



Red Hat OpenStack Platform 16.1

Keeping Red Hat OpenStack Platform Updated

Performing minor updates of Red Hat OpenStack Platform

Red Hat OpenStack Platform 16.1 Keeping Red Hat OpenStack Platform Updated

Performing minor updates of Red Hat OpenStack Platform

OpenStack Team
rhos-docs@redhat.com

Legal Notice

Copyright © 2023 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[®] is the registered trademark of Linus Torvalds in the United States and other countries.

Java[®] is a registered trademark of Oracle and/or its affiliates.

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

MySQL[®] is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js[®] 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[®] 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 procedures to perform a minor update of your Red Hat OpenStack Platform environment.

Table of Contents

MAKING OPEN SOURCE MORE INCLUSIVE	3
PROVIDING FEEDBACK ON RED HAT DOCUMENTATION	4
CHAPTER 1. INTRODUCTION	5
1.1. HIGH LEVEL WORKFLOW	5
1.2. KNOWN ISSUES THAT MIGHT BLOCK AN UPDATE	5
CHAPTER 2. PREPARING FOR A MINOR UPDATE	7
2.1. LOCKING THE ENVIRONMENT TO A RED HAT ENTERPRISE LINUX RELEASE	7
2.2. CHANGING FROM EUS REPOSITORIES TO TUS REPOSITORIES	8
2.3. UPDATING RED HAT OPENSTACK PLATFORM AND ANSIBLE REPOSITORIES	10
2.4. SETTING THE CONTAINER-TOOLS VERSION	11
2.5. UPDATING YOUR CONTAINER IMAGE PREPARATION FILE	12
2.6. UPDATING YOUR SSL/TLS CONFIGURATION	13
2.7. DISABLING FENCING IN THE OVERCLOUD	13
CHAPTER 3. UPDATING THE UNDERCLOUD	15
3.1. PERFORMING A MINOR UPDATE OF A CONTAINERIZED UNDERCLOUD	15
3.2. UPDATING THE OVERCLOUD IMAGES	15
3.3. UNDERCLOUD POST-UPGRADE NOTES	16
CHAPTER 4. UPDATING THE OVERCLOUD	17
4.1. RUNNING THE OVERCLOUD UPDATE PREPARATION	17
4.2. RUNNING THE CONTAINER IMAGE PREPARATION	18
4.3. OPTIONAL: UPDATING THE OVN-CONTROLLER CONTAINER ON ALL OVERCLOUD SERVERS	18
4.4. UPDATING ALL CONTROLLER NODES	19
4.5. UPDATING ALL COMPUTE NODES	19
4.6. UPDATING ALL HCI COMPUTE NODES	20
4.7. UPDATING ALL CEPH STORAGE NODES	21
4.8. PERFORMING ONLINE DATABASE UPDATES	22
4.9. FINALIZING THE UPDATE	23
CHAPTER 5. REBOOTING THE OVERCLOUD	24
5.1. REBOOTING CONTROLLER AND COMPOSABLE NODES	24
5.2. REBOOTING A CEPH STORAGE (OSD) CLUSTER	24
5.3. REBOOTING COMPUTE NODES	26

MAKING OPEN SOURCE MORE INCLUSIVE

Red Hat is committed to replacing problematic language in our code, documentation, and web properties. We are beginning with these four terms: master, slave, blacklist, and whitelist. Because of the enormity of this endeavor, these changes will be implemented gradually over several upcoming releases. For more details, see [our CTO Chris Wright's message](#).

PROVIDING FEEDBACK ON RED HAT DOCUMENTATION

We appreciate your input on our documentation. Tell us how we can make it better.

Using the Direct Documentation Feedback (DDF) function

Use the **Add Feedback** DDF function for direct comments on specific sentences, paragraphs, or code blocks.

