Red Hat OpenStack Platform 13

Back Up and Restore the Director Undercloud

Back up and restore the director undercloud

Last Updated: 2021-08-05

Red Hat OpenStack Platform 13 Back Up and Restore the Director Undercloud

Back up and restore the director undercloud

OpenStack Team
rhos-docs@redhat.com

Legal Notice

Abstract

A guide to backing up and restoring the undercloud in Red Hat OpenStack Platform director.

Table of Contents

CHAPTER 1. BACK UP THE UNDERCLOUD 3
1.1. BACKUP CONSIDERATIONS 3
1.2. HIGH AVAILABILITY OF THE UNDERCLOUD NODE 3
1.3. CREATING A BAREMETAL UNDERCLOUD BACKUP 3
1.4. VALIDATE THE COMPLETED BACKUP 5

PART I. RESTORE THE UNDERCLOUD 6

CHAPTER 2. RESTORING THE UNDERCLOUD 7

CHAPTER 3. RESTORING IMAGES FOR OVERCLOUD NODES 12

CHAPTER 4. VALIDATE THE COMPLETED RESTORE 13
4.1. CHECK IDENTITY SERVICE (KEYSTONE) OPERATION 13

CHAPTER 1. BACK UP THE UNDERCLOUD

This guide describes how to back up the undercloud used in the Red Hat OpenStack Platform director. The undercloud is usually a single physical node (although high availability options exist using a two-node pacemaker cluster that runs director in a VM) that is used to deploy and manage your OpenStack environment.

1.1. BACKUP CONSIDERATIONS

Formulate a robust back up and recovery policy in order to minimize data loss and system downtime. When determining your back up strategy, you will need to answer the following questions:

- *How quickly will you need to recover from data loss?* If you cannot have data loss at all, you should include high availability in your deployment strategy, in addition to using backups. You'll need to consider how long it will take to obtain the physical backup media (including from an offsite location, if used), and how many tape drives are available for restore operations.
- *How many backups should you keep?* You will need to consider legal and regulatory requirements that affect how long you are expected to store data.
- *Should your backups be kept off-site?* Storing your backup media offsite will help mitigate the risk of catastrophe befalling your physical location.
- *How often should backups be tested?* A robust back up strategy will include regular restoration tests of backed up data. This can help validate that the correct data is still being backed up, and that no corruption is being introduced during the back up or restoration processes. These drills should assume that they are being performed under actual disaster recovery conditions.
- *What will be backed up?* The following sections describe database and file-system backups for components, as well as information on recovering backups.

1.2. HIGH AVAILABILITY OF THE UNDERCLOUD NODE

You are free to consider your preferred high availability (HA) options for the Undercloud node; Red Hat does not prescribe any particular requirements for this. For example, you might consider running your Undercloud node as a highly available virtual machine within Red Hat Enterprise Virtualization (RHEV). You might also consider using physical nodes with Pacemaker providing HA for the required services.

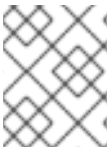
When approaching high availability for your Undercloud node, you should consult the documentation and good practices of the solution you decide works best for your environment.

1.3. CREATING A BAREMETAL UNDERCLOUD BACKUP

A full undercloud backup includes the following databases and files:

- All MariaDB databases on the undercloud node
- MariaDB configuration file on the undercloud (so that you can accurately restore databases)
- The configuration data: **/etc**
- Log data: **/var/log**
- Image data: **/var/lib/glance**

- Certificate generation data if using SSL: **/var/lib/certmonger**
- Any container image data: **/var/lib/docker** and **/var/lib/registry**
- All swift data: **/srv/node**
- All data in the stack user home directory: **/home/stack**



NOTE

Confirm that you have sufficient disk space available on the undercloud before performing the backup process. Expect the archive file to be at least 3.5 GB, if not larger.

Procedure

1. Log into the undercloud as the **root** user.
2. Back up the database:

```
[root@director ~]# mysqldump --opt --all-databases > /root/undercloud-all-databases.sql
```

3. Create a **backup** directory and change the user ownership of the directory to the **stack** user:

```
[root@director ~]# mkdir /backup
[root@director ~]# chown stack: /backup
```

You will use this directory to store the archive containing the undercloud database and file system.

