 0 192.0.2.2:28251 192.0.2.30:39796 ESTABLISHED 1783577/java
```

Diagnose Diameter No Response for Peer Message

```

tcp6      0      0 192.0.2.2:28251      192.0.2.21:46248     ESTABLISHED 1783577/java
tcp6      0      0 192.0.2.2:28251      192.0.2.36:32956     ESTABLISHED 1783577/java
tcp6      0      0 192.0.2.2:28251      192.0.2.31:52472     ESTABLISHED 1783577/java
tcp6      0      0 192.0.2.2:28251      192.0.2.36:49132     ESTABLISHED 1783577/java
tcp6      0      0 192.0.2.2:28251      192.0.2.26:34510     ESTABLISHED 1783577/java
tcp6      0      0 192.0.2.2:28251      192.0.2.34:42622     ESTABLISHED 1783577/java
tcp6      0      0 192.0.2.2:28751      192.0.2.3:49834      ESTABLISHED 1783577/java
    
```

- Send port (ZMQ PUSH queue ports): 2800x, 2900x

```

netstat -apn | grep 1783577 | egrep "28001|29001"
tcp6      0      0 192.0.2.2:28001      :::*                  LISTEN       1783577/java
tcp6      0      0 192.0.2.2:29001      :::*                  LISTEN       1783577/java
tcp6      0      0 192.0.2.2:28001      192.0.2.20:48572     ESTABLISHED 1783577/java
tcp6      0      0 192.0.2.2:28001      192.0.2.35:40748     ESTABLISHED 1783577/java
tcp6      0      0 192.0.2.2:29001      192.0.2.5:60350      ESTABLISHED 1783577/java
tcp6      0      0 192.0.2.2:28001      192.0.2.27:41736     ESTABLISHED 1783577/java
tcp6      0      0 192.0.2.2:28001      192.0.2.26:49958     ESTABLISHED 1783577/java
tcp6      0      0 192.0.2.2:28001      192.0.2.25:33866     ESTABLISHED 1783577/java
tcp6      0      0 192.0.2.2:28001      192.0.2.18:35812     ESTABLISHED 1783577/java
tcp6      0      0 192.0.2.2:28001      192.0.2.38:60088     ESTABLISHED 1783577/java
    
```

- For LDAP feature:

Receive port (ZMQ PULL Queue ports): 20250, 30250

```

netstat -apn | grep 1783650 | egrep "20250|30250"
tcp6      0      0 192.0.2.2:30250      :::*                  LISTEN       1783650/java
tcp6      0      0 192.0.2.2:20250      :::*                  LISTEN       1783650/java
tcp6      0      0 192.0.2.2:20250      192.0.2.24:44824     ESTABLISHED 1783650/java
tcp6      0      0 192.0.2.2:30250      192.0.2.20:50902     ESTABLISHED 1783650/java
tcp6      0      0 192.0.2.2:30250      192.0.2.32:43460     ESTABLISHED 1783650/java
tcp6      0      0 192.0.2.2:20250      192.0.2.22:55202     ESTABLISHED 1783650/java
tcp6      0      0 192.0.2.2:30250      192.0.2.30:51822     ESTABLISHED 1783650/java
tcp6      0      0 192.0.2.2:30250      192.0.2.26:33628     ESTABLISHED 1783650/java
tcp6      0      0 192.0.2.2:20250      192.0.2.25:51024     ESTABLISHED 1783650/java
tcp6      0      0 192.0.2.2:30250      192.0.2.25:58494     ESTABLISHED 1783650/java
    
```

- Send port (ZMQ PUSH queue ports):(ldapservers.zmq.send.port) 20200, 30000

```

netstat -apn | grep 1783650 | egrep "20200|30000"
tcp6      0      0 192.0.2.2:20200      :::*                  LISTEN       1783650/java
tcp6      0      0 192.0.2.2:30000      :::*                  LISTEN       1783650/java
tcp6      0      0 192.0.2.2:30000      192.0.2.37:59068     ESTABLISHED 1783650/java
tcp6      0      0 192.0.2.2:30000      192.0.2.19:58480     ESTABLISHED 1783650/java
tcp6      0      0 192.0.2.2:20200      192.0.2.24:53250     ESTABLISHED 1783650/java
tcp6      0      0 192.0.2.2:30000      192.0.2.23:50314     ESTABLISHED 1783650/java
tcp6      0      0 192.0.2.2:30000      192.0.2.20:48746     ESTABLISHED 1783650/java
tcp6      0      0 192.0.2.2:20200      192.0.2.35:46894     ESTABLISHED 1783650/java
tcp6      0      0 192.0.2.2:20200      192.0.2.26:59348     ESTABLISHED 1783650/java
tcp6      0      0 192.0.2.2:30000      192.0.2.28:59448     ESTABLISHED 1783650/java
tcp6      0      0 192.0.2.2:20200      192.0.2.19:45048     ESTABLISHED 1783650/java
    
```

### Successful Message Handling

The following snapshot shows filtered packets for a successful CCR/CCA message handling done for PD-3. Packet capture was taken using tcpdump on all Ethernet interfaces of active policy director (LB).



**Note** The port numbers provided in this section are an example and can differ based on the network deployment. For more information on port numbers, refer to [Table 2: Policy Director/Policy Server Listening Ports, on page 32.](#)

**Figure 6: Filtered Packet**

25	0.041183	DIAMETE	548	cmd=Credit-Control	Request(272)	Flags=RP--	app1=3GPP	Gx(16777238)	h2h=2a81e43f	e2e=54f2e3c3	
26	0.041236	TCP	548	36150-3870	[PSH, ACK]	Seq=1	Ack=1	win=3074	Len=480	TSval=948470183	TSecr=948470118
27	0.041594	TCP	984	50003-45025	[PSH, ACK]	Seq=1	Ack=1	win=23	Len=916	TSval=948470183	TSecr=866857036
28	0.041753	TCP	68	45025-50003	[ACK]	Seq=1	Ack=917	win=251	Len=0	TSval=866858866	TSecr=948470183
34	0.052556	DIAMETE	336	cmd=Credit-Control	Answer(272)	Flags=P---	app1=3GPP	Gx(16777238)	h2h=33e0e0ff	e2e=4312687d	
35	0.052568	TCP	68	41824-3868	[ACK]	Seq=573	Ack=269	win=6165	Len=0	TSval=948470194	TSecr=876673754
36	0.052612	DIAMETE	336	cmd=Credit-Control	Answer(272)	Flags=P---	app1=3GPP	Gx(16777238)	h2h=33e0e0ff	e2e=4312687d	
37	0.052619	DIAMETE	160	cmd=Device-watchdog	Answer(280)	Flags=----	app1=Diameter	Common Messages(0)	h2h=6bdda3c9	e2e=9216606	
38	0.052632	TCP	68	3868-48990	[ACK]	Seq=77	Ack=93	win=27	Len=0	TSval=948470194	TSecr=15319689
39	0.052689	TCP	160	33484-3870	[PSH, ACK]	Seq=1	Ack=77	win=133	Len=92	TSval=948470195	TSecr=948470179
40	0.052722	TCP	68	3870-33484	[ACK]	Seq=77	Ack=93	win=133	Len=0	TSval=948470195	TSecr=948470195
49	0.068600	DIAMETE	144	cmd=Device-watchdog	Request(280)	Flags=RP--	app1=Diameter	Common Messages(0)	h2h=6be1bde	e2e=1500016b	
74	0.079247	TCP	659	43422-51003	[PSH, ACK]	Seq=1	Ack=1	win=23	Len=591	TSval=866858904	TSecr=948468391
75	0.079255	TCP	68	51003-43422	[ACK]	Seq=1	Ack=592	win=251	Len=0	TSval=948470221	TSecr=866858904
76	0.079551	TCP	292	3870-36150	[PSH, ACK]	Seq=1	Ack=481	win=193	Len=224	TSval=948470221	TSecr=948470183
77	0.079589	TCP	68	36150-3870	[ACK]	Seq=481	Ack=225	win=3074	Len=0	TSval=948470221	TSecr=948470221
78	0.079622	DIAMETE	292	cmd=Credit-Control	Answer(272)	Flags=P---	app1=3GPP	Gx(16777238)	h2h=2a81e43f	e2e=54f2e3c3	

Packet Details

1. Packet#25 CCR message from PCEF to lbvip

**Figure 7: PCEF to lbvip CCR Message**

AVP Flags: 0x40
AVP Length: 63
Session-Id: GatewayService-3-14-0.44RDSAEGW01;1429527270;1711585016

2. Packet#26 CCR message sent to HaProxy port 3870 of PD-3

**Figure 8: CCR Message to HaProxy**

0060	47	61	74	65	77	61	79	53	65	72	76	69	63	65	2d	33	Gateways	ervice-3
0070	2d	31	34	2d	30	2e	34	34	52	44	53	41	45	47	57	30	-14-0.44	RDSAEGW0
0080	31	3b	31	34	32	39	35	32	37	32	37	30	3b	31	37	31	1;142952	7270;171
0090	31	35	38	35	30	31	36	00	00	00	01	02	40	00	00	00	1585016.	...@...
00a0	01	00	00	16	00	00	01	08	40	00	00	29	47	61	74	65	.....@..	Gate
00b0	77	61	79	53	65	72	76	69	63	65	2d	33	2d	31	34	2d	wayServi	ce-3-14-
00c0	30	2e	34	34	52	44	53	41	45	47	57	30	31	00	00	00	0.44RDSA	EGW01...

3. Packet#27 PD-3 sends message to policy server (QNS) VM by adding message to PUSH Queue port 50003

**Figure 9: PD-3 Message**

0100	81	00	74	00	31	00	53	00	42	00	39	00	00	01	12	00	U.._..S.	B.Y.....
01e0	00	00	04	07	01	00	00	00	00	00	00	37	00	00	00	47	.....	...7...G
01f0	61	74	65	77	61	79	53	65	72	76	69	63	65	2d	33	2d	atewaySe	rvice-3-
0200	31	34	2d	30	2e	34	34	52	44	53	41	45	47	57	30	31	14-0.44R	DSAEGW01
0210	3b	31	34	32	39	35	32	37	32	37	30	3b	31	37	31	31	;1429527	270;1711
0220	35	38	35	30	31	36	02	02	01	00	00	00	00	00	00	16	585016..	.....

4. Packet#74 policy server (QNS) VM sends response back to PD-3 on PULL Queue port 51003

**Figure 10: Policy Server (QNS) VM Response**

0110	47	00	61	00	74	00	65	00	77	00	61	00	79	00	53	00	G.a.t.e.	w.a.y.S.
0120	65	00	72	00	76	00	69	00	63	00	65	00	2d	00	33	00	e.r.v.i.	c.e.-.3.
0130	2d	00	31	00	34	00	2d	00	30	00	2e	00	34	00	34	00	-1.4.-.	0...4.4.
0140	52	00	44	00	53	00	41	00	45	00	47	00	57	00	30	00	R.D.S.A.	E.G.W.0.
0150	31	00	3b	00	31	00	34	00	32	00	39	00	35	00	32	00	1.;1.4.	2.9.5.2.
0160	37	00	32	00	37	00	30	00	3b	00	31	00	37	00	31	00	7.2.7.0.	;1.7.1.
0170	31	00	35	00	38	00	35	00	30	00	31	00	36	00	00	00	1.5.8.5.	0.1.6...

5. Packet#76 PD-3 sends CCA message to HaProxy port 3870

Figure 11: PD-3 Message

0050	2a 81 e4 3f 54 f2 e3 c3	00 00 01 07 40 00 00 3f	*..?T... ..@..?
0060	47 61 74 65 77 61 79 53	65 72 76 69 63 65 2d 33	GatewayService-3
0070	2d 31 34 2d 30 2e 34 34	52 44 53 41 45 47 57 30	-14-0.44 RDSAEGW0
0080	31 3b 31 34 32 39 35 32	37 32 37 30 3b 31 37 31	1;142952 7270;171
0090	31 35 38 35 30 31 36 00	00 00 01 a0 40 00 00 0c	1585016. ....@..
00a0	00 00 00 02 00 00 01 9f	40 00 00 0c 00 00 00 53	@

6. Packet#78 CCA sent to PCEF

Figure 12: CCA Message

AVP Flags:	0x40
AVP Length:	63
Session-Id:	GatewayService-3-14-0.44RDSAEGW01;1429527270;1711585016

All the above packets are co-related based on the “Diameter Session-Id” found in the Wireshark hex/bytes “ascii character” details as shown above.

Wireshark Filters for capturing messages between PCEF, lbvip, Policy Director and Policy Server (QNS) when tcpdump taken on all Ethernet interfaces of active policy director (LB):

- Filter PD-1 ---> “tcp.srport == 3868 || tcp.dstport == 3868 || tcp.srport == 50001 || tcp.dstport == 50001 || tcp.dstport == 3868 || tcp.srport == 3868 || tcp.srport == 51001 || tcp.dstport == 51001”
- Filter PB-2 ---> “tcp.srport == 3869 || tcp.dstport == 3869 || tcp.srport == 50002 || tcp.dstport == 50002 || tcp.dstport == 3868 || tcp.srport == 3868 || tcp.srport == 51002 || tcp.dstport == 51002”
- Filter PD-3 ---> “tcp.srport == 3870 || tcp.dstport == 3870 || tcp.srport == 50003 || tcp.dstport == 50003 || tcp.dstport == 3868 || tcp.srport == 3868 || tcp.srport == 51003 || tcp.dstport == 51003”

Message Drops at Diameter Interface

Based on the zabbix graphs (an example) if there are messages failing over to secondary then tcpdump taken on primary site active policy director (LB) VM should show the diameter messages for which no response was sent to PCEF. On a sample tcpdump we can apply following filter to check the number of messages dropped and find the list of corresponding peers

Filter in Wireshark - “!(diameter.answer\_in ) && !(diameter.answer\_to ) && diameter”

Figure 13: Message Drops

No.	Time	Source	Destination	Protocol	Length	Info
1906	2.939127			DIAMETE	858	cmd=Credit-Control Request (272) flags=RP-- app1=3GPP Gx(16777238) h2h=4a927b7d e2e=165e017
2102	2.163304			DIAMETE	850	cmd=Credit-Control Request (272) flags=RP-- app1=3GPP Gx(16777238) h2h=51403c4d e2e=a51d2b47
2278	2.358043			DIAMETE	858	cmd=Credit-Control Request (272) flags=RP-- app1=3GPP Gx(16777238) h2h=a9276dc e2e=3d114172
2467	2.580195			DIAMETE	850	cmd=Credit-Control Request (272) flags=RP-- app1=3GPP Gx(16777238) h2h=3b46ef43 e2e=60d376c2
2539	2.648706			DIAMETE	858	cmd=Credit-Control Request (272) flags=RP-- app1=3GPP Gx(16777238) h2h=20b67e92 e2e=6cb3b11b
2563	2.673613			DIAMETE	858	cmd=Credit-Control Request (272) flags=RP-- app1=3GPP Gx(16777238) h2h=b671ad e2e=5251fd87
2570	2.678431			DIAMETE	850	cmd=Credit-Control Request (272) flags=RP-- app1=3GPP Gx(16777238) h2h=2b6e09b6 e2e=5e030d02
2601	2.710351			DIAMETE	858	cmd=Credit-Control Request (272) flags=RP-- app1=3GPP Gx(16777238) h2h=2a927f32 e2e=7ea26928
2746	2.860198			DIAMETE	850	cmd=Credit-Control Request (272) flags=RP-- app1=3GPP Gx(16777238) h2h=912be0e e2e=63969aca
2853	2.975505			DIAMETE	858	cmd=Credit-Control Request (272) flags=RP-- app1=3GPP Gx(16777238) h2h=b671ae e2e=5251fd97
2934	3.045868			DIAMETE	858	cmd=Credit-Control Request (272) flags=RP-- app1=3GPP Gx(16777238) h2h=2a927ea6 e2e=4842e16e
2973	3.098941			DIAMETE	858	cmd=Credit-Control Request (272) flags=RP-- app1=3GPP Gx(16777238) h2h=194e8089 e2e=5800f8aa
3006	3.135687			DIAMETE	858	cmd=Credit-Control Request (272) flags=RP-- app1=3GPP Gx(16777238) h2h=20b67e93 e2e=6cb3b12d
3041	3.159617			DIAMETE	858	cmd=Credit-Control Request (272) flags=RP-- app1=3GPP Gx(16777238) h2h=2a927f33 e2e=7ea26837

Now filtered packets can be checked to find the number of packets dropped for each peer connections. All the packets dropped should be for a given list of Peers which are currently not being processed at primary-site.

### Message Dropped between Policy Director (LB)<->Policy Server (QNS)

The next step is to identify the PolicyDirector instance where these messages are being dropped.

1. top command output on active policy director (lb) should show that the PD instance not using any CPU as there are no messages being processed on the process-id, note the PD-instance.
2. Start a tcpdump on all Ethernet interfaces of the policy director VM which should contain all packets sent between lbvip, policy director instance and policy server (QNS) VMs. This tcpdump will also contain the requests which do not have any response from PCRF, so apply the filter “(!diameter.answer\_in) && !(diameter.answer\_to) && diameter” in wireshark and note a single request which was not processed.
3. This packet should be then forwarded to PD-instance HaProxy port.

**Figure 14: Forwarded Packets**

5499 1.786940	DIAMETE	860 cmd=Credit-control Request(272) flags=RF-- app=3GPP gx(16777238) h2h=b671ad eze=5251fd87
6500 1.786952	TCP	68 3868-52867 [ACK] Seq=1 Ack=1585 win=251 Len=0 TSval=244874711 TSecr=384075740
6501 1.787018	TCP	860 57817-3869 [PSH, ACK] Seq=793 Ack=1 win=1537 Len=792 TSval=244874711 TSecr=244874686

Packet 6499 CCR-I request from PCEF was not answered and the message is forwarded to HaProxy port 3869 which is PD-2 instance in packet 6501 but no subsequent forwarding to policy server (QNS) VMs occurred. Hence PD-2 was not processing and forwarding any requests from PCEF to policy server (QNS) VMs. Similarly, this can be verified for other filtered packets as identified in Step 2 above.

In such cases, your Cisco Technical Representative can be contacted to further diagnose the issue and find the cause for message drops at PD level. Similarly, above analysis can be applied to identify messages dropped at policy server (QNS) level if packets are forwarded from PD to policy server (QNS) on PUSH queue but no response from policy server (QNS) VM on PULL queue found.

### Recovering Hung Peers

Based on the above diagnosis from tcpdump and top command messages were dropped at the PD-2 instance. This caused all traffic for peers connected to this PD-2 instance to failover to secondary-site LoadBalancers as shown in Zabbix graphs. In order to recover from this situation the LoadBalancer processes should be restarted as follows:

1. Login to the active policy director (lb) of primary-site and execute the following:

```
monit status qnsXX
service heartbeat status
service monit status
```

2. Stop the services.

```
service heartbeat stop
service monit stop
monit stop qnsXX
```

3. Start the policy server (QNS) service and check its status.

```
monit start qnsXX
monit status qnsXX
```

4. Start the monit and heartbeat service.

```
service monit start
service heartbeat start
```

5. Repeat Step 1, on page 37 to Step 4, on page 37 on newly active policy director (lb).
6. Verify from Zabbix graphs or similar graphs that traffic has stopped failing over to secondary-site.

7. Take a tcpdump on all Ethernet interfaces of active policy director (lb) and verify that all the three Policy Directors are sending/receiving messages from policy server (QNS) instances as explained in [Successful Message Handling, on page 34](#).

## Diameter Result Codes and Scenarios

The following table describes some common diameter result codes and scenarios:

**Table 3: Common Diameter Result Codes and Scenarios**

Code	Name	CPS Scenarios
2001	DIAMETER_SUCCESS	Everything went well and Request processed successfully.
2002	DIAMETER_LIMITED_SUCCESS	The Request was successfully completed, but additional processing is required by the application in order to provide service to the user.
3001	DIAMETER_COMMAND_UNSUPPORTED	The Request contained a Command-Code that the receiver did not recognize or support. This MUST be used when a Diameter node receives an experimental command that it does not understand.
3002	DIAMETER_UNABLE_TO_DELIVER	Message cannot be delivered, either because no host within the realm supporting the required application was available to process the request or because Destination-Host AVP was given without the associated Destination-Realm AVP.
3003	DIAMETER_REALM_NOT_SERVED	The intended realm of the request is not recognized.
3004	DIAMETER_TOO_BUSY	Message got discarded by the overload handling mechanism. Note: CPS 7.5 adds the option to silently discard instead of sending DIAMETER_TOO_BUSY as discarding is often a better way to have other node back off instead of immediately resending the request in an overload scenario.
3005	DIAMETER_LOOP_DETECTED	An agent detected a loop while trying to get the message to the intended recipient. The message MAY be sent to an alternate peer, if one is available, but the peer reporting the error has identified a configuration problem.

Code	Name	CPS Scenarios
3006	DIAMETER_REDIRECT_INDICATION	A redirect agent has determined that the request could not be satisfied locally and the initiator of the request should direct the request directly to the server, whose contact information has been added to the response. When set, the Redirect-Host AVP MUST be present.
3007	DIAMETER_APPLICATION_UNSUPPORTED	A request was sent for an application that is not supported.
3008	DIAMETER_INVALID_HDR_BITS	A request was received whose bits in the Diameter header were either set to an invalid combination, or to a value that is inconsistent with the command code's definition.
3009	DIAMETER_INVALID_AVP_BITS	A request was received that included an AVP whose flag bits are set to an unrecognized value, or that is inconsistent with the AVP's definition.
3010	DIAMETER_UNKNOWN_PEER	A CER was received from an unknown peer.
4001	DIAMETER_AUTHENTICATION_REJECTED	The authentication process for the user failed, most likely due to an invalid password used by the user. Further attempts MUST only be tried after prompting the user for a new password.
4002	DIAMETER_OUT_OF_SPACE	A Diameter node received the accounting request but was unable to commit it to stable storage due to a temporary lack of space.
4003	ELECTION_LOST	The peer has determined that it has lost the election process and has therefore disconnected the transport connection.
4010	DIAMETER_END_USER_SERVICE_DENIED	The credit-control server denies the service request due to service restrictions. If the CCR contained used-service-units they are deducted, if possible.
4011	DIAMETER_CREDIT_CONTROL_NOT_APPLICABLE	The credit-control server determines that the service can be granted to the end user but no further credit-control is needed for the service (eg, service is free of charge).

Code	Name	CPS Scenarios
4012	DIAMETER_CREDIT_LIMIT_REACHED	The credit-control server denies the service request since the end-user's account could not cover the requested service. If the CCR contained used-service-units they are deducted, if possible.
4141	DIAMETER_PCC_BEARER_EVENT	When for some reason a PCC rule cannot be enforced or modified successfully in a network initiated procedure. The reason is provided in the Event Trigger AVP value.
4241	DIAMETER_ERROR_NO_AVAILABLE_POLICY_COUNTERS	Error used by the OCS to indicate to the PCRF that the OCS has no available policy counters for the subscriber.
5001	DIAMETER_AVP_UNSUPPORTED	The peer received a message that contained an AVP that is not recognized or supported and was marked with the Mandatory bit. A Diameter message with this error MUST contain one or more Failed-AVP AVP containing the AVPs that caused the failure.
5002	DIAMETER_UNKNOWN_SESSION_ID	The request contained an unknown Session-Id.
5003	DIAMETER_AUTHORIZATION_REJECTED	A request was received for which the user could not be authorized. No session created due to various reasons. For example, this error could occur if the service requested is not permitted to the user.
5004	DIAMETER_INVALID_AVP_VALUE	The request contained an AVP with an invalid value in its data portion. A Diameter message indicating this error MUST include the offending AVPs within a Failed-AVP AVP.



Code	Name	CPS Scenarios
5005	DIAMETER_MISSING_AVP	The request did not contain an AVP that is required by the Command Code definition. If this value is sent in the Result-Code AVP, a Failed-AVP AVP SHOULD be included in the message. The Failed-AVP AVP MUST contain an example of the missing AVP complete with the Vendor-Id if applicable. The value field of the missing AVP should be of correct minimum length and contain zeroes.
5006	DIAMETER_RESOURCES_EXCEEDED	A request was received that cannot be authorized because the user has already expended allowed resources. An example of this error condition is a user that is restricted to one dial-up PPP port, attempts to establish a second PPP connection.
5007	DIAMETER_CONTRADICTING_AVPS	The Home Diameter server has detected AVPs in the request that contradicted each other, and is not willing to provide service to the user. One or more Failed-AVP AVPs MUST be present, containing the AVPs that contradicted each other.
5008	DIAMETER_AVP_NOT_ALLOWED	A message was received with an AVP that MUST NOT be present. The Failed-AVP AVP MUST be included and contain a copy of the offending AVP.
5009	DIAMETER_AVP_OCCURS_TOO_MANY_TIMES	A message was received that included an AVP that appeared more often than permitted in the message definition. The Failed-AVP AVP MUST be included and contain a copy of the first instance of the offending AVP that exceeded the maximum number of occurrences
5010	DIAMETER_NO_COMMON_APPLICATION	When a CER message is received, and there are no common applications supported between the peers.
5011	DIAMETER_UNSUPPORTED_VERSION	A request was received, whose version number is unsupported.
5012	DIAMETER_UNABLE_TO_COMPLY	Message rejected as something else that went wrong and there's no specific reason.

Code	Name	CPS Scenarios
5013	DIAMETER_INVALID_BIT_IN_HEADER	An unrecognized bit in the Diameter header is set to one (1).
5014	DIAMETER_INVALID_AVP_LENGTH	The request contained an AVP with an invalid length. A Diameter message indicating this error MUST include the offending AVPs within a Failed-AVP AVP.
5015	DIAMETER_INVALID_MESSAGE_LENGTH	A request is received with an invalid message length.
5016	DIAMETER_INVALID_AVP_BIT_COMBO	The request contained an AVP with which is not allowed to have the given value in the AVP Flags field. A Diameter message indicating this error MUST include the offending AVPs within a Failed-AVP AVP.
5017	DIAMETER_NO_COMMON_SECURITY	A CER message is received, and there are no common security mechanisms supported between the peers. A Capabilities-Exchange-Answer (CEA) MUST be returned with the Result-Code AVP set to DIAMETER_NO_COMMON_SECURITY.
5030	DIAMETER_USER_UNKNOWN	The subscriber was not found in SPR.
5031	DIAMETER_RATING_FAILED	Informs the credit-control client that the credit-control server cannot rate the service request due to insufficient rating input, incorrect AVP combination or due to an AVP or an AVP value that is not recognized or supported in the rating.
5141	DIAMETER_ERROR_TRIGGER_EVENT	When the set of bearer/session information sent in a CCR originated due to a trigger event been met is incoherent with the previous set of bearer/session information for the same bearer/session.
5142	DIAMETER_PCC_RULE_EVENT	When for some reason the PCC rules cannot be installed/activated. The reason is provided in the Event Trigger AVP value.

Code	Name	CPS Scenarios
5143	DIAMETER_ERROR_BEARER_NOT_AUTHORIZED	Emergency service related - Used when the PCRF cannot authorize an IP-CAN bearer upon the reception of an IP-CAN bearer authorization request coming from the PCEF.
5144	DIAMETER_ERROR_TRAFFIC_MAPPING_INFO_REJECTED	Emergency service related - Used when the PCRF does not accept one or more of the traffic mapping filters.
5570	DIAMETER_ERROR_UNKNOWN_POLICY_COUNTERS	Error used by the OCS to indicate to the PCRF that the OCS does not recognize one or more Policy Counters specified in the request, when the OCS is configured to reject the request provided with unknown policy counter identifier(s).

## Diameter Experimental Result Codes

The following table describes some common Diameter experimental result codes and scenarios:

**Table 4: Common Diameter Experimental Result Codes**

Code	Name	CPS Scenarios
2001	DIAMETER_FIRST_REGISTRATION	The HSS informs the I-CSCF that: - The user is authorized to register this public identity; - A S-CSCF shall be assigned to the user.
2002	DIAMETER_SUBSEQUENT_REGISTRATION	The HSS informs the I-CSCF that: - The user is authorized to register this public identity; - A S-CSCF is already assigned and there is no need to select a new one.
2003	DIAMETER_UNREGISTERED_SERVICE	The HSS informs the I-CSCF that: - The public identity is not registered but has services related to unregistered state; - A S-CSCF shall be assigned to the user.
2004	DIAMETER_SUCCESS_SERVER_NAME_NOT_STORED	The HSS informs to the S-CSCF that: - The de-registration is completed; - The S-CSCF name is not stored in the HSS.
4100	DIAMETER_USER_DATA_NOT_AVAILABLE	The requested user data is not available at this time to satisfy the requested operation.

Code	Name	CPS Scenarios
4101	DIAMETER_PRIOR_UPDATE_IN_PROGRESS	The request to update the repository data at the HSS could not be completed because the related repository data is currently being updated by another entity.
4143	DIAMETER_AN_GW_FAILED	The policy decisions (i.e. installation/modification of PCC rules or provisioning of policy decisions not related to a PCC rule) received within a RAR initiated by the PCRF cannot be enforced by the PCEF because the AN-Gateway has failed. If one or more PCC Rules are affected, these PCC Rules will be provided in the Charging-Rule-Report AVP including the Rule-Failure-Code AVP set to AN_GW_FAILED (17), and PCC-Rule-Status AVP set to INACTIVE as described in Clause 4.5.12. Applicable only to 3GPP-EPS.
4144	TGPP_DIAMETER_PENDING_TRANSACTION	A node that supports the PendingTransaction feature receives an incoming request on a session while it has an ongoing transaction on the same session and cannot handle the request as described in Clause 8.2 of 3GPP TS 29.213 [8].
4196	DIAMETER_REQUESTED_SESSION_NOT_FOUND	Returned by PCEF when it doesn't find the session info for the requested session in SDR.
4197	DIAMETER_SESSION_RECOVERY_REQUESTED	
4198	DIAMETER_PENDING_TRANSACTION	The PCRF expects a response to a pending request that it initiated. The PCRF can also retry the request message if needed.
5001	DIAMETER_ERROR_USER_UNKNOWN	Message was received for a user or a wildcarded identity that is unknown.
5002	DIAMETER_ERROR_IDENTITIES_DONT_MATCH	Message was received with a public identity and a private identity for a user, and server determines that the public identity does not correspond to the private identity.

Code	Name	CPS Scenarios
5003	DIAMETER_ERROR_IDENTITY_NOT_REGISTERED	A query for location information is received for a public identity that has not been registered before. The user to which this identity belongs cannot be given service in this situation.
5004	DIAMETER_ERROR_ROAMING_NOT_ALLOWED	User is not allowed to roam in the visited network.
5005	DIAMETER_ERROR_IDENTITY_ALREADY_REGISTERED	Identity has already a server assigned and the registration status does not allow that it is overwritten.
5006	DIAMETER_ERROR_AUTH_SCHEME_NOT_SUPPORTED	Authentication scheme in an authentication request is not supported.
5007	DIAMETER_ERROR_IN_ASSIGNMENT_TYPE	Identity being registered has already the same server assigned and the registration status does not allow the server assignment type or the Public Identity type received in the request is not allowed for the indicated server-assignment-type.
5008	DIAMETER_ERROR_TOO_MUCH_DATA	Volume of the data pushed to the receiving entity exceeds its capacity.
5009	DIAMETER_ERROR_NOT_SUPPORTED_USER_DATA	The S-CSCF informs HSS that the received subscription data contained information which was not recognised/supported
5011	DIAMETER_ERROR_FEATURE_UNSUPPORTED	A request application message was received indicating that the origin host requests that the command pair would be handled using a feature which is not supported by the destination host.
5012	DIAMETER_ERROR_SERVING_NODE_FEATURE_UNSUPPORTED	The HSS supports the P-CSCF-Restoration-mechanism feature, but none of the user serving node(s) supports it.
5061	INVALID_SERVICE_INFORMATION	PCRF rejects new or modified service information the service information provided by the AF is invalid /insufficient for the server to perform the requested action.
5062	FILTER_RESTRICTIONS	PCRF rejects new or modified service information because the Flow-Description AVPs cannot be handled by the server.

Code	Name	CPS Scenarios
5063	REQUESTED_SERVICE_NOT_AUTHORIZED	PCRF rejects new or modified service information because the requested service is not consistent with the related subscription information /operator defined policy rules and/or the supported features in the IP-CAN network.
5064	DUPLICATED_AF_SESSION	PCRF rejects a new Rx session setup because the new Rx session relates to an AF session with another related active Rx session.
5065	IP_CAN_SESSION_NOT_AVAILABLE	PCRF rejects a new Rx session setup when it fails to associate the described service IP flows within the session information received from the AF to an existing IP-CAN session.
5066	UNAUTHORIZED_NON_EMERGENCY_SESSION	PCRF rejects new Rx session setup because the session binding function associated a non-Emergency IMS session to an IP-CAN session established to an Emergency APN.
5067	UNAUTHORIZED_SPONSORED_DATA_CONNECTIVITY	The PCRF rejects a new Rx session setup because the PCRF can't authorize the sponsored data connectivity based on the sponsored data connectivity profile or the operator policy.
5068	TEMPORARY_NETWORK_FAILURE	
5100	DIAMETER_ERROR_USER_DATA_NOT_RECOGNIZED	The data received by the AS is not supported or recognized.
5101	DIAMETER_ERROR_OPERATION_NOT_ALLOWED	The requested operation is not allowed for the user.
5102	DIAMETER_ERROR_USER_DATA_CANNOT_BE_READ	The requested user data is not allowed to be read.
5103	DIAMETER_ERROR_USER_DATA_CANNOT_BE_MODIFIED	The requested user data is not allowed to be modified.
5104	DIAMETER_ERROR_USER_DATA_CANNOT_BE_NOTIFIED	The requested user data is not allowed to be notified on changes

Code	Name	CPS Scenarios
5105	DIAMETER_ERROR_TRANSPARENT_DATA_OUT_OF_SYNC	The request to update the repository data at the HSS could not be completed because the requested update is based on an out-of-date version of the repository data. That is, the sequence number in the Sh-Update Request message, does not match with the immediate successor of the associated sequence number stored for that repository data at the HSS. It is also used where an AS tries to create a new set of repository data when the identified repository data already exists in the HSS.
5106	DIAMETER_ERROR_SUBS_DATA_ABSENT	The Application Server requested to subscribe to changes to Repository Data that is not present in the HSS.
5107	DIAMETER_ERROR_NO_SUBSCRIPTION_TO_DATA	The AS received a notification of changes of some information to which it is not subscribed
5108	DIAMETER_ERROR_DSAI_NOT_AVAILABLE	The Application Server addressed a DSAI not configured in the HSS.
5140	DIAMETER_ERROR_INITIAL_PARAMETERS	Used when the set of bearer or session or subscriber information needed by the PCRF for rule selection is incomplete/erroneous/not available for the decision to be made.
5141	DIAMETER_ERROR_TRIGGER_EVENT	The set of bearer/session information sent in a CCR originated due to a trigger event been met is incoherent with the previous set of bearer/session information for the same bearer/session. (E.g. event trigger met was RAT changed, and the RAT notified is the same as before)
5142	DIAMETER_PCC_RULE_EVENT	The PCC rules cannot be installed/activated. Affected PCC-Rules will be provided in the Charging-Rule-Report AVP including the reason and status as described in Clause 4.5.12. Absence of the Charging-Rule-Report means that all provided PCC rules for that specific bearer/session are affected.

Code	Name	CPS Scenarios
5143	DIAMETER_ERROR_BEARER_NOT_AUTHORIZED	The PCRF cannot authorize an IP-CAN bearer (e.g. the authorized QoS would exceed the subscribed QoS) upon the reception of an IP-CAN bearer authorization request coming from the PCEF. The affected IP-CAN bearer is the one that triggered the corresponding CCR. The PCEF shall reject the attempt to initiate or modify the bearer indicated in the related CCR command.
5144	DIAMETER_ERROR_TRAFFIC_MAPPING_INFO_REJECTED	The PCRF does not accept one or more of the traffic mapping filters (e.g. TFT filters for GPRS) provided by the PCEF in a CC Request.
5147	DIAMETER_ERROR_CONFLICTING_REQUEST	The PCRF cannot accept the UE-initiated resource request as a network-initiated resource allocation is already in progress that has packet filters that cover the packet filters in the received UE-initiated resource request. The PCEF shall reject the attempt for UE-initiated resource request.
5148	DIAMETER_ADC_RULE_EVENT	The ADC rules cannot be installed/activated. Affected ADC Rules shall be provided in the ADC-Rule-Report AVP including the reason and status as described in Clause 5b.3.6. Absence of the ADC-Rule-Report means that all provided ADC rules for that IP-CAN session are affected.
5199	DIAMETER_NEWER_SESSION_DETECTED	Received in the authentication response message. This result code is introduced to detect stale message requests and support session uniqueness.

## Frequently Encountered Scenarios

### Subscriber not Mapped on SCE

This issue was causing the subscriber to get no mapping on the SCE.

---

**Step 1** Write an awk script to perform the following grep to create a text file of over 1000 instances of this message:



```
grep "No member in system" policy.log* >
```

```
no_member_found.txt
```

This grep resulted in a file with these lines:

```
policy.log:2009-07-17 11:00:21,201 INFO
wikiimport:com.broadhop.sme.business.network.accounting.Ne
tworkAccountingUtil No member in system for d162818
policy.log:2009-07-17 11:02:06,108 INFO
wikiimport:com.broadhop.sme.business.network.accounting.Ne
tworkAccountingUtil No member in system for D02625
policy.log.1:2009-07-17 09:25:29,036 INFO
wikiimport:com.broadhop.sme.business.network.accounting.Ne
tworkAccountingUtil No member in system for D162346
policy.log.1:2009-07-17 09:27:28,718 INFO
wikiimport:com.broadhop.sme.business.network.accounting.Ne
tworkAccountingUtil No member in system for d162365
policy.log.1:2009-07-17 09:27:37,193 INFO
wikiimport:com.broadhop.sme.business.network.accounting.Ne
tworkAccountingUtil No member in system for d162365
policy.log.1:2009-07-17 09:27:42,257 INFO
wikiimport:com.broadhop.sme.business.network.accounting.Ne
tworkAccountingUtil No member in system for d162365
policy.log.1:2009-07-17 09:38:09,010 INFO
wikiimport:com.broadhop.sme.business.network.accounting.Ne
tworkAccountingUtil No member in system for d02116
policy.log.1:2009-07-17 09:38:12,618 INFO
wikiimport:com.broadhop.sme.business.network.accounting.Ne
tworkAccountingUtil No member in system for D163647
policy.log.1:2009-07-17 09:40:42,751 INFO
wikiimport:com.broadhop.sme.business.network.accounting.Ne
tworkAccountingUtil No member in system for d102096
```

**Step 2** Then use the following awk script to generate a new file that only has the user name. The script says print the 10th field:

```
awk '{print $10}' no_member_found.txt >
```

```
no_member_found_usernames_with_dupes.txt
```

**Step 3** Run the following command to remove duplicates:

```
sort no_member_found_usernames_with_dupes.txt | uniq >
```

```
uniq_sorted_no_member_found_usernames.txt
```

This resulted in a file with usernames only:

```
D00059
D00077
D001088
D00112
d001313
D00145
D001452
d00156
D00186
```

d00198  
D00200  
d00224

## CPS Server Will Not Start and Nothing is in the Log

If the CPS server does not start (or starts and immediately crashes) and no errors appear in `/var/log/broadhop/qns.log` file to give reasons it did not start check the following list

1. Check `/var/log/broadhop/service-qns-1.log` file.
2. Check `/etc/broadhop/servers`.
  - There should be an entry in this file for the current host name (Type 'hostname' in the console window to find the local hostname)
  - There must be directory that corresponds to the hostname entry with config files. That is if the servers file has `svn01=controlcenter` there must be a `/etc/broadhop/controlcenter` directory.
3. Attempt to start the server directly from the command line and look for errors.
  - Type: `/opt/broadhop/qns/bin/qns.sh`
  - The server should start up successfully and the command line should not return. If the command prompt returns then the server did not start successfully.
  - Look for any errors displayed in the console output.
4. Look for OSGi errors.
  - Look in `/opt/broadhop/qns/configuration` for a log file. If any exist examine the log file for error messages.

## Server returned HTTP Response Code: 401 for URL

A 401 type error means you're not logging in to SVN with proper credentials.

The server won't start and the following appears in the log:

```
2010-12-10 01:05:26,668 \[SpringOsgiExtenderThread-8\]
ERROR c.b.runtime.impl.RuntimeLoader - There was an error
initializing reference data\!
java.io.IOException: Server returned HTTP response code:
401 for URL: http://lbvip01/repos/run/config.properties
sun.net.www.protocol.http.HttpURLConnection.getInputStream
(HttpURLConnection.java:1313) \~\[na:1.6.0_20\]
org.springframework.core.io.UrlResource.getInputStream(Url
Resource.java:124) \~\[org.springframework.core_3.0.0.REL
```

To fix this error:

- Edit `/etc/broadhop/qns.conf`

- Ensure that the configuration URL and repository credentials hostnames match.

```
\-Dcom.broadhop.config.url=http://lbvip01/repos/run/
\Dcom.broadhop.repository.credentials=broadhop/
broadhop@lbvip01
```

## com.broadhop.exception.BroadhopException Unable to Find System Configuration for System

Symptoms server won't stay started and the log displays this:

```
com.broadhop.exception.BroadhopException: Unable to find system configuration for system:
The system that is set up in your Quantum Policy Builder (and cluster name) must match the
one
specified in /etc/broadhop/qns.conf. Either add or change this via the Quantum Policy Builder
interface, and then publish or update the system/clustername in /etc/broadhop/qns.conf
\Dcom.broadhop.run.systemId=poc-system
\Dcom.broadhop.run.clusterId=cluster-1
```

## Log Files Display the Wrong Time but the Linux Time is Correct

If log files or other dates are showing in the incorrect time zone despite the Linux time being set to the proper time zone, most likely the time zone that the JVM reads is incorrect.

**Step 1** In `/etc/sysconfig`, run the command `cat clock` to see this output:

```
ZONE="America/Denver"
UTC=false
ARC=false
```

**Step 2** Change the ZONE line to the time zone you desire, for instance you could change it to:

```
ZONE="Asia/Singapore"
UTC=false
ARC=false
```

to change the JVM time zone to Singapore time.

The value for ZONE is driven by the directories in `/usr/share/zoneinfo`

## REST Web Service Queries Returns an Empty XML Response for an Existing User

For example:

```
<subscriberProfile><content/></subscriberProfile>
```

Because there are multiple ways needed to return web service data, the BroadHop Web Service Blueprint doesn't return any XML by default. To fix this issue, configure the 'Default Web Service Query Response' blueprint under the 'BroadHop Web Services' Blueprint.

## Error in Datastore: "err": "E11000 Duplicate Key Error Index

Here mongo database has been used an example. The same steps can be replicated for all the databases.




---

**Note** This removes all the sessions.

---

Typically, duplicate keys like this happen when initially configuring policies and switching primary keys. In a production scenario, you may not want to remove all sessions.

---

**Step 1** SSH to sessionmgr01.

**Step 2** Open sessionmgr CLI usin the following command:

```
/usr/bin/mongo --port 27717
```

Using `/usr/bin/mong` indicates whether the mongo replica set is primary or secondary.

**Step 3** Enter following commands on the MongoDB CLI:

```
use session_cache;
db.session.remove({});
```

**Step 4** If it gives you a 'not master' error, log into sessionmgr02 and perform the same steps.

---

## Error Processing Request: Unknown Action

```
com.broadhop.policy.impl.RulesPolicyService - Error
processing policy request: Unknown action:
com.broadhop.pop3auth.actions.IPOP3AuthRequest and Remote
Actions are disabled.
```

If you see an error of the type above, it means that the implementation class it's looking for is not available on the server. This can be caused by:

- The component needed is not installed on the server.
- Ensure that the pop3auth service is installed in your server.
- Look for exceptions in the logs when starting up.
- Try restarting the service bundle (pop3auth service in this case) using the OSGi console and looking at the logs.

## How to check configured NTP sources?

You can check the configured NTP servers through `# chronyc sources`

## Memcached Server is in Error

```
ERROR c.b.d.impl.DiagnosticController - Diagnostic failed.
```

```
A problem exists with the system --> Common Services:
```

```
2:Memcached server is in error
```

**Step 1** Log on to the server where policy server (qns) is running

**Step 2** Telnet to the memcache server's IP and port 11211 (For example, `telnet lbvip01 11211`).

You can figure out which memcache server CPS is pointing to in Cisco Policy Builder. Look at: **Reference Data > Systems > System Name > Cluster Name**.

a) If you cannot telnet to the port, do the following:

Make sure memcache is running:

- Log on to server where memcache is running.

```
run service memcached status
[root@sessionmgr01 ~]# monit status memcached
memcached is stopped
```

- If the service is stopped, start it:

```
[root@sessionmgr01 ~]# monit start memcached
Starting a new distributed memory caching
(memcached) process for 11211:
```

b) Make sure firewall configuration is OK.

To check if this is the problem, stop the firewall.

```
/usr/bin/systemctl stop iptables
```

If it is the problem, add an exception in `/etc/sysconfig/iptables`. Look at other entries in the file for an example.

After adding an exception, restart the IP tables: `/usr/bin/systemctl start iptables`.

## Firewall Error: Log shows Host Not Reachable, or Connection Refused

In HA environment if we see some connection refused errors stop the firewall and execute

```
service iptables stop
```

to see if the problem is related to the iptables firewall issue.

## Unknown Error in Logging: License Manager

```
2010-12-12 18:51:32,258 [pool-4-thread-1] ERROR
c.b.licensing.impl.LicenseManager - Unknown error in
logging
java.lang.NullPointerException: null
at
com.broadhop.licensing.impl.LicenseManager.checkFeatures(L
icenseManager.java:311) ~[na:na]
```

This issue may occur if no license has been assigned yet.

Option 1: If this is for development or Proof Of Concept deployments you can turn on developer mode. This effectively gives you 100 users but is not for use in production.

1. Login to CPS.
2. Add the following to the `/etc/broadhop/qns.conf` file:

```
-Dcom.broadhop.developer.mode=true
```

3. Restart CPS

Option 2: Generate a real license. Have your Cisco technical representative send you the Technical Article *Tool com.broadhop.licensing.service - Creating a CPS License*.

Option 3: If we have license error in the logs, check the MAC address of the VM and compare that with the MAC address in the license file in `/etc/broadhop/license/`.

## Logging Does Not Appear to be Working

**Step 1** Run the JMX Command:

```
/opt/broadhop/qns/bin/jmxcmd.sh

ch.qos.logback.classic:Name=default,Type=ch.qos.logback
.classic.jmx.JMXConfigurator Statuses

or
```

**Step 2** Access that bean using JMX Term or JConsole to view the status of the Logback Appenders. To access JMX Term, follow these steps:

- a) Execute the script: `/opt/broadhop/qns-1/bin/jmxterm.sh`
- b) If user does not have permission to execute the command then change the permission using below command:

```
chmod 777 opt/broadhop/qns-1/bin/jmxterm.sh
```

- c) Again execute the script: `/opt/broadhop/qns-1/bin/jmxterm.sh`
- d) Once command is executed, JMX terminal opens up.
- e) Execute the following command to open connection:

```
$>open qns01:9045
```

- f) All beans can be seen using the following command:

```
$>beans
#domain = JMImplementation:
```

```

JMImplementation:type=MBeanServerDelegate
#domain = ch.qos.logback.classic:
ch.qos.logback.classic.Name=default,Type=ch.qos.logback.classic.jmx.JMXConfigurator
#domain = com.broadhop.action:
com.broadhop.action:name=AddSubscriberService,type=histogram
com.broadhop.action:name=AddSubscriberService,type=service
com.broadhop.action:name=GetSessionAction,type=histogram
com.broadhop.action:name=GetSessionAction,type=service
com.broadhop.action:name=GetSubscriberActionImpl,type=histogram
com.broadhop.action:name=GetSubscriberActionImpl,type=service
com.broadhop.action:name=LockSessionAction,type=histogram
com.broadhop.action:name=LockSessionAction,type=service
com.broadhop.action:name=LogMessage,type=histogram
com.broadhop.action:name=LogMessage,type=service
com.broadhop.action:name=OCSLoadBalanceState,type=histogram
com.broadhop.action:name=OCSLoadBalanceState,type=service
java.nio:name=mapped,type=BufferPool
#domain = java.util.logging:
java.util.logging:type=Logging

```

## Cannot Connect to Server Using JMX: No Such Object in Table

This is likely caused because the server's name is not set up in the hosts file with its proper IP address.

In `/etc/hosts` the hostname (e.g. `qns01`) SHOULD NOT be aliased to `127.0.0.1` or `localhost`.

If improperly aliased JMX tells the server it's connecting to connect back with the IP of it's hostname. If it's aliased to `localhost` (`127.0.0.1`) the server attempts to open connections with itself which is unfortunate.

Example Error:

```

ERROR com.broadhop.management.JmxClient -
Unable to connect to JmxClient iomgr019045. Cause no
such object in table Will attempt to reconnect.

```

## File System Check (FSCK) Errors

During machine boot `fsck` is run on file systems to check its consistency. This consistency check is done without user intervention and automatically fixes errors which it can. But sometimes if there is a hard reset to CPS VM/machine for example because of abrupt power failure then during `fsck` all the problems are not automatically fixed and user intervention is must to fix the errors reported by `fsck`. The table below describes the common `fsck` errors along with their description and solution.

**Table 5: File System Errors and Solutions**

SNo.	FSCK Error	Description/Solution
1	BAD SUPER BLOCK MAGIC NUMBER WRONG USE ALTERNATE SUPER-BLOCK TO SUPPLY NEEDED INFORMATION	This error comes when file system is cleanly unmounted. Some superblock corruptions can be automatically repaired. But for some like BAD MAGIC number fsck aborts and alternate superblock must be specified to fsck command to continue file system check.  Refer to the <a href="#">link</a> to fix the issue.
2	Block bitmap not in a group/inode bitmap not in a group	When this error occurs data on the device need to be restored using dd or any other device specific command.  Refer to the following links to fix the issue:  <a href="#">Link 1</a> , <a href="#">Link 2</a>
3	Inode table not in a group	When this error occurs data on the device need to be restored using dd or any other device specific command.  Refer to the <a href="#">link</a> to fix the issue
4	Primary superblock is corrupt	Please refer to Error 1 apart from bad magic number if fsck detects corruption in any static parameters of primary superblock (file system size inode list size etc) it requests operator to specify location of alternate superblock.
5	Journal superblock has an unknown read-only feature flag set	Please refer to Error 1 to 4 to fix this issue.
6	Resize inode is invalid	This error occurs after file system is resized.  Refer to the <a href="#">link</a> to fix this issue.
7	Last mount time is in the future	This error occurs after reboot system clock is not synchronized with UTC.  Refer to the <a href="#">link</a> to fix the issue.
8	Root directory is not an inode	If primary superblock is corrupt this error occurs alternate superblock needs to be specified to fsck in this case.  Refer to the following links to fix the issue:  <a href="#">Link 1</a> , <a href="#">Link 2</a>



SNo.	FSCK Error	Description/Solution
9	Duplicate '.' entry	<p>An indirect block is a pointer to a list of every block claimed by an inode. fsck checks every block number against a list of allocated blocks if two inodes claim the same block number that block number is added to a list of duplicate block numbers.</p> <p>The administrator may be asked to choose which inode is correct and usually time to verify files against backups. fsck additionally checks the integrity of the actual block numbers which can also become corrupt - it should always lie in the interval between the first data block and the last data block. If a bad block number is detected the inode is cleared.</p> <p>Similar to above example this issue is with file system synchronization with actual disk. If machine is powered OFF before fs synchronization to hardware disk on next reboot fsck will ask corrective questions to the user to take the action accordingly.</p> <p>For which manual intervention is needed as corrective actions will defer case to case. For example if one record is created by database operation and at the same time another record is deleted and same block number (of deleted record) is used for the newly created record duplicate block error might come.</p>
10	Error reading block <block_no> (Attempt to read from filesystem resulted in short read) while doing inode scan.	<p>This error stops the user from continuing with the fsck scan and correcting the problem. Disks that have physical hardware errors often report - being unable to read inodes error.</p> <p>To resolve this issue replace the disk rather than attempting any corrective action.</p>
11	Journal superblock has an unknown incompatible feature flag set	<p>Feature flag specifies what features a file system has. If this flag is corrupted fsck asks whether you want to abort the operation. You need to specify "no" and after this fix the superblock corruption.</p> <p>Refer to the <a href="#">link</a> to fix the issue.</p>

- The [link](#) gives list of all the errors which are automatically fixed by fsck as well as list of errors where user intervention is must -
- The [link](#) gives general idea about various phases in fsck.
- The [link](#) describes all the errors in case of UFS file system.

This link can be used as a reference to fix the errors reported by fsck on CPS file system which is ext3.

## CPS: Session Cache mongoDB Stuck in STARTUP2 after sessionMgr01/2 Reboot

There can be a situation where session cache mongoDB process is stuck after sessionMgr01/02 is rebooted. In this situation follow the steps below to bring up session cache database mongo processes from STARTUP2 state to PRIMARY/SECONDARY state specific to session database only.



**Note** The steps mentioned in this section are applicable to GR deployments. For HA, the mongo processes are recovered automatically. In case they are not recovered automatically, then only the steps mentioned in this section should be used.

- Step 1** Stop the CPS processes.
- Step 2** Log onto perclient01.
- Step 3** Execute the diagnostic.sh script to know which replica set (all members) have failed.

```
diagnostics.sh --get_replica_status
```

**Note** If a member is shown in an unknown state, it is likely that the member is not accessible from one of other members, mostly an arbiter. In that case, you must go to that member and check its connectivity with other members.

Also, you can login to mongo on that member and check its actual status.

The figure shows all replica set members of replica set set01 for session data are in STARTUP2 state.

**Figure 15: Replica Set Members**

SET NAME	PORT	IP ADDRESS	REPLICA STATE	HOST NAME	HEALTH	LAG TIME	PRIORITY
<b>SESSION/rs01</b>							
Member-1	27717	192.168.210.58	ARBITER	perclient01	ON-LINE	-----	0
Member-2	27717	192.168.210.59	STARTUP2	sessionmgr01	ON-LINE	No Primary	2
Member-3	27717	192.168.210.60	STARTUP2	sessionmgr02	ON-LINE	No Primary	2
Member-4	27717	192.168.210.65	STARTUP2	sessionmgr03	ON-LINE	No Primary	1
Member-5	27717	192.168.210.66	STARTUP2	sessionmgr04	ON-LINE	No Primary	1
<b>BALANCE/rs02</b>							
Member-1	27718	192.168.210.57	ARBITER	perclient01	ON-LINE	-----	0
Member-2	27718	192.168.210.59	PRIMARY	sessionmgr01	ON-LINE	-----	1
Member-3	27718	192.168.210.60	SECONDARY	sessionmgr02	ON-LINE	No Lag	1
Member-4	27718	192.168.210.65	SECONDARY	sessionmgr03	ON-LINE	No Lag	1
Member-5	27718	192.168.210.66	SECONDARY	sessionmgr04	ON-LINE	No Lag	1

- Step 4** Build the session replica sets. Select 2 for session non-sharded sets.

```
./build_set.sh --create --setname <setname>
```

- Step 5** Set the priority by executing the following command from Cluster Manager.

In case of GR: `set_priority.sh --db session --replSet <setname> --sitename <site1>`

In case of HA: `set_priority.sh --db session --replSet <setname>`

- Step 6** Verify if priority is set correctly for newly created replica set.

```
diagnostics.sh --get_replica_status
```

**Note** If a member is shown in an unknown state, it is likely that the member is not accessible from one of other members, mostly an arbiter. In that case, you must go to that member and check its connectivity with other members.

Also, you can login to mongo on that member and check its actual status.

- Step 7** To recover other failed sets, follow the recovery [Step 1, on page 58](#) to [Step 6, on page 58](#).
- Step 8** Restart CPS.

`/var/qps/bin/control/restartall.sh`

**Caution** Executing `restartall.sh` will cause messages to be dropped.

## Multi-user Policy Builder Errors

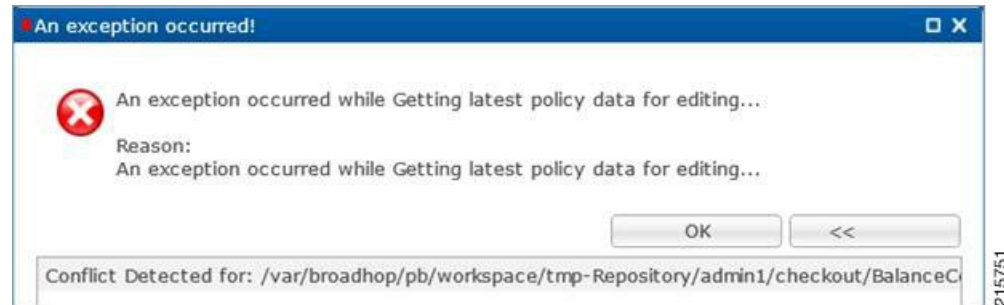
### Not able to do any edits after login

Verify the newly created SVN user has write permission. User should be specified under admins in `/var/www/svn/users-access-file` file.

### Error in login due to conflict

If error similar to below is seen during login, then revert the configuration and login again.

**Figure 16: Login Error**



### No configuration visible in Policy Builder after login

1. Verify the directory `/var/broadhop/pb/workspace/<username>/checkout` is created on `pcrfclient01` and it contains `.xmi` files.
2. If directory does not exist or does not have `.xmi` files then delete existing repository using `Remove` on login page and then add new repository using `Add` on login page.

**Figure 17: Delete Existing Repository**

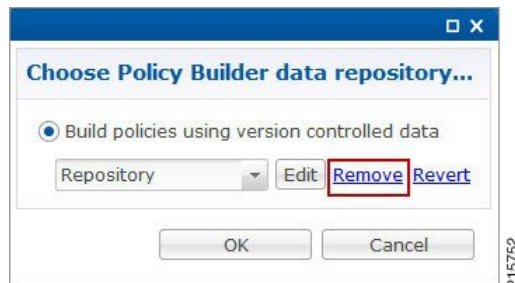
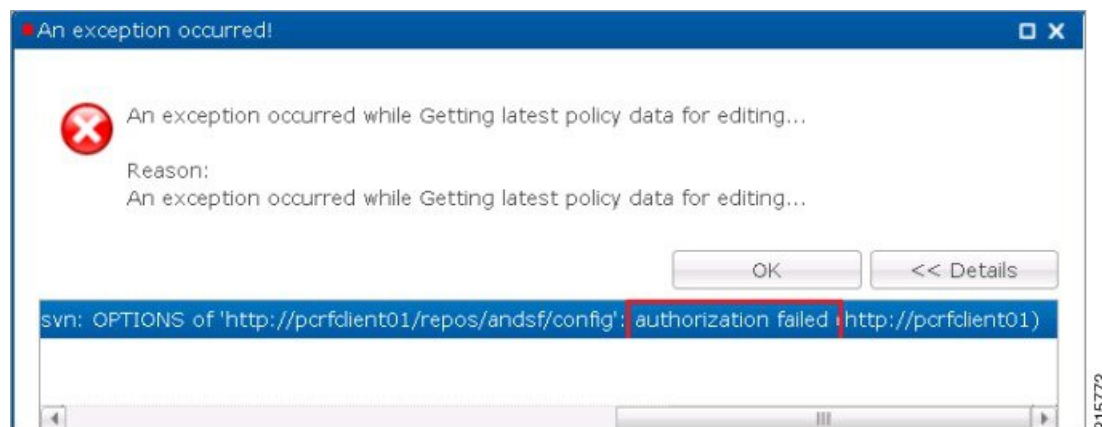


Figure 18: Add New Repository



### Exception Occurred During Login

Figure 19: Exception Occurred



This indicates user does not exist in SVN server.

**Debug:** Verify user exist in `/var/www/svn/.htpasswd` file.

#### Debug Details

Log Files: `/var/log/broadhop/qns-pb.log`

## Policy Reporting Configuration not getting updated post CPS Upgrade

During CPS upgrade from 5.5.1 to 7.0.1 it is observed that Policy Reporting configuration does not get updated as per configuration done in CPS 5.5.1.

All the configuration saved in Cisco Policy Builder are converted into XMI files which are added in the SVN repository. The XMI files based on the CPS 7.0.1 for Policy Reporting won't be fully compatible with the CPS 5.5 version.

To support backward compatibility a utility script `migrateCdrXmi_5_5_to_7_0.sh` can be implemented which upgrades the policy reporting configuration files (XMI files) to CPS 7.0.1.

**Step 1** Obtain the installer archive from the update site corresponding to the build deployed on the system.

**Step 2** Copy the archive into the `/tmp` directory of the CPS virtual machine `perfcient01`.

**Step 3** Log in as root to the same CPS virtual machine and run these commands.

```
mkdir /opt/broadhop/installer/migrate/
tar -zxvf /tmp/<installer archive anme> -C /opt/broadhop/ installer/migrate/
chown -R qns:qns /opt/broadhop/installer/migrate
chmod +x /opt/broadhop/installer/migrate/*.sh
```

**Step 4** Run these commands to execute the script:

```
cd /opt/broadhop/installer/migrate/
sh migrateCdrXmi_5_5_to_7_0.sh
```

The XMI files added or deleted from SVN configuration repository are displayed in the output.

**Step 5** Open the Policy Builder page to verify the configuration changes and publish to runtime.

The utility upgrades the Policy reporting fields, the policy reporting records and the Policy CDR configuration in Policy Reporting section of the Cisco Policy builder.

If an older CPS configuration had any ‘Reporting Server Configuration’ (in Policy Reporting Plugin Configuration) that used any existing policy CDRs, you have to recreate those reporting configurations using the newly created policy CDRs.

## CPS Memory Usage

CPS memory consumption can be monitored using appropriate KPIs in Grafana graphs or other monitoring tools. If memory consumption increases beyond the default threshold of 90% on any CPS VM CPS will generate a Low Memory alarm for that VM. This threshold is configurable in the CPS Deployment Template using the `free_mem_per` setting.

### Detect and Reclaim Cached Memory

In some cases a Low Memory alarm may be a result of Linux memory management allocating objects in cache.

To evaluate how much memory a VM has cached and to trigger Linux to free some of the cached memory

1. Compare the amount of memory cached on two or more CPS VMs by running the `free -m` command on each VM.

For example, on this `qns01` VM 1893 MB of memory is cached.

```
[root@qns01 ~]# free -m
              total        used         free      shared    buffers     cached
Mem:           7854         7719           135          0           311        1893
-/+ buffers/cache: 5514         2340
Swap:          4095           13         4082
```

However, on `qns02` only 1273 MB of memory is cached..

```
[root@qns02 ~]# free -m
              total        used         free       shared    buffers     cached
Mem:           7854         7175          678           0          321        1273
-/+ buffers/cache:  5580         2274
Swap:          4095           14         4081
```

From this example qns01 is storing 620 MB more memory in cache than qns02.

2. To reclaim some of the inactive cached memory execute the following command:

```
free && sync && echo 3 > /proc/sys/vm/drop_caches && echo "" && free
```



**Caution** Running this command will discard cache objects which can cause a temporary increase in IO and CPU usage, **so it is recommend to run this command during off-peak hours/maintenance window.**



**Note** This is a non-destructive command and will only free memory that is not in use.

## Errors while Installing HA Setup

**Step 1** Modify file `/var/qps/config/deploy/csv/AdditionalHosts.csv` to correct lbvip02 IP address and support sslvip01.

- a) Correct lbvip02 IP address.
- b) Add sslvip01 IP address.
- c) Convert to json `/var/qps/install/current/scripts/import/import_deploy.sh`.
- d) Synchronize host `/var/qps/bin/update/synchosts.sh`.
- e) Restart all CPS process using the following commands:

```
/var/qps/bin/control/stopall.sh
/var/qps/bin/control/startall.sh
```

- f) SSH to lbvip01 and update pcs resources.
- g) Delete lbvip02 resource.

```
/usr/sbin/pcs resource delete lbvip02
```

- h) Create lbvip02 and sslvip01 resources.

```
/var/broadhop/init_pacemaker_res.sh
```

- i) Restart httpd to use correct lbvip02 IP.

**Step 2** 27717 replica set members are in startup state, recreate replica set.

- a) Go to pcrfclient01, sessionmgr01 and sessionmgr02, and execute the following command:

```
/usr/bin/systemctl stop sessionmgr-27717
```

- b) Delete current data directory.

```
\rm -fr /var/data/sessions.1/*
```

c) Go to pcrfclient01, sessionmgr01 and sessionmgr02, and execute the following command:

```
/usr/bin/systemctl start sessionmgr-27717

/var/broadhop/initialize_replicaset.sh --port 27717 --hosts sessionmgr01,sessionmgr02 --arbiter
pcrfclient01 --set set01
```

**Step 3** Execute the following command to check errors:

```
/var/qps/install/current/scripts/bin/diag/diagnostics.sh (shows some memory and basic port unreachable errors)
```

For more information on `diagnostics.sh`, refer to **diagnostics.sh** section in *CPS Operations Guide*.

**Step 4** Install `bc` and `nc`, using the following commands:

```
yum install bc
yum install nc
```

Open port 6514 on pcrfclient01 and pcrfclient02, add highlighted bold mark line in `/etc/sysconfig/iptables` and restart iptables.

```
-A INPUT -i eth0 -p udp -m multiport --ports 6514 -m comment --comment "100 allow logstash syslog
access" -j ACCEPT
-A INPUT -i eth0 -p tcp -m multiport --ports 6514 -m comment --comment "100 allow logstash syslog
access tcp" -j ACCEPT
/usr/bin/systemctl start iptables
```

## Enable/disable Debit Compression

Debit compression can be used to identify what all the debits have happened for the subscriber. This data can also be used to cross check the debits with external entities.

- To disable compression add/edit the following flag in `/etc/broadhop/qns.conf` file.

```
-DcompressDebits=false
```

- To enable compression add/edit the following flag in `/etc/broadhop/qns.conf` file.