1. View the documentation in the *Multi-page HTML* format.
2. Ensure that you see the **Feedback** button in the upper right corner of the document.
3. Highlight the part of text that you want to comment on.
4. Click **Add Feedback**.
5. Complete the **Add Feedback** field with your comments.
6. Optional: Add your email address so that the documentation team can contact you for clarification on your issue.
7. Click **Submit**.

CHAPTER 1. INTRODUCTION

This document provides a workflow to help keep your Red Hat OpenStack Platform 16.1 environment updated with the latest packages and containers.

This guide provides an upgrade path through the following versions:

Old OpenStack Version	New OpenStack Version
Red Hat OpenStack Platform 16.0	Red Hat OpenStack Platform 16.1.z
Red Hat OpenStack Platform 16.1	Red Hat OpenStack Platform 16.1.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 16.1.z version.
Updating the overcloud	Update the overcloud to the latest OpenStack Platform 16.1.z version.
Updating the Ceph Storage nodes	Upgrade all Ceph Storage services.
Finalize the upgrade	Run the convergence command to refresh your overcloud stack.

If you have a multistack infrastructure, update each overcloud stack completely, one at a time. If you have a distributed compute node (DCN) infrastructure, update the overcloud at the central location completely, and then update the overcloud at each edge site, one at a time.

Considerations before you update your RHOSP environment

To help guide you during the update process, consider the following information:

- Red Hat recommends backing up the undercloud and overcloud control planes. For more information about backing up nodes, see [Backing up and restoring the undercloud and control plane nodes](#).
- Familiarize yourself with the known issues that might block an update.
- To identify your current maintenance release, run **\$ cat /etc/rhosp-release**. You can also run this command after updating your environment to validate the update.

1.2. KNOWN ISSUES THAT MIGHT BLOCK AN UPDATE

Review the following known issues that might affect a successful minor version update.

[BZ#1973660 - \(update\) from 16.1 to 16.2 breaks trying to configure the rabbitmq service.](#)

Overcloud nodes that run Pacemaker version **2.0.3-5.el8_2.4** might fail to update successfully because of a race condition that occurs when shutting down the cluster on a node.

If Pacemaker version **2.0.3-5.el8_2.4** is currently installed on any of the overcloud nodes, to avoid [BZ#1973660](#), you must upgrade Pacemaker before you can update the overcloud nodes. For more information, see the following Red Hat Knowledgebase solution [Update from OSP16.1 to OSP16.2 might fail to update certain HA containers.](#)

[BZ#1872404 - restarting nodes in parallel while maintaining quorum creates an unexpected node shutdown](#)

+For nodes that run RHEL 8.2, and are based on composable roles, you must update the **Database** role first before you can update any other role.

[BZ#2117179 - minor updates from \$\leq\$ OSP 16.1.6 causes Keystone LDAP connections to fail if old OSP13 configs at /etc/openldap are present](#)

A known issue causes LDAP connections to fail after an update from release 16.1.6 or earlier, to 16.1.7 or higher. With RHOSP 16.1.7, the Identity service (keystone) container mounts **/etc/openldap** on the host file system. If you have previously updated from RHOSP 13, old configuration files might be present in the **/etc/openldap** directory that will cause Identity service LDAP connection to fail.

As a workaround, run the following command on each of the controllers:

```
$ sudo cp /etc/openldap/ldap.conf.rpmnew /etc/openldap/ldap.conf
$ sudo podman restart keystone
```

[BZ#2158626 - Provider patch port is re-created on restart](#)

Network traffic that traverses from the provider network is disrupted during the ovn-controller restart if you meet the following conditions:

- Your environment has a provider network connected by floating IP or direct port on the provider network
- You update to any 16.1 release

The downtime can vary depending on the number of existing workloads. To avoid downtime, apply the workaround in Red Hat Knowledgebase solution [Data plane disruption during update/upgrade from OSP older than 16.2.2, including all 16.1 versions.](#)

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 16.1** to the latest minor release.