4. Change to the **backup** directory

```
[root@director ~]# cd /backup
```

5. Archive the database backup and the configuration files:

```
[root@director ~]# tar --xattrs --xattrs-include='*.*' --ignore-failed-read -cf \
  undercloud-backup-`date +%F`.tar \
  /root/undercloud-all-databases.sql \
  /etc \
  /var/log \
  /var/lib/glance \
  /var/lib/certmonger \
  /var/lib/docker \
  /var/lib/registry \
  /srv/node \
  /root \
  /home/stack
```

- The **--ignore-failed-read** option skips any directory that does not apply to your undercloud.
- The **--xattrs** and **--xattrs-include='*.*'** options include extended attributes, which are required to store metadata for Object Storage (swift) and SELinux.

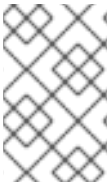
This creates a file named **undercloud-backup-<date>.tar.gz**, where **<date>** is the system date. Copy this **tar** file to a secure location.

1.4. VALIDATE THE COMPLETED BACKUP

You can validate the success of the completed back up process by running and validating the restore process. See the next section for further details on restoring from backup.

PART I. RESTORE THE UNDERCLOUD

This section describes how to restore the undercloud used in the Red Hat OpenStack Platform Director.



NOTE

This process contains steps to restore the data from the OpenStack Platform director backup to a fresh undercloud installation. As a result, the restored undercloud uses the latest packages.

CHAPTER 2. RESTORING THE UNDERCLOUD

The following restore procedure assumes your undercloud node has failed and is in an unrecoverable state. This procedure involves restoring the database and critical filesystems on a fresh installation. It assumes the following:

- You have re-installed the latest version of Red Hat Enterprise Linux 7.
- The hardware layout is the same.
- The hostname and undercloud settings of the machine are the same.
- The backup archive has been copied to the **root** directory.

Procedure

1. Log into your undercloud as the **root** user.
2. Register your system with the Content Delivery Network, entering your Customer Portal user name and password when prompted:

```
[root@director ~]# subscription-manager register
```

3. Attach the Red Hat OpenStack Platform entitlement:

```
[root@director ~]# subscription-manager attach --pool=Valid-Pool-Number-123456
```

4. Disable all default repositories, and enable the required Red Hat Enterprise Linux repositories:

```
[root@director ~]# subscription-manager repos --disable=*
[root@director ~]# subscription-manager repos --enable=rhel-7-server-rpms --enable=rhel-7-server-extras-rpms --enable=rhel-7-server-rh-common-rpms --enable=rhel-ha-for-rhel-7-server-rpms --enable=rhel-7-server-openstack-13-rpms
```

5. Perform an update on your system to ensure that you have the latest base system packages:

```
[root@director ~]# yum update -y
[root@director ~]# reboot
```

6. Ensure that the time on your undercloud is synchronized. For example:

```
[root@director ~]# yum install -y ntp
[root@director ~]# systemctl start ntpd
[root@director ~]# systemctl enable ntpd
[root@director ~]# ntpdate pool.ntp.org
[root@director ~]# systemctl restart ntpd
```

7. Copy the undercloud backup archive to the undercloud's **root** directory. The following steps use **undercloud-backup-$TIMESTAMP.tar** as the filename, where $TIMESTAMP is a Bash variable for the timestamp on the archive.
8. Install the database server and client tools:

```
[root@director ~]# yum install -y mariadb mariadb-server
```

9. Start the database:

```
[root@director ~]# systemctl start mariadb
[root@director ~]# systemctl enable mariadb
```

10. Increase the allowed packets to accommodate the size of our database backup:

```
[root@director ~]# mysql -uroot -e"set global max_allowed_packet = 1073741824;"
```

11. Extract the database and database configuration from the archive:

```
[root@director ~]# tar -xvC / -f undercloud-backup-$TIMESTAMP.tar
etc/my.cnf.d/*server*.cnf
[root@director ~]# tar -xvC / -f undercloud-backup-$TIMESTAMP.tar root/undercloud-all-
databases.sql
```

12. Restore the database backup:

```
[root@director ~]# mysql -u root < /root/undercloud-all-databases.sql
```

13. Extract a temporary version of the root configuration file:

```
[root@director ~]# tar -xvf undercloud-backup-$TIMESTAMP.tar root/.my.cnf
```

14. Get the old root database password:

```
[root@director ~]# OLDPASSWORD=$(sudo cat root/.my.cnf | grep -m1 password | cut -d'='
-f2 | tr -d "'")
```

15. Reset the root database password:

```
[root@director ~]# mysqladmin -u root password "$OLDPASSWORD"
```

16. Move the root configuration file from the temporary directory to the **root** directory:

```
[root@director ~]# mv ~/root/.my.cnf ~/.
[root@director ~]# rmdir ~/root
```

17. Get a list of old user permissions:

```
[root@director ~]# mysql -e 'select host, user, password from mysql.user;'
```

