



# Red Hat OpenStack Platform 16.0

## Keeping Red Hat OpenStack Platform Updated

Performing minor updates of Red Hat OpenStack Platform



# Red Hat OpenStack Platform 16.0 Keeping Red Hat OpenStack Platform Updated

---

Performing minor updates of Red Hat OpenStack Platform

OpenStack Team  
rhos-docs@redhat.com

## Legal Notice

Copyright © 2021 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux<sup>®</sup> is the registered trademark of Linus Torvalds in the United States and other countries.

Java<sup>®</sup> is a registered trademark of Oracle and/or its affiliates.

XFS<sup>®</sup> is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL<sup>®</sup> is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js<sup>®</sup> is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack<sup>®</sup> Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

## Abstract

This document provides the procedure to update your Red Hat OpenStack Platform 16.0 (Train) environment. This document assumes you will update a containerized OpenStack Platform deployment installed on Red Hat Enterprise Linux 8.

---

## Table of Contents

<b>CHAPTER 1. INTRODUCTION</b> .....	<b>3</b>
1.1. HIGH LEVEL WORKFLOW	3
<b>CHAPTER 2. PREPARING FOR A MINOR UPDATE</b> .....	<b>4</b>
2.1. LOCKING THE ENVIRONMENT TO A RED HAT ENTERPRISE LINUX RELEASE	4
2.2. CHANGING TO EXTENDED UPDATE SUPPORT (EUS) REPOSITORIES	5
<b>CHAPTER 3. UPDATING THE UNDERCLOUD</b> .....	<b>7</b>
3.1. PERFORMING A MINOR UPDATE OF A CONTAINERIZED UNDERCLOUD	7
3.2. UPDATING THE OVERCLOUD IMAGES	7
3.3. UNDERCLOUD POST-UPGRADE NOTES	8
3.4. NEXT STEPS	8
<b>CHAPTER 4. UPDATING THE OVERCLOUD</b> .....	<b>9</b>
4.1. RUNNING THE OVERCLOUD UPDATE PREPARATION	9
4.2. RUNNING THE CONTAINER IMAGE PREPARATION	9
4.3. UPDATING ALL CONTROLLER NODES	10
4.4. UPDATING ALL COMPUTE NODES	10
4.5. UPDATING ALL HCI COMPUTE NODES	11
4.6. UPDATING ALL CEPH STORAGE NODES	11
4.7. PERFORMING ONLINE DATABASE UPDATES	12
4.8. FINALIZING THE UPDATE	12
<b>CHAPTER 5. REBOOTING THE OVERCLOUD</b> .....	<b>14</b>
5.1. REBOOTING CONTROLLER AND COMPOSABLE NODES	14
5.2. REBOOTING A CEPH STORAGE (OSD) CLUSTER	14
5.3. REBOOTING COMPUTE NODES	15
5.4. REBOOTING HCI COMPUTE NODES	17



# CHAPTER 1. INTRODUCTION

This document provides a workflow to help keep your Red Hat OpenStack Platform 16.0 environment updated with the latest packages and containers.

This guide provides an upgrade path through the following versions:

Old Overcloud Version	New Overcloud Version
Red Hat OpenStack Platform 16.0	Red Hat OpenStack Platform 16.0.z

## 1.1. HIGH LEVEL WORKFLOW

The following table provides an outline of the steps required for the upgrade process:

Step	Description
Updating the undercloud	Update the undercloud to the latest OpenStack Platform 16.0.z version.
Updating the overcloud	Update the overcloud to the latest OpenStack Platform 16.0.z version.
Updating the Ceph Storage nodes	Upgrade all Ceph Storage services.
Finalize the upgrade	Run the convergence command to refresh your overcloud stack.

## CHAPTER 2. PREPARING FOR A MINOR UPDATE

You must follow some preparation steps on the undercloud and overcloud before you begin the process to update **Red Hat OpenStack Platform 16.0** to the latest minor release.