```
-DcompressDebits=true
```

We can also check directly in mongo how balance has been debited /credited for subscriber using the following queries

### Command to find subscriber:

- SPR database:

```
$use spr
$db.subscriber.find({
  "credentials_key" : [
    {
      "network_id_key" : "111111201"
    }
  ]});
```

Or

- `db.subscriber.find({"network_id_key": "886906007135"})db.subscriber.find({"network_id_key": "111111201"})`

**Output:**

```
{
  "_id" : ObjectId("001000009576290454afdc77"),
  "_id_key" : "001000009576290454afdc77",
  "name_key" : {
    },
  "end_date_key" : null,
  "realm_key" : null,
  "parent_id_key" : null,
  "billing_info_key" : {
    "rate_plan_code_key" : null,
    "charging_id_key" : null
  },
  "status_key" : "ACTIVE",
  "version_key" : 0,
  "start_date_key" : null,
  "credentials_key" : [
    {
      "network_id_key" : "111",
      "description_key" : null,
      "password_key" : null,
      "type_key" : null,
      "expiration_date_key" : null
    }
  ],
  "role_key" : "READ_ALL",
  "external_id_key" : null,
  "_transId" : "d2a3f602-69bb-4047-af6f-c979ec36732f-1"
}
```

Use “\_id\_key” output from the above command as subscriber id:

**Balance\_mgmt**

```
$use balance_mgmt
$db.account.find({"subscriberId" : "001000009576290454afdc77"}).pretty();
```

## Not able to Publish the Policy in Policy Builder

- Check whether you are getting any errors in diagnostics.sh and try to fix the error.
- Make sure that you have the correct URL for the run time environment. For example `http://pcrfclient01/repos/run`
- Make sure that you have following configuration on `/etc/broadhop/pb/pb.conf` configuration file. If not then do the changes as shown below and run the `synconfig.sh` and `restartall.sh` for the changes to come into effect:




---

**Caution** Executing `restartall.sh` will cause messages to be dropped.

---

**Sample Configuration:**

```
SESSION_TIMEOUT="-Dsession.timeout=9000"
QNS_SESSION_DATABASE="-Dsession.db.primary=sessionmgr01
-Dsession.db.secondary=sessionmgr02 -Dsession.db.port=27717
-Dua.client.submit.audit=false"
```



```

HA System Sample Configuration:
SESSION_TIMEOUT="-Dsession.timeout=9000"
QNS_SESSION_DATABASE="-Dsession.db.primary=sessionmgr01
-Dsession.db.secondary=sessionmgr02 -Dsession.db.port=27717
-Dua.client.submit.audit=true
-Dua.client.server.url=http://:8080

```




---

**Note** The IP-address is usually LBVIP01 where the SOAP requests are sent to Unified API for our configuration.

---

- If you still face issue collect and analyze the following logs:
  - /var/log/broadhop/qns-engine-pb.log
  - /var/log/broadhop/service-qns-pb.log

## CPS not sending SNMP traps to External NMS server

- Check whether the “snmpd” process is running in respective VM with the command `monit status snmpd`. If it is stopped start the snmpd process with the command `monit start snmpd`.
- Check whether all the IP tables have been turned off and check the status of UDP port 162 provided you are using the same UDP port 162 at the NMS as well.
- Check the external NMS IP is defined in policy director (lb) VM under `/etc/hosts` and also in the `/etc/snmp/scripts/component_trap_convert` in place of `corporate_nms_ip`.
- Check the file `cat /etc/snmp/snmpd.conf` has the line “rocommunity Broadhop” because all the internal traps from various policy server (QNS) VM to active policy director (lb) VM is been sent over this default community name “Broadhop” as mentioned above.
- Check the trap community name is same both in policy director and as well as in external NMS system. For example, `cat /etc/snmp/scripts/snmp_communities trap_community=cisco` (customer external NMS system should also have this same “cisco” community name).
- Check whether the traps from respective policy server (QNS) VM is properly reaching active policy director (lb) VM this can be checked under `/var/log/snmp/trap`.
- Check for `/var/log/messages` on active policy director (lb) for further analysis.

## Policy Builder Loses Repositories

When an hapoxy load balancer which forwards request to Policy Builder server on `perfclient01` is not available then it forwards the request to backup server on `perfclient02`.

Consider `perfcclient01` is up and a new repository is added using Policy Builder. This repository is saved on `perfcclient01` (on file at `/etc/broadhop/pb/policyRepositories.xml` `/etc/broadhop/pb/publishRepositories.xml`).

If pcrfclient01 becomes inaccessible haproxy sends request to pcrfclient02 where it does not find the above mentioned two files (`publishRepositories.xml` `policyRepositories.xml`) and does not display any repository on PB GUI.

### Fix

CPS does not currently support automatic synchronization of the two repository files  
`/etc/broadhop/pb/policyRepositories.xml`  
`/etc/broadhop/pb/publishRepositories.xml`

You must manually copy the two files from pcrfclient01 to pcrfclient02 or vice versa.

## Not able to access IPv6 Gx port from PCEF/GGSN

Make sure the IPv6 firewall is disabled on lb01 and lb02. If the firewall is not disabled then you can disable it by executing the command

```
service ip6tables stop
```

## Bring up sessionmgr VM from RECOVERY state to PRIMARY or SECONDARY State

When any sessionmgr VM mongo instance is stuck at RECOVERY state for a long time, perform the following steps to bring up sessionmgr VM mongo instance to PRIMARY or SECONDARY state.




---

**Note** The recovery steps must be performed during maintenance window only.

---

**Step 1** Execute the following command script to recover the member:

```
high_tps_db_recovery.sh <replica_setname>
```

### For Example:

```
high_tps_db_recovery.sh SPR-SET1
```

**Step 2** Execute `diagnostics.sh` command to check whether the RECOVERING member has recovered.

```
diagnostics.sh --get_replica_status
```

**Note** If a member is shown in an unknown state, it is likely that the member is not accessible from one of other members, mostly an arbiter. In that case, you must go to that member and check its connectivity with other members.

Also, you can login to mongo on that member and check its actual status.

---

After the replica set member is recovered, the state will change to SECONDARY and all the process logs are stored in a log file at the location:

```
/var/log/broadhop/scripts//high_tps_db_recovery_XXXXXXXXXX.log.
```

# ZeroMQ Connection Established between Policy Director and other site Policy Server

ZeroMQ connection established between Policy Director (lb) and other site Policy Server (qns).

## How to check

Execute `netstat -apn | grep 2800` on policy director (lb) and check if other site policy server (qns) are connected to this policy director (lb).

## You may also see the following logs:

```
L2-CA-SEC-lb01 2015-05-10 18:58:04,943 [pool-2-thread-1] ERROR c.b.d.impl.server.StackManager
- Stack
is Null and Realm is not found null:16777238 - realmToStacks [ocsl.sy.server.cisco.com:7,
ocs3.sy.server.cisco.com:7, ocs4.sy.server.cisco.com:7, cscf3.cisco.com:16777236,
ocs2.sy.server.cisco.com:7, cscf6.cisco.com:16777236, ocs6.sy.server.cisco.com:7,
ocs5.sy.server.cisco.com:7]
L2-CA-SEC-lb01 2015-05-10 18:58:04,944 [pool-3-thread-1] WARN c.b.d.impl.server.StackManager
-
Dropping message Outbound: Cmd: 272/1/0 E2E: 1431976189, HBH: 2798797074, Session-ID:
ds4;333241;2883160674, Result-Code: 2001
L2-CA-SEC-lb01 2015-05-10 18:58:04,944 [pool-2-thread-1] ERROR c.b.d.impl.server.StackManager
- Stack
is Null and Realm is not found null:16777238 - realmToStacks [ocsl.sy.server.cisco.com:7,
ocs3.sy.server.cisco.com:7, ocs4.sy.server.cisco.com:7, cscf3.cisco.com:16777236,
ocs2.sy.server.cisco.com:7, cscf6.cisco.com:16777236, ocs6.sy.server.cisco.com:7,
ocs5.sy.server.cisco.com:7]
L2-CA-SEC-lb01 2015-05-10 18:58:04,944 [pool-3-thread-1] WARN c.b.d.impl.server.StackManager
-
Dropping message Outbound: Cmd: 272/2/1 E2E: 1431980725, HBH: 2798442695, Session-ID:
ds1;333241;2799910481, Result-Code: 2001
```

## Fix

To fix the above mentioned problem, perform the following steps to clean the zmq endpoint registry.

1. Connect to admin database.

```
mongo adminDbIpAddress:adminDbPort
```

2. Delete endpoint registry

```
use queueing
db.endpoints.remove({});
```



**Caution** This step impacts the local and remote site services (in case of GR deployment) as queueing database is common to both the sites.

3. Restart the application on local as well as remote site (in case of GR deployment) to create the ZMQ connections by executing the following command on both sites:

```
/var/qps/bin/control/restartall.sh
```



**Caution** Executing `restartall.sh` impacts the service resulting in message drops. The command should be executed in Maintenance Window.

4. Verify by executing `netstat` command:

```
netstat -plan | grep 2800
tcp        0      0 :::ffff:172.20.7.18:28001 :::*          LISTEN
 32352/java
tcp        0      0 :::ffff:172.20.7.18:28002 :::*          LISTEN
 32294/java
tcp        0      0 :::ffff:172.20.7.18:28003 :::*          LISTEN
 32235/java
tcp        0      0 :::ffff:172.20.7.18:28003 :::ffff:172.20.7.26:35308 ESTABLISHED
 32235/java
tcp        0      0 :::ffff:172.20.7.18:28001 :::ffff:172.20.7.30:60045 ESTABLISHED
 32352/java
tcp        0      0 :::ffff:172.20.7.18:28003 :::ffff:172.20.7.34:46369 ESTABLISHED
 32235/java
tcp        0      0 :::ffff:172.20.7.18:28003 :::ffff:172.20.7.24:38216 ESTABLISHED
 32235/java
tcp        0      0 :::ffff:172.20.7.18:28002 :::ffff:172.20.7.32:55328 ESTABLISHED
 32294/java
tcp        0  1130 :::ffff:172.20.7.18:28001 :::ffff:172.20.7.28:58586 ESTABLISHED
 32352/java
tcp        0  1123 :::ffff:172.20.7.18:28002 :::ffff:172.20.7.30:49349 ESTABLISHED
 32294/java
tcp        0      0 :::ffff:172.20.7.18:28001 :::ffff:172.20.7.34:40201 ESTABLISHED
 32352/java
tcp        0      0 :::ffff:172.20.7.18:28002 :::ffff:172.20.7.34:40447 ESTABLISHED
 32294/java
tcp        0      0 :::ffff:172.20.7.18:28003 :::ffff:172.20.7.30:52127 ESTABLISHED
 32235/java
tcp        0      0 :::ffff:172.20.7.18:28001 :::ffff:172.20.7.24:34238 ESTABLISHED
 32352/java
tcp        0      0 :::ffff:172.20.7.18:28002 :::ffff:172.20.7.36:52364 ESTABLISHED
 32294/java
tcp        0      0 :::ffff:172.20.7.18:28003 :::ffff:172.20.7.28:38456 ESTABLISHED
 32235/java
tcp        0      0 :::ffff:172.20.7.18:28001 :::ffff:172.20.7.36:50427 ESTABLISHED
 32352/java
tcp        0      0 :::ffff:172.20.7.18:28003 :::ffff:172.20.7.22:44375 ESTABLISHED
 32235/java
tcp        0      0 :::ffff:172.20.7.18:28003 :::ffff:172.20.7.32:60651 ESTABLISHED
 32235/java
tcp        0      0 :::ffff:172.20.7.18:28002 :::ffff:172.20.7.22:45991 ESTABLISHED
 32294/java
tcp        0      0 :::ffff:172.20.7.18:28003 :::ffff:172.20.7.36:38120 ESTABLISHED
 32235/java
tcp        0      0 :::ffff:172.20.7.18:28001 :::ffff:172.20.7.26:46593 ESTABLISHED
 32352/java
tcp        0  1130 :::ffff:172.20.7.18:28002 :::ffff:172.20.7.28:56499 ESTABLISHED
 32294/java
tcp        0  1130 :::ffff:172.20.7.18:28002 :::ffff:172.20.7.24:57277 ESTABLISHED
 32294/java
tcp        0      0 :::ffff:172.20.7.18:28001 :::ffff:172.20.7.32:48030 ESTABLISHED
 32352/java
tcp        0      0 :::ffff:172.20.7.18:28001 :::ffff:172.20.7.22:36000 ESTABLISHED
 32352/java
tcp        0  1130 :::ffff:172.20.7.18:28002 :::ffff:172.20.7.26:53322 ESTABLISHED
 32294/java
```

## Incorrect Version after Upgrade from 7.0.0 System

If **Upgrade from Existing 7.0 System** does not show latest version then perform the following steps to show the latest version:

1. Reinitialize your environment by executing the following command from Cluster Manager:

```
/var/qps/install/current/scripts/upgrade/reinit.sh
```

2. To restart all the policy server (qns) services execute the following command from Cluster Manager:

```
/var/qps/install/current/scripts/bin/control/restartall.sh
```



**Caution** Executing `restartall.sh` will cause messages to be dropped.

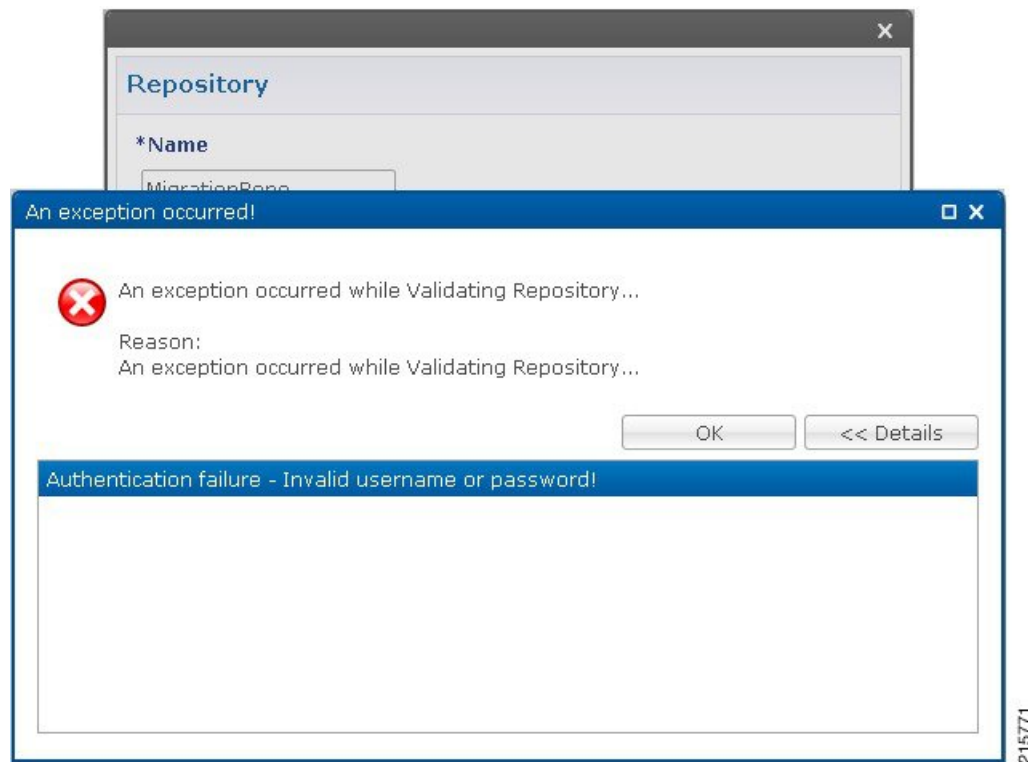
3. Verify CPS Status by running `diagnostics.sh` and `about.sh` scripts from Cluster Manager.

For more information on `diagnostics.sh`, refer to **diagnostics.sh** section in *CPS Operations Guide*.

## Not able to access Policy Builder

**Scenario 1:** When the svn-repos password expires the Policy Builder opens only in Read-only mode.

**Scenario 2:** Invalid username or password.



To resolve the errors described in Scenarios 1 and 2 above perform the following steps:

1. Login to Cluster Manager VM as the root user. The default credentials are root/cisco123.

- Execute `change -l <username>` to check the status of repository password.

For example,

```
[root@lab ~]# chage -l qns
Last password change : Jun 17, 2015
Password expires : Aug 16, 2015
Password inactive : never
Account expires : never
Minimum number of days between password change : 7
Maximum number of days between password change : 60
Number of days of warning before password expires : 7
```

- If the password has expired, execute `change_passwd.sh` to change the password.

```
/var/qps/bin/support/change_passwd.sh
```

- When prompted, enter `qns`.

```
Enter username whose password needs to be changed: qns
```

- When prompted, enter and reconfirm the desired password for the policy server (`qns`) user.

```
Enter new password:
```

```
Re-enter new password:
```

The script changes the policy server (`qns`) user password on all the CPS VMs in the cluster.

```
Changing password on $host...
```

```
Connection to $host closed.
```

```
Password for qns changed successfully on $host
```

- You can use the above steps to set or change the passwords for `root` and `qns-svn` users.




---

**Note** For more information about this and other CPS administrative commands, refer to the *CPS Operations Guide*.

---

## Graphs in Grafana are lost when time on VMs are changed

**Case:** Graphs in Grafana are lost when system time on VMs are changed.

**Solution:** Change the system timing on all VMs. Also change the browser time according to graphite server time and restart the `collectd` service on each VM.

The graphite server time and browser time should match then only we will be able to see graphs.

## Systems is not enabled for Plugin Configuration

**Case:** Systems configuration is not displayed for Plugin Configuration in Reference Data tab.

**Possible Cause:** This issue could occur if Systems plugin configuration is configured using `system.json` file.

**Solution:** Check whether your `system.json` file is valid or not using any json validator.

## Publishing is not Enabled

**Case:** Publishing is not available

**Possible Cause:** SVN configuration is manually exported and imported from one setup to another. While performing import user missed to import `.broadhopFileRepository` or deleted it unknowingly.

**Solution:** Check whether `.broadhopFileRepository` is present in `pcrclient01`. If it not present import the file.

## Added Check to Switch to Unknown Service if Subscriber is deleted Mid Session

**Problem:** There is an impact on 7.5.0 and higher releases with a new feature “**check to switch to unknown service if subscriber is deleted mid session**” due to the custom policies defined in some customer locations.

**Solution:** For the customers who are on pre-7.5.0 release and don't want the new feature a work around has been suggested with an addition of custom policy that will bypass this feature.

The custom policy has to be added in the call flow based on the conditions. This custom policy will add a “IgnoreSPR” AVP to policy state. If this AVP is present internally the code will skip the feature.

Below are screenshots of the policy where the customer wants to skip the feature in some specific conditions of flow like when “**Authenticating a subscriber on AAA server**” since we don't store subscriber information in SPR.

Without this custom policies we will see session switching from known to unknown service during SPR update/Accounting. This will be visible in information logs in engine and policy server (qns) with “**Session has switched from known to unknown as subscriber could not be found**”.

## Conditions

Figure 20: Conditions - 1

**Policies**

- Summary
- Initial Blueprint
  - Network Session
  - Autowire
  - Subscriber Data (SPR)
  - Pre session policies
  - Set keys to load session
  - Map session data from input
    - Set User Name from NAI
    - Set Mac Address from NAI
    - Handle Accounting No Gx
    - Accounting Stop Set Creden
  - Load subscriber data
    - Web-Auth TAL Session
      - Map TAL Service from CA
    - Map Acces-Request AVPs
      - Map Service from CAR
      - Set session as Guest
      - ignoreSPR
  - Post subscriber load

**Policy**

\*Name: ignoreSPR      Copy: Current Policy      Move: Reparent

Conditions | Actions | Advanced

**Conditions**

When all conditions are true, the actions on the adjacent tab are executed.

Name
An AVP retrieved from an authorization request exists
An IPolicyState exists
There does not exist an Avp

Add Remove ↑ ↓

Input Variables	Type	Operator	Value
code (String)	Literal	=	CISCO-ACCOUNT-INFO

Available Input Variables -  
Add All  
Add code (String) Add value (String)

Condition Outputs

215754

Figure 21: Conditions - 2

**Policies**

- Summary
- Initial Blueprint
  - Network Session
  - Autowire
  - Subscriber Data (SPR)
  - Pre session policies
  - Set keys to load session
  - Map session data from input
    - Set User Name from NAI
    - Set Mac Address from NAI
    - Handle Accounting No Gx
    - Accounting Stop Set Creden
  - Load subscriber data
    - Web-Auth TAL Session
      - Map TAL Service from CA
    - Map Acces-Request AVPs
      - Map Service from CAR
      - Set session as Guest
      - ignoreSPR
  - Post subscriber load

**Policy**

\*Name: ignoreSPR      Copy: Current Policy      Move: Reparent

Conditions | Actions | Advanced

**Conditions**

When all conditions are true, the actions on the adjacent tab are executed.

Name
An AVP retrieved from an authorization request exists
An IPolicyState exists
There does not exist an Avp

Add Remove ↑ ↓

**Input Variables**

**Condition Outputs**

IPolicyState (IPolicyState)

215755



Figure 22: Conditions - 3

**Policies**

- Summary
- Initial Blueprint
  - Network Session
  - Autowire
  - Subscriber Data (SPR)
  - Pre session policies
  - Set keys to load session
  - Map session data from input
    - Set User Name from NAI
    - Set Mac Address from NAI
    - Handle Accounting No Gx
    - Accounting Stop Set Creden
  - Load subscriber data
    - Web-Auth TAL Session
      - Map TAL Service from CA
    - Map Acces-Request AVPs
      - Map Service from CAR
      - Set session as Guest
      - ignoreSPR
  - Post subscriber load
  - Setup network access policies

**Policy**

**\*Name** ignoreSPR **Copy:** [Current Policy](#) **Move:** [Reparent](#)

Conditions | Actions | Advanced

**Conditions**  
When all conditions are true, the actions on the adjacent tab are executed.

Name
An AVP retrieved from an authorization request
An IPolicyState exists
<b>There does not exist an Avp</b>

Add Remove ↑ ↓

Input Variables	Type	Operator	Value
Code (String)	Literal	=	IgnoreSPR

**Available Input Variables -**  
[Add All](#)

- [Add](#) Code (String) [Add](#) Value (String)
- [Add](#) Next Evaluation Date (Date) [Add](#) Start Date (Date)
- [Add](#) Expiration Date (Date) [Add](#) Unique Key (String)
- [Add](#) Structure (Map)

215756

**Actions**

Figure 23: Actions - 1

**\*Name** ignoreSPR **Copy:** [Current Policy](#) **Move:** [Reparent](#)

Conditions | **Actions** | Advanced

**Actions**  
Executed when all conditions are true.

Name
<b>Add an Avp</b>
Policy tracking message

Add Remove ↑ ↓

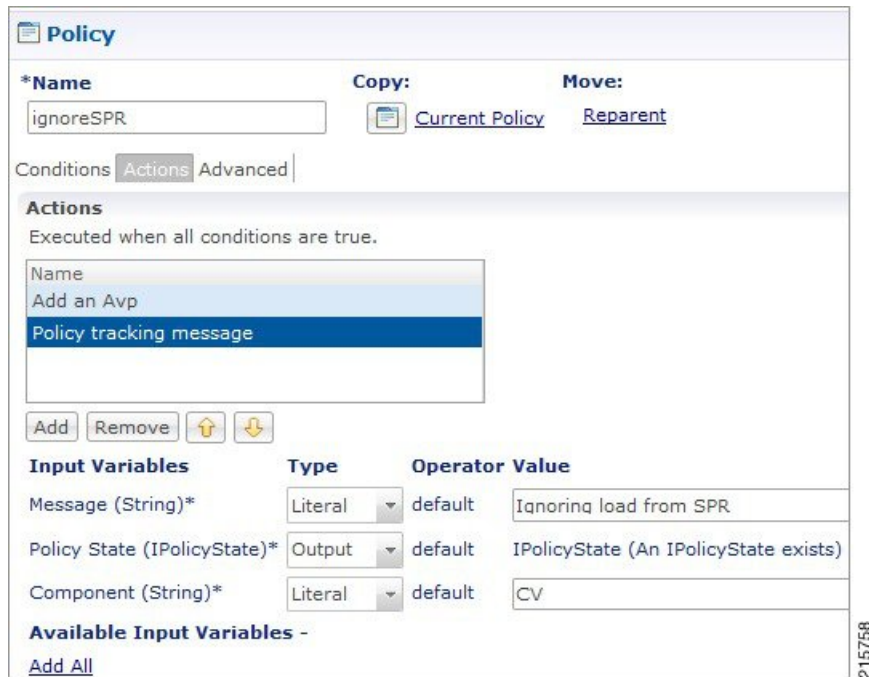
Input Variables	Type	Operator	Value
Code (String)	Literal	default	IgnoreSPR

**Available Input Variables -**  
[Add All](#)

- [Add](#) Value (String) [Add](#) Next Evaluation Date (Date)
- [Add](#) Start Date (Date) [Add](#) Expiration Date (Date)
- [Add](#) Structure (Map)

215757

Figure 24: Actions - 2



## Could not Build Indexes for Table

**Issue:** Policy Builder is not able to build indexes for table (Custom Reference Table).

**Case:** While publishing Policy Builder CPS logs below exception in policy server (qns) log.

For example,

```
ERROR c.b.custrefdata.impl.dao.GenericDao - Could not build indexes for table
QoS-Reference-Mapping
com.mongodb.CommandFailureException
```

**Possible Cause:** This could happen when CRD table key columns are changed from back-end (xmi) in Policy. Due to this underlying composite index on CRD table does not reflect new/changed key columns.

**Solution:** Drop the index on CRD table in MongoDB and publish the Policy Builder.

1. Drop index manually.
 

```
db.<crdtablename>.dropIndexes()
```
2. Make sure xmi (backend) and Policy Builder data of CRD table whose index you want to drop are in sync.
 

If both are not in sync, CPS displays There are uncommitted changes to the '<PBrepositoryname>' repository. Do you wish to discard those changes? while logging to the Policy Builder.

For example, if CRD table data gets modified via backend (xmi) then when you login, CPS shows uncommitted message. Choosing **Retain** will sync up the xmi and Policy Builder.

3. Publish Policy Builder.

4. Check the rebuilt index.

```
db.<crdtablename>.getindexes()
```

## Error Submitting Message to Policy Director (lb) during Longevity

**Case:** Messages timed out intermittently. CPS logs reports following exceptions

```
2015-10-11 145054918 [pool-2-thread-1] ERROR c.b.d.p.event.DiameterMessageDealer.? - Error
submitting message to lb
```

```
2015-10-11 145054918 [pool-2-thread-1] ERROR c.b.d.p.event.DiameterMessageDealer.? - Error
submitting message to lb
```

**Possible Cause:** Message timed out intermittently problem happens when a GC pause greater than 10 seconds is occurring on policy server (qns) and policy director (lb). Due to this pause queue gets overloaded and there are message drops and timeouts. This pause happens when the service-qns logs are getting rotated with size 100 M.

**Solution:** The following changes need to be done on cluster manager

- Change Daily > hourly, size 100M > 25M and rotate 5 > 20

```
cat /etc/logrotate.d/qps
/var/log/broadhop/determine_cluster_state.log
/var/log/broadhop/service-qns-*.log
/var/log/elasticsearch/*.log
{
    daily
    nodateext
    copytruncate
    size 25M
    rotate 20
    missingok
    compress
}
```

- Copy the changes to all the VMs using `copytoall` command.

## Mismatch between Statistics Count and Session Count

**Case:** There are no sessions on CPS but the statistics count still showing statistics.

```
#session_cache_ops.sh --count
Session cache operation script
Fri Nov 13 01:26:08 EST 2015
-----
Session Replica-set SESSION-SET1
-----
Session Database          : Session Count
-----
session_cache             : 0
session_cache_2           : 0
session_cache_3           : 0
session_cache_4           : 0
-----
No of Sessions in SET1   : 0
-----
```

```
Total Number of Sessions : 0

#session_cache_ops.sh --statistics-count
Session cache operation script
Fri Nov 13 01:26:31 EST 2015

-----
Sessions statistic counter on General
-----
Session Type          : Session Count
-----
ADMIN-SET1
RX_TGPP               : 364
GX_TGPP               : 983269
SY_PRIME              : 974457
-----
#
```

**Possible Cause:** CPS monitors the session count and updates the aggregation of message type into counters collection in the admin database. This query is performed on secondary databases. If due to some reason all secondary members are not in healthy state or are in recovering state, then we can incur that the discrepancy is in session count.

```
mongo rtpclabqps5g-sm01a:47721
MongoDB shell version: 2.6.3
set05:PRIMARY> use sharding
set05:PRIMARY> db.counters.find()
{ "_id" : 8, "db" : "session_cache_3", "session_type" : [ ] }
{ "_id" : 9, "db" : "session_cache_4", "session_type" : [ ] }
{ "_id" : 10, "db" : "session_cache", "session_type" : [ { "type" : "SY_PRIME", "count" :
246563 },
{ "type" : "GX_TGPP", "count" : 248921 }, { "type" : "RX_TGPP", "count" : 93 } ] }
{ "_id" : 11, "db" : "session_cache_2", "session_type" : [ { "type" : "SY_PRIME", "count"
: 247330 },
{ "type" : "GX_TGPP", "count" : 249614 }, { "type" : "RX_TGPP", "count" : 94 } ] }
{ "_id" : 12, "db" : "session_cache_3", "session_type" : [ { "type" : "SY_PRIME", "count"
: 227624 },
{ "type" : "GX_TGPP", "count" : 229542 }, { "type" : "RX_TGPP", "count" : 90 } ] }
{ "_id" : 13, "db" : "session_cache_4", "session_type" : [ { "type" : "SY_PRIME", "count"
: 252940 },
{ "type" : "GX_TGPP", "count" : 255192 }, { "type" : "RX_TGPP", "count" : 87 } ] }
{ "_id" : 18, "db" : "session_cache_2", "session_type" : [ ] }
```

**Diagnostic showing all secondary members are in bad shape:**

*Figure 25: Secondary Members*

```
SESSION:set02k
Member-1 - 37740 : 172.26.0.211 - ARBITER - rtpclabqps5g-cc01a - ON-LINE - ----- - 0
Member-2 - 27737 : 172.26.5.83 - PRIMARY - rtpclabqps5g-sm22a - ON-LINE - ----- - 5
Member-3 - 27737 : 172.26.5.73 - RECOVERING - rtpclabqps5g-sm21a - ON-LINE - 7 days - 4
Member-4 - 27737 : 172.26.6.83 - FATAL - rtpclabqps5g-sm22b - ON-LINE - 7 days - 3
Member-5 - 27737 : 172.26.6.73 - RECOVERING - rtpclabqps5g-sm21b - ON-LINE - 7 days - 2
```

**Consolidated CPS log throws below exception**

```
rtpclabqps5g-qns09b rtpclabqps5g-qns09b 2015-11-13 03:06:45,603 [pool-2-thread-1] WARN
c.b.c.m.dao.impl.ShardInterface - Unable to get direct connection for DB shard { "_id" :
10 ,
"seed_1" : "sessionmgr21" , "seed_2" : "sessionmgr22" , "port" : 27737 , "db" :
"session_cache" ,
"online" : true , "count" : 0 , "backup_db" : false , "lockTime" : { "$date" :
"2015-11-13T08:06:25.997Z"} , "isLocked" : false , "lockedBy" : null } - bypassing type
counts
rtpclabqps5g-qns09b rtpclabqps5g-qns09b 2015-11-13 03:06:45,605 [pool-2-thread-1] WARN
```

```
c.b.c.m.dao.impl.ShardInterface - Unable to get direct connection for DB shard { "_id" :
11 ,
"seed_1" : "sessionmgr21" , "seed_2" : "sessionmgr22" , "port" : 27737 , "db" :
"session_cache_2" ,
"online" : true , "count" : 0 , "backup_db" : false
```

**Solution:** Recovers all the secondary database members.

## Disk Statistics not Populated in Grafana after CPS Upgrade

**Case:** After CPS upgrade disk statistics are not populated in Grafana.

**Possible Cause:** Configurations are not refreshed after collectd package is upgraded.

**Solution:** Restart collectd service on respective VM/VMs.

## Re-create Session Shards

All sessions require to be cleared/removed from CPS.



**Note** Steps are NOT recommended to be performed in Production environment.

To delete all shards and then re-create shards, perform the following steps:

### Step 1

Take the backup of admin database.

- a) Run diagnostics command on pcrfclient01 and find admin database primary member and port

```
diagnostics.sh --get_replica_status
```

**Note** If a member is shown in an unknown state, it is likely that the member is not accessible from one of other members, mostly an arbiter. In that case, you must go to that member and check its connectivity with other members.

Also, you can login to mongo on that member and check its actual status.

**Table 6: Admin Database and Port Information**

SET NAME	PORT	IP ADDRESS	REPLICA STATE	HOST NAME	HEALTH	LAST SYNC	PRIORITY
ADMIN:set06							
Member-1	27721	192.167.82.35	ARBITER	pcrfclient01	ON-LINE	-----	0
Member-2	27721	192.167.82.29	PRIMARY	sessionmgr01	ON-LINE	-----	1
Member-3	27721	192.167.82.30	SECONDARY	sessionmgr02	ON-LINE	0	1

- b) Execute `mongo dump` command with primary member, and port to backup admin database:

```
mongodump --host sessionmgr01 --port 27721
```

This command creates the mongo dump files in the file system.

**Step 2** Clear all sessions from all the shards (execute command from pcrclient01)

```
session_cache_ops.sh --remove
```

When prompted for input, input **yes**

**Step 3** To recreate the shards you have two options:

a) Option-1: Delete or drop the “sharding” database and recreate the shards.

1. Stop all QNS process using `stopall.sh` script.

```
PRIMARY> use sharding
PRIMARY> Db.dropDatabase()
```

2. Start all QNS process using `startall.sh` script.

3. Once diagnostics shows green, you can start OSGi command to create the shards.

b) Option-2: Delete the collection entries in the “sharding” database.

1. Login to the ADMIN replica-set primary member by executing `mongo --sessionmgr01 --port 27721` and drop the “sharding” database.

```
PRIMARY> use sharding
PRIMARY> db.shards.remove({});
PRIMARY> db.buckets.remove({});
PRIMARY> db.config.remove({});
PRIMARY> db.instances.remove({});
```

2. Start and execute OSGi command to create the shards.

**Step 4** (Optional) Change default shards (skip this step if default shard does not need to be changed).

By default, one shard gets created. Default shard is `sessionmgr01/sessionmgr02 27717`.

In case user wants to change default shards, add/modify following parameter in `/etc/broadhop/qns.conf` file on cluster manager.

```
-Dsession.db.init.1=sessionmgr01
```

```
-Dsession.db.init.2=sessionmgr02
```

```
-Dsession.db.init.port=27717
```

Copy this file to all nodes (run script from Cluster Manager)

```
copytoall.sh /etc/broadhop/qns.conf /etc/broadhop/qns.conf
```

**Step 5** Restart policy server (QNS) services (execute script from cluster manager).

```
restartall.sh
```

**Caution** Executing `restartall.sh` will cause messages to be dropped.

**Step 6** Once policy servers (QNS) are UP, verify default shard is created in shard collection.

a. Login to admin database.

```
mongo - sessionmgr01 -port 27721
```

Check for default shard.

