Restore CPS

- Restore Cluster Manager VM in OpenStack, on page 1
- Restore Cluster Manager VM in VMware, on page 2
- Restore a CPS VM, on page 4
- Mongo Database Restore, on page 8
- Subversion Repository Restore, on page 11
- Validating the Restore , on page 11
- Restore Grafana Dashboard, on page 11

Restore Cluster Manager VM in OpenStack

Step 1 Copy the cluster manager VM snapshot to the controller blade as shown in the following command:

```
ls –ltr *snapshot*
```

Example output:

```
-rw-r--r--. 1 root root 10429595648 Aug 16 02:39 snapshot.raw
```

Step 2 Upload the snapshot image to OpenStack from Datastore:

```
glance image-create --name <snapshot_upload_image_name> --file <snapshot_file_path> --disk-format
qcow2 --container-format bare
```

Step 3 Verify whether the snapshot is uploaded with a Nova command as shown in the following example:

```
nova image-list
```

Figure 1: Example Output

```
+--------------------------------------+--------------------------------------+--------+--------------------------------------+
| ID                                   | Name                                 | Status | Server                               |
+--------------------------------------+--------------------------------------+--------+--------------------------------------+
| 146719e8-d8a0-4d5a-9b15-2a669cfab81f | CPS_10.9.9_20160803_100301_112.iso | ACTIVE |                                      |
| 1955d56e-4ecf-4269-b53d-b30e73ad57f0 | base_vm                              | ACTIVE |                                      |
| 2bbfb51c-cd05-4b7c-ad77-8362d76578db | cluman_snapshot                      | ACTIVE | 4842ae5a-83a3-48fd-915b-6ca6361adb2c |
| 5eebff44-658a-49a5-a170-1978f6276d18 | imported_image                       | ACTIVE |                                      |
+--------------------------------------+--------------------------------------+--------+--------------------------------------+
```

Step 4 Depending on whether the cluster manager VM exists or not, you can choose to create the cluman or rebuild the cluman:

- If the Cluster Manager VM instance does not exist, create the Cluman VM with an Heat or Nova command. An example is given below:

```
nova boot --config-drive true --image <imported_image_name>
--flavor "cluman" --nic net-id="6530560f-fc45-4ec0-86c1-810b5fee9a4e,
v4-fixed-ip=172.16.2.19" --nic net-id="99202a63-01ab-4594-b505-2e2ac818f12a,
v4-fixed-ip=10.81.69.181" --block-device-mapping
"/dev/vdb=f02b34cd-760d-4479-a60f-205e09f35f2e:::0" --availability-zone
"az-1:ch6-sr2-compute1.cisco.com" --security-groups cps_secgrp cluman
```

Note The block device mapping needs to be added with the Cinder ID of the previous ISO in which the Cluster Manager is planned to be restored. An example is given below:

```
cinder list | grep 10_0_CCO_ISO
| f02b34cd-760d-4479-a60f-205e09f35f2e | in-use
| 10_0_CCO_ISO | 3 | iscsi | true | fb19dd0f-d5f5-4dd0-b141-2ac976f361fa
|
```

- If the Cluster Manager VM instance exists, use a `nova rebuild` command to rebuild the Cluman VM instance with the uploaded snapshot as shown:

  ```
  nova rebuild <instance_name> <snapshot_image_name>
  ```

  For example:

  ```
  nova rebuild cps-cluman-5f3tujqvbi67 cluman_snapshot
  ```

Step 5 List all the instances as shown and verify that the new cluster manager instance is created and running:

```
nova list
```

Figure 2: Example Output

```
+--------------------------------------+--------+--------+------------+-------------+-----------------------------------------------+
| ID                                   | Name   | Status | Task State | Power State | Networks                                      |
+--------------------------------------+--------+--------+------------+-------------+-----------------------------------------------+
| ac3d2dbc-7b0e-4df4-a690-7f84ca3032bd | cluman | ACTIVE | -          | Running     | management=172.20.67.34; internal=172.20.70.34 |
```

Restore Cluster Manager VM in VMware

The following section describe how to restore the cluster manager VM using that OVF template.

Restore a Cluster Manager Using an OVF Template Backup

Note Before restoring the Cluster Manager, configure the ESXi server to have enough memory and CPU available. Confirm that the network port group is configured for the internal network.

1. Login to ESXi server using vSphere Web Client.
2. Right-click on the blade where you want to restore the Cluster Manager and select **Deploy OVF Template**. The Deploy OVF Template wizard opens.
3. Click **Browse...** to select all the files associated with an OVF template file. This includes files such as .ovf, .vmdk, and .iso. If you do not select all the required files, a warning message is displayed.
4. Click **Next**.