### 2.1. LOCKING THE ENVIRONMENT TO A RED HAT ENTERPRISE LINUX RELEASE

Red Hat OpenStack Platform 16.0 is supported on Red Hat Enterprise Linux 8.1. Prior to performing the update, lock the undercloud and overcloud repositories to the Red Hat Enterprise Linux 8.1 release to avoid upgrading the operating system to a newer minor release.

#### Procedure

1. Log into the undercloud as the **stack** user.
2. Source the **stackrc** file:

```
$ source ~/stackrc
```

3. Create a static inventory file of your overcloud:

```
$ tripleo-ansible-inventory --ansible_ssh_user heat-admin --static-yaml-inventory  
~/inventory.yaml
```

If you use an overcloud name different to the default overcloud name of **overcloud**, set the name of your overcloud with the **--plan** option.

4. Create a playbook that contains a task to lock the operating system version to Red Hat Enterprise Linux 8.1 on all nodes:

```
$ cat > ~/set_release.yaml <<'EOF'  
- hosts: overcloud,undercloud  
  gather_facts: false  
  tasks:  
    - name: set release to 8.1  
      command: subscription-manager release --set=8.1  
      become: true  
EOF
```

5. Run the **set\_release.yaml** playbook:

```
$ ansible-playbook -i ~/inventory.yaml -f 25 ~/set_release.yaml
```



#### NOTE

To manually lock a node to a version, log in to the node and run the **subscription-manager release** command:

```
$ sudo subscription-manager release --set=8.1
```



## 2.2. CHANGING TO EXTENDED UPDATE SUPPORT (EUS) REPOSITORIES

Your Red Hat OpenStack Platform subscription includes repositories for Red Hat Enterprise Linux 8.1 Extended Update Support (EUS). The EUS repositories include the latest security patches and bug fixes for Red Hat Enterprise Linux 8.1. Switch to the following repositories before performing a minor version update.

Standard Repository	EUS Repository
rhel-8-for-x86_64-baseos-rpms	rhel-8-for-x86_64-baseos-eus-rpms
rhel-8-for-x86_64-appstream-rpms	rhel-8-for-x86_64-appstream-eus-rpms
rhel-8-for-x86_64-highavailability-rpms	rhel-8-for-x86_64-highavailability-eus-rpms

### Procedure

1. Log into the undercloud as the **stack** user.
2. Source the **stackrc** file:
 

```
$ source ~/stackrc
```
3. Edit your overcloud subscription management environment file, which is the file that contains the **RhsmVars** parameter. The default name for this file is usually **rhsm.yml**.
4. Check the **rhsm\_repos** parameter in your subscription management configuration. If this parameter does not include the EUS repositories, change the relevant repositories to the EUS versions:

```
parameter_defaults:
  RhsmVars:
    rhsm_repos:
      - rhel-8-for-x86_64-baseos-eus-rpms
      - rhel-8-for-x86_64-appstream-eus-rpms
      - rhel-8-for-x86_64-highavailability-eus-rpms
      - ansible-2.9-for-rhel-8-x86_64-rpms
      - advanced-virt-for-rhel-8-x86_64-rpms
      - openstack-beta-for-rhel-8-x86_64-rpms
      - rhceph-4-osd-for-rhel-8-x86_64-rpms
      - rhceph-4-mon-for-rhel-8-x86_64-rpms
      - rhceph-4-tools-for-rhel-8-x86_64-rpms
      - fast-datapath-for-rhel-8-x86_64-rpms
```

5. Save the overcloud subscription management environment file.
6. Create a static inventory file of your overcloud:

```
$ tripleo-ansible-inventory --ansible_ssh_user heat-admin --static-yaml-inventory
~/inventory.yaml
```

If you use an overcloud name different to the default overcloud name of **overcloud**, set the name of your overcloud with the **--plan** option.

