

# Troubleshoot PCRF Session Manager VM Recovery - Openstack

## Contents

[Introduction](#)

[Troubleshoot](#)

[Session Manager Instance Recovery Procedures](#)

[Power on Session Manager from SHUTOFF State](#)

[Recover any Instance from ERROR State](#)

[Session Manager/MongoDB Recovery](#)

[Member of Replica Set in Offline State](#)

[Member\(s\) of Replica Set Stuck in Startup2/Recovering State for a Long Time State](#)

[Rebuild Replica Sets](#)

[Restore the Database from Backup Post Replica-Set](#)

## Introduction

This document describes the Session Manager Recovery procedure deployed on Ultra-M/Openstack deployments.

## Troubleshoot

### Session Manager Instance Recovery Procedures

#### Power on Session Manager from SHUTOFF State

If any instance is in SHUTOFF state due to a planned shutdown or some other reason, please use this procedure to start the instance and enable its monitoring in ESC.

1. Check the state of instance via OpenStack

```
source /home/stack/destackovsrc-Pcrf
nova list --fields name,host,status | grep sm-s1
| c5e4ebd4-803d-45c1-bd96-fd6e459b7ed6 | SVS1-tmo_sm-s1_0_2e5dbff5-a324-42ea-9a65-
beb005a4226 | destackovs-compute-2 | SHUTOFF|
```

2. Check if the compute is available and ensure that the state is up.

```
source /home/stack/destackovsrc
nova hypervisor-show destackovs-compute-2 | egrep 'status|state'
| state | up |
| status | enabled |
```

3. Login to Elastic Services Controller (ESC) Master as admin user and check the state of instance in opdata.

```
/opt/cisco/esc/esc-confd/esc-cli/esc_nc_cli get esc_datamodel/opdata | grep sm-s1_0
SVS1-tmo_sm-s1_0_2e5dbff5-a324-42ea-9a65-bebf005a4226 VM_ERROR_STATE
```

#### 4. Power on the instance from openstack

```
source /home/stack/destackovsrc-Pcrf
nova start SVS1-tmo_sm-s1_0_2e5dbff5-a324-42ea-9a65-bebf005a4226
```

#### 5. Wait five minutes for instance to boot up and come to active state.

```
source /home/stack/destackovsrc-Pcrf
nova list -fields name,status | grep sm-s1_0
| c5e4ebd4-803d-45c1-bd96-fd6e459b7ed6 | SVS1-tmo_sm-s1_0_2e5dbff5-a324-42ea-9a65-
bebf005a4226 | ACTIVE |
```

#### 6. Enable VM Monitor in ESC after instance is in active state.

```
/opt/cisco/esc/esc-confd/esc-cli/esc_nc_cli vm-action ENABLE_MONITOR SVS1-tmo_sm-s1_0_2e5dbff5-
a324-42ea-9a65-bebf005a4226
```

For Further recovery of instance configurations, refer instance type specific procedures provided in the next section.

### Recover any Instance from ERROR State

This procedure can be used if state of CPS instance in openstack is ERROR:

#### 1. Check the state of instance in OpenStack.

```
source /home/stack/destackovsrc-Pcrf
nova list --fields name,host,status | grep sm-s1
| c5e4ebd4-803d-45c1-bd96-fd6e459b7ed6 | SVS1-tmo_sm-s1_0_2e5dbff5-a324-42ea-9a65-
bebf005a4226 | destackovs-compute-2 | ERROR|
```

#### 2. Check if the compute is available and runs fine.

```
source /home/stack/destackovsrc
nova hypervisor-show destackovs-compute-2 | egrep "state|state"
| state | up |
| status | enabled |
```

#### 3. Login to ESC Master as admin user and check the state of instance in opdata.

```
/opt/cisco/esc/esc-confd/esc-cli/esc_nc_cli get esc_datamodel/opdata | grep sm-s1_0
SVS1-tmo_sm-s1_0_2e5dbff5-a324-42ea-9a65-bebf005a4226 VM_ERROR_STATE
```

#### 4. Reset the state of instance to force the instance back to an active state instead of an error state, once done, reboot your instance.

```
source /home/stack/destackovsrc-Pcrf
```

```
nova reset-state --active SVS1-tmo_sm-s1_0_2e5dbff5-a324-42ea-9a65-bebf005a4226  
nova reboot --hard SVS1-tmo_sm-s1_0_2e5dbff5-a324-42ea-9a65-bebf005a4226
```

5. Wait five minutes for the instance to boot up and come to active state.

```
source /home/stack/destackovsrc-Pcrf  
nova list --fields name,status | grep sm  
| c5e4ebd4-803d-45c1-bd96-fd6e459b7ed6 | SVS1-tmo_sm-s1_0_2e5dbff5-a324-42ea-9a65-  
bebf005a4226 | ACTIVE |
```

6. If, Cluster Manager changes state to ACTIVE after reboot, Enable VM Monitor in ESC after Cluster Manager instance is in active state.

```
/opt/cisco/esc/esc-confd/esc-cli/esc_nc_cli vm-action ENABLE_MONITOR SVS1-tmo_sm-s1_0_2e5dbff5-  
a324-42ea-9a65-bebf005a4226
```

Post recovery to running/active state, refer instance type specific procedure to recover config/data from backup.

