Cisco Policy Suite Backup and Restore Guide

Release 7.5.1

First Published: November 2, 2015
Last Updated: November 2, 2015

Preface

Welcome to Cisco Policy Suite Backup and Restore Guide.

This document describes common methods used to backup and restore various databases and configurations.

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 http://www.support.cisco.com and to other documents related to Cisco Policy Suite.

Terms and Definitions

This document uses certain terms and definitions specific to the CPS software application. For common Glossary of Terms, refer to http://wwwin.cisco.com/tech/EngCoE/cpdm/glossary.shtml.

Version Control Software

Cisco Policy Builder uses version control software to manage its various data repositories. The default installed version control software is Subversion, which is provided in your installation package.

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.
[ ]	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

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

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, using the Cisco Bug Search Tool (BST), submitting a service request, and gathering additional information, see *What's New in Cisco Product Documentation* at: http://www.cisco.com/c/en/us/td/docs/general/whatsnew/whatsnew.html.

Subscribe to *What's New in Cisco Product Documentation*, which lists all new and revised Cisco technical documentation as an RSS feed and delivers content directly to your desktop using a reader application. The RSS feeds are a free service.

Backup and Restore

First Published: November 2, 2015

Last Updated: November 2, 2015

This chapter covers the following sections:

Overview

There are various items from an CPS system that should be backed up. This document focuses on the following three items for backup and restore:

- Cluster Manager
- Databases
- Policy Builder Configuration

Before You Begin

- 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 the backups using either manual or automated methods.

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 backups on this schedule as a best practice.

Table 1 Backup Schedule

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

Back Up and Restore the Cluster Manager VM

The backup and restore procedures for the Cluster Manager described in this section 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 and restoring the Cluster Manager VM.

- Configuration Back Up and Restore, page 6
- VM Image Back Up and Restore Using VMware OVF Template, page 7

It is not recommended to perform backups of the other CPS VMs. Instead, these VMs can be redeployed from the Cluster Manager at any time. Refer to Restore a CPS VM, page 9 for more information.

Configuration Back Up and Restore

The following sections describe how to back up the configurations of the Cluster Manager to a remote server, and then restore those configurations after redeploying the Cluster Manager VM.

These steps do not make changes to the other VMs in the CPS cluster.

Back Up the Cluster Manager VM Configuration

1. Log into the new Cluster Manager VM and execute the following commands to create a back up file (.tar.gz) of the Cluster Manager configuration:

   ```
   /bin/mkdir -p /var/clustermgr/backup; cd /var/clustermgr/backup

   /bin/tar -P -czf files-$(date +"%Y-%m-%d-%H-%M-%S").tar.gz
   /var/qps/config/deploy/csv/ /var/qps/current_config/etc/broadhop/
   ```

   The commands above create a .tar.gz file in /var/clustermgr/backup/

   For example: `files-2015-09-02-22-56-01.tar.gz`

2. Export the .tar.gz to a remote server, for example:

   ```
   /usr/bin/scp /var/clustermgr/backup/files-YYYY-MM-DD-HH-MM-SS.tar.gz root@<remote server
   ip:>:/<remote server location>
   ```

   Note: Change *YYYY-MM-DD-HH-MM-SS* to your backup file name that you created in Step 1.

Redeploy the Cluster Manager VM and Restore the Configuration

Caution: Existing configurations will be lost if not properly backed up as described in the previous section.

To redeploy the entire Cluster Manager VM, follow the steps in the ***Deploy the Cluster Manager VM*** section of the *CPS Installation Guide* to redeploy a new Cluster Manager VM.

To restore the previously backed up Cluster Manager VM configuration to the new Cluster Manager VM:

1. Log into the new Cluster Manager VM and execute the following command to import the backed up files from the remote server:

   ```
   /bin/mkdir -p /var/clustermgr/restore; cd /var/clustermgr/restore

   /usr/bin/scp root@<remote server ip>:/<remote server
   location>/files-YYYY-MM-DD-HH-MM-SS.tar.gz /var/clustermgr/restore
   ```

2. Restore the backup files from the remote server:

   ```
   cd / ; /bin/tar -P -xzf /var/clustermgr/restore/files-YYYY-MM-DD-HH-MM-SS.tar.gz
   ```

   Note: Change *YYYY-MM-DD-HH-MM-SS* to your backup file name.

VM Image Back Up and Restore Using VMware OVF Template

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

Create a Cluster Manager OVF Template Backup

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 Client:

   a. Log in to the vSphere server that hosts the Cluster Manager using VSphere client.