7. Create a playbook that contains a task to set the repositories to Red Hat Enterprise Linux 8.1 EUS on all nodes:

```
$ cat > ~/change_eus.yaml <<'EOF'
- hosts: overcloud,undercloud
  gather_facts: false
  tasks:
    - name: change to eus repos
      command: subscription-manager repos --disable=rhel-8-for-x86_64-baseos-rpms --
disable=rhel-8-for-x86_64-appstream-rpms --disable=rhel-8-for-x86_64-highavailability-rpms
--enable=rhel-8-for-x86_64-baseos-eus-rpms --enable=rhel-8-for-x86_64-appstream-eus-
rpms --enable=rhel-8-for-x86_64-highavailability-eus-rpms
      become: true
EOF
```

8. Run the **change\_eus.yaml** playbook:

```
$ ansible-playbook -i ~/inventory.yaml -f 25 ~/change_eus.yaml
```

## CHAPTER 3. UPDATING THE UNDERCLOUD

This process updates the undercloud and its overcloud images to the latest **Red Hat OpenStack Platform 16.0** version.

### 3.1. PERFORMING A MINOR UPDATE OF A CONTAINERIZED UNDERCLOUD

The director provides commands to update the packages on the undercloud node. This allows you to perform a minor update within the current version of your OpenStack Platform environment.

#### Procedure

1. Log into the director as the **stack** user.
2. Run **dnf** to upgrade the director's main packages:

```
$ sudo dnf update -y python3-tripleoclient* openstack-tripleo-common openstack-tripleo-heat-templates tripleo-ansible
```

3. The director uses the **openstack undercloud upgrade** command to update the undercloud environment. Run the command:

```
$ openstack undercloud upgrade
```

4. Wait until the undercloud upgrade process completes.
5. Reboot the undercloud to update the operating system's kernel and other system packages:

```
$ sudo reboot
```

6. Wait until the node boots.

### 3.2. UPDATING THE OVERCLOUD IMAGES

You need to replace your current overcloud images with new versions. The new images ensure the director can introspect and provision your nodes using the latest version of OpenStack Platform software.

#### Prerequisites

- You have updated the undercloud to the latest version.

#### Procedure

1. Source the **stackrc** file:

```
$ source ~/stackrc
```

2. Remove any existing images from the **images** directory on the **stack** user's home (**/home/stack/images**):

```
$ rm -rf ~/images/*
```

3. Extract the archives:

```
$ cd ~/images
$ for i in /usr/share/rhosp-director-images/overcloud-full-latest-16.0.tar /usr/share/rhosp-
director-images/ironic-python-agent-latest-16.0.tar; do tar -xvf $i; done
$ cd ~
```

4. Import the latest images into the director:

```
$ openstack overcloud image upload --update-existing --image-path /home/stack/images/
```

5. Configure your nodes to use the new images:

```
$ openstack overcloud node configure $(openstack baremetal node list -c UUID -f value)
```

6. Verify the existence of the new images:

```
$ openstack image list
$ ls -l /var/lib/ironic/httpboot
```



### IMPORTANT

When deploying overcloud nodes, ensure the overcloud image version corresponds to the respective heat template version. For example, only use the OpenStack Platform 16 images with the OpenStack Platform 16 heat templates.



### IMPORTANT

The new **overcloud-full** image replaces the old **overcloud-full** image. If you made changes to the old image, you must repeat the changes in the new image, especially if you want to deploy new nodes in the future.

## 3.3. UNDERCLOUD POST-UPGRADE NOTES

- If using a local set of core templates in your **stack** users home directory, ensure you update the templates using the recommended workflow in [Using Customized Core Heat Templates](#) in the *Advanced Overcloud Customization* guide. You must update the local copy before upgrading the overcloud.

## 3.4. NEXT STEPS

The undercloud upgrade is complete. You can now update the overcloud.

## CHAPTER 4. UPDATING THE OVERCLOUD

This process updates the overcloud.

### Prerequisites

- You have updated the undercloud to the latest version.