```
set01:PRIMARY> use sharding
set01:PRIMARY> db.shards.count();
{
  "_id" : 1,
  "seed_1" : "sessionmgr01",
  "seed_2" : "sessionmgr02",
  "port" : 27717,
  "db" : "session_cache",
  "online" : true,
  "count" : NumberLong(0),
  "lockTime" : ISODate("2016-02-04T11:41:47.259Z"),
  "isLocked" : false,
  "lockedBy" : null
}
```

**Step 7** To add shard, refer to section Create Session Shards in *CPS Installation Guide for VMware*.

## Re-create SK Shards



**Note** Steps are NOT recommended to be performed in Production environment.

To delete all SK shards and then re-create SK shards, perform the following steps:

**Step 1** Take the backup of admin database.

a) Run diagnostics command on pcrfelient01 and find admin database primary member and port number.

```
diagnostics.sh --get_replica_status
```

**Note** If a member is shown in an unknown state, it is likely that the member is not accessible from one of other members, mostly an arbiter. In that case, you must go to that member and check its connectivity with other members.

Also, you can login to mongo on that member and check its actual status.

**Table 7: Admin Database and Port Information**

SET NAME	PORT	IP ADDRESS	REPLICA STATE	HOST NAME	HEALTH	LAST SYNC	PRIORITY
ADMIN:set06							
Member-1	27721	192.167.82.35	ARBITER	pcrfclient01	ON-LINE	-----	0
Member-2	27721	192.167.82.29	PRIMARY	sessionmgr01	ON-LINE	-----	1
Member-3	27721	192.167.82.30	SECONDARY	sessionmgr02	ON-LINE	0	1

- b) Execute `mongo dump` command with primary member, and port to backup admin database:

```
mongodump --host sessionmgr01 --port 27721
```

This command creates the mongo dump files in the file system.

**Step 2** Clear all the secondary keys from the SK shards.

- a) Login to admin database.

```
mongo --host sessionmgr01 --port 27721
```

- b) Delete all SK entries from all SK shards (this might take some time depending upon number of records).

```
set01:PRIMARY> use sharding
set01:PRIMARY> db.sk_shards.find({}, {_id:0, seed_1:1, seed_2:1, port:1, db:1}).forEach(
  function(rec) {
    var mongo = new Mongo(rec.seed_1, rec.port);
    var db = mongo.getDB(rec.db);
    var coll = db.getCollection("secondary_key");
    print(coll.remove({}));
  }
);
```

**Step 3** Drop all SK sharding collections:

```
set01:PRIMARY> use sharding
set01:PRIMARY> db.sk_shards.remove({});
set01:PRIMARY> db.sk_buckets.remove({});
set01:PRIMARY> db.sk_config.remove({});
set01:PRIMARY> db.sk_instances.remove({});
set01:PRIMARY> db.sk_order.remove({});
set01:PRIMARY> show collections
```

There should be no `sk_` collection after running drop command

**Step 4** Clear rebuildSkDb tasks.

```
set01:PRIMARY> use scheduler
set01:PRIMARY> db.tasks.remove({"type" : "rebuildSkDb"})
```

**Step 5** (Optional) Change default SK shards (skip this step if default SK shard does not need to be changed). By default, one SK shard gets created. Default SK shard is sessionmgr01/sessionmgr02 27717.

If you want to change the default SK shards, add/modify following parameter in `/etc/broadhop/qns.conf` file on Cluster Manager.



```
-Dsk.db.init.1=sessionmgr01
-Dsk.db.init.2=sessionmgr02
-Dsk.db.init.port=27717
```

Copy the updated `qns.conf` file to all the nodes by executing the following script from Cluster Manager:

```
copytoall.sh /etc/broadhop/qns.conf /etc/broadhop/qns.conf
```

**Step 6** Restart Policy Server (QNS) services by executing the following script from Cluster Manager:

```
restartall.sh
```

**Caution** Executing `restartall.sh` will cause messages to be dropped.

**Step 7** Once Policy Servers (QNS) are UP, verify default shard is created in shard collection.

a) Login to admin database.

```
mongo --host sessionmgr01 --port 27721
```

b) Check for default shard.

```
set01:PRIMARY> use sharding
set01:PRIMARY> db.sk_shards.find();
{
  "_id" : 1,
  "seed_1" : "sessionmgr01",
  "seed_2" : "sessionmgr02",
  "port" : 27717,
  "db" : "sk_cache",
  "online" : true,
  "count" : NumberLong(0),
  ...
}
```

**Step 8** To add SK shard again, refer to *Configuring SK DB in CPS Installation Guide for VMware*.

## Session Switches from Known to Unknown in CCR-U Request

**Case:** On running a load with Total TPS of 1200 for a CPS having four policy servers (qns) and for 500000 subscribers it was seen that for some of the CCR-U request the CPS sends “Session has switched from known to unknown as subscriber could not be found” causing the CCA-U to give a result code of 5012.

The call model that is used here is a simple Gx call model involving several CCR-U for Charging-Rule-Report were the subscribers are provisioned in a Auto-provisioning manner.

**Possible Cause:** The call model being run was auto provisioning call model with 200 TPS CCR-I 800 CCR-U and 200 CCR-T. On every CCR-I we had a subscriber being automatically provisioned and the balance provisioned automatically with the Automatic Balance Provisioning service.

We saw lot of locking errors for balance being the cause as there was a version mismatch being seen while updating balance.

```
qns02 qns02 2016-02-11 06:11:21,231 [pool-1315-thread-1] ERROR
c.b.b.i.d.i.MongoBalanceRepository -
Cache data is out of date for object 0057170054ce8d1a56bc6c59
qns03 qns03 2016-02-11 06:11:22,041 [pool-1308-thread-1] WARN
c.b.b.i.a.AutowireBalanceManagerBlueprint - Couldn't find a current Account Balance Status
for
template: daily
```

```

qns02 qns02 2016-02-11 06:11:21,232 [pool-1315-thread-1] WARN
c.b.d.p.g.t.DiameterGxTGPPDeviceMgr -
Issue getting reservation status for external reservation id: 1234ds1;338812;2613626736,
Balance
Code 1234, Subscriber Id: 0057170054ce8d1a56bc6c59
com.broadhop.exception.CachedDataIsOutOfDate: Optimistic Locking Error - the version number
does
not match the database version for subscriber: 0057170054ce8d1a56bc6c59

```

**Solution:** Thus with balance auto provisioning enabled and high TPS of balance provisioning (high CCR-I TPS which causes balance to be provisioned) it is suggested to keep the **Db Read Preference** as **Primary** or **PrimaryPreferred** under **Balance Configuration** plug-in in Policy Builder. This will avoid the balance locking errors.

## Intermittent BSON Object Size Error in createsub with Mongo v3.2.1

**Case:** While retrieving/searching subscriber profile using CPS Control Center/Unified API or using mongo client, the query results into BSONObj Size error. Due to this error, the subscriber is not displayed and an error is recorded in MongoDB.

**Example:**

```

set27:PRIMARY> db.subscriber.findOne({"credentials_key.network_id_key" : "910010100034"})
2016-02-17T02:42:18.263-0500 E QUERY [thread1] Error: error: {
"ok" : 0,
"errmsg" : "BSONObj size: 117440514 (0x7000002) is invalid. Size must be between 0 and
16793600(16MB) First element: id: ?type=95",
"code" : 10334
} :

```

**Possible Cause:** Data corruption can have many causes.

**Solution:** Repair all databases:

---

**Step 1** Repair all secondary databases.

```
mongo <hostname>:<port>/spr --eval "db.repairDatabase();"

```

**Step 2** Repair primary database.

```
/usr/bin/systemctl repair sessionmgr-<port#>

```

**Step 3** After stopping check another secondary has become primary or not.

```
/usr/bin/systemctl repair sessionmgr-<port#>

```

```
ps -ef | grep <port#>

```

**Step 4** After repairing, mongod process is stopped. Make sure it is not running.

```
/usr/bin/systemctl start sessionmgr-<port#>

```

**Note** Repairing database takes more time when database size is more (approx 30 sec for 1 GB database), so this activity should be performed in maintenance window (in non-peak hour).

---

## No Traps Generated When Number of Sessions Exceeds the Limit

**Case:** No traps are generated for license threshold when number of sessions exceeds the assigned limit.

**Possible Cause:** Parameter not added in qns.conf file.

**Solution:**

**Step 1** To generate license usage threshold trap, we need to configure the following parameter in `/etc/broadhop/qns.conf` file.

```
-Dcom.cisco.enforcementfree.mode=false
```

**Step 2** After adding the above entry in qns.conf file, execute `copytoall.sh` to synchronize the configuration changes to all VMs in the CPS cluster:

```
copytoall.sh /etc/broadhop/qns.conf /etc/broadhop/qns.conf
```

**Step 3** After modifying the configuration file, to make the changes permanent for future use (when any VM is redeployed or restarted...etc.), user needs to rebuild `etc.tar.gz`.

```
/var/qps/install/current/scripts/build/build_etc.sh
```

**Step 4** Restart the CPS service.

```
/var/qps/bin/control/restartall.sh
```

**Caution** Executing `restartall.sh` will cause messages to be dropped.

## RAR Message Not Received

**Case:** Sometimes the RAR message is not sent out from policy director (lb) even though CPS records in engine logs that the message (RAR, ASR and so on) has been sent. It is an intermittent behavior.

The following logs can be seen on the occurrence of this issue:

```
qns02 qns02 2016-02-22 18:07:31,634 [pool-2-thread-1] DEBUG c.b.d.p.registry.EndpointRegistry
- No
endpoint available and current endpoint is down lb01-4:diameter-lb site null
qns02 qns02 2016-02-22 18:07:31,634 [pool-2-thread-1] DEBUG c.b.d.p.registry.EndpointRegistry
-
Unable to get alternate endpoint for realm site-rx-client.com, host site-host-rx. Setting
destination to null.
```

**Possible Cause:** This issue may occur if the correct value of the parameter `-Ddiameter.peer.reload.interval` is not configured in `/etc/broadhop/qns.conf` file.

CPS reloads the peer endpoints after every 30 seconds. It also reloads endpoints whenever there is an occurrence of peer flapping or new peer tries to connect.

To avoid unnecessary reloading of endpoints CPS checks that if endpoint reload request comes within the 3 second interval after 30 seconds regular reload. If the reload request does not come within the stipulated time CPS does not allow to reload.

Sometimes if request comes within this 3 second interval then the request is not processed and endpoints are not loaded due to which the message in question at that time will not be sent out from policy director (lb) though it is visible in engine logs that the message has been sent out.

**Solution:** This 3 second interval can be tweaked using `-Ddiameter.peer.reload.interval` parameter in `/etc/broadhop/qns.conf` file.

If it is kept to default value (0 millisecond) then there is a very less probability of collision so a 0 millisecond or very small value is advisable.

## Time Zone and Location Information Not Received

**Case:** Sometimes the time zone information and the location information are not received in AAA response message from CPS.

**Solution:**

- Ensure that you have Rx-Client configured under the Diameter Client (in Reference Data tab) matching the realm received in the AAR message. The Rx Client should have **Send timezone and location info in AAA Response** check box checked.
- Enable logging is set to debug level for Diameter and check for the following log messages:
  - `Unable to decode timezone value {} in gx-session with msg {}` indicates that the time zone information received from PCEF is not in correct hex format.
  - `3gpp-ms-timezone value not found on Gx session to send in AAA response` indicates that CPS did not receive the 3GPP-MS-TimeZone AVP in the CCR message.
  - `3gpp-user-location-info value not found on Gx session to send in AAA response` indicates that CPS did not receive the 3GPP-User-Location-Info AVP in the CCR message.
- Ensure that the `UE_LOCATION_CHANGE` and `UE_TIMEZONE_CHANGE` event triggers are enabled to receive the updated time zone and location information.

## Admin Database shows Problem in Connecting to the Server

**Case:** Admin database shows problem in connecting to the server in diagnostics. It throws the following error message while checking replica set status.

```
diagnostics.sh --get_replica_status
```

```
Current setup have problem while connecting to the server on port 27721
```

**Possible Cause:** The oplog collection is a circular capped collection. It is possible that the corruption occurred due to abrupt failure of VM and exception comes when the collection wrapped around to the corrupted region.

- Check which specific replica-set member is corrupted. It can be verified using `rs.status()` command.

For example,

```
mongo sessionmgr01:27721
>rs.status()
  "_id" : 3,
  "name" : "L3-CA-SEC-sessionmgr01:27721",
  "health" : 1,
  "state" : 2,
  "stateStr" : "SECONDARY",
  "lastHeartbeatMessage" : "syncThread: 17322 write to oplog failed: InternalError no space in capped collection",
  "syncingTo" : "L3-CA-SEC-sessionmgr02:27721"
```

- Also verify if mongo logs shows the following related errors:

```

2016-03-09T14:33:24.101+0530 [rsHealthPoll] replSet member L3-CA-SEC-sessionmgr01:27721
  is up
2016-03-09T14:33:24.101+0530 [rsHealthPoll] replSet member L3-CA-SEC-sessionmgr01:27721
  is now in state SECONDARY
2016-03-09T14:33:29.801+0530 [rsSync] couldn't make room for new record (len: 172) in
  capped ns local.oplog.rs
2016-03-09T14:33:29.801+0530 [rsSync]   Extent 0
2016-03-09T14:33:29.801+0530 [rsSync]   (capExtent)
2016-03-09T14:33:29.801+0530 [rsSync]
2016-03-09T14:33:29.801+0530 [rsSync]   magic: 41424344 extent->ns: local.oplog.rs
2016-03-09T14:33:29.801+0530 [rsSync]   fr: null lr: 1:1b8dedd4 extent->len: 1073741824
2016-03-09T14:33:29.801+0530 [rsSync] local.oplog.rs Assertion failure len * 5 >
  _lastExtentSize
src/mongo/db/structure/catalog/namespaces_details.cpp 366

```

**Solution:** Make sure there is at least one surviving member that is primary database member using `rs.status()` command.

1. Stop mongo process.

```
/etc/init.d/sessionmgr-27721 stop
```

2. Go to the data directory.

For example,

```
cd /var/data/sessions.3
```

3. Take backup of local file at a temporary location.

```

ls -l local*

-rw----- 1 root root 67108864 Jan 7 2253 local.0
-rw----- 1 root root 2146435072 Jan 27 0251 local.1
-rw----- 1 root root 16777216 Jan 27 0251 local.ns

```

4. Remove the local files.

```
rm -rf local.*
```

5. Start the mongo process.

```
/etc/init.d/sessionmgr-27719 start
```

6. Check whether the local files have been re-created again.

```

ls -l local*

-rw----- 1 root root 67108864 Jan 7 2253 local.0
-rw----- 1 root root 2146435072 Jan 27 0251 local.1
-rw----- 1 root root 16777216 Jan 27 0251 local.ns

```

7. Repeat Step 1 to Step 6 for other corrupted member.

## Locale MAC Error

**Case:** Locale environment variables related errors are observed by MAC users while running CPS scripts/logging into CPS VMs, and so on.

**Possible Cause:** CPS VMs show `-bash: warning: setlocale: LC_CTYPE: cannot change locale (UTF-8): No such file or directory` warning while login through SSH and if we login to such terminal then there is some issue while creating replica-sets.

**Solution:**

1. Go to **Terminal > Preferences > Profiles > Advanced** tab.

*Figure 26: Advanced Tab*



2. Uncheck **Set locale environment variables on startup** and restart your terminal.

## Sessions Stored in a Single Shard

**Issue:** If the user has recently added/modified shards but is seeing all the sessions created in first shard only.

**Symptoms:** Run the `session_cache_ops.sh` script and check the session count. Here in this case all sessions are stored in shard 1 (102543) as the rest of the shards have count 0.

```
session_cache_ops.sh --count
```

```

Session cache operation script
Thu Jun 16 02:07:45 EDT 2016
-----
Session Replica-set SESSION-SET1
-----
Session Database      : Session Count
-----
session_cache         : 102543
session_cache_2       : 0
session_cache_3       : 0
session_cache_4       : 0
-----
No of Sessions in SET1 : 0
-----

-----
Session Replica-set SESSION-SET2
-----
Session Database      : Session Count
-----
session_cache         : 0
session_cache_2       : 0
session_cache_3       : 0
session_cache_4       : 0
-----
No of Sessions in SET2 : 0
-----

Total Number of Sessions : 0

```

**Solution:**

1. Find admin database in use.

Open Policy Builder, check for **Admin Database** under Cluster configuration.

2. Login to admin database and run the following query:

For example, mongo sessionmgr01:27721

```

set01:PRIMARY> use sharding
switched to db sharding
set01:PRIMARY> db.buckets.count({"shard":1})
8192

```

If the bucket count is 8192 (means all the buckets are in shard 1 and rebalance is required).

3. Run `rebalance` and `migrate` OSGi commands from perfclient01.




---

**Note** For production environment, this needs to be done in maintenance window.

---

```
echo "rebalance" | nc qns01 9091
```

```
echo "migrate" | nc qns01 9091
```

4. After rebalance is completed, verify by re-running `session_cache_ops.sh --count` script.

## Licensing not Throwing Traps or Diagnostic Errors upon Breach

**Issue:** Licensing is not throwing traps or diagnostic errors upon breach.

**Symptoms:** Application traps are not generated.

**Solution:** Check log level for logger `com.broadhop.eventlogging` in `/etc/broadhop/qns.conf` file.

```
<logger name="com.broadhop.eventlogging" level="info">
  <appender-ref ref="JSON-LOGGER" />
</logger>
```




---

**Important** Log level must be set to *info*.

---

If log level is set to *warn/error*, SNMP traps related to licensing will not be generated. You need to change the log level to *info* to generate traps related to licensing.

After changing the log level to *info*, execute the following commands:

```
/var/qps/install/current/scripts/build_all.sh
/var/qps/install/current/scripts/upgrade/reinit.sh
```

## Corosync Process Taking lot of Time to Unload and is Stuck

**Issue:** The corosync process is taking a lot of time to unload and is stuck.

**Solution:** If user finds corosync process is stuck, while doing `monit restart corosync` OR `monit stop corosync`, perform the following steps:

---

**Step 1** Exit from the process by pressing `Ctrl+c`.

**Step 2** Note down corosync process *pid* by executing the following command:

```
cat /var/run/corosync.pid
```

**Step 3** Stop corosync and its child processes by executing the following command:

```
kill -2 <corosync process pid>
```

**Step 4** Check whether all the corosync and all the child processes are stopped by executing the following command:

```
ps -ef | grep "corosync\|pacemaker"
```

**Step 5** If you are still seeing that the processes are UP then kill all the processes (corosync and pacemaker), which are shown in [Step 4, on page 88](#) by executing the following commands:

```
kill -9 <all pid of processes, space seprated>
```

---

## Issue related to Firewall

**Issue:** When firewall state is changed from enabled to disabled state or vice versa, sometimes firewall is not completely purged on some VMs.



**Solution:** Perform the following when you are changing firewall state:

- If the firewall state is being changed from disabled to enabled, execute the following command twice after SSH to VM:

```
vm-init-client.sh
```

- If the firewall state is being changed from enabled to disabled, check if `/etc/sysconfig/iptables` has any old rules by executing the following command by SSH to VM:

```
/etc/init.d/vm-init.sh
```

If Yes, delete it manually and execute the following command again on a VM:

```
vm-init-client.sh
```

## CPS Setup cannot Handle High TPS

**Issue:** Too many request to query mongo.

**Case:** Consider user is running a very basic call model (Gx only) on CPS setup. On reaching close to 15K TPS (1 CCR-I, 3- CCR-U and 1-CCR-T), timeouts are observed in grafana and average response time goes up.

Response time:

```
[root@pcrfclient01 ~]# mongostat --host sessionmgr01 --port 27717
connected to: sessionmgr01:27717
insert query update delete getmore command flushes mapped vsize res faults locked db idx
miss % qr|qw ar|aw netIn
netOut conn set repl time
1055 17691 3868 921 248 328|0 0 5.03g 5.84g 4.12g 0 session_cache:32.0% 0 73|0 0|4 19m 33m
 137 set01 PRI 15:26:15
973 16848 3568 752 200 257|0 0 5.03g 5.84g 4.12g 0 session_cache_2:29.7% 0 4|0 6|1 18m 31m
 138 set01 PRI 15:26:16
1049 17023 4162 809 215 283|0 0 5.03g 5.84g 4.12g 0 session_cache_3:33.7% 0 74|1 2|3 20m
35m 138 set01 PRI 15:26:17
1010 17200 3956 804 211 278|0 0 5.03g 5.84g 4.12g 0 session_cache_4:32.1% 0 18|0 3|1 19m
33m 137 set01 PRI 15:26:18
975 17027 3912 816 206 275|0 0 5.03g 5.84g 4.13g 0 session_cache:31.8% 0 34|0 0|4 19m 33m
138 set01 PRI 15:26:19
990 16631 3643 904 204 279|0 0 5.03g 5.84g 4.13g 0 session_cache_2:34.4% 0 58|0 0|4 18m 31m
 138 set01 PRI 15:26:20
913 15475 3806 881 185 257|0 0 5.03g 5.84g 4.13g 0 session_cache_4:30.7% 0 40|0 1|3 18m 32m
 140 set01 PRI 15:26:21
```

**Possible Cause:** Selecting the following settings in Policy Builder, system sends multiple queries to mongo which impacts the performance. (Timeouts are observed and average response time goes up.)

1. **Enable Multi Primary Key** check box is selected for your system.
2. **Load By Nai, Imsi Based Nai, Limit with Requested QoS on modification failure** check boxes are selected under your Diameter client.
3. **Max Timer T P S** value has been configured to 2000 for your cluster.
4. **Re-evaluation diffusion interval** value has been configured to 20000 for your cluster.

**Solution:**

1. Login to Cisco Policy Builder.

2. From left side, select *name of your system* and uncheck **Enable Multi Primary Key** check box.
3. Under **Diameter Clients**, expand Gx Clients and select the *name of your Gx client*. For example, Gx.
4. Uncheck **Load By Nai, Imsi Based Nai, Limit with Requested QoS on modification failure** check boxes.
5. From left side, select *name of your cluster* under *name of your system*.
6. From right side pane, update **Max Timer T P S** value to 100.
7. Also, change the **Re-evaluation diffusion interval** value to 20 milliseconds for your cluster.

## CPS System is Crashing when Running More than 6K TPS

**Issue:** High response time is observed when system is running with all the default features installed and has Gx traffic with 6K TPS.

**Consideration:** It is recommended to create session replica-set as per performance requirements for scaling.

**Solution:**

- 
- Step 1** Create/update `/etc/broadhop/mongoConfig.cfg` file on Cluster Manager VM to create session cache shards in criss-cross manner.

```
[SESSION-SET1]
SETNAME=set01
OPLOG_SIZE=5120
ARBITER1=arbitervip:27717
ARBITER_DATA_PATH=/var/data/sessions.1
MEMBER1=sessionmgr01:27717
MEMBER2=sessionmgr02:27717
DATA_PATH=/var/data/sessions.1/1
[SESSION-SET1-END]

[SESSION-SET2]
SETNAME=set07
OPLOG_SIZE=5120
ARBITER1=arbitervip:27727
ARBITER_DATA_PATH=/var/data/sessions.7
MEMBER1=sessionmgr02:27727
MEMBER2=sessionmgr01:27727
DATA_PATH=/var/data/sessions.1/2
[SESSION-SET2-END]
```

- Step 2** Refer to *Create Specific Replica-set* and *Session Cache Replica-set* sections in *CPS Installation Guide for VMware* for further information on how to create replica sets.

- Step 3** Set session database priority so that the PRIMARY members will be on separate VM:

```
cd /var/qps/bin/support/mongo
./set_priority.sh --db session
```

For more information on `set_priority.sh` script, refer to *CPS Operations Guide* and *CPS Geographic Redundancy Guide*.

**Step 4** To create session shards, refer to the *Create Session Shards* section in *CPS Installation Guide for VMware*.

---

## Old VIP is not deleted After Modifying VIP Name

If VIP name is modified then user has to manually delete old VIP from active policy director (lb)/OAM (pcrfclient) using the following below command:

```
pcs resource delete <old-vip-name>
```

where, <old-vip-name> is the old VIP name.

## lbvip not moving to Secondary Policy Director (lb) VM

**Issue:** lbvip does not move cleanly to the secondary policy director (lb) VM when the network on primary policy director (lb) VM is stopped.

**Scenario:** For example, consider lbvip is on lb01 VM.

To stop the network on lb01 VM, execute the following command:

```
service network stop
```

lbvip moves to lb02 VM immediately but it is not pingable from anywhere which stops the traffic and grafana.

After performing `service network restart` on lb02 VM, the traffic restored partially with lot of errors (and lbvip is pingable from everywhere).

After stopping the network on lb01 VM, lbvip was seen on both the lb VMs (even after doing network restart on lb02 VM).

**Solution:**

Before executing `service network stop`, stop corosync from the node using `monit stop corosync` command.



---

**Note** This is needed since corosync has the functionality to bring up an interface if they are down. So after `service network stop` is executed all interfaces are down and corosync brings up the interfaces (like, eth0:0, eth1:0, and so on).

---

## Internal Session Sharding not Recovered on Power Outage

**Case:** `session_cache_ops.sh --count` shows the error: `There is no session db found`.

**Symptoms:** On power outage, if all the members of session database replica were down and came up, session database replica-set will get automatically re-created. However after this if you run `session_cache_ops.sh --count` script, it may show the error: `There is no session db found`.

This is because, script do not create `session_cache` databases on its own. `session_cache` databases would get created by application automatically when calls would run.

Here is an example:

```
Session cache operation script
Mon Oct 17 05:40:34 EDT 2016
```

There is no session db found for site SITE1

There is no session db found for site SITE1

There is no session db found for site SITE1

There is no session db found for site SITE1

**Validate:** Run `listshards` command to verify the internal shards are not deleted.

For HA, run the following command:

```
echo "listshards" | nc qns01 9091
```

For Active active GR, run the following command:

```
echo "listshards <site name>" | nc qns01 9091
```

This command displays all the shards configured in the system.

Here is an example:

```
echo "listshards clusterA_PRI" | nc qns01 9091
```

```
osgi>
```

Shard Id	Mongo DB	State	Backup DB	Removed	Session Count
1	sessionmgr01:27717/session_cache	online	false	false	261758
2	sessionmgr01:27717/session_cache_2	online	false	false	261439
3	sessionmgr01:27717/session_cache_3	online	false	false	261139
4	sessionmgr01:27717/session_cache_4	online	false	false	262069
5	sessionmgr03:27722/session_cache	online	false	false	261984
6	sessionmgr03:27722/session_cache_2	online	false	false	260759
7	sessionmgr03:27722/session_cache_3	online	false	false	262147
8	sessionmgr03:27722/session_cache_4	online	false	false	262087
9	sessionmgr05:27723/session_cache	online	false	false	261627
10	sessionmgr05:27723/session_cache_2	online	false	false	262118
11	sessionmgr05:27723/session_cache_3	online	false	false	262088
12	sessionmgr05:27723/session_cache_4	online	false	false	261775
13	sessionmgr09:47717/session_cache	online	true	false	0
14	sessionmgr09:47717/session_cache_2	online	true	false	0
15	sessionmgr09:47717/session_cache_3	online	true	false	0
16	sessionmgr09:47717/session_cache_4	online	true	false	0

Rebalance Status: Rebalanced

## Flow Information Parameters Not Derived As Per Actions

If the Flow-Information parameters are not derived as per the Actions (Enforce/Mirror), check for the following configuration details:

- Ensure that the AVP name defined in the **ColumnAndAvpPair** in the RxSTGConfiguration service option is the correct action name ("Flow Status" or "Flow Description") to derive the action. Also check the 3GPP AVP name to derive the CRD value for the field.
- Confirm that the action values in the CRD are correctly entered (Mirror/Enforce).
- Check if the Flow-Description value defined in the CRD for Enforce action is in correct format. If the `IPFilterRule` syntax is not proper then CPS logs a warn message to indicate the same.
- Enable logs at debug level and confirm the CRD evaluation and logs for applying the action of the individual fields in the flow information.

## Mapped Target AVPs Not Received In Diameter Message

If the source AVP is a Session/Policy State Data Retriever, the target AVPs are sent in the outbound diameter messages only if the data to be retrieved is available in the Session/Policy State data.

If the mapped target AVPs are not received in the respective diameter message, check for the following:

- Ensure that you have correctly configured **Custom Avp Table** and **Avp Mappings** under **Reference Data > Diameter Defaults**. Verify that the mandatory fields (marked with asterisk) are configured correctly.
- Enable logs at debug level for Diameter and check for the following log messages:
  - `AVP Mappings are not defined`  
Indicates that no AVP mappings have been defined in Policy Builder.
  - `No mappings found for {}`  
Indicates that no target AVP mappings were found for a particular policy derived attribute.
  - `submit object is null or not a message`  
Indicates that CPS did not receive a Diameter message.

## Running Puppet on Cluster Manager in HA Setup

**Issue:** After applying patch or updating kernel in HA setup, when you run `puppet apply` command `/etc/httpd/conf/httpd.conf` file was modified, not all VMs are configured with the modified `httpd.conf` file:

**Solution:** After applying a patch or updating kernel in HA setup, run the following command from Cluster Manager:

```
puppet apply --logdest=/var/log/cluman/puppet-custom-run.log --
modulepath=/opt/cluman/puppet/modules --config=/opt/cluman/puppet/ puppet.conf
/opt/cluman/puppet/nodes/node_repo.pp
```



**Note** Manually enter `puppet apply` command in your system.

After applying the `puppet apply` command, run the following command from Cluster Manager to update the `/etc/httpd/conf/httpd.conf` file on all VMs:

```
/var/qps/install/current/scripts/modules/update_httpd_conf.py
```

## Not Able to Rebalance and Migrate after Shards Recreation

**Case:** Unable to complete rebalance operation displaying following error:

```
osgi> rebalance GR-S1
Rebalancing ...
Unable to complete operation.
```

**Reason:**

- Instance `site1-qns90-1` is on older version (1) of sharding configuration. Latest version is: (10)
- Make sure all the instances are up and running and try running `rebalance` command again
- If the instance in the error does not exist, manually clear it from instances collection (`admindb/sharding`) and try running `rebalance` command again

**Possible Cause:** `qns90` VM is either deleted or is no more in use or policy server (QNS) process is down

**Solution:**

If QNS process was down unintentionally, perform the following steps:

1. Start the policy server (`qns`) process.
2. Verify that the diagnostics is clear.
3. Run the `rebalance` command again.

If `qns90` VM is deleted or is no more in use, perform the follow steps:

1. Login to the admin database.
2. Clear the instance in error from "instances" collection.

```
use sharding
```

- a. Delete the instance in error.

```
db.instances.find({"_id":"Site1-qns90-1"})
db.instances.remove({"_id":"Site1-qns90-1"})
```

- b. Reload configuration.

```
db.config.update({},{$inc:{"version": NumberInt(1)}})
db.changes.insert({"ts":new Timestamp(), "change" : "Manually deleting qns90"})
```

3. Run the `rebalance` command again.

## pcrfclient01 Automatically Becomes Unresponsive

**Case:** pcrfclient01 automatically becomes unresponsive after some time in OpenStack environment

**Condition:** You are not able to SSH to any VM.

**Solution:** Refer to the following fix:

The Errata for OSP10 (Newton) is:

RHSA-2018\*0058 - Security Advisory Issued:2018-01-05 Updated: 2018-01-05 Errata:

<https://access.redhat.com/errata/RHSA-2018T0058>

Fixes:

- BZ - 1519780 - CVE-2017-5715 hw: cpu: speculative execution branch target injection
- BZ - 1525502 - QEMU's subsystem gets stuck inhibiting all I/O operations on virtio-blk-pci devices [OSP 10]

## Primary Member Isolated from all Arbiters Displaying Incorrect State

**Issue:** If the primary database member gets isolated from all the arbiters then diagnostics output displays incorrect state for the primary member.

**Solution:** If a member is displayed in an unknown state when using `diagnostics.sh --get_replica_status` command, it is likely that the member is not accessible from one of other members, mostly an arbiter. In that case, you must go to that member and check its connectivity with other members.

Also, you can login to mongo on that member and check its actual status.

## No Alarm is Generated When Mongo Process Stop/Restart

**Case:** No SNMP alarm is generated when Mongo process is stopped or restarted.

**Possible Cause:** This can happen due to the timing difference between AIDO functionality and SNMP alarm schedule.

- When the Mongo process is stopped, it gets restarted by AIDO.
- Before SNMP scheduler detects the Mongo failure, it is restarted by AIDO. Hence, no alarm is generated.

**Solution:**

If you want to stop AIDO from restarting the Mongo process, you need to create `/var/tmp/stopped-PORT` file on the AIDO client node.

This stops AIDO from monitoring or handling the Mongo process. The alarm is generated when Mongo process is stopped or restarted.

**Example:** During ISSU, you need to stop and start mongod process which is done by upgrade script. So, you need to execute the following steps:

1. Create `/var/tmp/stopped-PORT` file on AIDO client node.
2. Stop mongod using `/etc/init.d/sessionmgr-PORT stop`

3. Upgrade your system (ISSU). AIDO does not restart the mongod process as /vat/tmp/stopped-PORT file is created on AIDO client node.
4. Start mongod using /etc/init.d/sessionmgr-PORT start
5. Remove /vat/tmp/stopped-PORT file from AIDO client node

## Zookeeper becoming Unavailable on Cluster Manager

**Case:** Zookeeper in an endless restart cycle. In monit summary, the zookeeper-server is displayed in Initializing state.

```
Process 'zookeeper-server'           Initializing
```

**Possible Cause:** Zookeeper server with version 3.4.6 fails to start.

**Solution:** Go to /var/opt/zookeeper/version-2 and delete zero byte file or move the file to some other location.