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

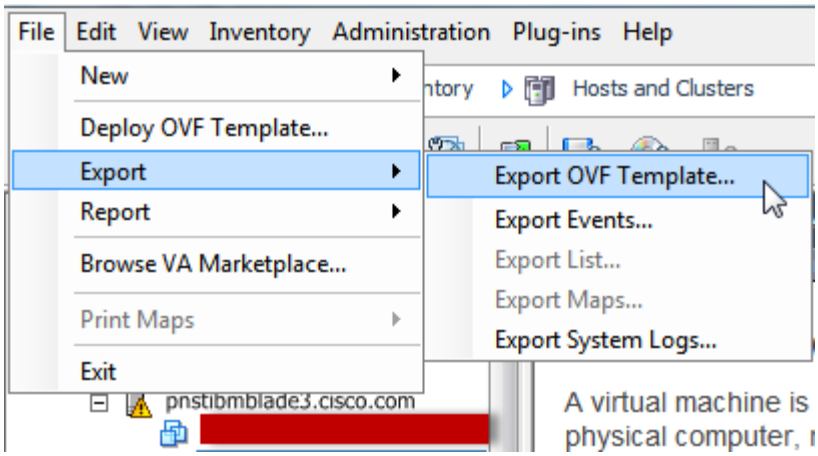
      A **Confirm Power Off** message appears. Click **Yes** to confirm the Power Off.

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

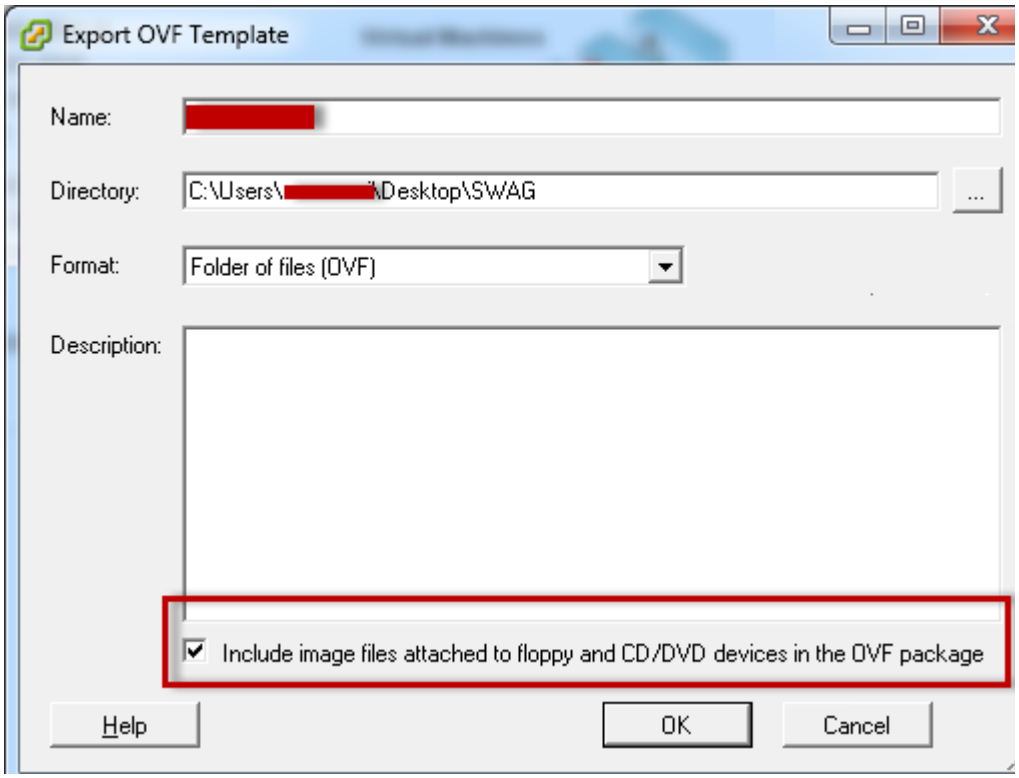
2. Export the OVF template of the Cluster Manager VM.

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

b. From the menu, select **File** > **Export** > **Export OVF Template**.



The Export OVF Template dialog opens



c. Enter the VM name in **Name** field. Also set the directory path where you want to save the backup.

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

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

f. After the export succeeds, save the OVF file.