5. In the name and location window, do the following:
   a. Specify the name that the virtual machine will have when it is deployed at the target location.

      The name defaults to the selected template. If you change the default name, it must be unique within each vCenter Server virtual machine folder.

   b. Select or search for a datacenter or folder for the virtual machine.

      The default location is based on where you started the wizard. For example, if you started the wizard from a datastore, that datastore is preselected.

6. Click **Next**.
7. Search or browse for the host, cluster, or resource pool on which you want to deploy the OVF template and click **Next**.

Note If deploying the OVF template to the selected location might cause compatibility problems, the problems appear at the bottom of the window.

8. Review the OVF template details and click **Next**.

Note If some details are not as per your requirements, click **Back** and repeat the steps.

9. Select the virtual disk format to store the files for the deployed template and click **Next**.

Table 1: Disk Formats

Format	Description
Thick Provisioned Lazy Zeroed	Creates a virtual disk in a default thick format. Space required for the virtual disk is allocated when the virtual disk is created. Data remaining on the physical device is not erased during creation, but is zeroed out on demand at a later time on first write from the virtual machine.
Thick Provision Eager Zeroed	A type of thick virtual disk that supports clustering features such as Fault tolerance. Space required for the virtual disk is allocated at creation time. In contrast to the flat format the data remaining on the physical device is zeroed out when the virtual disk is created. it might take much longer to create disks in this format than to create other types o disks.
Thin Provision	Use this format to save storage space. For the thin disk, you provision as much datastore space as the disk would require based on the value that you enter for the disk size. However, the thin disk starts small and at first, uses only as much datastore space as the disk needs for its initial operations.

10. Select the network (map the networks used in OVF template to the network in your inventory) and click **Next**.
11. Verify the settings from **Ready to Complete** window and click **Finish**.

12. After the OVF template is successfully deployed, power ON the VM. The Cluster Manager VM is displayed successfully.

Restore a CPS VM

The Cluster Manager VM is the cluster deployment host that maintains all the necessary CPS software and deployment configurations. If a VM in the CPS cluster becomes corrupted, the VM can be recreated. For more information, see the *CPS Installation Guide for OpenStack*.

Note Because of its role in the cluster, the Cluster Manager cannot be redeployed using these steps. To restore the Cluster Manager VM, refer to one of the previous two sections.

Restore a Single VM in the Cluster

The following sections describe how to restore/redeploy specific VM in the CPS cluster (other than the Cluster Manager).

pcrfclient01 VM

To redeploy the pcrfclient01 VM:

Step 1 Log in to the Cluster Manager VM as the root user.

Step 2 Note the UUID of SVN repository using the following command:

```
svn info http://pcrfclient02/repos | grep UUID
```

The command outputs the UUID of the repository. For example:

```
Repository UUID: ea50bbd2-5726-46b8-b807-10f4a7424f0e
```

Step 3 Import the backup Policy Builder configuration data on the Cluster Manager, as shown in the following example:

```
config_br.py -a import --etc-oam --svn --stats --grafanadb --auth-htpasswd --users
/mnt/backup/oam_backup_27102016.tar.gz
```

Note Many deployments run a cron job that backs up configuration data regularly. See Subversion Repository Backup, for more details.

Step 4 To generate the VM archive files on the Cluster Manager using the latest configurations, execute the following command:

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

Step 5 To deploy the pcrfclient01 VM, perform one of the following:

- In VMware, execute the following command: `/var/qps/install/current/scripts/deployer/deploy.sh pcrfclient01`
- In OpenStack, use the HEAT template or the Nova command to re-create the VM. For more information, see *CPS Installation Guide for OpenStack*.

Step 6 Re-establish SVN master/slave synchronization between the pcrfclient01 and pcrfclient02 with pcrfclient01 as the master by executing the following series of commands.

Note If SVN is already synchronized, do not issue these commands. To check if SVN is in sync, run the following command from pcrfclient02. If a value is returned, then SVN is already in sync:

```
/usr/bin/svn propget svn:sync-from-url --revprop -r0 http://pcrfclient01/repos
```

Execute the following commands from pcrfclient01:

```
/bin/rm -fr /var/www/svn/repos

/usr/bin/svnadmin create /var/www/svn/repos

/usr/bin/svn propset --revprop -r0 svn:sync-last-merged-rev 0
http://pcrfclient02/repos-proxy-sync

/usr/bin/svnadmin setuuid /var/www/svn/repos/ "Enter the UUID captured in step 2"

/etc/init.d/vm-init-client

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

Step 7 If pcrfclient01 is also the arbiter VM, then execute the following steps:

a) Create the mongodb start/stop scripts based on the system configuration.

Note Not all deployments have all these databases configured. Refer to `/etc/broadhop/mongoConfig.cfg` to determine which databases need to be set up.

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

build_set.sh --session --create-scripts

build_set.sh --admin --create-scripts

build_set.sh --spr --create-scripts

build_set.sh --balance --create-scripts

build_set.sh --audit --create-scripts

build_set.sh --report --create-scripts
```

b) Start the mongo process:

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

c) Wait for the arbiter to start, then run `diagnostics.sh --get_replica_status` to check the health of the replica set.