```
ls -l
total 1988
-rw-r--r-- 1 root root 67108880 Feb 20 2018 log.1
-rw-r--r-- 1 root root 67108880 Aug 23 04:58 log.1f43
-rw-r--r-- 1 root root          0 Aug 23 04:58 log.3lfd
```

After moving out zero byte file from /var/opt/zookeeper/version-2, check monit summary. The zookeeper-server must be in running state.

## Upgrade Fails due to monit Race Condition

**Case:** Upgrade fails when running puppet.

During upgrade, the following error is encountered:

```
2018-12-10 05:10:17,909 INFO [__main__.run_recipe] Performing installation stage:
ApplyClumanPuppet
2018-12-10 05:10:17,910 INFO [cluman_puppet.run] Applying puppet on cluman
2018-12-10 05:10:53,133 ERROR [cluman_puppet.run] Puppet failed! For details check log
/var/log/cluman/puppet-run.log
2018-12-10 05:10:53,133 ERROR [__main__.<module>] Error during installation
2018-12-10 05:10:53,133 ERROR [__main__.<module>]
Traceback (most recent call last):
  File "/mnt/iso/modules/install/__main__.py", line 780, in <module>
    main(sys.argv[1:])
  File "/mnt/iso/modules/install/__main__.py", line 770, in main
    install(argv[1:])
  File "/mnt/iso/modules/install/__main__.py", line 208, in install
    run_recipe(install_recipe)
  File "/mnt/iso/modules/install/__main__.py", line 763, in run_recipe
    stage.run()
  File "install/cluman_puppet.py", line 55, in run
    puppet_cmd.execute()
  File "util/command.py", line 106, in execute
    raise RuntimeError(' '.join(self.command) + ' returned ' + str(self.exitcode) + ' instead
of ' + str(self.expected_exitcode))
RuntimeError: /usr/bin/puppet apply --detailed-exitcodes --debug --verbose --logdest
/var/log/cluman/puppet-run.log --modulepath=/opt/cluman/puppet/modules --config
/opt/cluman/puppet/puppet.conf /opt/cluman/puppet/nodes/node_repo.pp returned 6 instead of
[0, 2]
2018-12-10 05:10:53,134 INFO [__main__.<module>] =====
```



```
2018-12-10 05:10:53,134 INFO [__main__.<module>] FAILURE
2018-12-10 05:10:53,134 INFO [__main__.<module>] ===== END =====
2018-12-10 05:10:53,134 INFO [__main__.<module>] To have the environment variable updated,
please logout and login from all opened shell on the current system
```

**Solution:** Re-run the `install.sh`.

## Messages Timed Out When Running Heap Dump

**Case:** Messages timed out when running Heap Dump of Policy Server (QNS)/Policy Director (LB) process on Policy Server (QNS)/Policy Director (LB) VM.

**Condition:** Taking heap dump of Policy Server (QNS)/Policy Director (LB) process on Policy Server (QNS)/Policy Director (LB) VM. Heap dumps taken results in full GC. This in turn results in application pause which causes message time out.

**Solution:** It is recommended to take the heap dump during Maintenance Window (MW).

## mongod Process Not Running on both pcrfclient after Fresh Install

**Case:** mongod process not running on both pcrfclients after fresh installation.

`pcs status` throwing not installed error as follows:

```
pcs status
Cluster name: cps
WARNING: corosync and pacemaker node names do not match (IPs used in setup?)
Stack: corosync
Current DC: pcrfclientXX (version 1.1.18-11.el7_5.3-2b07d5c5a9) - partition with quorum
Last updated: Thu Jan 17 14:09:38 2019
Last change: Thu Jan 17 12:27:52 2019 by root via cibadmin on pcrfclientXX
```

```
2 nodes configured
10 resources configured
```

```
Online: [ pcrfclientXX pcrfclientXX ]
```

Full list of resources:

```
Resource Group: mongod
  arbitervip (ocf::heartbeat:IPaddr2):          Started pcrfclientXX
  sessionmgr-27721 (systemd:sessionmgr-27721):  Stopped
  sessionmgr-27717 (systemd:sessionmgr-27717):  Stopped
  sessionmgr-27727 (systemd:sessionmgr-27727):  Stopped
  sessionmgr-27017 (systemd:sessionmgr-27017):  Stopped
  sessionmgr-27718 (systemd:sessionmgr-27718):  Stopped
  sessionmgr-27719 (systemd:sessionmgr-27719):  Stopped
  sessionmgr-27720 (systemd:sessionmgr-27720):  Stopped
Clone Set: PingIP-clone [PingIP]
  Started: [ AB-RT-pcrfclient01 AB-RT-pcrfclient02 ]
```

Failed Actions:

```
* sessionmgr-27721_start_0 on pcrfclientXX 'not installed' (5): call=12, status=Not installed,
  exitreason='',
  last-rc-change='Thu Jan 17 12:25:40 2019', queued=0ms, exec=41ms
* sessionmgr-27721_start_0 on pcrfclientXX 'not installed' (5): call=39, status=Not installed,
  exitreason='',
  last-rc-change='Thu Jan 17 12:36:59 2019', queued=0ms, exec=40ms
* sessionmgr-27717_start_0 on pcrfclientXX 'not installed' (5): call=40, status=Not installed,
  exitreason='',
  last-rc-change='Thu Jan 17 12:36:59 2019', queued=1ms, exec=77ms
```

```
* sessionmgr-27727_start_0 on pcrfclientXX 'not installed' (5): call=41, status=Not installed,
exitreason='',
  last-rc-change='Thu Jan 17 12:36:59 2019', queued=0ms, exec=115ms
* sessionmgr-27017_start_0 on pcrfclientXX 'not installed' (5): call=42, status=Not installed,
exitreason='',
  last-rc-change='Thu Jan 17 12:36:59 2019', queued=0ms, exec=151ms
* sessionmgr-27718_start_0 on pcrfclientXX 'not installed' (5): call=44, status=Not installed,
exitreason='',
  last-rc-change='Thu Jan 17 12:36:59 2019', queued=0ms, exec=159ms

Daemon Status:
corosync: active/disabled
pacemaker: active/disabled
pcsd: inactive/disabled
```

**Solution:** On pcrfclient, run `crm_resource --clean` command.

## Replica-set ID is Getting Changed after pcrfclient (arbiter) Failover

**Case:** When the pcrfclient and the sessionmgr VM's are rebooted, it is observed that some of the session managers go offline.

**Reason:** Some session managers are not able to connect to arbiter because of `InvalidReplicaSetConfig: replica-set IDs do not match` error. This is because the replica-set ID in arbiter and sessionmgr's does not match.

**Solution:** Perform the following steps to change database path in `mongoConfig.cfg` file:

1. Remove reporting replica-set.

```
build_set.sh --report --remove-replica-set --setname replica-setID
```

For example, `build_set.sh --report --remove-replica-set --setname set03`

2. Update `mongoConfig.cfg` file with the new database path.
3. Run the `/var/qps/install/current/scripts/build/build_etc.sh` script from the Cluster Manager to finalize `mongoConfig.cfg` file.
4. Create replica-set with force option.

```
build_set.sh --report --create --setname replica-setID --force
```

For example, `build_set.sh --report --create --setname set03 --force`

5. Verify the new database path in mongod instance.

## Errors/Warnings Observed during PS Node Warmup

**Case:** The warmup solution warms up the PS node(s) by internally initiating configured number of warm up messages prior to connecting to the load balancer node and processing external calls. The warm up messages are processed by policy engine like regular messages and sessions are created/deleted in session database. The actual response/request messages generated in response to incoming warm up requests are not sent towards Policy Director (lb) node, as there is no connectivity established towards Policy Director (lb). As Policy Director (lb) nodes accept traffic during processing node(s) warm up phase, it can result in some call failures (3004) towards DRA/Gateway.

Due to non-connectivity towards Policy Director (Ib), if PS nodes are restarted (`restartall.sh` or individual restart) the following errors/warnings can be observed.



**Caution** Executing `restartall.sh` will cause messages to be dropped.



**Note** The errors/warnings are not seen after all the nodes are warmed up.

1. Some calls can be rejected with 3004 (DIAMETER\_TOO\_BUSY) error code.
2. The following Warnings/Errors can be seen in qns logs.

```
WARN c.b.d.p.event.DiameterMessageDealer - Unable to send message
healthCheckHost.healthCheckRealm;1523905102;13492
    site-1-pps06 site-1-pps06 2019-02-13 09:41:06,775
[pool-2-thread-1] ERROR c.b.u.zmq.nodes.PushConnection - Exception sending message
com.broadhop.exception.BroadhopException: No channels available
    at
com.broadhop.utilities.zmq.nodes.PushConnection.getNextChannel(PushConnection.java:252)
~[com.broadhop.utility_14.0.1.r132522.jar:na]
    at
com.broadhop.utilities.zmq.nodes.PushConnection.send(PushConnection.java:285)
~[com.broadhop.utility_14.0.1.r132522.jar:na]
    at
com.broadhop.utilities.zmq.WorkerNode.send(WorkerNode.java:265)
[com.broadhop.utility_14.0.1.r132522.jar:na]
    at
com.broadhop.diameter2.policy.event.DiameterMessageDealer.processOutboundMessage(DiameterMessageDealer.java:671)
[com.broadhop.diameter2.policy.endpoint_14.0.1.r132523.jar:n]
    at
com.broadhop.diameter2.policy.event.DiameterMessageDealer.submit(DiameterMessageDealer.java:645)
[com.broadhop.diameter2.policy.endpoint_14.0.1.r132523.jar:na]
    at
com.broadhop.diameter2.policy.actions.SendDiameterResponse.execute(SendDiameterResponse.java:63)
[com.broadhop.diameter2.policy.endpoint_14.0.1.r132523.jar:na]
    at
com.broadhop.utilities.policy.async.PolicyLocalAsyncActionRunnable.run(PolicyLocalAsyncActionRunnable.java:33)
[com.broadhop.utility_14.0.1.r132522.jar:na]
    at
java.util.concurrent.Executors$RunnableAdapter.call(Executors.java:511) [na:1.8.0_72]
    at
java.util.concurrent.FutureTask.run(FutureTask.java:266) [na:1.8.0_72]

ERROR c.b.policy.remote.jms.JmsReceiver - Error processing and deserializing incoming
message com.esotericsoftware.kryo.SerializationException: Unable to deserialize object
of type: com.broadhop.policy.remote.RemoteActionRequest
    at
com.esotericsoftware.kryo.Kryo.readClassAndObject(Kryo.java:571) ~[na:na]
    at
com.esotericsoftware.kryo.ObjectBuffer.readClassAndObject(ObjectBuffer.java:206) ~[na:na]
    at
com.broadhop.policy.remote.jms.Jms.deserialize(Jms.java:271)
~[com.broadhop.policy.remote.jms_14.0.1.r131154.jar:na]
    at
com.broadhop.policy.remote.jms.JmsReceiver.onMessage(JmsReceiver.java:170)
~[com.broadhop.policy.remote.jms_14.0.1.r131154.jar:na]
    at
org.apache.activemq.ActiveMQMessageConsumer.dispatch(ActiveMQMessageConsumer.java:1361)
```

Total Number of Session Exceeding Allowed Limit

```
[activemq-all-5.9.0.jar:5.9.0]
    at
org.apache.activemq.ActiveMQSessionExecutor.dispatch(ActiveMQSessionExecutor.java:131)
[activemq-all-5.9.0.jar:5.9.0]
    at
org.apache.activemq.ActiveMQSessionExecutor.iterate(ActiveMQSessionExecutor.java:202)
[activemq-all-5.9.0.jar:5.9.0]
    at
org.apache.activemq.thread.PooledTaskRunner.runTask(PooledTaskRunner.java:129)
[activemq-all-5.9.0.jar:5.9.0]
    at
org.apache.activemq.thread.PooledTaskRunner$1.run(PooledTaskRunner.java:47)
[activemq-all-5.9.0.jar:5.9.0]
    at
java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1142)
[na:1.8.0_72]
    at
java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:617)
[na:1.8.0_72]
    at java.lang.Thread.run(Thread.java:745)
[na:1.8.0_72]
    Caused by: com.esotericsoftware.kryo.SerializationException:
Unable to find class: com.broadhop.diameter2.actions.ICreateClientSession
```

## Total Number of Session Exceeding Allowed Limit

**Case:** Session Limit Overload Protection is set to 400000 in Policy Builder. But when the calls are run at 2K TPS (CCR-I), the total number of sessions are growing till 475049 which is more than the allowed limit. However, CPS does not allow the sessions to grow after this, but still this is about 20% more than the allowed limit.

Mon Mar 4 09:55:52 UTC 2019

\*\*\* End-of-Collection \*\*\*

-----  
Host Detail:

qns10, qns07, qns13, qns05, qns16, qns08, qns12  
qns01, qns17, qns11, qns14, qns04, qns02, qns03  
qns19, qns06, qns20, qns18, qns15, qns09  
Measurement timer: 1      QNS Count: 20  
-----

Average	Success	TPS	Error	Time Used	Messages
16.9670	2100	2100.0000	0	35.6306	diameter_Gx_CCR-I
13.1713	2005	2005.0000	0	26.4084	diameter_Rx_AAR

Average	Success	TPS	Error	Time Used	Actions
3.2845	6358	6358.0000	0	20.8826	
6.7885	2100	2100.0000	0	14.2558	com.broadhop.session.CreateEntry
4.6131	2218	2218.0000	0	10.2319	com.broadhop.session.UpdateEntry
0.6030	8551	8551.0000	0	5.1565	ResolveServices
0.2736	8550	8550.0000	0	2.3397	ResolveServiceOptions
0.1966	2223	2223.0000	0	0.4371	send.diameter_Gx_RAR
0.1638	2223	2223.0000	0	0.3640	send.diameter_Rx_AAA
0.1436	2100	2100.0000	0	0.3016	send.diameter_Gx_CCA-I
0.0404	4231	4231.0000	0	0.1709	
0.0148	8544	8544.0000	0	0.1266	BundleVirtualServices
0.0181	4231	4231.0000	0	0.0766	
					com.broadhop.policyintel.impl.actions.StartPolicyReporting

```
0.0049      4290  4290.0000      0      0.0211
com.broadhop.policyintel.impl.actions.StopPolicyReporting
```

Mon Mar 4 09:55:53 UTC 2019

\*\*\* End-of-Collection \*\*\*

^C

\*\*\* Exiting \*\*\*

```
[root@pcrfclient01 ~]# session_cache_ops.sh --count|grep Total;date
Total Number of Sessions : 393478
Mon Mar 4 09:56:03 UTC 2019
[root@pcrfclient01 ~]# top_qps.sh
```

-----

Host Detail:

```
qns02,qns17,qns15,qns20,qns04,qns14,qns07
qns11,qns08,qns01,qns12,qns06,qns05,qns16
qns09,qns19,qns13,qns18,qns10,qns03
Measurement timer: 1      QNS Count: 20
```

-----

Average	Success	TPS	Error	Time Used	Messages
15.4398	2100	2100.0000	0	32.4236	diameter_Gx_CCR-I
16.1731	1848	1848.0000	0	29.8879	diameter_Rx_AAR

-----

Average	Success	TPS	Error	Time Used	Actions
2.8085	6600	6600.0000	0	18.5362	
com.broadhop.cache.impl.actions.GetSessionAction					
5.6463	2100	2100.0000	0	11.8572	com.broadhop.session.CreateEntry
4.4516	2398	2398.0000	0	10.6750	com.broadhop.session.UpdateEntry
0.5664	9000	9000.0000	0	5.0973	ResolveServices
0.2597	9000	9000.0000	0	2.3369	ResolveServiceOptions
0.1828	2398	2398.0000	0	0.4384	send.diameter_Gx_RAR
0.1513	2398	2398.0000	0	0.3628	send.diameter_Rx_AAA
0.1549	2100	2100.0000	0	0.3252	send.diameter_Gx_CCA-I
0.0371	4500	4500.0000	0	0.1670	
com.broadhop.policy.impl.actions.FormatTimeWithOffsetAction					
0.0137	9000	9000.0000	0	0.1234	BundleVirtualServices
0.0166	4500	4500.0000	0	0.0747	
com.broadhop.policyintel.impl.actions.StartPolicyReporting					
0.0047	4500	4500.0000	0	0.0214	
com.broadhop.policyintel.impl.actions.StopPolicyReporting					

Mon Mar 4 09:56:24 UTC 2019

\*\*\* End-of-Collection \*\*\*

^C

\*\*\* Exiting \*\*\*

```
[root@pcrfclient01 ~]# session_cache_ops.sh --count|grep Total;date
Total Number of Sessions : 452341
Mon Mar 4 09:56:36 UTC 2019
[root@pcrfclient01 ~]# session_cache_ops.sh --count|grep Total;date
Total Number of Sessions : 462917
Mon Mar 4 09:56:46 UTC 2019
[root@pcrfclient01 ~]# session_cache_ops.sh --count|grep Total;date
Total Number of Sessions : 475049
Mon Mar 4 09:57:16 UTC 2019
[root@pcrfclient01 ~]# tailf /var/log/broadhop/consolidated-sessions.log
2019-03-04 15:13:45 - TPS_COUNT:          SESSION_COUNT:
                      LICENSE_COUNT: 200000
2019-03-04 15:15:15 - TPS_COUNT:          SESSION_COUNT:
                      LICENSE_COUNT: 200000
```

Total Number of Session Exceeding Allowed Limit

```

2019-03-04 15:16:45 - TPS_COUNT:                SESSION_COUNT:
                    LICENSE_COUNT: 200000
2019-03-04 15:18:15 - TPS_COUNT:                SESSION_COUNT:
                    LICENSE_COUNT: 200000
2019-03-04 15:19:45 - TPS_COUNT:                SESSION_COUNT:
                    LICENSE_COUNT: 200000
2019-03-04 15:21:15 - TPS_COUNT:                SESSION_COUNT:
                    LICENSE_COUNT: 200000
2019-03-04 15:22:45 - TPS_COUNT:                SESSION_COUNT:
                    LICENSE_COUNT: 200000
2019-03-04 15:24:15 - TPS_COUNT:                SESSION_COUNT:
                    LICENSE_COUNT: 200000
2019-03-04 15:25:45 - TPS_COUNT:                SESSION_COUNT:
                    LICENSE_COUNT: 200000
2019-03-04 15:27:15 - TPS_COUNT:                SESSION_COUNT:
                    LICENSE_COUNT: 200000

^C
[root@pcrfclient01 ~]# grep 'sessionLimitOverloadProtection'
/var/broadhop/checkout/pcrfclient01-1/*
/var/broadhop/checkout/pcrfclient01-1/System-default-_GrTxALOLEeWR0MXs_g7BPA.xmi:
sessionLimitOverloadProtection="400000">
[root@pcrfclient01 ~]# session_cache_ops.sh --count|grep Total;date
Total Number of Sessions : 475049
Mon Mar  4 09:58:00 UTC 2019
[root@pcrfclient01 ~]# grep 'sessionLimitOverloadProtection'
/var/broadhop/checkout/pcrfclient01-1/*;date
/var/broadhop/checkout/pcrfclient01-1/System-default-_GrTxALOLEeWR0MXs_g7BPA.xmi:
sessionLimitOverloadProtection="400000">
Mon Mar  4 09:58:03 UTC 2019
[root@pcrfclient01 ~]# cat /etc/broadhop/qns.conf
QNS_COMMON_OPTS="
-Dcom.broadhop.run.systemId=system-1
-DapirouterContextPath=/ua/soap
-Dapi.ua.context.path=/ua/backend
-Dcom.broadhop.run.clusterId=cluster-scale
-Dcom.broadhop.run.instanceId=$HOSTNAME-$QNS_INSTANCE
-Dcom.broadhop.config.url=http://lbvip02/repos/run/
-Dcom.broadhop.repository.credentials.isEncrypted=true
-Dcom.broadhop.repository.credentials=qns-svn/3300901EA069E81CE29D4F77DE3C85FA@lbvip02
-Dcom.broadhop.referencedata.local.location=/var/broadhop/checkout
-DjmsRebalanceClients=true
-Denable.compression=true
-Denable.dictionary.compression=true
-DuseZlibCompression=true
-Dcom.broadhop.locking.autodiscovery=true
-DlookasideThreshold=3
-DcompressDebits
-Dnetworkguard.tcp.local=eth0
-DrefreshOnChange=true
-DenableRuntimePolling=true
-DdefaultNasIp=127.0.0.1
-Dua.version.2.0.compatible=true
-DsessionPadding=1200
-DnodeHeartBeatInterval=9000
-Dcom.mongodb.updaterIntervalMS=400
-Dcom.mongodb.updaterConnectTimeoutMS=600
-Dcom.mongodb.updaterSocketTimeoutMS=600
-DdbSocketTimeout=1000
-DdbSocketTimeout.balance=1000
-DdbConnectTimeout=1200
-DdbConnectTimeout.balance=1200
-Dmongo.client.thread.maxWaitTime=1200
-Dmongo.client.thread.maxWaitTime.balance=1200
-Dstatistics.step.interval=1

```

```

-Dmongo.connections.per.host=10
-Dmongo.connections.per.host.balance=12
-Dmongo.threads.allowed.to.wait.for.connection=12
-Dmongo.threads.allowed.to.wait.for.connection.balance=12
-DmaxLockAttempts=3
-DretryMs=3
-DmessageSlaMs=1500
-DshardPingLoopLength=3
-DshardPingCycle=200
-DshardPingerTimeoutMs=75
-Ddiameter.default.timeout.ms=1500
-DmemcacheClientTimeout=200
-Dlocking.disable=true
-Dcontrolcenter.disableAndsf=true
-DenableQueueSystem=false
-Dredis.keystore.connection.string=lb01:lb02:6379:6381
-Dcom.cisco.balance.dbs=6
-Dcom.cisco.balance.compression=true
-Duse.pre.v11.service.resolution=false
-Ddo.service.bundling.without.profiles=true
-Dpolicystate.optimize.inserts=true
-Dvirtualservice.optimize.cdr=true
-Duse.ldap.vs.evaluation.order=true
-DPCRf_Name=TMOSITE1
-Djdiameter.accept.unknown_desthost=true
-Djdiameter.replace.unknown_desthost=true
-DmaxHash=2
-DdbSocketTimeout.cdrrep=1000
-DdbConnectTimeout.cdrrep=1200
-Dmongo.client.thread.maxWaitTime.cdrrep=1200
-Dmongo.connections.per.host.cdrrep=10
-Dmongo.threads.allowed.to.wait.for.connection.cdrrep=10
-DdbSocketTimeout.cdr=1000
-DdbConnectTimeout.cdr=1200
-Dmongo.client.thread.maxWaitTime.cdr=1200
-Dmongo.connections.per.host.cdr=10
-Dmongo.threads.allowed.to.wait.for.connection.cdr=10
-Dcisco.cdr.compression=true
-Dcisco.cdr.disableBlocking=true
-DapirouterContextPath=/ua/soap
-Dua.context.path=/ua/backend
-Dapi.ua.context.path=/ua/backend
-Dredis.keystore.connectionTimeout=20000
-DdbSocketTimeout.remoteBalance=1000
-DdbConnectTimeout.remoteBalance=1200
-Dmongo.client.thread.maxWaitTime.remoteBalance=1200
-Dmongo.connections.per.host.remoteBalance=12
-Dmongo.threads.allowed.to.wait.for.connection.remoteBalance=12
-Dmessage.buffer.early.processing.time=5
-Dlog.cdr.csv=true
-Dcom.broadhop.cdr.dir=/var/broadhop/cdr
-Dcom.broadhop.cdr.rollover.dir=/var/broadhop/cdr/rollover
-DuseV1ClientId=true
-DaddOnlySvcPlanAsVirtualService=true
-Dsk.db.replicateSessionSharding=true
-Dsk.db.audit.perShardTPS=1000
"
[root@pcrfclient01 ~]#

```

**Solution:** In Policy Builder, **Session Limit Overload Protection** must be configured such that there should be buffer/considerable difference between the configured value and the total system capacity. In session database, additional sessions are created beyond value configured in **Session Limit Overload Protection** until the Policy Server (QNS) process checks the current session count of the system in the next interval.

## Application Bundles or Plugins Unable to Start After Site Recovery

**Issue:** Diagnostics script output displays "application bundle or plugins failed to start".

**Solution:** During the start or initialization of application, it loads the configured data from the Policy Builder and brings up the bundles or plugins. The following table lists reasons and corresponding actions for solving the issue.

Reason	Solution
If the data is not configured properly in Policy Builder for specific feature plugin then the application fails to bring-up or start the bundle.	Correct the wrongly configured details in the Policy Builder.
Database details are incorrect in Policy Builder configuration.	Correct the database configuration details and publish the changes.
Application fails to connect to the configured database due to network issues.	Verify if the firewalls are blocking the connections. If the firewalls are blocking the connections, stop the firewall.  Verify database sessionmgr VMs are down. If the database sessionmgr VMs are down, bring-up the replica-set members.
Application fails to connect to the configured database due to replica-set members not available (PRIMARY might not available during the recovery procedure).	Restart the qns process.  If the application still fails to connect to the database, open the published configuration in Policy Builder and publish the same (without any change).
Application fails to connect to the configured database if configuration for some required features is not done in Policy Builder.	Configure the missing features in Policy Builder and publish the changes.

## CPS System Stuck in Rebalancing

Perform the following steps to resolve the issue in HA environment:

1. Stop the QNS process running on the Policy Server (QNS) VMs.

If this is a lab or no traffic is running, use `stopall.sh` script to stop all the QNS processes.

2. Drop the scheduler database from the ADMIN replica-set.

```
use scheduler
db.dropDatabase()
```

3. Clear the migration shards list from "cache\_config" collection under sharding database from the ADMIN replica-set.

```
use sharding
db.cache_config.update({"_id" : 1},{$unset : {"migratingShards" : 1}})
db.cache_config.update({"_id" : 2},{$unset : {"migratingShards" : 1}})
```

4. Set migration field to false and unset the prev\_shard field in "buckets" collections under sharding database from the ADMIN replica-set.



- For GR setup, collection name is the name of the bucket (for example, bucket\_1 and bucket\_2).
- For SK DB in HA setup, collection name is the SK DB bucket name (for example, db.sk\_buckets).
- For SK DB in GR setup, collection name is the SK DB bucket name in GR setup (for example, db.sk\_buckets\_1 and db.sk\_buckets\_1).

```
use sharding
db.buckets.update({"prev_shard" : <x>}, { $unset: { "prev_shard" : <x>}, $set : {"migration"
: false} }, { multi: true })
db.buckets.update({"prev_shard" : <x>}, { $unset: { "prev_shard" : <x>}, $set : {"migration"
: false} }, { multi: true })
```

where, <x> is the output on the bucket to identify the prev\_shard.

- To find <x> in GR setup, execute `db.buckets_1.find({"migration":true}) & db.buckets_2.find({"migration":true})` command.
- To find <x> in HA setup, execute `db.buckets.find({"migration":true})` command.
- In SK DB enabled setups,
  - To find <x> in HA, execute `db.skbuckets.find({"migration":true})` command.
  - To find <x> in GR, execute `db.skbuckets_1.find({"migration":true}) & db.buckets_2.find({"migration":true})` command.

5. Start the QNS processes which are stopped in [1, on page 104](#).

If this is a lab or no traffic is running, use `startall.sh` script to start all the QNS processes.

6. Login to the OSGI console and use `listshards` command to list the shards.

7. If rebalance is still in running or InProgress state or rebalance is required, execute the following commands from OSGi console:

- For GR setup,
 

```
migrate <site_name>
rebalance <site_name>
```
- For HA SK DB,
 

```
"migratesk"
"rebalancesk"
```
- For GR SK DB,
 

```
migratesk <site_name>
rebalancesk <site_name>
```

8. Run `diagnostics.sh` command.

## System Timeouts

**Issue:** When Session Managers with memcached (using default UDP protocol) are rebooted while processing traffic, System Timeouts are observed.

**Solution:** To recover from this issue, you must can configure memcache to use TCP protocol by setting `-Denable.memcache.on.tcp=true` in `qns.conf` file.

## High Swap Memory Usage during Resiliency Event

**Issue:** High swap memory usage is observed on `pcrfclient01` VM during resiliency event.

diagnostics output:

```
Checking swap space for all VMs...
Checking swap memory usage on pcrfclient01...[FAIL]
Swap usage is 1835 MB. This will likely lead to a slowdown in your system!
Please ensure your memory is provisioned properly.
If systems memory usage is no longer high, you can reset swap with: swap2ram.sh
```

**Condition:** Diagnostics fails after running the `diagnostic.sh` due to low swap memory issue in the `pcrfclient` VM.

**Solution:**

1. SSH to the `pcrfclient` VM and check the disk memory by using the `du -sh *` command.
2. Check `/var/lib/carbon/whisper/cisco/quantum` directory for memory usage. If an increase in the memory usage is observed, then issue is due to the statistics files.
3. To recover memory, delete the statistics (`.wsp`) files or increase the VM disk memory.
4. To recover the memory, delete the statistics manually from the folder, `/var/lib/carbon/whisper/cisco/quantum` or delete the statistics using the following steps:

a. Run `df -h` command and note down the disk spaces.

b. Run the following commands:

```
du -h --max-depth=0 /var/lib/carbon/whisper/cisco/quantum/qps
monit stop grafana-server
monit stop carbon-cache
vi /etc/carbon/storage-schemas.conf
```

c. Update the retention period (from 90 days to 30 days) which is located at the end of the file. This results in more aggressive retention period.

**Older - retentions:** 10s:1d,60s:90d

**Update to - retentions:** 10s:1d,60s:30d

d. After updating the retention period, run the following commands:

```
monit start carbon-cache
monit start grafana-server
```




---

**Note** Alternatively, you can use `systemctl start/stop/restart <service_name>` command instead of `monit start/stop/restart <service_name>` command.

---

e. Create a file named `resize.sh` and add the following:

```
monit stop carbon-cache
cd /var/lib/carbon/whisper/cisco/quantum/qps
find ./ -type f -name '*.wsp' -exec whisper-resize --nobackup {} $1 \;
```

```
chown -R carbon:carbon *
monit start carbon-cache
monit restart grafana-server
```

f. Add the necessary permissions by running the `chmod 777 resize.sh` command.

g. Run `./resize.sh "updated_retention_value"`.

**Example from 4.c, on page 106:** `./resize.sh "10s:1d 60s:30d"`

h. Once the script is completed, run the following commands to confirm the amount of data has reduced by comparing the previous and current result.

```
run df -h: disk space should reduce or less than previous result
du -h --max-depth=0 /var/lib/carbon/whisper/cisco/quantum/qps
```

i. Verify that the Grafana displays the last 30 days statistics and not more than 30 days. (Updated retention period is 30 days).

5. If you do not want to delete the statistics, then increase the disk size by referring to *Attach and Detach External Disk to VM* section in the *CPS Operations Guide*. You need to map all the ESXi hosts to the vCenter.

## config\_br.py execution Fails to Export Data

**Issue:** When `config_br.py -a export --stats /var/tmp/stats.tar.gz` command is executed to export statistics, it takes long time to complete the task.

During the command execution, if linux terminal is closed, the execution fails to export the data.

**Solution:** Open a new terminal and perform the following steps:

1. Grep the PID of the process.

```
ps -ef | grep config_br
```

2. If `config_br` process exists, kill the process.

```
Kill -9 PID
```

3. Remove the stats.tar.gz.

```
rm -rf /var/tmp/stats.tar.gz
```

## MongoDB Member not Coming Up after Reboot

**Issue:** MongoDB member is not coming up as Secondary after reboot. The member state as UNKNOWN when executing `diagnostics.sh --get_replica_status` command.

**Condition:** Sometimes after restart, the replica set members do not come up as expected. Few of them move to REMOVED state with "errmsg" : "Our replica set config is invalid or we are not a member of it", "code" : 93, and "codeName" : "InvalidReplicaSetConfig" when `rs.status()` is executed on the available member.

**Possible Cause:** This happens because during restart, when loading the configuration from the disk, MongoDB tries to discover itself from the configuration. It would iterate through each enlisted member from the configuration and execute a simple `isSelf` test command against every member. If the node cannot determine or resolve its own hostname in the course of this process then it would assume that it is not the part of the

configuration file. At this point MongoDB does not retry to discover itself until it is restarted, or it receives a new config through `rs.reconfig()`.

### Solution

As this is one point failure or intermittent issue, Primary node is available and the issue does not affect the traffic. To resolve this issue, you can:

**Solution1:** Restart the MongoDB process to get the node back to secondary.

Stop the MongoDB process using `/etc/init.d/sessionmgr-<portNum> stop` command.

Stopped process is restarted via AIDO.

**Solution2:** Run `rs.reconfig(rs.config())` command on the available primary node. If the primary node of replica set is not available, execute `rs.reconfig(cfg, {force : true})` on any available secondary node of the set.

## Remove Traces of Old Policy Director (LB) VIPs

**Case:** User has extra lbvips which are not needed anymore and wants to remove the extra lbvips. Even after removing old lbvip using `pcs resource delete RESOURCE_ID` command, the old lbvips keep appearing in `crm_mon -l` output. The following section provides the steps to remove the extra lbvips so that they don't come up again in the future.

**Solution:** For solution, see *Remove Traces of Old Policy Director (LB) VIPs* section in the *CPS Operations Guide*.

## DiameterPeerDown Alarms Stuck When Active Policy Director (LB) VM is Rebooted

**Issue:** When Diameter peers connected to Policy Director (LB) VM (with VIPs) is rebooted, 3001 DiameterPeerDown alarm is stuck and not cleared even though the peers connect back successfully.

```
[root@FPSNE3-04-1-ClusterManager2 alarm-test]# date; diagnostics.sh --get_a
Wed Jun 24 14:28:47 -03 2020
CPS Diagnostics GR Multi-Node Environment
-----
Ping check for lb02 Adding to IGNORED_HOSTS...[FAIL]
Active Application Alarm Status
-----

id=1000 sub_id=3001 event_host=diameter-int1-vip status=down date=2020-06-24,14:28:51,235-0300

msg="3001:Host: nvblm1.1.lte.tim.br Realm: lte.tim.br PeerIP: 10.46.37.18 Interface: Gx is
down"
```

**Analysis:** When LB (with VIP) is rebooted, lbvip moves from active LB (say, lb01) to the other LB (say, lb02). All qns processes on LB write the events relating to peer connect/disconnect to rsyslog proxy which listens on lbvip02:5544.

When lb01 reboots, there is a delay for the lbvip02 to come up from lb01 to lb02. During this time qns processes on lb02 wait for the lbvip02 to come up. When lbvip02 comes up each of the qns processes connect to rsyslog proxy and write the peer connect/disconnect to lbvip02:5544.

In some race conditions, qns process to which peer is connected, event is written first to rsyslog proxy and then followed by the event from qns process where the peer is disconnected. In such scenarios, Fault Management Service which is responsible for sending the CLEAR alarm does not generate the alarm since the UP is followed DOWN.

**Possible Cause:** qns process connect to rsyslog proxy and in some race conditions, events are written to the rsyslog proxy out of order.

**Solution:** To reboot the Policy Director (LB) VM with VIP addresses, perform the following steps:

1. Check the number of Diameter connections before reboot. This command can be executed from Cluster Manager or PCRF client.

