CPS Backup and Restore Guide, Release 24.1.0

First Published: 2024-03-14

Americas Headquarters

CONTENTS

Preface

- About This Guide, on page v
- Audience, on page v
- Additional Support, on page vi
- Conventions (all documentation), on page vi
- Communications, Services, and Additional Information, on page vii
- Important Notes, on page viii

About This Guide

Note The documentation set for this product strives to use bias-free language. For purposes of this documentation set, bias-free is defined as language that does not imply discrimination based on age, disability, gender, racial identity, ethnic identity, sexual orientation, socioeconomic status, and intersectionality. While any existing biased terms are being substituted, exceptions may be present in the documentation due to language that is hardcoded in the user interfaces of the product software, language used based on RFP documentation, or language that is used by a referenced third-party product.

This document is a part of the Cisco Policy Suite documentation set.

For information about available documentation, see the *CPS Documentation Map* for this release at Cisco.com.

Note The PATS/ATS, ANDSF, and MOG products have reached end of life and are not supported in this release. Any references to these products (specific or implied), their components or functions in this document are coincidental and are not supported. Full details on the end of life for these products are available at: https://www.cisco.com/c/en/us/products/wireless/policy-suite-mobile/eos-eol-notice-listing.html.

Audience

This guide is best used by these readers:

- Network administrators

- Network engineers

- Network operators

- System administrators

This document assumes a general understanding of network architecture, configuration, and operations.

Additional Support

For further documentation and support:

- Contact your Cisco Systems, Inc. technical representative.

- Call the Cisco Systems, Inc. technical support number.

- Write to Cisco Systems, Inc. at support@cisco.com.

- Refer to support matrix at https://www.cisco.com/c/en/us/support/index.html and to other documents related to Cisco Policy Suite.

Conventions (all documentation)

This document uses the following conventions.

Conventions	Indication
bold font	Commands and keywords and user-entered text appear in **bold** font.
italic font	Document titles, new or emphasized terms, and arguments for which you supply values are in *italic* font.
[ ]	Elements in square brackets are optional.
{x	y	z }	Required alternative keywords are grouped in braces and separated by vertical bars.
[x	y	z]	Optional alternative keywords are grouped in brackets and separated by vertical bars.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.
courier font	Terminal sessions and information the system displays appear in courier font.
< >	Nonprinting characters such as passwords are in angle brackets.

Conventions	Indication
[ ]	Default responses to system prompts are in square brackets.
!, #	An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.

Note Means reader take note. Notes contain helpful suggestions or references to material not covered in the manual.

Caution Means reader be careful. In this situation, you might perform an action that could result in equipment damage or loss of data.

Warning IMPORTANT SAFETY INSTRUCTIONS.

Means danger. You are in a situation that could cause bodily injury. Before you work on any equipment, be aware of the hazards involved with electrical circuitry and be familiar with standard practices for preventing accidents. Use the statement number provided at the end of each warning to locate its translation in the translated safety warnings that accompanied this device.

SAVE THESE INSTRUCTIONS

Note Regulatory: Provided for additional information and to comply with regulatory and customer requirements.

Communications, Services, and Additional Information

- To receive timely, relevant information from Cisco, sign up at Cisco Profile Manager.

- To get the business impact you're looking for with the technologies that matter, visit Cisco Services.

- To submit a service request, visit Cisco Support.

- To discover and browse secure, validated enterprise-class apps, products, solutions and services, visit Cisco Marketplace.

- To obtain general networking, training, and certification titles, visit Cisco Press.

- To find warranty information for a specific product or product family, access Cisco Warranty Finder.

Cisco Bug Search Tool

Cisco Bug Search Tool (BST) is a web-based tool that acts as a gateway to the Cisco bug tracking system that maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. BST provides you with detailed defect information about your products and software.

Important Notes

☞

Important   Any feature or GUI functionality that is not documented may not be supported in this release or may be customer specific, and must not be used without consulting your Cisco Account representative.