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.

pcrfclient02 VM

To redeploy the pcrfclient02 VM:

Step 1 Log in to the Cluster Manager VM as the root user.

Step 2 To generate the VM archive files on the Cluster Manager using the latest configurations, execute the following command:

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

Step 3 To deploy the pcrfclient02 VM, perform one of the following:

- In VMware, execute the following command: `/var/qps/install/current/scripts/deployer/deploy.sh pcrfclient02`
- In OpenStack, use the HEAT template or the Nova command to re-create the VM. For more information, see *CPS Installation Guide for OpenStack*.

Step 4 Secure shell to the pcrfclient01:

```
ssh pcrfclient01
```

Step 5 Run the following script to recover the SVN repos from pcrfclient01:

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

sessionmgr VMs

To redeploy a sessionmgr VM:

Step 1 Log in to the Cluster Manager VM as the root user.

Step 2 To deploy the sessionmgr VM and replace the failed or corrupt VM, perform one of the following:

- In VMware, execute the following command: `/var/qps/install/current/scripts/deployer/deploy.sh sessionmgrXX`
- In OpenStack, use the HEAT template or the Nova command to re-create the VM. For more information, see *CPS Installation Guide for OpenStack*.

Step 3 Create the mongodb start/stop scripts based on the system configuration.

Note Not all deployments have all these databases configured. Refer to `/etc/broadhop/mongoConfig.cfg` to determine which databases need to be set up.

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

build_set.sh --session --create-scripts

build_set.sh --admin --create-scripts

build_set.sh --spr --create-scripts

build_set.sh --balance --create-scripts

build_set.sh --audit --create-scripts

build_set.sh --report --create-scripts
```

Step 4 Secure shell to the sessionmgr VM and start the mongo process:

```
ssh sessionmgrXX
```

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

Step 5 Wait for the members to start and for the secondary members to synchronize, then run `diagnostics.sh --get_replica_status` to check the health of the database.

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 6 To restore Session Manager database, use one of the following example commands depending on whether the backup was performed with `--mongo-all` or `--mongo` option:

- `config_br.py -a import --mongo-all --users /mnt/backup/sm_backup_27102016.tar.gz`
- `config_br.py -a import --mongo --users /mnt/backup/sm_backup_27102016.tar.gz`

Policy Director (Load Balancer) VM

To redeploy the Policy Director (Load Balancer) VM:

Step 1 Log in to the Cluster Manager VM as the root user.

Step 2 To import the backup Policy Builder configuration data on the Cluster Manager, execute the following command:

```
config_br.py -a import --network --haproxy --users /mnt/backup/lb_backup_27102016.tar.gz
```

Step 3 To generate the VM archive files on the Cluster Manager using the latest configurations, execute the following command:

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

Step 4 To deploy the lb01 VM, perform one of the following:

- In VMware, execute the following command: `/var/qps/install/current/scripts/deployer/deploy.sh lb01`
- In OpenStack, use the HEAT template or the Nova command to re-create the VM. For more information, see *CPS Installation Guide for OpenStack*.

QNS VM

To redeploy the Policy Server (QNS) VM:

Step 1 Log in to the Cluster Manager VM as the root user.

Step 2 Import the backup Policy Builder configuration data on the Cluster Manager, as shown in the following example:

```
config_br.py -a import --users /mnt/backup/qns_backup_27102016.tar.gz
```

Step 3 To generate the VM archive files on the Cluster Manager using the latest configurations, execute the following command:

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

Step 4 To deploy the qns VM, perform one of the following:

- In VMware, execute the following command: `/var/qps/install/current/scripts/deployer/deploy.sh qns`
- In OpenStack, use the HEAT template or the Nova command to re-create the VM. For more information, see *CPS Installation Guide for OpenStack*.

Mongo Database Restore

To restore databases in a production environment that use replica sets with or without sharding, a maintenance window is required as the CPS software on all the processing nodes and the sessionmgr nodes must be stopped. A database restore is needed after an outage or problem with the system and/or its hardware. In that case, service has been impacted and to properly fix the situation, service will need to be impacted again. From a database perspective, the main processing nodes must be stopped so that the system is not processing incoming requests while the databases are stopped and restored. If replica sets are used with or without sharding, then all the database instances must be stopped to properly restore the data and have the replica set synchronize from the primary to the secondary database nodes.

For reference, the following Mongo DB documentation was used to develop the CPS restore procedures:

- http://docs.mongodb.org/manual/tutorial/restore-sharded-cluster/#restore-sh-cl-dmp
- http://docs.mongodb.org/manual/tutorial/restore-replica-set-from-backup/
- http://docs.mongodb.org/manual/tutorial/resync-replica-set-member/

Determine Health of Database After Outage

The following SNMP Notifications (Alarms) are indicators of issues with the CPS databases.

- **All DB Member of a replica set Down**- CPS is unable to connect to any member of the replica set.
- **No Primary DB Member Found**- CPS is unable to find the primary member for a replica set.
- **Secondary DB Member Down**- In a replica set, a secondary DB member is not able to connect.

To determine the status of the databases, run the following command:

```
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.

If the mongod process is stopped on any VM, try to manually start it using the following command, where (XXXXX is the DB port number):

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

If the mongod process does not start (stops immediately), or reports errors (either command line or in monogdb log file), refer to the following sections for more information:

- Repair a Damaged Database
- Rebuild a Damaged Database

General Procedure for Database Restore

The following steps describe how to import data from a previous backup (as described in General Procedure for Database Restore).

If the database is damaged, refer to Repair a Damaged Database, or Rebuild a Damaged Database, before proceeding with these database restoration steps.

Step 1 Execute the following command to restore the database:

```
config_br.py -a import --mongo-all /mnt/backup/backup_$date.tar.gz
```

where $date is the timestamp when the export was made.

For example,

```
config_br.py -a import --mongo-all /mnt/backup/backup_27092016.tgz
```

Step 2 Log in to the database and verify whether it is running and is accessible:

a. Log into session manager:

```
mongo --host sessionmgr01 --port $port
```

where $port is the port number of the database to check. For example, 27718 is the default Balance port.

b. Display the database by executing the following command:

```
show dbs
```

c. Switch the mongo shell to the database by executing the following command:

```
use $db
```

where $db is a database name displayed in the previous command. The 'use' command switches the mongo shell to that database.

For example,

```
use balance_mgmt
```

d. To display the collections, execute the following command:

```
show collections
```

e. To display the number of records in the collection, execute the following command:

```
db.$collection.count()
```

For example,

```
db.account.count()
```

The above example will show the number of records in the collection “account” in the Balance database (balance_mgmt).

Repair a Damaged Database

After an outage, the database may be in a state where the data is present but damaged. When you try to start the database process (mongod), it will start, and then stop immediately. You can also observe a “repair required” message in the /var/log/mongodb log file.

If this occurs, you can attempt to repair the database using the following commands:

Note Because the session database (session_cache - 27717) stores only transient session data of active network sessions, you should not try to repair this database. If the session database is damaged, refer to Rebuild a Damaged Database, on page 10 to rebuild it.

Run the following commands:

```
/usr/bin/systemctl stop sessionmgr-$port
```

```
/usr/bin/systemctl repair sessionmgr-$port
```

Verify that the mongod process is running on the VM:

```
ps –ef | grep mongo | grep $port
```

If it is not running, then start the mongod process:

```
/usr/bin/systemctl start sessionmgr-$port
```

After repairing the database, you can proceed to import the most recent data using `config_br.py` as described in General Procedure for Database Restore.

Rebuild a Damaged Database

If the existing data in the database is damaged and cannot be repaired/recovered (using the steps in Repair a Damaged Database), the database must be rebuilt.

Step 1 Secure shell to the pcrfclient01 VM as the root user:

```
ssh pcrfclient01
```

Step 2 To rebuild the failed database:

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

Step 3 To rebuild a specific replica-set:

```
build_set.sh --$db_name --create
```

where:

`$db_name`: Database name

Step 4 After repairing the database, you can proceed to import the most recent data using config_br.py as described in General Procedure for Database Restore.

Subversion Repository Restore

To restore the Policy Builder Configuration Data from a backup, execute the following command:

```
config_br.py -a import --svn /mnt/backup/backup_$date.tgz
```

where, $date is the date when the cron created the backup file.

Validating the Restore

After restoring the data, verify the working system by executing the following command:

```
/var/qps/bin/diag/diagnostics.sh
```

Restore Grafana Dashboard

You can restore Grafana dashboard using the following command:

```
config_br.py -a import --grafanadb /mnt/backup/<backup_filename>
```