```
ssh pcrfclient01 "python /var/qps/bin/support/show_peers.py" | grep -c OKAY
```

2. Reallocate the VIP addresses from the Policy Director (LB) VM (lb01). This command can be executed from Cluster Manager or PCRF client.

```
ssh lbvip01 "monit restart corosync"
```

3. Check whether all the Diameter connections are successfully re-established by executing the following command from Cluster Manager or PCRF client.

```
ssh pcrfclient01 "python /var/qps/bin/support/show_peers.py" | grep -c OKAY
```

4. Reboot the target Policy Director (LB) VM (now without VIP addresses). This command can be executed from Cluster Manager or PCRF client.

```
ssh lb01 reboot
```




---

**Caution** The user will lose the current state of all the alarms when the qns processes on pcrfclient01 VM is restarted.

---

## Replica-sets Recovery in Case of Upgrade Warnings

**Issue:** `/var/log/mongo-<port>.log` shows the following error messages:

```
IMPORTANT: UPGRADE PROBLEM: The data files need to be fully upgraded to version 3.4 before attempting an upgrade to 3.6
```

**Solution:**

1. Find the system where the replica-set status are not coming up and confirm if it's showing the upgrade problem message.

Example: If the issue was with the admin replica-set then fetch the admin replica details and port information from `/etc/broadhop/mongoConfig.cfg` file. Also, login to the respective sessionmgr VM and confirm if the service for the respective port is down using `systemctl status sessionmgr-<portnumber>` command.

where, `<portnumber>` is the port configured for admin replica sets in `mongoConfig.cfg` file.

2. Post confirmation, perform the following steps:
  - a. Execute `monit stop aido_client` command on the machine where the admin replica services are not coming up.

- b. Locate the data path, which can be obtained from `/etc/broadhop/mongoConfig.cfg` file. For example, the datapath for admin replica set is `ARBITER_DATA_PATH=/var/data/sessions.5/set13`.

- c. Login to sessionmgr and take the backup of the exiting data path for futur use.

```
mkdir /tmp/backup-folder
cp -rf /var/data/sessions.5/set13/ /tmp/backup-folder
```

- d. After taking the backup, delete the data directories.

```
rm -rf /var/data/sessions.5/set13/
```

- 3. Execute `monit start aido_client` to start AIDO client and wait for it to bring the sessionmgr-27721 service.

## Incoming Traffic is Dropped and not Processed

**Issue:** No Gx sessions are created and all incoming traffic is dropped.

The following message is printed in qns logs continously:

```
WARN c.b.d.p.event.DiameterMessageDealer - StaleSessionEndToEndSla breached at QNS,
Time taken 1555DROPPING message RequestReceivedTime 1594136320622 currentTime 1594136322177
```

**Analysis:** When **Stale Session Message Handling Configuration** is configured in Policy Builder, request processing happens within the given SLA time period for the incoming request. The request or responses which cross the configured SLA are dropped.

**Possible Cause:** The incoming traffic is dropped if there is a clock skew on the Policy Server (qns) VMs from the Policy Director (lb) VM. If **Stale Session Message Handling Configuration** is configured, it is mandatory that all the VMs should be in time sync with the Policy Director (lb) VM.

**Solution:** Execute `sync_times.sh ha` command on the Cluster Manager to make sure all the VMs are in time sync with Policy Director (lb) VM.

## MongoDB Processes not Coming Up on Arbitervip

**Issue:** If MongoDB processes on arbitervip are not coming up when enabling/disabling the MongoDB authentication.

**Possible Cause:** Due to corosync issue.

**Solution:**

1. Check where arbitervip is running and login to that host.
2. Execute `ps -ef | grep mongod` command.
3. Execute `pkill mongod` command.
4. Execute `pcs resource cleanup` command and wait for some time.
5. Check the status using `pcs status` command.
6. If the failed resource action entries still exist, again execute `pcs resource cleanup` command.

## Issue with Policy Builder Publishing Time

**Issue:** It takes longer time to publish the Policy Builder configuration in HA clusters.

**Condition:** SVN source and destination repositories are on different hosts/clusters rather than on the same host/cluster.

**Solution:** This is SVN server behavior and not CPS issue. If you are publishing on same host then use `svn copy` command and if host is different than use `svn import` command. As mentioned in the SVN docs, copy is faster than import.

For example, if you are logged in using `http://lbvip02/repos/configuration` and publishing to `http://lbvip02/repos/run` then both the hosts are same (lbvip02) and you can use `svn copy` command.

But if you are logged in using `http://lbvip02/repos/configuration` and publishing to `http://<different_host>/repos/run` then you can use `svn import` command.

SVN import takes more time than copy command. So, this is expected SVN server behavior.

The recommendation is that if you want to publish on different host or cluster, then open Policy Builder of other cluster and use other Cluster's run repository to publish.

1. Export policy configurations from hostA (clusterA) and push the same on hostB (clusterB) in `/repos/configuration` using SVN import command.
2. Open Policy Builder with other Cluster's IP address.
3. Login to Policy Builder with `http://lbvip02/repos/configuration`.
4. Publish to Cluster's to run repository using `http://lbvip02/repos/run`.

## Reporting Replica-set not Coming Up

**Issue:** `diagnostics.sh` execution is stuck at one place for a longer time.

**Analysis:** This issue comes in the MongoDB replicas when the secondary replica lagging behind the primary replica. The reason is the higher CDR rate (approx. 8 k) in the reporting replica-set primary member. Due to this, diagnostics is stuck in hang state.

**Scenario:** To confirm whether the issue is due to reporting replica-set or not use the following steps.

1. Check the diagnostics by using `diagnostics.sh --get_replica_status` command.
2. If the `diagnostics.sh --get_replica_status` command execution is stuck, issue might be due to reporting replica-set. To cross verify, check whether the readonly user is able to login in to the MongoDB secondary member or not.

```
mongo -u readonly -p '<password>' --authenticationDatabase admin sessionmgr<num>:<portNum>
```

3. Repeat 2, on page 111 for all the secondary members.
4. If any of the readonly logins are stuck, check whether the inserts are high or not (approx. 8k) in the Primary member using `mongostat` command.

```
mongostat -u admin -p '<password>' --authenticationDatabase admin --host sessionmgr<num> --port <portnum>
```

**Solution:** To unblock the hanged secondary replica-set member, use the following steps:

1. Stop the sessionmgr mongod process by using `/usr/bin/systemctl stop sessionmgr-XXXXX` command. where, XXXXX is the database port number
2. Clear data directory of that sessionmgr by using `\rm -fr <data directory path of that mongod>` command.
3. Start the sessionmgr mongod process by using `/usr/bin/systemctl start sessionmgr-XXXXX` command. where, XXXXX is the same database port number mentioned in 1, on page 112.
4. When this database member goes to 'SECONDARY' state, check the replica-set status by using `diagnostics.sh --get_replica_status` command.




---

**Note** If a member is shown in an unknown state, it is likely that the member is not accessible from one of other members, mostly an arbiter. In that case, you must go to that member and check its connectivity with other members. Also, you can login to mongo on that member and check its actual status.

---

After unblock the hanged secondary replica-set member. the load balancing and incoming CDR rate must be configured as a policy change in the Policy Builder based on the customer setup. For more information, contact your Cisco Account representative.

## Rebalance and Migrate SK OSGi Commands Unable to Complete

**Issue:** rebalance or migrate SK database commands are running but unable to complete.

**Condition:** Run `rebalancesk` from OSGi CLI. After hours, `rebalancesk` status shows following output instead of Rebalanced:

Remaining buckets: <number greater than 0>

### Expected Output:

```
osgi> rebalanceskstatus
Rebalanced
```

### Solution:

1. Login to the ADMIN replica-set PRIMARY member.

```
mongo <sessionmgr_VM_hostname>:<Port_Number>
```

2. Check the records in the tasks collection.

```
PRIMARY> use scheduler
PRIMARY> db.tasks.find()
```

3. If any tasks are present, remove those tasks.

```
PRIMARY> db.tasks.remove({})
```

4. Connect to sharding database.

```
use sharding
```

5. Run the following queries.

```
db.sk_buckets.update({"migration": true}, {'$set' : {"migration" : false }}, {multi:true})
db.sk_buckets.updateMany({}, { $unset: { "prev_shard": "" } })
```



6. Restart one of the qns service using `monit restart qns-x` command.
7. Once qns is up, execute `rebalancesk` command to rebalance the SK database shards.

Example:

```
OSGI> rebalancesk
Rebalancing ...
All versions up to date
Obtaining lock
Current version: xxx. Validating all instances on same version
Min bucket size: xxxx
Balance calculations completed successfully
All versions up to date
All versions up to date
Obtaining lock
Completed: 100%
Migration completed successful
```




---

**Note** Once rebalance command is complete, execute `migratesk` command.

---

8. Execute `migratesk` command to migrate the SK database shards.

Example:

```
OSGI> migratesk
Migrate ...
All versions up to date
Obtaining lock
Completed: 100%
Migration completed successfully
```

## CRD Import Failure

**Issue:** Import operation fails when user tries to import the same CRD through GUI and `curl` command immediately. The same MongoDB tables are changed which result in bad CRD state which is not recommended.

**Solution:** Since there is a limitation on updating json/bson files, `-DuseMongoCLI=true` configuration should not be used while exporting CRD.

Either set the parameter `-DuseMongoCLI=false` or remove the parameter from the setup where CRD export/import operations are being executed.

## Unauthorised User Alert Displayed When Performing CRD Import

**Issue:** During import, `Unauthorized User` alert is displayed on the browser when performing import operation through GUI.

**Solution:** Clear the browser cookies when `Unauthorized User` alert is displayed while performing CRD import operation through GUI.

## Connection Reset by HAProxy

**Issue:** Frequent TCP connection reset seen in Policy Director (LB) nodes on port 3868, 3869, 3870 by HAProxy.

**Analysis:** User is running haproxy on Policy Director (LB) nodes with keepalive (check interval) for every 2 sec and forwarding data received on port 3868 to 3868 port on server lb01-A lb01:3868 and vice versa. But in tcpdump packet captured on LB01 VM, there is a connection reset on port 3868 every 2 seconds in a sequence: **syn > syn,ack > rst,ack**.

```
listen diameter-int1-vip
bind 172.17.50.128:3868
mode tcp
option tcpka
balance leastconn
server lb01-A lb01:3868 check
server lb01-B lb01:3869 check
server lb01-C lb01:3870 check
server lb02-A lb02:3868 check
server lb02-B lb02:3869 check
server lb02-C lb02:3870 check
```

The sequences of packets including reset is completely normal. This is the way haproxy performs health checks efficiently. As soon as haproxy has discovered that the endpoint is up, there is no point in wasting any further resources at either end. It turns out that using TCP RST is the most efficient way for kernels at both ends of the connection to finish their conversation and free up those resources. This is an expected behavior and no traffic loss is observed.

## Troubleshoot Manage SSH keys

**Issue:** There is mismatch between given password and VM's password.

**Condition:** Encountered following error when running

`/var/qps/install/current/scripts/bin/support/manage_sshkey.sh` script.

```
"Unable to check password"
```

**Solution:** Make sure root password is same for all CPS VMs.

Execute `/var/qps/bin/support/change_passwd.sh` from installer VM to change the password.

## Failure when Restoring Cluster Manager during Rollback on Openstack Environment

**Issue:** During rollback procedures, restoring Cluster Manager on OpenStack environment fails displaying the following error message:

Considering rollback from CPS 19.4.0 to CPS 21.1.0 as an example.

```
[root@cm~]# /mnt/iso/migrate.sh restore cluman
migrate_cluman_20210408_175613.tar.gz
2021-04-21 15:33:03.191 INFO [__main__.install] imported datetime
2021-04-21 15:33:03.191 INFO [__main__.install] imported sys
2021-04-21 15:33:03.197 INFO [__main__.install] imported argparse
```

```

2021-04-21 15:33:03.199 INFO [__main__.install] imported subprocess
2021-04-21 15:33:03.200 INFO [__main__.install] imported logging
2021-04-21 15:33:03.201 INFO [__main__.install] imported shutil
2021-04-21 15:33:03.275 INFO [__main__.install] imported managetmpfile
2021-04-21 15:33:03.277 INFO [__main__.install] imported Validate
2021-04-21 15:33:03.278 INFO [__main__.install] imported RemoveClumanApiMarker
2021-04-21 15:33:03.278 INFO [__main__.install] imported RemoveRepoFile
2021-04-21 15:33:03.278 INFO [__main__.install] imported CreateRepoFile
2021-04-21 15:33:03.280 INFO [__main__.install] imported CopyFilesPrePuppetStage
2021-04-21 15:33:03.281 INFO [__main__.install] imported CopyFilesPostPuppetStage
2021-04-21 15:33:03.281 INFO [__main__.install] imported ExtractClumanArtifacts
2021-04-21 15:33:03.282 INFO [__main__.install] imported ExtractRpmsArtifacts
2021-04-21 15:33:03.283 INFO [__main__.install] imported GenerateClumanFacts
2021-04-21 15:33:03.283 INFO [__main__.install] imported local
2021-04-21 15:33:03.284 INFO [__main__.install] imported ApplyClumanPuppet
2021-04-21 15:33:03.284 INFO [__main__.install] imported MarkClumanStarting
2021-04-21 15:33:03.284 INFO [__main__.install] imported MarkClumanReady
2021-04-21 15:33:03.284 INFO [__main__.install] imported LegacyInstall
2021-04-21 15:33:03.288 INFO [__main__] Executing main
2021-04-21 15:33:03.288 INFO [__main__.main] Executing main
2021-04-21 15:33:03.288 INFO [__main__.main] Calling migrate function
2021-04-21 15:33:03.288 INFO [__main__.migrate] Doing migration
2021-04-21 15:33:03,292 INFO [__main__.migrate]
=====
|CPS Migration 2021-04-21 15:33:03.291914
|Arguments:
| --log_path: /var/log
| --log_file: inservice_migration
| subcommand: restore
|
| Logging to: /var/log/in-service_migration_20210421_153303.log
=====

[localhost] local: /bin/grep NODE_TYPE /etc/broadhop.profile | cut -d= -f2
2021-04-21 15:33:03,320 INFO [__main__.extra_banner]
=====
| Restoring from file:
/var/tmp/cps_194_to_211/migrate_cluman_20210408_175613.tar.gz
=====

2021-04-21 15:33:03,320 INFO [__main__.run_recipe] Performing installation stage:
Restore backup Tar
2021-04-21 15:33:03,323 INFO [__main__.run_recipe] Performing installation stage:
Restore Cluman.
2021-04-21 15:33:03,323 INFO [restore_cluman.backup_logback_xml_files] Restoring
logback xml files.
2021-04-21 15:33:03,324 INFO [restore_cluman.backup_logback_xml_files]
/var/tmp/logback_backup already exist.. Recreating it..
2021-04-21 15:33:03,332 INFO [restore_cluman.restore_config_br] Restore cluman
config_br files.
2021-04-21 15:33:03,333 INFO [backup.handleRequest] Action Import
2021-04-21 15:33:03,334 INFO [backup.etc] Restore: etc
2021-04-21 15:33:03,377 INFO [restore_cluman.restore_config_deploy] Restoring
cluman orch api files.
2021-04-21 15:33:03,379 INFO [restore_cluman.restore_config_deploy] Converting
system.json for arbiters.
2021-04-21 15:33:03,384 INFO [restore_cluman.restore_config_deploy] (stdout):

```

```

Updating system.json to support multiple arbiters

2021-04-21 15:33:03,384 INFO [restore_cluman.restore_config_deploy] Regenerating
cluman files.
2021-04-21 15:33:03,503 INFO [command.execute] (stdout): HTTP/1.1 200 OK
Date: Wed, 21 Apr 2021 15:33:03 GMT
Content-Type: application/json
Content-Length: 20

{"state":"deployed"}
2021-04-21 15:33:03,503 INFO [restore_cluman.restore_config_deploy] Trying
regenerate API (CPS 12.0+)
2021-04-21 15:33:03,561 INFO [command.execute] (stderr): curl: (22) The requested
URL returned error: 500 Internal Server Error

2021-04-21 15:33:03,562 ERROR [__main__.<module>] Error during installation
2021-04-21 15:33:03,562 INFO [__main__.<module>] =====
2021-04-21 15:33:03,562 INFO [__main__.<module>] FAILURE
2021-04-21 15:33:03,562 INFO [__main__.<module>] ===== END =====
2021-04-21 15:33:03,563 INFO [__main__.<module>] To have the environment variable
updated, please logout and login from all opened shell on the current system
Traceback (most recent call last):
File "/usr/lib64/python2.7/runpy.py", line 174, in _run_module_as_main
  "__main__", fname, loader, pkg_name)
File "/usr/lib64/python2.7/runpy.py", line 72, in _run_code
exec code in run_globals
File "/mnt/iso/modules/install/__main__.py", line 895, in <module>
rtnCode = main(sys.argv[1:])
File "/mnt/iso/modules/install/__main__.py", line 885, in main
rtnCode = migrate(argv[1:])
File "/mnt/iso/modules/install/__main__.py", line 813, in migrate
args.func(args)
File "/mnt/iso/modules/install/__main__.py", line 317, in
migrate_restore_type_cluman
run_recipe(recipe)
File "/mnt/iso/modules/install/__main__.py", line 863, in run_recipe
stage.run()
File "install/restore_cluman.py", line 40, in run
self.restore_config_deploy()
File "install/restore_cluman.py", line 128, in restore_config_deploy
exe.execute()
File "util/command.py", line 116, in execute
raise RuntimeError(' '.join(self.command) + ' returned ' + str(self.exitcode) +
' instead of ' + str(self.expected_exitcode))
RuntimeError: /usr/bin/curl --fail --silent --show-error -i -X POST
http://localhost:8457/api/system/config/action/regenerate returned 22 instead of
[0]

```

**Condition:** Rollback procedure is executed from CPS 21.x to CPS 19.x version. The error occurs as older release (for example, CPS 19.x) Cluster Manager is not able to process the backed up configuration JSON files from the higher release (for example, CPS 21.x).

**Solution:**



**Note** The solution steps are provided for the rollback from CPS 21.1.0 to CPS 19.4.0. If the issue occurs on other versions, then change the respective versions accordingly.

1. Check the orchestration API logs for the following errors during the current time in which the restore command was executed.

```
$ cat /var/log/orchestration-api-server.log | grep -i "Unrecognized field"
WARN [2021-05-10 06:26:37,019] com.cisco.bobcat.apiframework.plugin.ApiFrameworkPlugin:
  Caught IOException in mapToObject method
! com.fasterxml.jackson.databind.exc.UnrecognizedPropertyException: Unrecognized field
"preventPrimaryFlappingEnabled" (class
```

2. If the error contains any unrecognized field such as, `preventPrimaryFlappingEnabled` and so on, then this issue is occurring because the original backed up migration tar files contains a flag which is not available in CPS 19.x orchestration API jar. So, it is necessary to replace the API jar from the last applied platform patch on CPS 19.x systems to recover from this issue. Stop the Orchestration API service by executing the following command:

```
$ monit stop orchestration-api-server
```

3. Verify that the service is stopped by executing the following command:

```
$ systemctl status orchestration-api-server
```

4. Create a backup directory folder and copy the original files.

```
$ mkdir /tmp/api-jar-backup
$ cp /opt/orchestration_api_server/plugins/core-plugin-19.4-a-SNAPSHOT.jar
/tmp/api-jar-backup
$ cp /qsb_config/features/system/system.json /tmp/api-jar-backup
$ cp /var/qps/config/deploy/json/Configuration.js /tmp/api-jar-backup
```

5. Take `core-plugin-19.4-a-snapshot.jar` from the last applied platform patch on CPS 19.4.0 and copy it in the following location

```
$ cp core-plugin-19.4-a-snapshot.jar /opt/orchestration_api_server/plugins/
```

6. Start the Orchestration API service by executing the following command:

```
$ monit start orchestration-api-server
```

7. Verify that the service is up and running by executing the following command:

```
$ systemctl status orchestration-api-server
```

8. Restore Cluster Manager by executing the following command:

```
$ [root@cm~]# /mnt/iso/migrate.sh restore cluman migrate_cluman_20210408_175613.tar.gz
```

## Corrupted Admin Database Sharding

**Issue:** During creation of PCRf or UDC shards, if there's an issue that isn't monitored, the Admin sharding database gets created partially. This leads to an issue in the index creation and instance/configuration collection of sharding database.

### Symptoms:

Policy Server (QNS) VM or UDC VM keeps restarting. This restarting of VMs leads to failure during shard creation.

OR

Rebuilding SK database operation is failing.

OR

There is only one index in shards collection of sharding database. You can confirm by logging into the Mongo Shell of Admin DB and checking the database shards.

#### Example:

```
mongo sessionmgr01:27721
set06:PRIMARY> use sharding
switched to db sharding
set06:PRIMARY> db.shards.getIndexes()
db.shards.getIndexes()
[
  {
    "v" : 2,
    "key" : {
      "_id" : 1
    },
    "name" : "_id_",
    "ns" : "sharding.shards"
  }
]
```

#### Solution:

1. Login to Mongo Shell of Admin DB. Refer to `/etc/broadhop/mongoConfig.cfg` file to get the admin primary shard details.

For example, `mongo sessionmgr01:27721`

2. Drop the shards collection using `db.shards.drop()` command.




---

**Note** If UDC shard has issue, UDC admin shard needs to be dropped and on `udc01` VM. After shards are dropped, restart `qns-1` process.

---

3. Restart `qns01` process on any Policy Server (QNS) VM using `monit restart qns-01` command.

## Passwordless Blade Access not Working

**Issue:** Passwordless access not working between Cluster Manager and ESXi hosts after changing SSH keys using `manage_sshkeys.sh` script.

**Solution:** Execute `/var/qps/install/current/scripts/deployer/support/jvalidate.py` to synchronize the keys with ESXi hosts and restore passwordless access.

## pcrfclient VM Unable to get Configuration

**Issue:** PCRFCLIENT VM is deployed and powered ON successfully but is not reachable from Internal CPS VMs. Communication works for Management/OAM interface properly.

**Possible Cause:** As internal IP is not reachable, Cluster Manager is not able to push puppet configuration of the respective node.

**Condition:** This issue is applicable to VMware based installation.

Security settings on Vswitch port connected to Internal Interface on PCRFCLIENT VM has:

- **Allow MAC Address Change** set to false
- **Allow Forged Transmits** set to false

**Solution:** For fixing the issue it is recommended to set security settings on Vswitch port connected to Internal Interface on PCRFCLIENT VM as:

- **Allow MAC Address Change** to true
- **Allow Forged Transmits** to true

## Unreachable Time Source in Chronyd

**Issue:** An unreachable source is selected as a time source in chronyd.

**Cause:** When chronyd is configured with multiple time sources, it tries to select the most accurate and stable source for synchronization of the system clock. When the best source becomes unreachable, chronyd doesn't immediately switch to the next available best source in an attempt to minimize the clock error.

Chronyd lets the clock run free for as long as its estimated error (in terms of root distance) based on previous measurements is smaller than the estimated error of the next available source, and there is still an interval which contains some measurements from both sources. If the first source was better than the next available source, it can take hours before the available source is selected, depending on its polling interval.

**Solution:** This issue is resolved automatically depending on the configured NTP source.

## CPS not Recovering from `System - CRD is BAD`

**Issue:** As CRD is in BAD state, all CRD APIs (except import all, list and query) are blocked. User is not allowed to use the CRD APIs.

**Analysis:** When user tries to import CRD data using **CPS Central > Import Custom Reference Data**, CPS marks `System-CRD is BAD`. This is because underlying CRD schema is not compatible with CRD Data being imported. Hence, all CRD APIs (except import all, list and query), are blocked and user is not allowed to use them.

**Possible Cause:** In some conditions, though user made sure the underlying CRD schema is compatible with CRD data being imported but the system state is still doesn't recover from BAD.

**Solution:** Connect to admin database and update the `isSystemBad` flag to false manually. Refer to the following sample commands:

```
use admin
switched to db admin
> db.state.find()
{ "_id" : "state", "isSystemBad" : true, "lastUpdatedDate" :
ISODate("2021-06-18T09:13:00.961Z") }
> db.state.updateOne({_id:"state"},{$set:{isSystemBad:false}})
{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }
> db.state.find()
{ "_id" : "state", "isSystemBad" : false, "lastUpdatedDate" :
ISODate("2021-06-18T09:13:00.961Z") }
>
```

## Recurring Quota not Working

**Issue:** Recurring quota creation or refresh does not happen when **Recurrence Frequency** is configured as Bill Cycle under **Recurring Quota Template** in the **Policy Builder**. This impacts new subscribers creation after migration to CPS 20.2 or later releases.

**Possible Cause:** When the Policy Builder with recurring quota template configured with **Bill Cycle (RFamt ignored)** is imported into the CPS 20.2 or later release, the recurrence frequency defaults to **Month(s)** instead of **Bill Cycle**. This issue is applicable for Policy Builder based Balance templates and not for CRD based Balance templates.

**Solution:** Before importing the Policy Builder from previous releases (prior to CPS 20.2), execute the following command

```
for FILE in *; do `sed -i 's/ (RFamt ignored)//g' $FILE`; done
```

on the folder containing PB .xmi files. This script ensures all the references to **Bill Cycle (RFamt ignored)** is changed to **Bill Cycle**. This folder can be imported to CPS 20.2 or later version for publishing.

## pcrfclient Disk Space and Memory Issue after ISSM

**Issue:** After ISSM is complete, pcrfclient disk space and memory are getting filled.

**Possible Cause:** This issue occurs when the bulkstats files are copied during ISSM and are kept on hold by the collectd service even though they are removed from the system. This causes load on the system and ends up increasing the system storage and memory simultaneously.

**Solution:** Confirm if the collectd service is consuming high memory.

```
service collectd status | grep -i mem
Redirecting to /bin/systemctl status collectd.service
Memory: 70.2G
```

The memory is at 70+ GB which is not normal for collectd service running on pcrfclient VM.

Perform the following operations on pcrfclient 01 and 02 VMs. If your setup is a GR or a dual cluster, make sure that the following steps are performed on both sites pcrfclient VMs.

The following is an example of how to restart the collectd on pcrfclient01 VM. You must follow the same procedures on pcrfclient02 VM and on other site pcrfclient VMs if your setup is GR or a dual cluster.

1. SSH to pcrfclient01.

```
ssh root@pcrfclient01
```

2. Stop collectd service.

```
monit stop collectd
```

3. Confirm that the collectd service is no longer monitored and also the service is successfully stopped.

```
monsum | grep -i collectd
collectd                               Not monitored                               Process

systemctl status collectd
```

4. Once the service is successfully stopped, bring the collectd service back on.

```
monit start collectd
```

5. Confirm that the service is up and running using `monsum` and `systemctl` commands.

```
monsum | grep -i collectd
```



```
systemctl status collectd
```

# Troubleshoot REDIS

## Troubleshooting REDIS Reporting Database

The following errors are outputted to the logs if there is an issue regardless of log level:

**Table 8: Errors Outputted to Logs**

Log Level	Source
<b>ERROR</b>	<b>Policy Server (QNS)</b>
Unknown message received, no type information {cdrSession}	
Unable to determine message type for message {message}	
<b>Errors specific to converting the records to bytes and submitting to REDIS</b>	
Exception in creating byte array output stream: {exception}	
Exception in closing ObjectOutputStream: {exception}	
Exception in closing ByteArrayOutputStream: {exception}	
<b>ERROR</b>	<b>Policy Director (LB)</b>
log.error("ERROR: dispatchMsg from Redis to reporting: Deserialization problem: ", ex);	
ERROR Unable to create temp file: {FileName} for collection {Collection}	
File handle not found for collection {collectionName}	
Unable to create temp file: { tmpFileName } for collection {collectionName }	
Throws broad hop exception	
Exception during cleanup and removal of the oldest files from the directory {finalOutputDirectory}	
Exception while closing files: {exception}	
Unable to close properly output file for collection { collectionName }	

To troubleshoot reporting issues, logging levels, in most cases, must be turned up. To turn up the logging level on one Policy Server (QNS) node rather than system wide do the following:

**Step 1** Modify the logging levels in `/etc/broadhop/logback.xml` on one Policy Server (QNS) node if reporting does not occur.

```
<logger name="com.broadhop.policyintel" level="trace" />
```

Reporting does not occur

```
<logger name="com.broadhop.reporting" level="trace" />
```

**Step 2** Set the logging levels to TRACE. This shows the records being submitted and read by REDIS, and those records being submitted to CSV files.

- Changing to TRACE on a Policy Server (QNS) node shows all reporting records being submitted to REDIS.
- Changing to TRACE on an Policy Director (LB) node shows all reporting records being read from REDIS and written to CSV files.

**Table 9: Logging Levels**

Log Level	Source
<b>TRACE</b>	<b>Policy Server (QNS)</b>
Submitting message to redis: [record]	
Example: TRACE c.b.reporting.impl.dao.ReportingDao - Submitting message to redis: {timestamp=Tue May 24 22:50:13 MDT 2016, type=cdr-2, sessionId=John_Doe, ....}	
<b>TRACE</b>	<b>Policy Director (LB)</b>
CsvReplicationRunner - Writing to file [record]	
Example: 22:50:13.775 [message-cluster-recv-queue-0] TRACE c.b.r.i.r.csv.CsvReplicationRunner - Writing to file [John_Doe, 2122.3232.3434, 20160524105013000615123, Tue May 24 22:50:13 MDT 2016, Tue May 24 22:50:13 MDT 2016, cdr4]	

**Step 3** Change the node back to the original levels after troubleshooting.

**Caution** This creates a lot of logging if TPS is high and can cause system performance issues.

## Reporting does not occur

**Step 1** Restart a Policy Server (QNS) node. If the production environment is set to **INFO**, the following logs appear in the output on startup.

Typical logback.xml default setting for production:

```
<logger name="com.broadhop.policyintel" level="info" />
<logger name="com.broadhop.reporting" level="info" />
```

**Step 2** Examine one Policy Server (QNS) node, restart the Policy Server (QNS) service on that node and look for the following logs for success or reporting start up issues:

Table 10: Log Information

Log Level	Source
<b>INFO</b>	<b>Policy Director (LB)</b>
ReplicationRunner - ReplicationRunner: Redis: Added Listener: {table name} {replication class}	
Example: INFO c.b.r.i.r.ReplicationRunner - ReplicationRunner: Redis: Added Listener: cdr-1 class com.broadhop.refdata.policyintel.impl.CsvReplicationImpl	
Writing out CSV files for {table name} to {output directory location}	
Example: INFO CsvReplicationRunner - Writing out CSV files for cdr-1 to /var/tmp/csv1	
Writing out temporary CSV files for {table name} to {tmp output directory location}	
Example: INFO CsvReplicationRunner - Writing out temporary CSV files for cdr-1 to /var/tmp/csv1/tmp	
<b>INFO</b>	<b>Policy Server (QNS)</b>
Starting replication for destination {collectionName}	
Example: INFO c.b.r.i.r.ReplicationManager - Starting replication for destination cdr-1	
Defining cdrs reporting queue with queue capacity : {cdrsQueueCapacity} and semaphore permits : {cdrsSemaphorePermits}	
Example: Defining cdrs reporting queue with queue capacity : 2000 and semaphore permits : 2000	
<b>ERROR</b>	<b>Policy Server (QNS)</b>
Unable to set date format for output CSV. Pattern is not valid: [Pattern]	
Example: ERROR CsvReplicationRunner Unable to set date format for output CSV. Pattern is not valid: yyyyMMddhmmssSSSSSS	
No reporting configuration found (Check PB reporting configuration if you see this error)	
Example: No reporting configuration found	

# REDIS does not receive or push out CDR records

- Step 1** Check the `redisTopology.ini` file to locate the primary location of the REDIS database:
- `/etc/broadhop/qns.conf` for `-DredisTopologyFile=[location]`
  - If parameter is absent the default location is `/etc/broadhop/redisTopology.ini`

```
redisTopology.ini example:
policy.redis.qserver.1=lb02:6379
policy.redis.qserver.2=lb01:6379
policy.local-control-plane.redis.1=lb02:6379
```

- Step 2** Go to `qserver.1` location, in this case `lb02`.

- Verify that REDIS is running:
 

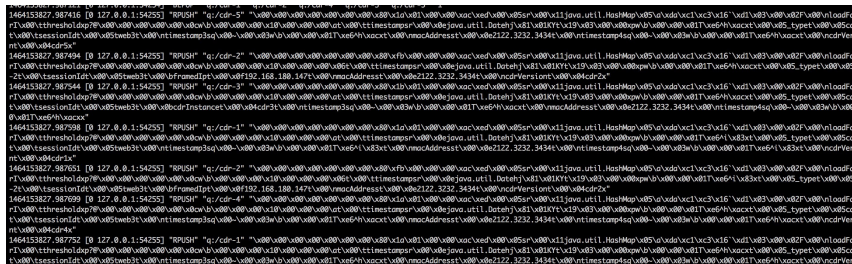
```
monit status redis
```
- Login to REDIS using the following commands:

```
redis-cli
lb02:6379> monitor
```

- Step 3** If you see the following output, REDIS is receiving and pushing out CDR records. In this case the CDRs are named `cdr-1`, `cdr-2`, `cdr-3`, `cdr-4`.

**Note** In a production environment, the output will reflect the names of the customers CDR table names.

*Figure 27: REDIS Output*



- If you do not see this output, move the `qserver.2` location and perform the same `monitor` command to determine if records are getting written and pushed to REDIS.
- If you cannot determine if any records are reaching REDIS, check if REDIS is online and there is not a system wide failure occurring.
- If you do not see a particular CDR in the REDIS output from the `monitor` command, check for CPS logging.

## Troubleshooting Graphite Database

### Default Password Change for graphite\_default User

**Case:** Traps were not generated after changing the `graphite_default` user password.

**Issue:** Traps are not generated for Gx and LDAP from `gen-gx-drop-trap.sh` and `gen-ldap-trap.sh`. The event log (`/var/log/broadhop/scripts/gen-gx-drop-trap.log`) shows There is no Gx Message traffic on \$HOST VM error for Gx and There are no Ldap queries processing on \$HOST VM error for LDAP even when the traffic is running on the setup.

**Analysis:** To have a common password for all the Grafana users, customer changed the default password for `graphite_default` user and updated `/var/www/html/htpasswd`. They missed to update `/root/.graphite_default` with the latest password which is being referred to access Graphite database.

When `gen-gx-drop-trap.sh` is trying to get data from Graphite database using old `graphite_default` user credentials (by referring from `/root/.graphite_default`) it is throwing 401 Unauthorized error.

**Request:** Executed the following commands from `pcrfclient01` & `pcrfclient02` VM and got the 401 Unauthorized error for the `graphite_default` credentials.