Back up CPS

Overview

Various items in a Cisco Policy Suite (CPS) cluster require periodic backups.

This document describes the procedure to use the config_br.py script to back up the configuration and the data from a CPS cluster, and the steps to restore the backups, in case of any failures.

For general information about the config_br.py script and all the options, see the About the Backup and Restore Script, on page 21.

Before You Begin

Before you begin any backup and restore procedures, ensure you have performed the following tasks:

- Install CPS and have it running successfully. Backups are stored on customer-provided hardware, preferably in a location apart from where CPS is currently running.

- Initiate backups from the cluster manager VM. Ensure that there is adequate storage space on the cluster manager VM before taking a backup.

- pcrfclient01/OAM01 VM is up and running for SVN repository backup.

- lbvip01 VIP is available on Policy Director (LB) VMs.

Note Cisco recommends that the backup of the cluster manager be performed by taking a snapshot of the cluster manager VM. This operation requires administrator access to OpenStack or VMware (depending on the environment that CPS is deployed in).

Backup Schedule

Your first backup operation should occur after a successful installation and configuration. This provides a baseline and tests your backup procedures with respect to hardware, software, and protocols.

Then, do backup on this schedule as a best practice.

Table 1: Backup Schedule

Backup this...	...this often
Cluster Manager VM	Monthly and after any configuration changes, patch updates, or upgrades
Databases	Daily
Policy Builder Configurations	Weekly or after any changes

Back Up the Cluster Manager VM

The backup and restore procedures for the Cluster Manager do not require a maintenance window. The CPS cluster can continue to operate successfully without an operational Cluster Manager. Any CPS administrative scripts which use the Cluster Manager would not be usable while the Cluster Manager is offline.

The following sections describe two options for backing up the Cluster Manager VM:

- Back Up the Cluster Manager VM in OpenStack
- Back Up the Cluster Manager VM in VMware

It is not recommended to perform backups of the other CPS VMs. Instead, these VMs can be redeployed at any time. For more information, refer to Restore a CPS VM.

Back Up the Cluster Manager VM in OpenStack

To back up the cluster manager VM in OpenStack, you must first create a snapshot of the VM.

Back up Cluster Manager VM

Step 1 Use the following command to view the nova instances and note the name of the cluster manager VM instance:

```
nova list
```

Step 2 Create a nova snapshot image as shown in the following command:

```
nova image-create --poll <cluman_instance_name> <cluman_snapshot_name>
```

Note Ensure that you have enough disk space for the snapshot.

Important In case if VM becomes unreachable after snapshot creation, check status of VM using `nova list` command. If it is in "SHUTOFF" state, you need to start the VM manually.

Step 3 View the image list with the following command:

```
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

+------------------------------------+--------------------------------+-------+------------------------------------+
```

Step 4 When a snapshot is created, the snapshot image is stored in OpenStack Glance. To store the snapshot in a remote data store, download the snapshot and transfer the file. To download the image, use the following command in OpenStack:

```
glance image-download --file <snapshot_name_on_filesystem> <snapshot_id>
```

For example:

```
glance image-download --file snapshot.raw 2bbfb51c-cd05-4b7c-ad77-8362d76578db
```

Step 5 List the downloaded images 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 6 Store the snapshot of the Cluster Manager VM to restore in the future.

Back Up the Cluster Manager VM in VMware

The following section describe how to back up the entire Cluster Manager VM to a VMware OVF template. Backing up a Cluster Manager VM backs up all configurations and software applications.

Back up Cluster Manager Using OVF Template

To take the backup of the Cluster Manager, perform the following steps:

1. Shutdown the Cluster Manager VM using either of the following methods:

    From the Cluster Manager VM: Log in to the Cluster Manager VM and run the following command to shutdown the VM.