Restore a Cluster Manager Using an OVF Template Backup

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

1. Login to ESX server using vSphere Client.
2. Select File > Deploy OVF Template... The Deploy OVF Template wizard opens.
3. Click **Browse** to select the required OVF template file and click **Next**.
4. Enter the name for the VM in the **Name** filed and click **Next**.
5. Select the required destination datastore from the **Storage** windows where the OVF template will be deployed and click **Next**.
6. From **Disk Format** window, select the format in which you want to store the template and click **Next**.
7. From **Network Mapping** window, select the network (map the networks used in OVF template to the network in your inventory) and click **Next**.
8. Verify the settings from **Ready to Complete** window and click **Finish**.
9. After the OVF template is deployed is successfully deployed, power on the VM. The Cluster Manager VM is deployed 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 by deploying a new VM from the Cluster Manager. Because the Cluster Manager is already staged with the correct software image and configuration, deploying a new VM node is very simple.

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 Back Up and Restore the Cluster Manager VM, page 6.

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
- pcrfclient02 VM
- sessionmgr VMs
- All Other VMs

pcrfclient01 VM

To redeploy the **pcrfclient01** VM:

1. Log in to the Cluster Manager VM as the root user.
2. Note the UUID of SVN repository using the following command:

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

The command will output the UUID of the repository. For example:

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

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

   /var/qps/bin/support/env/env_import.sh –svn-cm /var/tmp/env_export_svn_*$date*.tgz

   where *$date* is the date when the backup file was created.

   Note: Many deployments run a cron job that backs up configuration data regularly. See Subversion Repository Backup, page 17 for more details.

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

5. To deploy the pcrfclient01 VM, execute the following command:

   /var/qps/install/current/scripts/deployer/deploy.sh pcrfclient01

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

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:

   /etc/init.d/sessionmgr-XXXXX start

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

pcrfclient02 VM

To redeploy the **pcrfclient02** VM:

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

3. Execute the following command to redeploy and replace the failed or corrupt VM:

   /var/qps/install/current/scripts/deployer/deploy.sh pcrfclient02

4. Secure shell to the pcrfclient01.

   ssh pcrfclient01

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:

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

3. Execute the following command to redeploy and replace the failed or corrupt VM:

   /var/qps/install/current/scripts/deployer/deploy.sh sessionmgr*XX*

4. 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

5. Secure shell to the sessionmgr VM and start the mongo process:

   ssh sessionmgr*XX*

   /etc/init.d/sessionmgr-*XXXXX* start

6. 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.

All Other VMs

To redeploy any other CPS VMs:

1. Log in to the Cluster Manager VM as the root user and execute the following command:

   /var/qps/install/current/scripts/deployer/deploy.sh *host*

   where *`host`* is the short alias name and **not** the full host name.

   Note: This command uses the short alias name (qns01, qns02, etc.) of the VM as defined in the Hosts tab of the CPS Deployment Template. It will not work if you enter the host name of the VM.

   For example,

   `/var/qps/install/current/scripts/deployer/deploy.sh qns01` < === succeeds

   `/var/qps/install/current/scripts/deployer/deploy.sh NDC2BSND2QNS01` < === fails

   Once the VM is deployed, the VM will boot and all CPS processes will be started automatically.

Restore All VMs in the Cluster

To restore/redeploy all VMs in the cluster (except for Cluster Manager):

1. Log in to the Cluster Manager VM and execute the following command:

   python /var/qps/install/current/scripts/deployer/support/deploy_all.py

2. Stop all the CPS processes by executing the following command:

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

3. Create replica-sets for all sets defined in configuration file, `/etc/broadhop/mongoConfig.cfg`:

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

4. Transfer all database dumps to the Cluster Manager. Refer to General Procedure for Database Backup, page 14 for steps on creating the database dumps.

5. Restore all database dumps by executing the following command:

   /var/qps/bin/support/env/env_import.sh --mongo /var/tmp/env_export_*$date*.tgz

6. Verify database status by executing the following command:

   /var/qps/bin/diag/diagnostics.sh --get_replica_status

7. To restore policy information, refer to the section Subversion Repository Restore, page 18.

8. Start all CPS processes by executing the following command:

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

9. Verify working system by executing the following command:

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

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.

This section covers the following topics:

- Mongo Database Backup, page 13
- Mongo Database Restore, page 15

Mongo Database Backup

CPS uses Mongo DB for primary system databases. These include:

- Admin
- Audit
- Balance
- Customer Reference Data
- Policy Reporting
- Portal
- Radius
- Sharding
- SPR
- Vouchers

The Session database (session_cache) runs on 27717 and can be backed up as well, but we strongly discourage backing up the session_cache because it is not useful for production environments. The session database represents transient session data of active network sessions of subscribers on the network and therefore is never the same as what is actually occurring in the network when restoring the data.