```
[root@pcrfclient01 ~]# graphite_default_passwd=`/var/qps/bin/support/mongo/decrypt_passwd.sh
\`cat /root/.graphite_default\`
[root@pcrfclient01 ~]# HOST=`awk '/#BEGIN_QPS_LOCAL_HOSTS/,/#END_QPS_LOCAL_HOSTS/' /etc/hosts
| egrep 'qns01' | awk '{ print $3}' | strings`
[root@pcrfclient01 ~]# curl -u graphite_default:$graphite_default_passwd -G
"http://localhost/graphite/render?&target=alias(transformNull
(cisco.quantum.qps.$HOST.nodel.messages.diameter_Gx_CCR-I.success,0),'CCR-I')&format=csv&from=-120s&to=-10s"
```

#### Output:

```
GX_CCR_I in dc193-qns01
<html><head><title>401 Unauthorized</title></head><body>
<h1>Unauthorized</h1>
<p>This server could not verify that you
are authorized to access the document
requested. Either you supplied the wrong
credentials (e.g., bad password), or your
browser doesn't understand how to supply
the credentials required.</p>
</body></html>
```

**Solution:** For solution, see *Changing Default default\_graphite User Password* section in the *CPS Operations Guide*.

## Unable to Access Graphite DB Using Default Graphite User

You need to verify that the encrypted password for `graphite_default` user in `/root/.graphite_default` file on cluster manager VM and in the “`factor | grep graphite_default`” command output on `pcrfclient` VM is the same.

The following files should contain entries for `graphite_default` user if not it means the default user is not created:

`/var/www/html/htpasswd` file (cluster manager VM)

`/var/broadhop/.htpasswd` (`pcrfclient` VM)

Perform the following steps to create graphite default user:

**Step 1** Run the following script on the cluster manager VM to create default user.

```
/var/qps/install/current/scripts/create_graphite_default_user.py
```

**Step 2** Run `import_deploy.sh` to update Factor file.

```
/var/qps/install/current/scripts/import/import_deploy.sh
```

**Step 3** Apply current configuration to all VMs.

```
/var/qps/install/current/scripts/upgrade/reinit.sh
```

## Grafana UI displays Continuous Prompt for Username and Password

Grafana UI displays continuous prompt for user credentials for any of the following reasons:

1. No user who is configured in graphite data source.
2. Current configured user in graphite data source does not have access to Graphite DB.

To confirm the cause behind the display of the continuous prompt for user credentials, you need to verify that the configured Graphite/Grafana user exists in `/var/broadhop/.htpasswd` file.

If the user does not exist, ensure that first graphite user is created using step that is mentioned in GRAPHITE/GRAFANA NEW USER CREATION and update graphite data source accordingly.

## Graphite Queries to Fetch Diameter Statistics

**Case:** Graphite queries to fetch Diameter statistics (one particular chart alone) is not working after performing chassis down scenario.

**Issue:** Issue has been observed in retrieving the diameter statistics after rebooting all the VMs in the setup as a part of chassis down scenario. It has been observed that bulk statistics and WSP files are getting generated in Graphite database directories.

**Impact:** Graphite queries are not yielding any results.

Error Observed for Grafana Dashboard:

```
IS Error
&#10;<center&#10;<h2&#10;</h2><p>Graphite encountered an unexpected error while handling your request.</p>&#10;<p>Please contact your site administrator if the problem
persists.</p>&#10;&#10;<br>&#10;<div>&#10;&#10;<div>&#10;&#10;<pre>Traceback (most recent call last)&#10; File &#34;/usr/lib/python2.7/site-packages/django/core/handlers/base.py&#34;, line 112, in
get_response&#10; response = wrapped_callback(request, *callback_args, **callback_kwargs)&#10; File &#34;/usr/lib/python2.7/site-packages/graphite/render/views.py&#34;, line 123, in renderView&#10; seriesList
= evaluateTarget(requestContext, target)&#10; File &#34;/usr/lib/python2.7/site-packages/graphite/render/evaluator.py&#34;, line 10, in evaluateTarget&#10; result = evaluateTokens(requestContext, tokens)&#10; File
&#34;/usr/lib/python2.7/site-packages/graphite/render/evaluator.py&#34;, line 21, in evaluateTokens&#10; return evaluateTokens(requestContext, tokens.expression)&#10; File &#34;/usr/lib/python2.7/site-
packages/graphite/render/evaluator.py&#34;, line 28, in evaluateTokens&#10; return evaluateTokens(requestContext, arg) for arg in tokens.call_args]&#10; File &#34;/usr/lib/python2.7/site-
packages/graphite/render/evaluator.py&#34;, line 21, in evaluateTokens&#10; return evaluateTokens(requestContext, tokens.expression)&#10; File &#34;/usr/lib/python2.7/site-
packages/graphite/render/evaluator.py&#34;, line 21, in evaluateTokens&#10; return evaluateTokens(requestContext, tokens.expression)&#10; File &#34;/usr/lib/python2.7/site-
packages/graphite/render/evaluator.py&#34;, line 21, in evaluateTokens&#10; return evaluateTokens(requestContext, tokens.expression)&#10; File &#34;/usr/lib/python2.7/site-
packages/graphite/render/evaluator.py&#34;, line 21, in evaluateTokens&#10; return evaluateTokens(requestContext, tokens.expression)&#10; File &#34;/usr/lib/python2.7/site-
packages/graphite/render/evaluator.py&#34;, line 21, in evaluateTokens&#10; return evaluateTokens(requestContext, tokens.expression)&#10; File &#34;/usr/lib/python2.7/site-
packages/graphite/render/evaluator.py&#34;, line 21, in evaluateTokens&#10; return evaluateTokens(requestContext, tokens.expression)&#10; File &#34;/usr/lib/python2.7/site-
packages/graphite/render/evaluator.py&#34;, line 24, in evaluateTokens&#10; return fetchData(requestContext, tokens.pathExpression)&#10; File &#34;/usr/lib/python2.7/site-
packages/graphite/render/datalib.py&#34;, line 378, in fetchData&#10; dbResults = dbFile.fetch(startTime, endTime, now)&#10; File &#34;/usr/lib/python2.7/site-packages/graphite/storage.py&#34;, line 347, in
fetch&#10; return whisper.fetch(self.fs_path, startTime, endTime, now)&#10; File &#34;/usr/lib/python2.7/site-packages/whisper.py&#34;, line 721, in fetch&#10; return file_fetch(fh, fromTime, untilTime,
now)&#10; File &#34;/usr/lib/python2.7/site-packages/whisper.py&#34;, line 758, in file_fetch&#10; return __archive_fetch(fh, archive, fromTime, untilTime)&#10;UnboundLocalError: local variable 'archive' referenced
before assignment&#10;&#10;</pre>&#10;</div>&#10;&#10;</center>
```

**Root Cause Analysis (RCA):** During reboot of VM (not all the time), some of the WSP files are getting corrupted/ broken.

**Solution:**

1. Confirm by switching the Grafana page from one perclient to other perclient VM to identify in which server the broken WSP files are present.

- By default, the Grafana fetches the data from pcrfclient01 VM. To check the data from pcrfclient02 VM, use the steps mentioned.
  - If public IP address is assigned to pcrfclient02 VM, then fetch the graph from pcrfclient02 IP (instead of lbvip01), else stop httpd service on pcrfclient01 VM (`/usr/bin/systemctl stop httpd`) and verify the data from Grafana. Once verified, start httpd service again on pcrfclient01 VM and verify the URLs.
2. Once determined, take the backup of the Grafana data (by moving the content to some other location) from the affected server and sync Grafana data from unaffected server to make it working. Assume if pcrfclient01 has corrupted WSP files, refer the following steps to sync Grafana data from pcrfclient02 VM to pcrfclient01 VM.

- a. Login to pcrfclient01 and take the backup of Grafana data (`/var/lib/carbon`) by moving the content to some other location.

```
ssh pcrfclient01
monit stop carbon-aggregator
monit stop carbon-aggregator-b
monit stop carbon-cache
monit stop carbon-cache-b
monit stop carbon-cache-c
mv /var/lib/carbon/* "PathForFilesToBeMoved"
```

- b. Synchronize the Grafana data from pcrfclient02 (healthy) VM to pcrfclient01(affected) VM.

```
rsync -a root@pcrfclient02:/var/lib/carbon /var/lib
monit start carbon-aggregator
monit start carbon-aggregator-b
monit start carbon-cache
monit start carbon-cache-b
monit start carbon-cache-c
```

## Grafana Statistics Missing after Stopping Carbon Cache

**Case:** Observed that some Grafana statistics are missing when stopping one of the carbon\_cache.

Same behavior is observed when stopping carbon-relay.

```
[root@pcrfclient01 ~]# monit stop carbon-aggregator-b
[root@pcrfclient01 ~]# monit status carbon-aggregator-b
Monit 5.25.1 uptime: 1d 1h 14m
```

```
Process 'carbon-aggregator-b'
  status                Not monitored
  monitoring status     Not monitored
  monitoring mode       active
  on reboot             start
  data collected        Tue, 12 Mar 2019 08:19:48
```

**Solution:** For any graph in Grafana, randomly if you see no statistics, then one reason can be that carbon-cache process is not running on pcrfclient VMs.

Check if carbon-cache, carbon-cache@b, carbon-cache@c processes are running using `systemctl` command.

For example, `systemctl status carbon-cache@b`

## No Data is Displayed in Grafana Dashboard after Rebooting pcrfclient

**Issue:** No data is populating in Grafana dashboard after pcrfclient VM nodes are rebooted.

**Possible Cause:** After rebooting pcrfclient VM, Graphite and network plugins are loaded successfully by collectd process but the carbon daemon is shutting down unexpectedly. Also, connections are closed on port 2013 due to which carbon fails to write the data to Graphite database and collectd starts displaying write\_graphite plugin errors. This is rare case scenario for carbon-0.9.6 version where MAX\_UPDATES\_PER\_SECOND\_ON\_SHUTDOWN doesn't work when the service starts for the first time.

The following is a sample output:

```
[root@pcrfclient01 ~]# shutdown -r now

Connection to pcrfclient01 closed by remote host.

[root@pcrfclient01 ~]# systemctl status collectd

collectd.service - Collectd statistics daemon

   Loaded: loaded (/etc/systemd/system/collectd.service; disabled; vendor preset: disabled)

   Active: active (running) since Fri 2020-08-07 04:29:38 UTC; 3min 4s ago

     Docs: man:collectd(1)
           man:collectd.conf(5)

 Main PID: 18661 (collectdmon)

  CGroup: /system.slice/collectd.service

          18661 /usr/sbin/collectdmon -P /var/run/collectdmon.pid -c /usr/sbin/collectd
-- -C /etc/collectd.conf

          20596 /usr/sbin/collectd -C /etc/collectd.conf -f
          21433 /usr/sbin/collectd -C /etc/collectd.conf -f
          21438 /usr/sbin/collectd -C /etc/collectd.conf -f
          21443 /usr/sbin/collectd -C /etc/collectd.conf -f
          21449 /usr/sbin/collectd -C /etc/collectd.conf -f
          21454 /usr/sbin/collectd -C /etc/collectd.conf -f
          21457 /usr/sbin/collectd -C /etc/collectd.conf -f
          21459 /usr/sbin/collectd -C /etc/collectd.conf -f
          21462 /usr/sbin/collectd -C /etc/collectd.conf -f
          21464 /usr/sbin/collectd -C /etc/collectd.conf -f
          21468 /usr/sbin/collectd -C /etc/collectd.conf -f
          21469 /usr/sbin/collectd -C /etc/collectd.conf -f
          21472 /usr/sbin/collectd -C /etc/collectd.conf -f
          21477 /usr/sbin/collectd -C /etc/collectd.conf -f
```



```

21482 /usr/sbin/collectd -C /etc/collectd.conf -f
21487 /usr/sbin/collectd -C /etc/collectd.conf -f
21491 /usr/sbin/collectd -C /etc/collectd.conf -f
21499 /usr/sbin/collectd -C /etc/collectd.conf -f
21501 /usr/sbin/collectd -C /etc/collectd.conf -f
21528 /bin/bash /etc/collectd.d/session_count.sh
25962 sleep 9

Aug 07 04:32:16 pcrfclient01 collectd[20596]: write_graphite plugin: getaddrinfo (localhost,
2013, tcp) failed: Name or service not known

Aug 07 04:32:19 pcrfclient01 collectd[20596]: write_graphite plugin: getaddrinfo (localhost,
2013, tcp) failed: Name or service not known

Aug 07 04:32:19 pcrfclient01 collectd[20596]: network plugin: getaddrinfo (pcrfclient02,
25826) failed: Name or service not known

Aug 07 04:32:20 pcrfclient01 collectd[20596]: write_graphite plugin: getaddrinfo (localhost,
2013, tcp) failed: Name or service not known

[root@hnpcrfoam01 ~]# netstat -plan | grep 2013

tcp        0      0 127.0.0.1:2013          0.0.0.0:*                LISTEN      20091/python2

/var/log/carbon/console.log :

07/08/2020 04:28:36 :: Received SIGTERM, shutting down.
07/08/2020 04:28:36 :: Carbon shutting down. Changed the update rate to: 1000
07/08/2020 04:28:36 :: (TCP Port 7102 Closed)
07/08/2020 04:28:36 :: Stopping factory <twisted.internet.protocol.ServerFactory instance
at 0x7f9500295e18>
07/08/2020 04:28:36 :: (TCP Port 2104 Closed)
07/08/2020 04:28:36 :: Stopping factory <twisted.internet.protocol.ServerFactory instance
at 0x7f9500298098>
07/08/2020 04:28:36 :: (TCP Port 2103 Closed)
07/08/2020 04:28:36 :: Stopping factory <twisted.internet.protocol.ServerFactory instance
at 0x7f9500299d40>
07/08/2020 04:28:48 :: Main loop terminated.
07/08/2020 04:28:48 :: Warning: No permission to delete pid file
07/08/2020 04:28:48 :: Server Shut Down.

07/08/2020 04:28:36 :: Exception in metricGenerated event handler:
args=('cisco.quantum.qps.pcrfcs05.nodel.actions.IldapAddition.success', (1596774516.0, 0.0))
kwargs={}
Traceback (most recent call last):
  File "/usr/lib/python2.7/site-packages/carbon/protocols.py", line 117, in stringReceived
    self.metricReceived(metric, datapoint)
  File "/usr/lib/python2.7/site-packages/carbon/protocols.py", line 64, in metricReceived
    events.metricReceived(metric, datapoint)
  File "/usr/lib/python2.7/site-packages/carbon/events.py", line 20, in __call__
    handler(*args, **kwargs)
  File "/usr/lib/python2.7/site-packages/carbon/aggregator/receiver.py", line 36, in process
    events.metricGenerated(metric, datapoint)
--- <exception caught here> ---

```

```

File "/usr/lib/python2.7/site-packages/carbon/events.py", line 20, in __call__
    handler(*args, **kwargs)
File "/usr/lib/python2.7/site-packages/carbon/client.py", line 267, in sendDataPoint
    for destination in self.router.getDestinations(metric):
File "/usr/lib/python2.7/site-packages/carbon/routers.py", line 78, in getDestinations
    for (count, node) in enumerate(self.ring.get_nodes(key)):
File "/usr/lib/python2.7/site-packages/carbon/hashing.py", line 42, in get_nodes
    assert self.ring
exceptions.AssertionError:

```

**Solution:** Restart the collectd process using `/usr/bin/systemctl restart collectd` command.

## SNMP Traps and Key Performance Indicators (KPIs)

### Full (HA) Setup

**Step 1** Check whether `snmpd` service is running on all VMs. If the service is not running then start it by executing the command:

```
monit start snmpd
```

**Step 2** Check whether `snmptrapd` is running on policy director (lb) VMs. If the service is not running then start it by executing the command:

```
monit start snmptrapd
```

**Step 3** On `pcrfclient01`:

a) Verify whether `/etc/broadhop/<server_name>/snmp/manager.xml` file has the following content. If the content is not present, add the following content to the file:

**Note** `server_name` details can be found from `/etc/broadhop/server` file.

```

<manager-list>
  <manager>
    <address>localhost</address>
    <port>162</port>
    <version>1</version>
  </manager>
</manager-list>

```

b) Execute the command `syncconfig.sh` so that the change done in Step 3.a, on page 130 gets synchronized to all VMs.

c) Execute the command `restartall.sh` to restart all policy server (qns) processes.

**Caution** Executing `restartall.sh` will cause messages to be dropped.

d) Verify whether service `monit` is running or not. If the service is not running then start it by executing the command:

```
service monit start
```

**Note** If `monit` is not installed on OAM (pcrfclient) VMs, then you need to get the `monit rpm` and install it in on all OAM (pcrfclient) VMs.

e) Verify whether `monit.conf` file has entries of `check_program` executing different traps generating script. If the entries are not present, then get the latest `monit.conf` file for OAM (pcrfclient) VMs and update it on all OAM (pcrfclient) VMs setup.

- f) Restart monit service.

```
service monit start
```

**Step 4** On policy director (lb) VMs:

- a) Verify whether `/etc/hosts` file has the entry as `corporate_nms_ip <ip_address>`.

**Note** `<ip_address>` is the NMS address.

- b) Verify whether service monit is running or not, If the service is not running then start it by executing the command:

```
service monit start
```

**Note** If monit is not installed on policy director (lb) VMs then you need to get the monit rpm and install it on all policy director (lb) VMs.

- c) Verify whether `monit.conf` file has entries of `check_program` executing different traps generating script. If the entries are not present then get the latest `monit.conf` file for policy director (lb) VMs and update it on all policy director (lb) VMs,
- d) Restart monit service.