### 4.1. RUNNING THE OVERCLOUD UPDATE PREPARATION

The update requires running **openstack overcloud update prepare** command, which performs the following tasks:

- Updates the overcloud plan to OpenStack Platform 16.0
- Prepares the nodes for the update

### Procedure

1. Source the **stackrc** file:

```
$ source ~/stackrc
```

2. Run the update preparation command:

```
$ openstack overcloud update prepare \
  --templates \
  -r <ROLES DATA FILE> \
  -n <NETWORK DATA FILE> \
  -e <ENVIRONMENT FILE> \
  -e <ENVIRONMENT FILE> \
  --stack <STACK_NAME> \
  ...
```

Include the following options relevant to your environment:

- Custom configuration environment files (**-e**)
  - If using your own custom roles, include your custom roles (**roles\_data**) file (**-r**)
  - If using custom networks, include your composable network (**network\_data**) file (**-n**)
  - If the name of your overcloud stack is different to the default name **overcloud**, include the **--stack** option in the update preparation command and replace **<STACK\_NAME>** with the name of your stack.
3. Wait until the update preparation completes.

### 4.2. RUNNING THE CONTAINER IMAGE PREPARATION

The overcloud requires the latest OpenStack Platform 16.0 container images before performing the update. This involves executing the **container\_image\_prepare** external update process. To execute this process, run the **openstack overcloud external-update run** command against tasks tagged with the **container\_image\_prepare** tag. These tasks:

- Automatically prepare all container image configuration relevant to your environment.
- Pull the relevant container images to your undercloud, unless you have previously disabled this option.

### Procedure

1. Source the **stackrc** file:

```
$ source ~/stackrc
```

2. Run the **openstack overcloud external-update run** command against tasks tagged with the **container\_image\_prepare** tag:

```
$ openstack overcloud external-update run --stack STACK_NAME --tags  
container_image_prepare
```

## 4.3. UPDATING ALL CONTROLLER NODES

This process updates all the Controller nodes to the latest OpenStack Platform 16.0 version. The process involves running the **openstack overcloud update run** command and including the **--limit Controller** option to restrict operations to the Controller nodes only.



### NOTE

If you are not using the default stack name (**overcloud**), set your stack name with the **--stack *STACK\_NAME*** option replacing **STACK\_NAME** with the name of your stack.

### Procedure

1. Source the **stackrc** file:

```
$ source ~/stackrc
```

2. Run the update command:

```
$ openstack overcloud update run --stack STACK_NAME --limit Controller --playbook all
```

3. Wait until the Controller node update completes.

## 4.4. UPDATING ALL COMPUTE NODES

This process updates all Compute nodes to the latest OpenStack Platform 16.0 version. The process involves running the **openstack overcloud update run** command and including the **--nodes Compute** option to restrict operations to the Compute nodes only.



### NOTE

If you are not using the default stack name (**overcloud**), set your stack name with the **--stack *STACK\_NAME*** option replacing **STACK\_NAME** with the name of your stack.

### Procedure

1. Source the **stackrc** file:

```
$ source ~/stackrc
```

2. Run the update command:

```
$ openstack overcloud update run --stack STACK_NAME --limit Compute --playbook all
```

3. Wait until the Compute node update completes.

## 4.5. UPDATING ALL HCI COMPUTE NODES

This process updates the Hyperconverged Infrastructure (HCI) Compute nodes. The process involves:

- Running the **openstack overcloud update run** command and including the **--nodes ComputeHCI** option to restrict operations to the HCI nodes only.
- Running the **openstack overcloud external-update run --tags ceph** command to perform an update to a containerized Red Hat Ceph Storage 4 cluster.



### NOTE

If you are not using the default stack name (**overcloud**), set your stack name with the **--stack STACK\_NAME** option replacing **STACK\_NAME** with the name of your stack.