18. Remove the old user permissions for each host listed. For example:

```
[root@director ~]# HOST="192.0.2.1"
[root@director ~]# USERS=$(mysql -Nse "select user from mysql.user WHERE user !=
\"root\" and host = \"$HOST\";" | uniq | xargs)
[root@director ~]# for USER in $USERS ; do mysql -e "drop user \"$USER\"@\"$HOST\"" ||
true ;done
[root@director ~]# for USER in $USERS ; do mysql -e "drop user $USER" || true ;done
[root@director ~]# mysql -e 'flush privileges'
```

Perform this for all users accessing through the host IP and any host ("**%**").

NOTE

The IP address in the HOST parameter is the undercloud's IP address in control plane.

19. Restart the database:

```
[root@director ~]# systemctl restart mariadb
```

20. Create the **stack** user:

```
[root@director ~]# useradd stack
```

21. Set a password for the user:

```
[root@director ~]# passwd stack
```

22. Disable password requirements when using **sudo**:

```
[root@director ~]# echo "stack ALL=(root) NOPASSWD:ALL" | tee -a /etc/sudoers.d/stack
[root@director ~]# chmod 0440 /etc/sudoers.d/stack
```

23. Restore the **stack** user home directory:

```
# tar -xvC / -f undercloud-backup-$TIMESTAMP.tar home/stack
```

24. Install the **policycoreutils-python** package:

```
[root@director ~]# yum -y install policycoreutils-python
```

25. Install the **openstack-glance** package and restore its data and file permissions:

```
[root@director ~]# yum install -y openstack-glance
[root@director ~]# tar --xattrs --xattrs-include='*.*' -xvC / -f undercloud-backup-
$TIMESTAMP.tar var/lib/glance/images
[root@director ~]# chown -R glance: /var/lib/glance/images
[root@director ~]# restorecon -R /var/lib/glance/images
```

26. Install the **openstack-swift** package and restore its data and file permissions:

```
[root@director ~]# yum install -y openstack-swift
[root@director ~]# tar --xattrs --xattrs-include='*.*' -xvC / -f undercloud-backup-
$TIMESTAMP.tar srv/node
[root@director ~]# chown -R swift: /srv/node
[root@director ~]# restorecon -R /srv/node
```

27. Install the **openstack-keystone** package and restore its configuration data:

```
[root@director ~]# yum -y install openstack-keystone
[root@director ~]# tar -xvC / -f undercloud-backup-$TIMESTAMP.tar etc/keystone
[root@director ~]# restorecon -R /etc/keystone
```

28. Install the **openstack-heat** and restore configuration:

```
[root@director ~]# yum install -y openstack-heat*
[root@director ~]# tar -xvC / -f undercloud-backup-$TIMESTAMP.tar etc/heat
[root@director ~]# restorecon -R /etc/heat
```

29. Install puppet and restore its configuration data:

```
[root@director ~]# yum install -y puppet hiera
[root@director ~]# tar -xvC / -f undercloud-backup-$TIMESTAMP.tar etc/puppet/hieradata/
```

30. If you use SSL in the undercloud, refresh the CA certificates. Depending on your undercloud configuration, use either the steps for user-provided certificates or the steps for the auto-generated certificates:

- If the undercloud is configured with user-provided certificates, complete the following steps:

    a. Extract the certificates:

    ```
    [root@director ~]# tar -xvC / -f undercloud-backup-$TIMESTAMP.tar etc/pki/instack-certs/undercloud.pem
    [root@director ~]# tar -xvC / -f undercloud-backup-$TIMESTAMP.tar etc/pki/ca-trust/source/anchors/*
    ```

    b. Restore the SELinux contexts and manage the file system labelling:

    ```
    [root@director ~]# restorecon -R /etc/pki
    [root@director ~]# semanage fcontext -a -t etc_t "/etc/pki/instack-certs(/.*)?"
    [root@director ~]# restorecon -R /etc/pki/instack-certs
    ```

    c. Update the certificates:

    ```
    [root@director ~]# update-ca-trust extract
    ```

- If you use **certmonger** to auto-generate certificates for the undercloud, complete the following steps:

    a. Extract certificates, CA certificate and certmonger files:

    ```
    [root@director ~]# tar -xvC / -f undercloud-backup-$TIMESTAMP.tar var/lib/certmonger/*
    [root@director ~]# tar -xvC / -f undercloud-backup-$TIMESTAMP.tar etc/pki/tls/*
    [root@director ~]# tar -xvC / -f undercloud-backup-$TIMESTAMP.tar etc/pki/ca-trust/source/anchors/*
    ```

    b. Restore the SELinux contexts:

```
[root@director ~]# restorecon -R /etc/pki
[root@director ~]# restorecon -R /var/lib/certmonger
```

c. Remove the **/var/lib/certmonger/lock** file:

```
[root@director ~]# rm -f /var/lib/certmonger/lock
```

31. Switch to the **stack** user:

```
[root@director ~]# su - stack
[stack@director ~]$
```

32. Install the **python-tripleoclient** package:

```
$ sudo yum install -y python-tripleoclient
```

33. Run the undercloud installation command. Ensure that you run it in the **stack** user's home directory:

```
[stack@director ~]$ openstack undercloud install
```

When the install completes, the undercloud automatically restores its connection to the overcloud. The nodes continue to poll OpenStack Orchestration (heat) for pending tasks.

34. Switch to the root user:

```
$ sudo su -
```

35. Synchronize the container data with backup content:

```
[root@director ~]$ tar -xvC / -f undercloud-backup-$TIMESTAMP.tar var/lib/docker/
[root@director ~]$ tar -xvC / -f undercloud-backup-$TIMESTAMP.tar var/lib/registry/
[root@director ~]$ tar -xvC / -f undercloud-backup-$TIMESTAMP.tar etc/docker/
[root@director ~]$ tar -xvC / -f undercloud-backup-$TIMESTAMP.tar etc/docker-distribution/
[root@director ~]$ tar -xvC / -f undercloud-backup-$TIMESTAMP.tar etc/sysconfig/docker*
[root@director ~]$ systemctl restart docker docker-distribution
```

CHAPTER 3. RESTORING IMAGES FOR OVERCLOUD NODES

The director requires the latest disk images for provisioning new overcloud nodes. Follow this procedure to restore these images.

Procedure

1. Source the **stackrc** file to enable the director's command line tools:

```
[stack@director ~]$ source ~/stackrc
```

2. Install the **rhosp-director-images** and **rhosp-director-images-ipa** packages:

```
(undercloud) [stack@director ~]$ sudo yum install rhosp-director-images rhosp-director-images-ipa
```

3. Extract the images archives to the **images** directory in the **stack** user's home (/**home**/**stack**/**images**):

```
(undercloud) [stack@director ~]$ cd ~/images
(undercloud) [stack@director images]$ for i in /usr/share/rhosp-director-images/overcloud-full-latest-13.0.tar /usr/share/rhosp-director-images/ironic-python-agent-latest-13.0.tar; do tar -xvf $i; done
```

4. Import these images into the director:

```
(undercloud) [stack@director images]$ cd ~/images
(undercloud) [stack@director images]$ openstack overcloud image upload --image-path /home/stack/images/
```

5. Configure nodes in your environment to use the new images:

```
(undercloud) [stack@director images]$ for NODE in $(openstack baremetal node list -c UUID -f value) ; do openstack overcloud node configure $NODE ; done
```

CHAPTER 4. VALIDATE THE COMPLETED RESTORE

Use the following commands to perform a healthcheck of your newly restored environment:

4.1. CHECK IDENTITY SERVICE (KEYSTONE) OPERATION

This step validates Identity Service operations by querying for a list of users.

```
# source stackrc
# openstack user list
```

When run from the controller, the output of this command should include a list of users created in your environment. This action demonstrates that keystone is running and successfully authenticating user requests. For example:

```
# openstack user list
+----------------------------------+------------+---------+----------------------+
|                id                |    name    | enabled |        email         |
+----------------------------------+------------+---------+----------------------+
| 9e47bb53bb40453094e32eccce996828 |   admin    |   True  |    root@localhost    |
| 9fe2466f88cc4fa0ba69e59b47898829 | ceilometer |   True  | ceilometer@localhost |
| 7a40d944e55d422fa4e85daf47e47c42 |   cinder   |   True  |   cinder@localhost   |
| 3d2ed97538064f258f67c98d1912132e |    demo    |   True  |                      |
| 756e73a5115d4e9a947d8aadc6f5ac22 |   glance   |   True  |   glance@localhost   |
| f0d1fcee8f9b4da39556b78b72fdafb1 |  neutron   |   True  |  neutron@localhost   |
| e9025f3faeee4d6bb7a057523576ea19 |    nova    |   True  |    nova@localhost    |
| 65c60b1278a0498980b2dc46c7dcf4b7 |   swift    |   True  |   swift@localhost    |
+----------------------------------+------------+---------+----------------------+
```