```
service monit start
```

## Testing Traps Generated by CPS

The following tables describe the SNMP notifications (traps) generated by CPS as well as the procedures that can be used to test their operation.

For a complete list of CPS traps, including detailed descriptions, refer to the *CPS SNMP and Alarms Guide*, Release 9.1.0 and prior releases or *CPS SNMP, Alarms and Clearing Procedures Guide*, Release 10.0.0 and later releases.

### Component Notifications

**Table 11: Component Notifications**

Alarm Name	Procedure to Test
DiskFull	<ol style="list-style-type: none"> <li>1. In <code>/etc/snmp/snmpd.conf</code>, set "disk / 90%". (So when disk remaining is 90% i.e. Disk occupied is 10%, alarm is generated.)</li> <li>2. Restart the snmpd process. <b>monit restart snmpd</b></li> <li>3. Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active Policy Director (lb).</li> <li>4. Trap have messages like <code>:dskErrorMsg.1 = STRING: /: less than 90% free (= 100%)</code></li> </ol>

Alarm Name	Procedure to Test
DiskFull	<ol style="list-style-type: none"> <li>In <code>/etc/snmp/snmpd.conf</code>, set "disk / X%". (X should just less than actual remaining space. For example, if drive / is 25% full, put 74% as value of X).</li> <li>Restart the snmpd process. <b>monit restart snmpd</b></li> <li>Now dump a big file which consumes at least 2-3 % space on drive /. This generates diskful alarm first.</li> <li>Delete this file. This generates clear alarm.</li> <li>Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active Policy Director (lb).</li> </ol>
HighLoadAlert	<ol style="list-style-type: none"> <li>In <code>/etc/snmp/snmpd.conf</code>, set "load 1 1 1". (first digit corresponds to average 1 min load. Second digit is for 5 minutes average load. Third is for 15 mins. When it crosses 1 %, alarm is generated.)</li> <li>Restart the snmpd process. <b>monit restart snmpd</b></li> <li>Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active Policy Director (lb).</li> <li>Trap have message like 1 min Load Average too high (= 1.41)</li> </ol>
HighLoadClear	<ol style="list-style-type: none"> <li>In <code>/etc/snmp/snmpd.conf</code>, set "load 1 1 1". (first digit corresponds to average 1 min load. Second digit is for 5 minutes average load. Third is for 15 mins. When load is below value (as mentioned 1 % ), clear alarm is generated.)</li> <li>Restart the snmpd process. <b>monit restart snmpd</b></li> <li>Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active Policy Director (lb).</li> </ol>
LowSwapAlert	<ol style="list-style-type: none"> <li><b>swapoff -a</b> This command disables all swap areas. Use the top command to see that the swap has been disabled: "Swap: 0k total".</li> <li>Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active Policy Director (lb): "QNS component notification Running out of swap space".</li> </ol>

Alarm Name	Procedure to Test
LowSwapClear	<ol style="list-style-type: none"> <li>1. <b>swapon -a</b>                      This command enables all swap areas again.                      The top command output shows the correct swap memory size (not 0k total). The clear trap gets generated if swap alarms was generated earlier.</li> <li>2. Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb): “QNS component notification Swap space recovered”.</li> </ol>
Link Down	<ol style="list-style-type: none"> <li>1. <code>ifconfig &lt;interface_name&gt; down</code> (For example, <code>ifconfig eth2 down</code>)</li> <li>2. Within 1 minute interval interface down trap gets generated.</li> <li>3. Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>
Link Up	<ol style="list-style-type: none"> <li>1. <code>ifconfig &lt;interface_name&gt; up</code> (For example, <code>ifconfig eth2 up</code>)</li> <li>2. Within 1 minute interval interface up trap gets generated</li> <li>3. Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>
LowMemoryAlert	<ol style="list-style-type: none"> <li>1. In output of top command find out the current free RAM memory value.</li> <li>2. Update <code>snmpd.conf</code> file monitor entry for Low Memory Alert to have value just less than the current free RAM memory value.</li> <li>3. Restart the <code>snmpd</code> process.  <b>monit restart snmpd</b></li> <li>4. Do some activity on VM such as running some command or starting some process so that free RAM value goes below the configured value.</li> <li>5. The low memory alert alarm gets generated within a minute interval.</li> <li>6. Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>

Alarm Name	Procedure to Test
LowMemoryClear	<ol style="list-style-type: none"> <li>1. In output of top command find out the current free RAM memory value.</li> <li>2. Update <b>snmpd.conf</b> file monitor entry for Low Memory Clear to have value just more than the current free RAM memory value.</li> <li>3. Restart the snmpd process. <b>monit restart snmpd</b></li> <li>4. Kill some processes on VM so that free RAM memory value is more than the configured value.</li> <li>5. The low memory clear alarm gets generated within a minute interval.</li> <li>6. Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>
ProcessDown	<ol style="list-style-type: none"> <li>1. On the Load Balancer VMs, issue the following command to stop the corosync process: <b>monit stop corosync</b></li> <li>2. Within 5 minutes of interval process down trap is generated.</li> <li>3. Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb): “QNS component notification corosync process is down”.</li> </ol>
ProcessUp	<ol style="list-style-type: none"> <li>1. Issue the following command to restart the <b>corosync</b> process: <b>monit start corosync</b></li> <li>2. Within 5 minutes of interval process up trap is generated.</li> <li>3. Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb):“QNS component notification corosync process is up”.</li> </ol>
HIGH CPU USAGE Alert	<ol style="list-style-type: none"> <li>1. Change the threshold value for the CPU usage alert (<code>cpu_usage_alert_threshold</code>) to a lower value. The default value is 80 percent.  Refer to the <i>CPS SNMP, Alarms and Clearing Procedures Guide</i> for steps to configure this threshold.</li> <li>2. The system generates an Alert trap whenever the CPU usage of the VM goes above be higher than this value.</li> <li>3. Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>

Alarm Name	Procedure to Test
HIGH CPU USAGE Clear	<ol style="list-style-type: none"> <li>1. Change the clear threshold value for CPU usage (cpu_usage_clear_threshold) to a higher value. The default value is 40 percent. Refer to the <i>CPS SNMP, Alarms and Clearing Procedures Guide</i> for steps to configure this threshold.</li> <li>2. The system generates a Clear trap whenever the CPU usage of the VM drops below this threshold value. It is generated only when a High CPU Usage Alert was generated earlier.</li> <li>3. Verify the generated alarm on NMS server and /var/log/snmp/trap of active policy director (lb).</li> </ol>
Critical File Operation Alert	<ol style="list-style-type: none"> <li>1. Configure critical file monitoring configuration on VMWare/OpenStack. For more information, refer to <i>CPS Installation Guide for VMware</i> and <i>CPS Installation Guide for OpenStack</i>.</li> <li>2. Modify/change attributes in any monitored file.</li> <li>3. The system generates a trap as per SNMP configuration (SNMP v2c/v3) on lbvip02 VM and similar trap is forwarded to NMS server configured.</li> </ol>

## Application Notifications

Table 12: Application Notifications

Alarm Name	Procedure to Test
MemcachedConnect Error	<ol style="list-style-type: none"> <li>1. Kill the memcached process running on active policy director (lb).</li> <li>2. Within 5 minutes of interval memcached Connect Error trap gets generated from policy server (QNS) VMs.</li> <li>3. Verify the generated alarm on NMS server and /var/log/snmp/trap of active policy director (lb).</li> </ol>
ApplicationStartError	<p><b>Note</b> Take the configuration backup before applying the procedure.</p> <ol style="list-style-type: none"> <li>1. Remove the balance configuration from Policy Builder and publish the changes.</li> <li>2. Restart the policy server (QNS) process.</li> <li>3. Within 5 minute of interval ApplicationStartError trap gets generated on active policy director (lb).</li> <li>4. Verify the generated alarm on NMS server and /var/log/snmp/trap of active policy director (lb).</li> </ol>

Alarm Name	Procedure to Test
License Usage Threshold Exceeded	<ol style="list-style-type: none"> <li>1. Create the license having small number of Usage Threshold limit.</li> <li>2. Install the above created license on setup.</li> <li>3. Restart all policy server (QNS) processes.</li> <li>4. Send multiple request so that it crosses the threshold limit.</li> <li>5. The License Usage Threshold Exceeded alarm gets generated.</li> <li>6. Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>
LicensedSessionCreation	<ol style="list-style-type: none"> <li>1. Create the license having small number of Session Usage Threshold limit.</li> <li>2. Install the above created license on setup.</li> <li>3. Restart all policy server (QNS) processes.</li> <li>4. Send multiple request so that it crosses session threshold limit.</li> <li>5. For the next request after the limit over LicenseSessionCreation alarm gets generated.</li> <li>6. Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>
InvalidLicense	<ol style="list-style-type: none"> <li>1. Copy the license of <code>perfclient02</code> on <code>perfclient01</code> or create a license for <code>perfclient02</code> and install it on <code>perfclient01</code>.</li> <li>2. Restart <b>lmgrd</b> service.</li> <li>3. Restart the policy server (QNS) process.</li> <li>4. Within 5 minutes of interval the License invalid trap gets generated.</li> <li>5. Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>
PolicyConfiguration	<ol style="list-style-type: none"> <li>1. Configure some wrong policy in Policy Builder under the Policies tab.</li> <li>2. Publish the configuration.</li> <li>3. <b>restartall.sh</b>. <b>Caution</b> Executing <code>restartall.sh</code> will cause messages to be dropped.</li> <li>4. Last policy configuration failed with the following message:xxx trap gets generated.</li> <li>5. Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>



Alarm Name	Procedure to Test
PoliciesNotConfigured	<ol style="list-style-type: none"> <li>1. Create the invalid blueprint (java code having syntax error) in Policy Builder under the Policies tab.</li> <li>2. Assign the created blueprint to some policies.</li> <li>3. Publish the configuration.</li> <li>4. Restart all policy server (QNS) processes.</li> <li>5. PoliciesNotConfigured trap gets generated.</li> <li>6. Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>
DiameterPeerDown	<ol style="list-style-type: none"> <li>1. Make a seagull diameter call.</li> <li>2. After seagull script terminate it generates the diameter peer down trap.</li> <li>3. Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>
DiameterAllPeersDown	<ol style="list-style-type: none"> <li>1. Integrate CPS with two Seagull/SITE Instances.</li> <li>2. Make a seagull diameter call.</li> <li>3. Simultaneously make a diameter call from another Seagull/SITE Instance.</li> <li>4. After two Seagull/SITE scripts terminate it generates the DiameterAllPeersDown trap.</li> <li>5. Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>
HA_Failover	<ol style="list-style-type: none"> <li>1. Cat <code>/etc/broadhop/mongoConfig.cfg</code>.</li> <li>2. If there are two or more sessionmgr ports configured as replica set then find out the one acting as a primary member using <code>rs.isMaster().primary</code>.</li> <li>3. Shutdown the primary instance of sessionmgr.</li> <li>4. Within 1 minute of interval HA Failover trap gets generated.</li> <li>5. Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>

Alarm Name	Procedure to Test
GR_Failover	<ol style="list-style-type: none"> <li>1. Cat /etc/broadhop/mongoConfig.cfg.</li> <li>2. There should be primary and secondary member set for each replica set. Find the current active sessionmgr instance of a replica set using rs.isMaster().primary.</li> <li>3. Shutdown all sessionmgr instances of active sessionmgr instance set.</li> <li>4. Within 1 minute of interval Geo Failover trap gets generated.</li> <li>5. Verify the generated alarm on NMS server and /var/log/snmp/trap of active policy director (lb).</li> </ol>
All DB Member of replica Down	<ol style="list-style-type: none"> <li>1. Get all members of replica set from /etc/broadhop/mongoconfig.cfg.</li> <li>2. Go to each sessionMgr of a replica set and stop the sessionmgr service or shutdown the sessionmgr VM.</li> <li>3. Within 5 minutes of interval All replicas of DB Down trap gets generated.</li> <li>4. Verify the generated alarm on NMS server and /var/log/snmp/trap of active policy director (lb).</li> </ol>
All DB Member of replica Up	<ol style="list-style-type: none"> <li>1. Perform the steps above to generate the All DB Member of replica Down trap.</li> <li>2. Once that trap is generated, start the session manager service or bring up the sessionmanager VM.</li> <li>3. Within 5 minutes of interval All DB Member of replica Up trap gets generated.</li> <li>4. Verify the generated alarm on NMS server and /var/log/snmp/trap of active policy director (lb).</li> </ol>
No Primary DB Member Found	<ol style="list-style-type: none"> <li>1. Run <b>diagnostics.sh --get_replica_status</b>. <ul style="list-style-type: none"> <li><b>Note</b> If a member is shown in an unknown state, it is likely that the member is not accessible from one of other members, mostly an arbiter. In that case, you must go to that member and check its connectivity with other members.</li> <li>Also, you can login to mongo on that member and check its actual status.</li> </ul> </li> <li>2. Choose any set which has arbiter and primary and secondary database member.</li> <li>3. Shutdown Arbiter VM.</li> <li>4. Shutdown Primary Session Manager VM.</li> <li>5. Within 5 minutes of interval No primary Member found trap gets generated.</li> <li>6. Verify the generated alarm on NMS server and /var/log/snmp/trap of active policy director (lb).</li> </ol>

Alarm Name	Procedure to Test
Primary DB Member Found	<ol style="list-style-type: none"> <li>1. Run <b>diagnostics.sh --get_replica_status</b>.                             <p style="margin-left: 20px;"><b>Note</b> If a member is shown in an unknown state, it is likely that the member is not accessible from one of other members, mostly an arbiter. In that case, you must go to that member and check its connectivity with other members.</p> <p style="margin-left: 20px;">Also, you can login to mongo on that member and check its actual status.</p> </li> <li>2. Choose any set which has arbiter and primary and secondary database member.</li> <li>3. Shutdown Arbiter VM.</li> <li>4. Shutdown Primary Session Manager VM or stop the corresponding mongo set process.</li> <li>5. After 5 minutes, power on the Primary Session Manager VM.</li> <li>6. Within 5 minutes of interval, verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>
DB Member Down	<ol style="list-style-type: none"> <li>1. Cat <code>/etc/broadhop/mongoConfig.cfg</code>.</li> <li>2. Shutdown any of the sessionmgr VM listed in the configuration as database member of replica set.</li> <li>3. Within 5 minutes of interval database down trap gets generated.</li> <li>4. Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>
DB Member Up	<ol style="list-style-type: none"> <li>1. Perform the steps above to generate the DB Member Down trap.</li> <li>2. After 5 minutes, power on the sessionmgr VM (the secondary database) that was shutdown earlier.</li> <li>3. Within 5 minutes of interval, verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>
Arbiter Down	<ol style="list-style-type: none"> <li>1. Cat <code>/etc/broadhop/mongoConfig.cfg</code>.</li> <li>2. Shutdown any of the Arbiter VMs listed in the configuration.</li> <li>3. Within 5 minutes of interval Arbiter down trap gets generated.</li> <li>4. Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>

Alarm Name	Procedure to Test
Arbiter Up	<ol style="list-style-type: none"> <li>1. Perform the steps above to generate the Arbiter Down trap.</li> <li>2. After 5 minutes, power on the Arbiter VM that was shutdown earlier.</li> <li>3. Within 5 minutes of interval, verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>
DB resync is needed	<ol style="list-style-type: none"> <li>1. Cat <code>/etc/broadhop/mongoConfig.cfg</code>.</li> <li>2. Shutdown any of the sessionmgr VM (the secondary database) listed in the configuration as database member of replica set.</li> <li>3. From Primary member find out oplog holding seconds, using below command:  <b>mongo --host &lt;primary host name&gt; --port &lt;DB port number&gt; --eval 'rs.printReplicationInfo()'   grep 'log length start to end'</b> </li> <li>4. Wait till oplog holding seconds and check shutdown database member is in the RECOVERING state, using below command:  <b>diagnostics.sh --get_replica_status</b> <p><b>Note</b> If a member is shown in an unknown state, it is likely that the member is not accessible from one of other members, mostly an arbiter. In that case, you must go to that member and check its connectivity with other members.</p> <p>Also, you can login to mongo on that member and check its actual status.</p> </li> <li>5. When this database member goes to 'RECOVERING' state. After 5 minutes of interval 'DB resync is needed' trap gets generated.</li> <li>6. Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>

Alarm Name	Procedure to Test
DB resync is not needed	<ol style="list-style-type: none"> <li>1. Power on the sessionmgr VM (the secondary database) that was shutdown early.</li> <li>2. Stop the sessionmgr mongod process, using below command (XXXXXX change to database port number). <b>/usr/bin/systemctl stop sessionmgr-XXXXXX</b></li> <li>3. Clear data directory of that sessionmgr (specify correct data directory path). <b>\rm -fr &lt;data directory path of that mongod&gt;</b></li> <li>4. Start the sessionmgr mongod process, using below command (XXXXXX change to database port number). <b>/usr/bin/systemctl start sessionmgr-XXXXXX</b></li> <li>5. When this database member goes to 'SECONDARY' state. After 5 minutes of interval 'DB resync is not needed' trap gets generated, using below command: <b>diagnostics.sh --get_replica_status</b>   <p><b>Note</b> If a member is shown in an unknown state, it is likely that the member is not accessible from one of other members, mostly an arbiter. In that case, you must go to that member and check its connectivity with other members.</p> <p>Also, you can login to mongo on that member and check its actual status.</p> </li> <li>6. Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>
Config Server Down	<ol style="list-style-type: none"> <li>1. Cat <code>/etc/broadhop/mongoConfig.cfg</code>.</li> <li>2. Shutdown any of the Config Server VMs listed in the configuration.</li> <li>3. Within 5 minutes of interval, Config Server Down trap gets generated.</li> <li>4. Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>
Config Server Up	<ol style="list-style-type: none"> <li>1. Perform the steps above to generate the Config Server Down trap.</li> <li>2. After 5 minutes, power on the Config Server VM that was shutdown earlier.</li> <li>3. Within 5 minutes of interval, verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>

Alarm Name	Procedure to Test
<p>MongoPrimaryDB fragmentation exceeded the threshold value</p>	<p><b>1.</b> Start dumping the static sessions in DB.</p> <p>When there is no static session in DB, NoFrag (use <code>diagnostics.sh --get_frag_status</code> command) is displayed. Once session creation starts and reach up to few million, some fragmentation is induced at some point in any or all the session shards. Stop the static sessions once the fragmentation is induced and start to delete the session operation.</p> <p><b>2.</b> Delete sessions per shard using the following command and check the rise in Fragmentation percentage and continue to delete until 40% Frag is reached. The down alarm "MongoPrimaryDB fragmentation exceeded the threshold value" is triggered immediately in few seconds once percentage exceeds the threshold value of 40%. The alarm is triggered per shard per replica set.</p> <ul style="list-style-type: none"> <li>• <code>mongo sessionmgr01:(use mongo port)</code>: Run from Cluster Manager</li> <li>• <code>show dbs</code>: Displays all the session shards.</li> <li>• <code>use session_cache_3</code>: Use any session shard cache that has fragment induced.</li> <li>• <code>db.session.count()</code>: Displays the list of active sessions Example: <code>db.session.count({_v : { \$gt:0}})</code></li> <li>• <code>db.session.remove({ input the query/condition that removes a chunk of sessions})</code> Example: <code>db.session.remove({_v : { \$gt:0}})</code></li> </ul> <p><b>Note</b> Fragmentation percent threshold values are configured in <code>/etc/collectd.d/dbMonitorList.cfg</code> file (present on sessionmgr VMs) for all the databases. Default threshold value for all the databases is configured as 40%.</p> <p>The default fragmentation threshold value can be changed as required. For more information, refer to <i>Configure Custom Database Fragmentation Threshold Percentage</i> section in the <i>CPS Operations Guide</i>.</p>

Alarm Name	Procedure to Test
<p>MongoPrimaryDB fragmentation conforms to the threshold value</p>	<ol style="list-style-type: none"> <li data-bbox="621 296 1528 352">1. To reduce fragmentation percentage, database can be shrunk by following steps in <i>Resync Member of a Replica Set</i> section in <i>CPS Operations Guide</i>.</li> </ol> <p data-bbox="667 373 1528 430"><b>Note</b> If bulk deletes are done due to any maintenance activity, then database shrink is recommended to reduce the fragmentation.</p> <ol style="list-style-type: none"> <li data-bbox="621 472 1528 562">2. Within few minutes of fragmentation reduction, the Up alarm "MongoPrimaryDB fragmentation conforms to the threshold value" gets generated for the primary member of the set.</li> <li data-bbox="621 583 1528 640">3. Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol> <p data-bbox="667 661 760 695"><b>Sample:</b></p> <pre data-bbox="667 716 1528 1325"># tail -f /var/log/snmp/trap   grep 7107   grep 2020-01-28   grep MongoPrimaryDB  2020-01-28T13:52:13.387025+00:00 lb01 snmptrapd[23220]: 2020-01-28 13:52:13 pcrfclient01 [192.166.22.5] (via UDP: [192.166.22.5]:33745-&gt;[192.166.22.27]:162) TRAP, SNMP v1, community public#012#011BROADHOP-MIB::broadhopNotificationPrefix Enterprise Specific Trap (BROADHOP-MIB::broadhopClearAlarm) Uptime: 964645484#012#011BROADHOP-MIB: :broadhopAlarmDeviceName = STRING: QNS#011BROADHOP-MIB::broadhopAlarmErrorNumber = INTEGER: 7100#011BROADHOP-MIB::broadhopAlarmErrorText = STRING: KpiEvent [id=7100,values={sub_id=7107, event_host=sessionmgr01, status=up, msg="MongoPrimaryDB fragmentation conforms to the threshold value, CURR_FRAG = 37%, THRESHOLD = 40% at sessionmgr01:27717 for session_cache of set01\"]]#011BROADHOP-MIB::broadhopAlarmDateAndTime = STRING: 2020-01-28 at 13:52:13 +0000#011BROADHOP-MIB::broadhopAlarmProbableCause = STRING: #011BROADHOP-MIB: :broadhopAlarmAdditionalInfo = STRING:</pre>
<p>VM Down</p>	<ol style="list-style-type: none"> <li data-bbox="621 1367 1528 1400">1. Cat <code>/etc/hosts</code> file on policy director (lb) VM.</li> <li data-bbox="621 1421 1528 1455">2. Shutdown and power off any of the VMs listed under <code>/etc/hosts</code>.</li> <li data-bbox="621 1476 1528 1509">3. Within 5 minutes of interval VM down trap gets generated.</li> <li data-bbox="621 1530 1528 1585">4. Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>
<p>VM Up</p>	<ol style="list-style-type: none"> <li data-bbox="621 1627 1528 1661">1. Perform the steps above to generate the VM Down trap.</li> <li data-bbox="621 1682 1528 1715">2. After 5 minutes, power on the VM that was shutdown earlier.</li> <li data-bbox="621 1736 1528 1791">3. Within 5 minutes of interval, verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>

Alarm Name	Procedure to Test
QNS Process Down	<ol style="list-style-type: none"> <li>1. Stop the policy server (QNS) process using the command: <b>monit stop qnsXX</b>.</li> <li>2. Within 5 minutes of interval CPS process down trap gets generated.</li> <li>3. Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>
QNS Process Up	<ol style="list-style-type: none"> <li>1. Perform the steps above to generate the CPS Process Down trap.</li> <li>2. After 5 minutes, start the process again using the command: <b>monit start qnsXX</b>.</li> <li>3. Within 5 minutes of interval, verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>
Admin Logged In	<ol style="list-style-type: none"> <li>1. Create a new telnet session for any VM and login with root user on it.</li> <li>2. Within 1 minute interval Admin User logged in trap gets generated.</li> <li>3. Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>
Developer Mode	<ol style="list-style-type: none"> <li>1. Use developer mode by adding the following in <code>qns.conf</code> file: <code>-Dcom.broadhop.developer.mode.</code></li> <li>2. Restart the policy server (QNS) process.</li> <li>3. Within 5 minutes interval the Developer Mode License gets generated.</li> <li>4. Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>
Developer Mode Clear	<ol style="list-style-type: none"> <li>1. Perform the steps above to generate the Developer Mode License trap.</li> <li>2. Now remove the following line from the <code>qns.conf</code> file: <code>-Dcom.broadhop.developer.mode.</code></li> <li>3. Restart the policy server (QNS) process.</li> <li>4. Within 5 minutes of interval, verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>
ZeroMQConnectionError	<ol style="list-style-type: none"> <li>1. Start policy server (QNS).</li> <li>2. Start Messaging Load (CCR-I,CCR-U,CCR-T) scenario at high TPS.</li> <li>3. The trap will be seen if message sending over socket between policy director (lb) and policy server (QNS) fails (Due to socket send errors). For subsequent failures there is no further trap raised.</li> <li>4. Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>



Alarm Name	Procedure to Test
ZeroMQConnectionError Clear	<ol style="list-style-type: none"> <li>1. This trap will be sent when message send on socket succeeds after the prior failure.</li> <li>2. Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>
VirtualInterfaceDown	<ol style="list-style-type: none"> <li>1. Login to active policy director (lb) VM.</li> <li>2. Run command <b>ifconfig eth1:0 down</b>.</li> <li>3. VirtualInterface Down trap with the interface name gets generated.</li> <li>4. You can see this trap on NMS server.</li> </ol>
VirtualInterfaceUp	<ol style="list-style-type: none"> <li>1. Login to active policy director (lb) VM.</li> <li>2. Run command <b>ifconfig eth1:0 up</b>.</li> <li>3. VirtualInterface Up trap with the interface name gets generated.</li> <li>4. You can see this trap on NMS server.</li> </ol>
LdapAllPeersDown	<ol style="list-style-type: none"> <li>1. Configure LDAP in CPS and verify the connection between CPS and LDAP. <b>netstat -an   grep 389</b> Let us say you configure 2 LDAP servers.</li> <li>2. Bring down the LDAP server: Kill the LDAP process on LDAP server or break the connectivity between LDAP and CPS (for example, block the port through firewall).</li> <li>3. Verify the LdapAllPeersDown alarm was generated on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol> <p>This alarm will be generated only when all the LDAP servers configured in the CPS are down.</p>
LdapAllPeersDown Clear	<ol style="list-style-type: none"> <li>1. Perform the steps above to generate the LdapAllPeersDown trap.</li> <li>2. Bring up any one or both the LDAP servers.</li> <li>3. Verify the LdapAllPeersDown Clear alarm was generated on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>

Alarm Name	Procedure to Test
LdapPeerDown	<ol style="list-style-type: none"> <li>1. Configure LDAP in CPS and verify the connection between CPS and LDAP. <b>netstat -an   grep 389</b> Let us say you configure 2 LDAP servers.</li> <li>2. Bring down any one LDAP server: Kill the LDAP process on LDAP server or break the connectivity between LDAP and CPS (for example, block the port through firewall).</li> <li>3. Verify the LdapPeersDown alarm was generated on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb). Verify that the IP address of the LDAP server is correct in the alarm.</li> </ol> <p>So, this alarm is generated per LDAP server.</p>
LdapPeerDown Clear	<ol style="list-style-type: none"> <li>1. Perform the steps above to generate the LdapPeerDown trap.</li> <li>2. Bring up the LDAP server.</li> <li>3. Verify the LdapPeerDown Clear alarm was generated on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb). Verify that the IP address of the LDAP server is correct in the alarm.</li> </ol>
Percentage of LDAP retry threshold Exceeded	<ol style="list-style-type: none"> <li>1. CPS HA is deployed as per guidelines provided in the <i>CPS Installation Guide for VMware</i>.</li> <li>2. Run Gx diameter calls and LDAP (configure multiple LDAP servers).</li> <li>3. Verify Call Model is stable using <b>top_qps.sh</b> command.</li> <li>4. Check for latest log: <code>/var/log/broadhop/scripts/gen-ldap-trap.log</code>.</li> <li>5. If system (all policy server (QNS) VMs) is processing Gx and LDAP messages normal, then normal text message will be logged into the log file.</li> <li>6. Abruptly shutdown LDAP server.</li> <li>7. Within 30 seconds of interval, trap (dropped alarm) is generated. Verify receipt of the alarm on NMS and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>
Percentage of LDAP retry threshold Normal	<ol style="list-style-type: none"> <li>1. After dropped trap alarm is generated, restart LDAP server.</li> <li>2. Within 30 seconds of interval, trap (clear indicator) is generated</li> <li>3. Verify the clear indicator was generated on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>
LDAP Requests as percentage of CCR-I Dropped	Refer to steps for <i>Percentage of LDAP retry threshold Normal</i> alarm.

Alarm Name	Procedure to Test
LDAP Requests as percentage of CCR-I Normal	Refer to steps for <i>Percentage of LDAP retry threshold Normal</i> alarm.
LDAP Request Dropped	Refer to steps for <i>Percentage of LDAP retry threshold Normal</i> alarm.
LDAP Requests Normal	Refer to steps for <i>Percentage of LDAP retry threshold Normal</i> alarm.
LDAP Query Result Dropped	Refer to steps for <i>Percentage of LDAP retry threshold Normal</i> alarm.
LDAP Query Result Normal	Refer to steps for <i>Percentage of LDAP retry threshold Normal</i> alarm.
Gx Message processing Dropped	<ol style="list-style-type: none"> <li>1. CPS HA is deployed as per guidelines provided in the Cisco Policy Suite Installation Guide.</li> <li>2. Configure Message Handling Rules in Policy Builder.</li> <li>3. Run Gx diameter calls (CCR-I, U or T).</li> <li>4. Verify Call Model is stable using top_qps.sh command.</li> <li>5. Check for latest log: /var/log/broadhop/scripts/gen-gx-drop-trap.log.</li> <li>6. If system (all policy server (QNS) VMs) is processing Gx messages normally, then normal text messages will be logged into the log file.</li> <li>7. Increase the Gx message load beyond system capacity, such that threshold configured in Policy Builder should be breached.</li> <li>8. Within 30 seconds of interval, trap (dropped alarm) is generated. Verify receipt of the alarm on NMS and /var/log/snmp/trap of active policy director (Ib).</li> </ol>
Gx Message processing Normal	<ol style="list-style-type: none"> <li>1. After Gx Message Dropped trap alarm is generated, reduce traffic within system capacity.</li> <li>2. Within 30 seconds of interval, trap (clear indicator) is generated</li> <li>3. Verify the Gx Message processing Normal alarm was generated on NMS server and /var/log/snmp/trap of active policy director (Ib).</li> </ol>

Alarm Name	Procedure to Test
Average Gx Message processing Dropped	<ol style="list-style-type: none"> <li>1. CPS HA is deployed as per guidelines provided in the Cisco Policy Suite Installation Guide.</li> <li>2. Configure Message Handling Rules in Policy Builder.</li> <li>3. Run Gx diameter calls (CCR-I, U or T).</li> <li>4. Verify Call Model is stable using <b>top_qps.sh</b> command.</li> <li>5. Check for latest log: /var/log/broadhop/scripts/gen-gx-drop-trap.log.</li> <li>6. If system (all policy server (QNS) VMs) is processing Gx messages normally, then normal text messages will be logged into the log file.</li> <li>7. Increase the Gx message load beyond system capacity, such that threshold configured in Policy Builder should be breached.</li> <li>8. Within 30 seconds of interval, trap (dropped alarm) is generated. Verify receipt of the alarm on NMS and /var/log/snmp/trap of active policy director (lb).</li> </ol>
Average Gx Message processing Normal	<ol style="list-style-type: none"> <li>1. After Average Gx Message processing Dropped alarm is generated, reduce traffic within system capacity.</li> <li>2. Within 30 seconds of interval, trap (clear indicator) is generated</li> <li>3. Verify the Gx Message processing Normal alarm was generated on NMS server and /var/log/snmp/trap of active policy director (lb).</li> </ol>
AllSMSCNotification ServerDown	<ol style="list-style-type: none"> <li>1. Stop all the Active SMSC servers.</li> <li>2. Verify receipt of the alarm on NMS and /var/log/snmp/trap of active policy director (lb).</li> </ol>
AtLeastOneSMSC NotificationServerUp	<ol style="list-style-type: none"> <li>1. Start any of the configured SMSC servers.</li> <li>2. Verify receipt of the alarm on NMS and /var/log/snmp/trap of active policy director (lb).</li> </ol>
SMSCNotification ServerDown	<ol style="list-style-type: none"> <li>1. Stop one of the active SMSC servers.</li> <li>2. Verify receipt of the alarm on NMS and /var/log/snmp/trap of active policy director (lb).</li> </ol>
SMSCNotification ServerUp	<ol style="list-style-type: none"> <li>1. Start one of the down and configured SMSC servers.</li> <li>2. Verify receipt of the alarm on NMS and /var/log/snmp/trap of active policy director (lb).</li> </ol>

Alarm Name	Procedure to Test
AllEmailNotification ServerDown	<ol style="list-style-type: none"> <li>1. Close all SMTP servers defined.</li> <li>2. In Wireshark trace, Major alarm will be triggered along with Critical alarm as 'Email server not reachable' and 'All Email servers not reachable'.</li> </ol>
AtLeastOneEmail NotificationServerUp	<ol style="list-style-type: none"> <li>1. Perform the steps above to generate the Email server not reachable and All Email servers not reachable traps.</li> <li>2. Since all the servers are down, try bringing up only one SMTP server.</li> <li>3. Verify receipt of the alarm on NMS and /var/log/snmp/trap of active policy director (lb).</li> </ol>
EmailNotification ServerDown	<ol style="list-style-type: none"> <li>1. Consider multiple SMTP servers are defined in CPS under 'Multiple Email Server Configuration' with different ports.</li> <li>2. Close any one of the SMTP servers (this will make the SMTP server not reachable), and keep the Wireshark trace ON.</li> <li>3. Filter out the Wireshark trace with SNMP.</li> <li>4. Verify receipt of the alarm on NMS and /var/log/snmp/trap of active policy director (lb).</li> </ol>
EmailNotification ServerUp	<ol style="list-style-type: none"> <li>1. Perform the steps above to generate the Email server not reachable trap.</li> <li>2. Now bring up the SMTP server that was powered OFF.</li> <li>3. In Wireshark trace another alarm (Clear Alarm) will be triggered as 'Email server reachable'.</li> </ol>
SPR_DB_ALARM	<ol style="list-style-type: none"> <li>1. Introduce a network failure or latency from Policy Server (qns) nodes to the remoteSpr databases or decrease -DserverSelectionTimeout.remoteSpr value in qns.conf file. Observe if the alarm is raised.</li> <li>2. Correct the failures introduced and restart Policy Server (qns) services. The alarm should get cleared.</li> </ol>
Binding Not Available at Policy DRA	<ol style="list-style-type: none"> <li>1. Enable Binding-db-health-check feature for the APN.</li> <li>2. PCRF sends health check AAR to Policy DRA.</li> <li>3. Alarm is generated immediately if Policy DRA sends AAA with error cause code configured in Policy Builder (<b>Diameter Configuration &gt; PolicyDRA Health Check &gt; Alarm Config &gt; Alarm Clearance Interval &gt; Policy Dra Resultcode</b>).</li> <li>4. Verify the generated alarm on NMS server and /var/log/snmp/trap of active policy director (lb).</li> </ol>

Alarm Name	Procedure to Test
DiameterQnsWarmupError	<ol style="list-style-type: none"> <li>1. Enable the warmup feature by configuring the <code>qns.node.warmup</code> to true in <code>qns.conf</code> file.</li> <li>2. Do not configure <code>qns.node.warmup.hostname.substring</code> parameter in <code>qns.conf</code>.</li> <li>3. The <b>DiameterQnsWarmupError</b> alarm is generated.</li> </ol>
SPRNodeNotAvailable	<ol style="list-style-type: none"> <li>1. Run <code>diagnostics.sh --get_replica_status</code> command to get the status of replica sets.</li> <li>2. Select the SPR repset configured under <b>USuM Configuration &gt; Shard Configuration</b>.</li> <li>3. Go to each sessionmgr of a replica-set and stop the sessionmgr service for that replica-set or shutdown the sessionmgr VM.</li> <li>4. Run <code>diagnostics.sh -get_active_alarms</code> to verify the alarm is generated.</li> <li>5. Verify the generated alarm on NMS server and <code>/var/log/snmp/trap</code> of active policy director (lb).</li> </ol>
GC State	There is no procedure to test this alarm. This alarm is generated when there are memory leak issues in software.
OldGen State	There is no procedure to test this alarm. This alarm is generated when there are memory leak issues in software.
SessionLimitOverload ProtectionNotSet	Set the recommended value for <b>Session limit Overload Protection</b> under <b>System</b> configuration in Policy Builder and publish it. During the upgrade to the build where the parameter is introduced, the value will be defaulted to 0. In this case, alarm is raised so that you can change the value to recommended value (the value differs for each deployment).
SessionLimitOverload ProtectionExceeded	Send 'n' number of CCR-I's to CPS so that 'n' sessions are created in session database (this is total count of sessions from all the session replica sets which can be found using <code>session_cache_ops.sh --count</code> ). Assume value set for <b>Session Limit Overload Protection</b> as 'm'. If $n > m$ , then the alarm will be raised within 30 seconds.
SESSION_SHARD_ UNREACHABLE	Bring down the VM. Wait for few seconds for the alarm to show up in <code>diagnostics -get_active_alarms</code> .
ADMIN_DB_MISSING_ SHARD_ENTRIES	<p>Either do not create shards after a fresh install in GR environment or remove all shard entries in the ADMIN replica-set &gt; sharding database &gt; shards collection.</p> <p>It takes atleast "x" number of seconds before which this alarm is generated, the default being 30 seconds.</p>
MISSING_SESSION_ INDEXES	Drop an existing index from any one of the session collections using mongo CLI. This alarm shows up on a Policy Server (qns) restart on any one of the nodes, only when an index creation fails. In most cases the indexes are successfully created during startup time hence one might not see the alarm. Only when there is an index creation failure, the alarm is generated.

Alarm Name	Procedure to Test
MISSING_SPR_INDEXES	<p>Drop an existing index from any one of the SPR collections using mongo CLI. This alarm shows up only on a Policy Server (qns) restart on any one of the nodes, as indexes get created during the process startup time.</p>
ProcessRestarted	<ol style="list-style-type: none"> <li>1. Stop qns processes on any of the VM where qns processes are running by running the following command.   <pre>kill -9 &lt;process_pid&gt;</pre> </li> <li>2. Wait for some time (approximately 30-35 seconds) till process get restarted.</li> <li>3. Verify the ProcessRestarted trap generated on NMS and /var/log/snmp/trap file on active lb VM.</li> </ol>
Database Operation	<ol style="list-style-type: none"> <li>1. Reboot the primary Session Manager VM which is having the highest priority in a replica-set.   <pre>reboot</pre> </li> <li>2. Once the primary Session Manager VM is up, change the time stamp of the VM to 00:00:00.   <pre>date -s "DD MM YYYY HH:MM:SS"</pre> <p><b>Example:</b> <code>date -s "19 APR 2019 00:00:00"</code></p> <p>Once the primary state of replica-set is taken over by same session manager, application fails to detect the session manager's role as primary. As a result, application fails to perform the write operation and an alarm is generated.</p> <p>You can see the following exception in the application logs:</p> <pre>2019-03-28 14:05:37,449 [pool-1008134-thread-1] ERROR c.b.i.a.impl.IpstaticApiServiceImpl.? - Error occurred while calling the addStaticIpForSinglePartner Timed out after 1200 ms while waiting for a server that matches WritableServerSelector. Client view of cluster state is {type=REPLICA_SET, servers= { Mongo Client details } }</pre> </li> </ol>
SVNnotinsync	<p>Turn off httpd services on one perfcient VM.</p> <p>This causes the SVN revision not to be available on the corresponding VM and critical event is triggered. The event contains the event id, sub id, event host and the corresponding message showing the revision number details.</p> <p>As a result, an alarm is triggered.</p> <p>The event is sent to NMS and the corresponding log is captured in /var/log/snmp/trap on Policy Director (lb) VM.</p>

Alarm Name	Procedure to Test
Realtime Notification ServerDown	<ol style="list-style-type: none"> <li>1. Consider Realtime Notification server is defined in CPS under <b>Realtime Notification Server Configuration</b>.</li> <li>2. Shutdown the Realtime Notification server and trigger the Realtime Notification Server not reachable alarm.</li> <li>3. Verify that the event is sent to NMS and the corresponding log is captured in <code>/var/log/snmp/trap</code> of active Policy Director (lb).</li> </ol>
Realtime Notification ServerUp	<ol style="list-style-type: none"> <li>1. Perform the steps above to generate the Realtime Notification server not reachable alarm.</li> <li>2. Bring up the Realtime Notification server that was powered OFF.</li> <li>3. Verify that the event is sent to NMS and the corresponding log is captured in <code>/var/log/snmp/trap</code> of active Policy Director (lb).</li> </ol>

## SNMP System and Application KPI Values

- [SNMP System KPIs, on page 152](#)
- [Application KPI Values, on page 153](#)

### SNMP System KPIs

In this table, the system KPI information is provided:

*Table 13: SNMP System KPIs*

Component	Information
lb01/lb02	CpuUser
perfcient01/perfcient02	CpuSystem
sessionMgr01/sessionMgr02	CpuIdle
QNS01/QNS02/QNS03/QNS04...	CpuIdle
	LoadAverage1
	LoadAverage5
	LoadAverage15
	MemoryTotal
	MemoryAvailable
	SwapTotal
	SwapAvailable



## Application KPI Values

Table 14: Application KPI Values

KPI Values	
lb01/lb02	<p><b>snmpwalk -v 2c -c broadhop -M +BROADHOP-MIB:CISCO-QNS-MIB &lt;lb01&gt; &lt;OIDvalue&gt;</b></p> <p>For example, <b>snmpwalk -v 2c -c broadhop -M +BROADHOP-MIB:CISCO-QNS-MIB lb01 .1.3.6.1.4.1.26878.200.3.3.70.11</b></p> <p>List all KPIs value of load balancer (lb), if all values are 0 then</p> <p>For ExternalCurrentSession:</p> <ol style="list-style-type: none"> <li>1. Open another terminal.</li> <li>2. Enter the following command: <b>telnet &lt;lbvip01&gt; 8443</b></li> <li>3. On previous terminal run the above <b>snmpwalk</b> command again.</li> <li>4. This time it will display the externalCurrentSession KPIs value to be 1.</li> <li>5. Repeat the process with more telnet session open on lbvip01 8080 port</li> </ol> <p>For InternalCurrentSession:</p> <ol style="list-style-type: none"> <li>1. Open another terminal.</li> <li>2. Enter the following command: <b>telnet &lt;lbvip02&gt; 8080</b></li> <li>3. On previous terminal run the above <b>snmpwalk</b> command again.</li> <li>4. This time it will display the internalCurrentSession KPIs value to be 1.</li> <li>5. Repeat the process with more telnet sessions open on lbvip01 8080 port.</li> </ol>

KPI Values	
qns01/qns02/qns03/qns04...	<p><b>snmpwalk -v 2c -c broadhop -M +BROADHOP-MIB:CISCO-QNS-MIB &lt;qns01/02/03/04...&gt; &lt;OIDvalue&gt;</b></p> <p>For example, <b>snmpwalk -v 2c -c broadhop -M +BROADHOP-MIB:CISCO-QNS-MIB qns01.1.3.6.1.4.1.26878.200.3.3.70.15</b></p> <p>List all KPIs value of load balancer (lb), if all values are 0 then</p> <p>For ExternalCurrentSession:</p> <ol style="list-style-type: none"> <li>1. Open another terminal.</li> <li>2. Enter the following command: <b>telnet &lt;lbvip01&gt; 8443</b></li> <li>3. On previous terminal run the above <b>snmpwalk</b> command again.</li> <li>4. This time it will display the externalCurrentSession KPIs value to be 1.</li> <li>5. Repeat the process with more telnet sessions open on lbvip01 8080 port</li> </ol> <p>For InternalCurrentSession:</p> <ol style="list-style-type: none"> <li>1. Open another terminal.</li> <li>2. Enter the following command: <b>telnet &lt;lbvip02&gt; 8080</b></li> <li>3. On previous terminal run the above <b>snmpwalk</b> command again.</li> <li>4. This time it will display the internalCurrentSession KPIs value to be 1.</li> <li>5. Repeat the process with more telnet session open on lbvip01 8080 port.</li> </ol>

KPI Values	
qns01/qns02/qns03/qns04...	<pre>snmpwalk -v 2c -c broadhop -M +BROADHOP-MIB:CISCO-QNS-MIB &lt;qns01/02/03/04...&gt; &lt;OIDvalue&gt;</pre> <p>For example, <b>snmpwalk -v 2c -c broadhop -M +BROADHOP-MIB:CISCO-QNS-MIB qns01.1.3.6.1.4.1.26878.200.3.3.70.15</b></p> <p>List all KPIs value of policy server (QNS) VM.</p> <p>For example, the output will be displayed as below:</p> <pre>SNMPv2-SMI::enterprises.26878.200.3.3.70.15.20 = STRING: "11" SNMPv2-SMI::enterprises.26878.200.3.3.70.15.20.0 = STRING: "11" SNMPv2-SMI::enterprises.26878.200.3.3.70.15.21 = STRING: "0" SNMPv2-SMI::enterprises.26878.200.3.3.70.15.21.0 = STRING: "0" SNMPv2-SMI::enterprises.26878.200.3.3.70.15.22 = STRING: "0" SNMPv2-SMI::enterprises.26878.200.3.3.70.15.22.0 = STRING: "0" SNMPv2-SMI::enterprises.26878.200.3.3.70.15.23 = STRING: "0" SNMPv2-SMI::enterprises.26878.200.3.3.70.15.23.0 = STRING: "0" SNMPv2-SMI::enterprises.26878.200.3.3.70.15.24 = STRING: "0" SNMPv2-SMI::enterprises.26878.200.3.3.70.15.24.0 = STRING: "0" SNMPv2-SMI::enterprises.26878.200.3.3.70.15.25 = STRING: "3204764880" SNMPv2-SMI::enterprises.26878.200.3.3.70.15.25.0 = STRING: "3204764880"</pre>

## Troubleshooting Scenarios in OpenStack Environment

### Unable to Call API due to Puppet Time-out

**Issue:** In case of asynchronous API requests, though API returns 200 OK in the background API execution continues.

**Case:** In case when API execution fails (for example, puppet timeout error) then API state remains busy.

In this case further API requests cannot be made until API server state is changed. Many CPS orchestration APIs are accepted only when the CPS system is in a particular state.

**Solution:** To change the state of API, `/api/system` API can be used. For more information, refer to `/api/system` section in *CPS Installation Guide for OpenStack*.




---

**Note** Contact Cisco Technical Representative on the usage of this API.

---

## FAQs

- Q.** Where to check if traps are getting generated or not?
- A.** On active policy director (lb) VMs tail the below log file `/var/log/snmp/trap` to get the generated trap.
- Q.** Traps are getting generated from different VMs such as OAM (pcrfclient) or policy server (QNS) VMs but not getting logged to `/var/log/snmp/trap` and not appear on NMS receiver?
- A.** Check on active policy director (lb) VM if `/etc/snmp/scripts/application_trapv1_convert` and `component_trap_convert` files are present or not. If the files are present but traps are not getting generated then try to execute the following commands and test it again.
- ```
dos2unix /etc/snmp/scripts/application_trapv1_convert
dos2unix /etc/snmp/scripts/component_trap_convert
```
- Q.** The traps are getting logged in `/var/log/snmp/trap` but not receive on NMS?
- A.**
1. Check the setup configuration is correct or not as per the instruction given above.
  2. Perform the steps given in the previous question.
  3. Check if NMS IP is accessible from policy director (lb) VMs. Using command such as `ping <nms_ip>`.
- Q.** Database related traps not getting generated?
- A.**
1. Check the setup is configured and running as per instruction given above.
  2. On `pcrfclient/lb` VMs all the scripts generating the traps are logging the details inside `/var/log/broadhop/script/<script_name><date>.log` file. Open log file to check if there is any error in the script or is it generating the traps successfully or not. If not generated by script then contact system administrator team to resolve the issue.
- Q.** What is the difference between `pcrfclient01` and `pcrfclient02` virtual machines?
- A.**
- `pcrfclient01` --Master / Standby
  - `pcrfclient02` ---Slave / Standby
  - `pcrfclient02` support high availability of policy related services but it may not replicate all the services which were present in `pcrfclient01`.
- Q.** What is the ideal threshold limit for processor load in particular VM?
- A.** **A.** Ideally the threshold limit should be equal to number of vCPU that are present in the VM.
- You can check the vCPU on a particular VM using the following command: `grep ^processor /proc/cpuinfo | wc -l`.
- So if we have 12 vCPU, threshold limit for processor load is 12.
- Q.** I have multiple release trains (software releases) in my repository file (`cat /etc/broadhop/repositories`). Which one will take high precedence?
- A.** The highest version number is always selected and it is all merged. The versions are classified as follows and each type of versions will have version number and highest version takes high precedence:
1. Major

2. Minor
3. Patch
4. Build

## Reference Document

For more information on SNMP traps and KPIs, refer to *CPS SNMP, Alarms and Clearing Procedures Guide*.