### Procedure

1. Source the **stackrc** file:

```
$ source ~/stackrc
```

2. Run the update command:

```
$ openstack overcloud update run --stack _STACK_NAME_ --limit ComputeHCI --playbook all
```

3. Wait until the node update completes.
4. Run the Ceph Storage update command. For example:

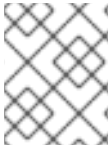
```
$ openstack overcloud external-update run --stack _STACK_NAME_ --tags ceph
```

5. Wait until the Compute HCI node update completes.

## 4.6. UPDATING ALL CEPH STORAGE NODES

This process updates the Ceph Storage nodes. The process involves:

- Running the **openstack overcloud update run** command and including the **--nodes CephStorage** option to restrict operations to the Ceph Storage nodes only.
- Running the **openstack overcloud external-update run** command to run **ceph-ansible** as an external process and update the Red Hat Ceph Storage 4 containers.



## NOTE

If you are not using the default stack name (**overcloud**), set your stack name with the **--stack STACK\_NAME** option replacing **STACK\_NAME** with the name of your stack.

### Procedure

1. Source the **stackrc** file:

```
$ source ~/stackrc
```

2. Run the update command:

```
$ openstack overcloud update run --stack STACK_NAME --limit CephStorage --playbook all
```

3. Wait until the node update completes.

4. Run the Ceph Storage container update command:

```
$ openstack overcloud external-update run --stack _STACK_NAME_ --tags ceph
```

5. Wait until the Ceph Storage container update completes.

## 4.7. PERFORMING ONLINE DATABASE UPDATES

Some overcloud components require an online upgrade (or migration) of their databases tables. This involves executing the **online\_upgrade** external update process. To execute this process, run the **openstack overcloud external-update run** command against tasks tagged with the **online\_upgrade** tag. This performs online database updates to the following components:

- OpenStack Block Storage (cinder)
- OpenStack Compute (nova)

### Procedure

1. Source the **stackrc** file:

```
$ source ~/stackrc
```

2. Run the **openstack overcloud external-update run** command against tasks tagged with the **online\_upgrade** tag:

```
$ openstack overcloud external-update run --stack STACK_NAME --tags online_upgrade
```

## 4.8. FINALIZING THE UPDATE

The update requires a final step to update the overcloud stack. This ensures the stack's resource structure aligns with a regular deployment of OpenStack Platform 16.0 and allows you to perform standard **openstack overcloud deploy** functions in the future.

### Procedure



1. Source the **stackrc** file:

```
$ source ~/stackrc
```

2. Run the update finalization command:

```
$ openstack overcloud update converge \  
  --templates \  
  -r <ROLES DATA FILE> \  
  -n <NETWORK DATA FILE> \  
  -e <ENVIRONMENT FILE> \  
  -e <ENVIRONMENT FILE> \  
  --stack <STACK_NAME> \  
  ...
```

Include the following options relevant to your environment:

+ \* Custom configuration environment files (**-e**) \* If using your own custom roles, include your custom roles (**roles\_data**) file (**-r**) \* If using custom networks, include your composable network (**network\_data**) file (**-n**) \* If the name of your overcloud stack is different to the default name **overcloud**, include the **--stack** option in the update preparation command and replace **<STACK\_NAME>** with the name of your stack.

3. Wait until the update finalization completes.

## CHAPTER 5. REBOOTING THE OVERCLOUD

After a minor Red Hat OpenStack version update, reboot your overcloud. The reboot refreshes the nodes with any associated kernel, system-level, and container component updates. These updates may provide performance and security benefits.

Plan downtime to perform the following reboot procedures.

### 5.1. REBOOTING CONTROLLER AND COMPOSABLE NODES

Complete the following steps to reboot Controller nodes and standalone nodes based on composable roles, excluding Compute nodes and Ceph Storage nodes.