    ```
    shutdown -h now
    ```

    From the vSphere Web Client:

    a. Log in to the vSphere server that hosts the Cluster Manager using vSphere Web Client.

    b. Right-click the Cluster Manager VM and select **Power** > **Power Off**.

    A Confirm Power Off message appears. Click **Yes** to confirm the Power Off.

    c. Verify the Cluster VM is powered off from the vSphere Web Client UI.

2. Export the OVF template of the Cluster Manager VM.

a. Select the Cluster Manager from the VM list on the left column.

b. Select **Edit Settings...**.

c. Uncheck **Connected** near CD/DVD drive where you have linked the data store ISO file.

d. Right-click the Cluster Manager and select **Template** > **Export OVF Template**.

   The **Export OVF Template** dialog opens.

e. In *Name* field, type the name of the template.

f. (Optional) In the *Annotation* field, type a description.

g. Uncheck "Include image files attached to floppy and CD/DVD devices in the OVF package". By default, it is checked.

h. Select the **Enable advanced options** checkbox if you want to include additional information or configurations in the exported template. The advanced settings include information about the BIOS UUID, MAC addresses, boot order, PCI Slot numbers, and configuration settings used by other applications.

i. After selecting the required parameters, click **OK**. The backup starts.

j. After the export succeeds, you are prompted to save each file associated with the template (.ovf, .vmdk, .iso). Save the files to the desired location.

Back up CPS VMs

To back up CPS VMs, see the following table for a description of the options that you use with the `config_br.py` script:

Table 2: `config_br.py` Script Options

VM	Options to include with script
Policy Director (LB)	`--network --haproxy --users`
OAM (pcrfclient)	`--etc-oam --svn --stats --grafanadb --auth-htpasswd --users`
QNS	`--users`
Session Manager	`--mongo --mongo-all --users`
All VMs	`--all`

Note For more information about each of the command options, see Script Options, on page 21.

The following examples illustrate how to use the command from Cluster Manager and options for various VMs:

- Session Manager VM: `config_br.py -a export --mongo-all --users /mnt/backup/sessionmgr_backup_27102016.tar.gz`

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

- Policy Director (LB) VM: `config_br.py -a export --network --haproxy --users /mnt/backup/lb_backup_27102016.tar.gz`

- QNS VM: `config_br.py -a export --users /mnt/backup/qns_backup_27102016.tar.gz`

- All VMs: `config_br.py -a export --all /mnt/backup/cps_backup_27102016.tar.gz`

To restore data from CPS VMs, see .

Mongo Database

In a production environment, databases need to use replication to help guarantee data integrity. Mongo DB calls its replication configuration replica sets as opposed to Master/Slave terminology used for Relational Database Management System (RDBMS).

Replica sets create a group of database nodes that work together to provide the data backup. There is a primary (the master) and 1..n secondaries (the slaves). Additionally, each replica set requires another node called the Arbiter. The Arbiter is used as a non-data-processing node that helps decide which node becomes the primary in the case of failure. For example, if there are four nodes: primary, secondary1, secondary2 and the arbiter, and if the primary fails, the remaining nodes "vote" for which of the secondary nodes becomes the primary. Since there are only two secondaries, there would be a tie and failover would not occur. The arbiter solves that problem and "votes" for one node breaking the tie.

Mongo DB has another concept called Sharding that helps redundancy and speed for a cluster. Shards separate the database into indexed sets which allow for much greater speed for writes which improves overall database performance. Sharded databases are often setup so that each shard is a replica set. Replica Sets and Sharding both require some special handling for backup. Mongo DB recommends that for each replica set being backed up, one secondary is shut down and that node is used for the backup. After backup, that node is brought back up and integrated back into the replica set.

Mongo Database Backup

✎

Note   RADIUS-based policy control is no longer supported in CPS 14.0.0 and later releases as 3GPP Gx Diameter interface has become the industry-standard policy control interface.