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 2 Default Ports List

Module Name	Database Name	Default Ports
Core	admin	27721
Audit	audit	27725
Balance	balance_mgmt	27718
Customer 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

All-in-One (AIO) Environment:

By default, in AIO environment all databases run on port 27017 except the portal which runs on port 27749 like in the multi-node environment.

General Procedure for Database Backup

The Mongo DB provides various tools to assist with database backups. Mongodump is the recommended tool for the CPS environment.

The following CPS administrative script utilizes mongodump to generate a backup of the CPS databases.

/var/qps/bin/support/env/env_export.sh --mongo /var/tmp/env_export_*$date*.tgz

where *$date* is the date when the backup file was created.

For example,

env_export.sh --mongo /var/tmp/env_export_*2014-10-30*.tgz

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 Database 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 database backups on any CPS node. Move them immediately to the Cluster Manager for removal to external storage like a Storage Area Network (SAN).

Note: Because the export script may be used by multiple cron entries, it implements a waiting function. If the script detects that another process is already running, it will wait until the other process has completed and then continue. To avoid conflicts and overwriting data due to multiple cron entries running the script for automated backups, make sure that each cron entry has a unique export file name that includes a timestamp. For example, /var/tmp/export_mongo_$(date +%Y-%m-%d).tgz.

The following example creates a backup of the default database set (Admin, Balance, Customer Reference Data, Sharding, and SPR) every night at 10:00 pm:

1. Login to the Cluster Manager (ssh or console login through VMware client) as the root user.
2. To edit the root user's cron tab, execute the command:

   crontab -e
3. Add the following line:

   22 * * * /var/qps/bin/support/env/env_export.sh --mongo /var/tmp/env_export_mongo_$(date +%Y-%m-%d).tgz

   Note: The crontab editor is VI.
4. Save the file and the new cron tab is installed.

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/

General Procedure for Database Restore

The following steps describe how to import data from a previous backup (as described in General Procedure for Database Backup, page 14).

If the database is damaged, refer to Repairing a Damaged Database, page 16 or Rebuilding a Damaged Database, page 17 before proceeding with these database restoration steps.

1. Execute the following command to restore the database:

   /var/qps/bin/support/env/env_import.sh --mongo /var/tmp/env_export_mongo_*$date*.tgz

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

   For example,

   env_import.sh --mongo /var/tmp/env_export_*2014-10-30*.tgz

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).

Repairing 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 the next section to rebuild it.

Run the following commands:

/etc/init.d/sessionmgr-*$port* stop

/etc/init.d/sessionmgr-*$port* repair

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:

/etc/init.d/sessionmgr-*$port* start

After repairing the database, you can proceed to import the most recent data using env_import.sh as described in General Procedure for Database Backup, page 14.

Rebuilding a Damaged Database

If the existing data in the database is damaged and cannot be repaired/recovered (using the steps in Repairing a Damaged Database, page 16), the database must be rebuilt.

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

   ssh pcrfclient01

2. To rebuild the failed database:

   cd /var/qps/bin/support/mongo

   build_set.sh --*$db_name* --create

3. To rebuild a specific replica-set:

   build_set.sh --*$db_name* --create --setname *$setname*

   where:

   $db_name: Database name

   $setname: Set name of the replica-set which can found from /etc/broadhop/mongoConfig.cfg file.

4. After repairing the database, you can proceed to import the most recent data using env_import.sh as described in General Procedure for Database Backup, page 14.

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 Repository 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 the Cluster Manager for removal to external storage like a Storage Area Network (SAN).

Note: Because the export script may be used by multiple cron entries, it implements a waiting function. If the script detects that another process is already running, it will wait until the other process has completed and then continue. To avoid conflicts and overwriting data due to multiple cron entries running the script for automated backups, make sure that each cron entry has a unique export file name that includes a timestamp. For example, /var/tmp/export_mongo_$(date +%Y-%m-%d).tgz.

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

1. Login to the Cluster Manager (ssh or console login through VMware client) as the root user.
2. To edit the root user's cron tab, execute the command:

   crontab -e
3. Add the following line:

   22 * * * /var/qps/bin/support/env/env_export.sh --svn /var/tmp/env_export_svn_$(date +%Y-%m-%d).tgz

   Note: The crontab editor is VI.

Save the file and the new cron tab is installed.

Subversion Repository Restore

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

/var/qps/bin/support/env/env_import.sh --svn /var/tmp/env_export_svn_*$date*.tgz

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

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.

Grafana Dashboard Backup and Restore

For more information on exporting and importing Grafana Dashboards, refer to *CPS Operations Guide*.