#### Procedure

1. Log in to the node that you want to reboot.
2. Optional: If the node uses Pacemaker resources, stop the cluster:

```
[heat-admin@overcloud-controller-0 ~]$ sudo pcs cluster stop
```

3. Reboot the node:

```
[heat-admin@overcloud-controller-0 ~]$ sudo reboot
```

4. Wait until the node boots.
5. Check the services. For example:
  - a. If the node uses Pacemaker services, check that the node has rejoined the cluster:

```
[heat-admin@overcloud-controller-0 ~]$ sudo pcs status
```

- b. If the node uses Systemd services, check that all services are enabled:

```
[heat-admin@overcloud-controller-0 ~]$ sudo systemctl status
```

- c. If the node uses containerized services, check that all containers on the node are active:

```
[heat-admin@overcloud-controller-0 ~]$ sudo podman ps
```

### 5.2. REBOOTING A CEPH STORAGE (OSD) CLUSTER

Complete the following steps to reboot a cluster of Ceph Storage (OSD) nodes.

#### Procedure

1. Log in to a Ceph MON or Controller node and disable Ceph Storage cluster rebalancing temporarily:

```
$ sudo podman exec -it ceph-mon-controller-0 ceph osd set noout  
$ sudo podman exec -it ceph-mon-controller-0 ceph osd set norebalance
```

2. Select the first Ceph Storage node that you want to reboot and log in to the node.

3. Reboot the node:

```
$ sudo reboot
```

4. Wait until the node boots.

5. Log in to the node and check the cluster status:

```
$ sudo podman exec -it ceph-mon-controller-0 ceph status
```

Check that the **pgmap** reports all **pgs** as normal (**active+clean**).

6. Log out of the node, reboot the next node, and check its status. Repeat this process until you have rebooted all Ceph storage nodes.

7. When complete, log into a Ceph MON or Controller node and re-enable cluster rebalancing:

```
$ sudo podman exec -it ceph-mon-controller-0 ceph osd unset noout
$ sudo podman exec -it ceph-mon-controller-0 ceph osd unset norebalance
```

8. Perform a final status check to verify that the cluster reports **HEALTH\_OK**:

```
$ sudo podman exec -it ceph-mon-controller-0 ceph status
```

## 5.3. REBOOTING COMPUTE NODES

Complete the following steps to reboot Compute nodes. To ensure minimal downtime of instances in your Red Hat OpenStack Platform environment, this procedure also includes instructions about migrating instances from the Compute node that you want to reboot. This involves the following workflow:

- Decide whether to migrate instances to another Compute node before rebooting the node.
- Select and disable the Compute node you want to reboot so that it does not provision new instances.
- Migrate the instances to another Compute node.
- Reboot the empty Compute node.
- Enable the empty Compute node.

### Prerequisites

Before you reboot the Compute node, you must decide whether to migrate instances to another Compute node while the node is rebooting.

If for some reason you cannot or do not want to migrate the instances, you can set the following core template parameters to control the state of the instances after the Compute node reboots:

### **NovaResumeGuestsStateOnHostBoot**

Determines whether to return instances to the same state on the Compute node after reboot. When set to **False**, the instances will remain down and you must start them manually. Default value is: **False**

### NovaResumeGuestsShutdownTimeout

Number of seconds to wait for an instance to shut down before rebooting. It is not recommended to set this value to **0**. Default value is: 300

### NovaResumeGuestsShutdownTimeout

Number of seconds to wait for an instance to shut down before rebooting. It is not recommended to set this value to **0**. Default value is: 300

For more information about overcloud parameters and their usage, see [Overcloud Parameters](#).

### Procedure

1. Log in to the undercloud as the **stack** user.
2. List all Compute nodes and their UUIDs:

```
$ source ~/stackrc
(undercloud) $ openstack server list --name compute
```

Identify the UUID of the Compute node that you want to reboot.