CPS uses MongoDB for primary system databases. These include:

- Admin

- Audit

- Balance

- Custom Reference Data

- Policy Reporting

- Portal

- Radius

- Sharding

- SPR

- Vouchers

The session database (session_cache) represents the transient session data of the active network sessions of subscribers on the network. Due to the transient nature of this data, it does not make sense to backup or restore this database. The script does not provide an option to back up the Session Cache database (configured by default on port 27717).

Full Environment:

The following table lists the Module Name, the database name, and the default ports when using Replica Sets via the `/etc/broadhop/mongoConfig.cfg` file.

Table 3: Using Replica-set

Module Name	Database Name	Default Ports
Core	admin	27721
Audit	audit	27725
Balance	balance_mgmt	27718
Custom Reference Data	cust_ref_data	27717
Policy Intel	policy_trace	27719
Portal	portal	27749
Radius	radius	27717
Core	sharding	27717
SPR	spr	27720
Voucher	vouchers	27717

General Procedure for Database Backup

Use the following command to generate a backup of the CPS databases.

```
config_br.py –a export --mongo-all /mnt/backup/backup_28092016.tar.gz
```

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

http://docs.mongodb.org/manual/tutorial/backup-sharded-cluster-with-database-dumps/

Automatic Backup via Cron

Using a cron job, it is possible to automate backups. It is best to schedule automated backups when least amount of traffic is running through the CPS system.

> **✎**
>
> | **Note** | Do not store repository backups on any CPS node. Move them immediately to an external storage like a Storage Area Network (SAN). |

The following example procedure creates a backup of the policy configuration subversion repository every night at 10:00pm:

1. Login to the Cluster Manager as the root user.

2. To edit the root user's cron tab, execute the command:

   ```
   crontab -e
   ```

3. Add the following line:

> **✎**
>
> | **Note** | The crontab editor is VI. |

   ```
   00 22 * * * /var/qps/install/current/scripts/modules/config_br.py -a export --svn
   /mnt/backup/backup_$(date +\%Y-\%m-\%d).tar.gz
   ```

> **✎**
>
> | **Note** | Manually enter the crontab entries in your system. |

Save the file and the new cron tab is installed.

Policy Builder Configuration Data

The Policy Builder uses a Subversion (SVN) repository to store the policy configurations. The following sections outline the backup and restore procedures for the Subversion repository.

Subversion Repository Backup

The Subversion repository is setup like a master/slave with the master repository in pcrfclient01 and the slave repository in pcrfclient02. All commits go to the master and are replicated to the slave using the Subversion hooks process. Hooks are scripts that get executed by the SVN binary automatically. Typically in deployments, policy configuration does not change very often once the system is live, so automated weekly backups of the repository are usually sufficient.

Automatic Backup via Cron

Using a cron job, it is possible to automate backups. It is best to schedule automated backups when least amount of traffic is running through the CPS system.

> **Note** Do not store repository backups on any CPS node. Move them immediately to an external storage like a Storage Area Network (SAN).

The following example procedure creates a backup of the policy configuration subversion repository every night at 10:00pm:

1. Login to the Cluster Manager as the root user.

2. To edit the root user's cron tab, execute the command:

   ```
   crontab -e
   ```

3. Add the following line:

> **Note** The crontab editor is VI.

   ```
   00 22 * * * /var/qps/install/current/scripts/modules/config_br.py -a export --svn
   /mnt/backup/backup_$(date +\%Y-\%m-\%d).tar.gz
   ```

> **Note** Manually enter the crontab entries in your system.

Save the file and the new cron tab is installed.

Validating the Backup

After you make a backup of any database, you can check these things to make sure the backup is valid:

- Observe and correct any errors or warnings during the backup. For example, the backup may be aborted if there is not enough file space available or if the media is corrupt.

- Make sure that the file size of the backup is the same as the original, and that it is not zero.

Open the backup database with an appropriate third-party tool.

With these instructions, your backup routines should be adequate and timely. If in doubt, try to restore backups to a test environment and gauge your success. Please contact your Cisco technical representative at any time with questions or concerns.