## Session Manager/MongoDB Recovery

Session Manager provides the Database layer to Cluster Policy Suite in this section, recovery of databases on a recently recovered instance of session manager is discussed:

### Member of Replica Set in Offline State

If member(s) of a replica set are in offline state, use this procedure:

1. Check the status of replica set using this command on Cluster Manager.

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

2. List down all OFF-LINE members in all replica sets.
3. Run the command on Cluster Manager.

```
cd /var/qps/bin/support/mongo  
build_set.sh --all --create-scripts
```

4. Secure shell to the sessionmgr VM(s) and start the mongo process.

```
ssh sessionmgrXX  
/etc/init.d/sessionmgr-XXXXX start
```

## Member(s) of Replica Set Stuck in Startup2/Recovering State for a Long Time State

If member(s) of a replica set are stuck in startup2 or recovering state and primary is available in replica set, use this procedure:

1. Check the status of replica set using this command on Cluster Manager.

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

2. List down all members in all replica sets.
3. Secure shell to the sessionmgr VM(s) and get the storage location of mongo process. As shown in the example dbpath is /var/data/sessions.1/b for mongo process running on sessionmgr01 at port 37717.

```
ssh sessionmgr01
ps -ef | grep mongo | grep 37717
root      2572      1 25 Feb11 ?          24-11:43:43 /usr/bin/mongod --ipv6 --nojournal --
storageEngine mmapv1 --noprealloc --smallfiles --port 37717 --dbpath=/var/data/sessions.1/b --
replSet set01b --fork --pidfilepath /var/run/sessionmgr-37717.pid --oplogSize 5120 --logpath
/var/log/mongod-37717.log --logappend --quiet --slowms 500
```

4. Stop the mongo process and cleanup the contents in dbpath:

```
/etc/init.d/sessionmgr-xxxxx stop
rm -rf /var/data/sessions.1/b/*
```

5. Start the mongo process, this causes the replica set member to sync all data from primary database and not the oplog.

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

Step 5 might take considerable time to sync in all the data from primary, depending upon the database size.

## Rebuild Replica Sets

Due to some outages, it might become necessary to rebuild some or all replica sets. However, before decision is taken to rebuild some or all replica sets, it might be noted that all data in these replica sets could be lost. Availability of backups must be cross verified for these databases:

- Admin (generally on 27721)
- Balance (generally on port 27718)
- SPR (generally on port 27720)

Once backups are cross-verified and decision for recreating database replica sets is taken, use this procedure:

1. Check contents of `/etc/broadhop/mongoConfig.cfg`, LLD must have information on what configuration must be present in this file or you can use a backed up file.
2. The command `build_set.sh --<db-name> --create` must run on Cluster Manager, which depends upon the database that you intend to rebuild. It creates all replica sets relating to that DB.

**Note:** The command to create all dbs in a replica set cleans up the database. All contents of the replica set would be lost.

3. If you wish to rebuild a specific replica set for one database, use this command:

```
build_set.sh --<db-name> --create --setname <set-name>
```

4. If you wish to rebuild all replica sets for all databases, use this command:

```
build_set.sh --all --create
```

## Restore the Database from Backup Post Replica-Set

Once all members of replica set are online and one of the members is primary, mongoDB can be restored from backup through this procedure.

1. In order to restore all DBs from backup use this command:

```
config_br.py --action import --mongo-all /mnt/backup/<file-name.tar.gz>
```

2. In order to restore a specific DB from backup through `config_br.py` these options are available:

- SPR

```
config_br.py --action import --mongo-all --spr /mnt/backup/<file-
```

- Admin

```
config_br.py --action import --mongo-all --admin /mnt/backup/<file-
```

- Balance

```
config_br.py --action import --mongo-all --balance /mnt/backup/<file-name.tar.gz>
```

- Reporting

```
config_br.py --action import --mongo-all --report /mnt/backup/<file-
```

If mongodump is used to backup databases, this explains it's usage through mongo restore:

1. Extract the backup **tar.gz** file.

```
tar -zxvf /mnt/backup/<file-name.tar.gz>
```

2. Locate the folder containing mongo dump of the database you want to recover and change directory to enter it.

```
ls -ltr /mnt/backup/cd /mnt/backup/27721_backup_$(date +%Y-%m-%d)/dump
```

3. Restore the replica set from backup.

```
mongorestore --host <primary-member> --port <db-port>
```

4. Optionally to restore a specific collection or a db use this command:

```
mongorestore --host <primary-member> --port <db-port> --db <db-name> --
```