3. From the undercloud, select a Compute node. Disable the node:

```
$ source ~/overcloudrc
(overcloud) $ openstack compute service list
(overcloud) $ openstack compute service set [hostname] nova-compute --disable
```

4. List all instances on the Compute node:

```
(overcloud) $ openstack server list --host [hostname] --all-projects
```

5. If you decide not to migrate instances, skip to [this step](#).
6. If you decide to migrate the instances to another Compute node, use one of the following commands:

- Migrate the instance to a different host:

```
(overcloud) $ openstack server migrate [instance-id] --live [target-host]--wait
```

- Let **nova-scheduler** automatically select the target host:

```
(overcloud) $ nova live-migration [instance-id]
```

- Live migrate all instances at once:

```
$ nova host-evacuate-live [hostname]
```



### NOTE

The **nova** command might cause some deprecation warnings, which are safe to ignore.

7. Wait until migration completes.
8. Confirm that the migration was successful:

```
(overcloud) $ openstack server list --host [hostname] --all-projects
```

9. Continue to migrate instances until none remain on the chosen Compute node.

10. Log in to the Compute node and reboot the node:

```
[heat-admin@overcloud-compute-0 ~]$ sudo reboot
```

11. Wait until the node boots.

12. Re-enable the Compute node:

```
$ source ~/overcloudrc
(overcloud) $ openstack compute service set [hostname] nova-compute --enable
```

13. Check that the Compute node is enabled:

```
(overcloud) $ openstack compute service list
```

## 5.4. REBOOTING HCI COMPUTE NODES

The following procedure reboots Compute hyperconverged infrastructure (HCI) nodes.

### Procedure

1. Log in to a Ceph MON or Controller node and disable Ceph Storage cluster rebalancing temporarily:

```
$ sudo podman exec -it ceph-mon-controller-0 ceph osd set noout
$ sudo podman exec -it ceph-mon-controller-0 ceph osd set norebalance
```

2. Log in to the undercloud as the **stack** user.

3. List all Compute nodes and their UUIDs:

```
$ source ~/stackrc
(undercloud) $ openstack server list --name compute
```

Identify the UUID of the Compute node you aim to reboot.

4. From the undercloud, select a Compute node and disable it:

```
$ source ~/overcloudrc
(overcloud) $ openstack compute service list
(overcloud) $ openstack compute service set [hostname] nova-compute --disable
```

5. List all instances on the Compute node:

```
(overcloud) $ openstack server list --host [hostname] --all-projects
```

- 6. Use one of the following commands to migrate your instances:

- a. Migrate the instance to a specific host of your choice:

```
(overcloud) $ openstack server migrate [instance-id] --live [target-host]--wait
```

- b. Let **nova-scheduler** automatically select the target host:

```
(overcloud) $ nova live-migration [instance-id]
```

- c. Live migrate all instances at once:

```
$ nova host-evacuate-live [hostname]
```



#### NOTE

The **nova** command might cause some deprecation warnings, which are safe to ignore.

- 7. Wait until the migration completes.

- 8. Confirm that the migration was successful:

```
(overcloud) $ openstack server list --host [hostname] --all-projects
```

- 9. Continue migrating instances until none remain on the chosen Compute node.

- 10. Log in to a Ceph MON or a Controller node and check the cluster status:

```
$ sudo podman exec $CEPH_MON_CONTAINER ceph -s
```

Check that the **pgmap** reports all **pgs** as normal (**active+clean**).

- 11. Reboot the Compute HCI node:

```
$ sudo reboot
```

- 12. Wait until the node boots.

- 13. Enable the Compute node again:

```
$ source ~/overcloudrc
(overcloud) $ openstack compute service set [hostname] nova-compute --enable
```

- 14. Verify that the Compute node is enabled:

```
(overcloud) $ openstack compute service list
```

- 15. Log out of the node, reboot the next node, and check its status. Repeat this process until you have rebooted all Ceph storage nodes.

16. When complete, log in to a Ceph MON or Controller node and enable cluster rebalancing again:

```
$ sudo podman exec $CEPH_MON_CONTAINER ceph osd unset noout  
$ sudo podman exec $CEPH_MON_CONTAINER ceph osd unset norebalance
```

17. Perform a final status check to verify the cluster reports **HEALTH\_OK**:

```
$ sudo podman exec $CEPH_MON_CONTAINER ceph status
```