Back up Grafana Dashboard

You can back up Grafana dashboard using following command:

```
config_br.py -a export --grafanadb /mnt/backup/backup_$(date +%Y-%m-%d).tar.gz
```

CHAPTER **2**

Restore CPS

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 2: 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 3: 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 4: 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 18 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>
```

Restore Grafana Dashboard

Backup and Restore Script

About the Backup and Restore Script

The backup and restore script is a Python script that is used to take the backup of the requested configuration item, available locally on Cluster Manager VM or on other VMs. When a restore is required, the configuration supplied is copied to the requested location within Cluster Manager VM or the specific VM.

Name : config_br.py

Path : `/var/qps/install/current/scripts/modules`

VM : Cluster Manager

When you run this script, you provide options and specify the location for the backup file. For a description of the script options, see Script Options, on page 21.

Usage Examples

- `config_br.py -a export --etc --etc-oam --svn --stats /mnt/backup/backup_27092016.tar.gz`

  Back up /etc/broadhop configuration data from OAM (pcrfclient) VM, Policy Builder configuration, and logstash

- `config_br.py -a import --etc --etc-oam --svn --stats /mnt/backup/backup_27092016.tar.gz`

  Restore data from /etc/broadhop configuration from OAM (pcrfclient) VM, Policy Builder configuration, and logstash

Script Options

The following table describes the options you can use with the backup and restore script:

Option	Description
Positional Arguments	

Option	Description
`tar`	Location of the tarball of the configuration data. The backup and restore script (config_br.py) creates a tarball in tar.gz format and the same tar.gz type is used to restore configuration data.
Optional Arguments	
`-h` or `--help`	Displays help message.
`--action ACTION` or `-a ACTION` where `ACTION` is either `export` or `import`	Specifies whether you want to export (backup) or import (restore) VM configuration data.
`--quiet` or `-q`	Specifies whether you want to be notified if the tarball with same name exists. The same tarball name can be used for multiple backup commands. If tarball with the configuration data already exists, the tar is overwritten with a warning. You can choose to avoid the display of the warning with the `--quiet` or `-q` option. The script will automatically overwrite the existing tar. Not applicable for restore operations.
`--all`	Specifies backup or restore of all VMs. This option is equivalent to `--auth-htpasswd --etc --etc-oam --grafanadb --haproxy --mongo-all --network --svn --users --stats`
`--auth-htpasswd`	Specifies the backup or restore of .htpasswd file (`/var/broadhop/.htpasswd`).
`--etc`	Specifies the backup or restore of `/etc/broadhop` cluster configuration.
`--etc-oam`	Specifies the backup or restore of `/etc/broadhop` configuration from OAM (pcrfclient) VM.
`--grafanadb`	Specifies the backup or restore of the Grafana databases (data such as dashboard and user information).
`--haproxy`	Specifies the backup or restore of HAproxy configurations on Policy Director (Load Balancer) VM.
`--mongo`	Specifies the backup and restore of the default database and the data about Admin, Balance, Cluster, Customer Reference Data, Diameter Peers and Diameter Queuing, Sharding, SPR.
`--mongo-all`	Specifies the backup or restore of default database and the Audit, Policy Reporting, Portal, Radius, Voucher databases. **Note** RADIUS-based policy control is no longer supported in CPS 14.0.0 and later releases as 3GPP Gx Diameter interface has become the industry-standard policy control interface.
`--network`	Specifies the backup or restore of network configuration `/etc/hosts`, `/etc/sysconfig, /network-scripts/route-*`

Option	Description
`--svn`	Specifies the backup or restore of Policy Builder configuration data.
`--users`	Specifies the backup or restore of user and password information on the deployed VMs. Required for backup of all VMs.
`--stats`	Specifies the backup or restore of Logstash (`/var/broadhop/stats`, `/var/lib/carbon`)