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

Red Hat OpenStack Platform 16.1 is supported on Red Hat Enterprise Linux 8.2. Prior to performing the update, lock the undercloud and overcloud repositories to the Red Hat Enterprise Linux 8.2 release to avoid upgrading the operating system to a newer minor release.

Procedure

1. Log in to 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 your subscription management configuration for the **rhsm_release** parameter. If this parameter is not set, add this parameter and set the parameter to 8.2:

```
parameter_defaults:
  RhsmVars:
    ...
    rhsm_username: "myusername"
    rhsm_password: "p@55w0rd!"
    rhsm_org_id: "1234567"
    rhsm_pool_ids: "1a85f9223e3d5e43013e3d6e8ff506fd"
    rhsm_method: "portal"
    rhsm_release: "8.2"
```

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 lock the operating system version to Red Hat Enterprise Linux 8.2 on all nodes:

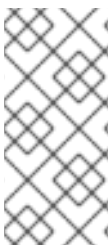
```
$ cat > ~/set_release.yaml <<'EOF'
- hosts: all
  gather_facts: false
  tasks:
    - name: set release to 8.2
```

```
command: subscription-manager release --set=8.2
become: true
EOF
```

8. Run the **set_release.yaml** playbook:

```
$ ansible-playbook -i ~/inventory.yaml -f 25 ~/set_release.yaml --limit <undercloud>,
<Controller>,<Compute>
```

- Use the **--limit** option to apply the content to all RHOSP nodes. Replace **<undercloud>**, **<Controller>**, **<Compute>** with the Ansible groups in your environment that contain those nodes.
- You cannot run this playbook against Ceph Storage nodes if you are using a different subscription for these nodes.



NOTE

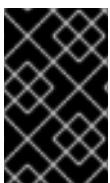
To manually lock a node to a version, log in to the node and run the **subscription-manager release** command:

```
$ sudo subscription-manager release --set=8.2
```

2.2. CHANGING FROM EUS REPOSITORIES TO TUS REPOSITORIES

Your Red Hat OpenStack Platform subscription includes repositories for Red Hat Enterprise Linux 8.2 Extended Update Support (EUS). After April 30, 2022, you must enable the RHEL 8.2 Telecommunications Update Service (TUS) repositories for Maintenance Support. The TUS repositories include the latest security patches and bug fixes for Red Hat Enterprise Linux 8.2. Switch to the following repositories before performing an update.

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



IMPORTANT

You must use TUS repositories to retain compatibility with a specific version of Podman. Later versions of Podman are untested for the Red Hat Open Stack Platform 16.1 release and can cause unexpected results.

Prerequisites

- RHOSP 16.1 EUS subscription

Procedure

1. Log in to 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 TUS repositories, change the relevant repositories to the TUS versions:

```
parameter_defaults:
  RhsmVars:
    rhsm_repos:
      - rhel-8-for-x86_64-baseos-tus-rpms
      - rhel-8-for-x86_64-appstream-tus-rpms
      - rhel-8-for-x86_64-highavailability-tus-rpms
      - ansible-2.9-for-rhel-8-x86_64-rpms
      - advanced-virt-for-rhel-8-x86_64-rpms
      - openstack-16.1-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 that is different from the default overcloud name of **overcloud**, set the name of your overcloud by using the **--plan** option.

7. Create a playbook that contains a task to set the repositories to Red Hat Enterprise Linux 8.2 TUS on all nodes:

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

8. Run the **change_tus.yaml** playbook:

```
$ ansible-playbook -i ~/inventory.yaml -f 25 ~/change_tus.yaml --limit <undercloud>,
<Controller>,<Compute>
```

- Use the **--limit** option to apply the content to all RHOSP nodes. Replace **<undercloud>**, **<Controller>**, **<Compute>** with the Ansible groups in your environment that contain those nodes.
- You cannot run this playbook against Ceph Storage nodes if you are using a different subscription for these nodes.

2.3. UPDATING RED HAT OPENSTACK PLATFORM AND ANSIBLE REPOSITORIES

Update your repositories to use Red Hat OpenStack Platform 16.1 and Ansible 2.9 packages.

Procedure

1. Log in to 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 the **rhsm_repos** parameter is using the Red Hat OpenStack Platform 16.0 and Ansible 2.8 repositories, change the repository to the correct versions:

```
parameter_defaults:
  RhsmVars:
    rhsm_repos:
      - rhel-8-for-x86_64-baseos-tus-rpms
      - rhel-8-for-x86_64-appstream-tus-rpms
      - rhel-8-for-x86_64-highavailability-tus-rpms
      - ansible-2.9-for-rhel-8-x86_64-rpms
      - advanced-virt-for-rhel-8-x86_64-rpms
      - openstack-16.1-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 that is different from the default overcloud name of **overcloud**, set the name of your overcloud by using the **--plan** option.

7. Create a playbook that contains a task to set the repositories to Red Hat OpenStack Platform 16.1 on all nodes:

```
$ cat > ~/update_rhosp_repos.yaml <<'EOF'
- hosts: all
  gather_facts: false
```

```

tasks:
  - name: change osp repos
    command: subscription-manager repos --disable=openstack-16-for-rhel-8-x86_64-rpms --
enable=openstack-16.1-for-rhel-8-x86_64-rpms --disable=ansible-2.8-for-rhel-8-x86_64-rpms
--enable=ansible-2.9-for-rhel-8-x86_64-rpms
    become: true
EOF

```

8. Run the **update_rhosp_repos.yaml** playbook:

```
$ ansible-playbook -i ~/inventory.yaml -f 25 ~/update_rhosp_repos.yaml --limit <undercloud>,
<Controller>,<Compute>
```

- Use the **--limit** option to apply the content to all RHOSP nodes. Replace **<undercloud>**, **<Controller>**, **<Compute>** with the Ansible groups in your environment that contain those nodes.
 - You cannot run this playbook against Ceph Storage nodes if you are using a different subscription for these nodes.
9. Create a playbook that contains a task to set the repositories to Red Hat OpenStack Platform 16.1 on all nodes:

```
$ cat > ~/update_ceph_repos.yaml <<'EOF'
- hosts: all
  gather_facts: false
  tasks:
    - name: change ceph repos
      command: subscription-manager repos --disable=openstack-16-deployment-tools-for-
rhel-8-x86_64-rpms --enable=openstack-16.1-deployment-tools-for-rhel-8-x86_64-rpms --
disable=ansible-2.8-for-rhel-8-x86_64-rpms --enable=ansible-2.9-for-rhel-8-x86_64-rpms
      become: true
EOF

```

10. Run the **update_ceph_repos.yaml** playbook:

```
$ ansible-playbook -i ~/inventory.yaml -f 25 ~/update_ceph_repos.yaml --limit CephStorage
```

Use the **--limit** option to apply the content to Ceph Storage nodes.

2.4. SETTING THE CONTAINER-TOOLS VERSION

Set the **container-tools** module to version **2.0** to ensure you use the correct package versions on all nodes.

Procedure

1. Log in to 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 set the **container-tools** module to version **2.0** on all nodes:

```
$ cat > ~/container-tools.yaml <<'EOF'
- hosts: all
  gather_facts: false
  tasks:
    - name: disable default dnf module for container-tools
      command: dnf module reset -y container-tools
      become: true
    - name: set dnf module for container-tools:2.0
      command: dnf module enable -y container-tools:2.0
      become: true
EOF
```

5. Run the **container-tools.yaml** playbook against all nodes:

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

2.5. UPDATING YOUR CONTAINER IMAGE PREPARATION FILE

Your container preparation file is the file that contains the **ContainerImagePrepare** parameter. You use this file to define the rules for obtaining container images for the undercloud and overcloud. Before you update your environment, check the file to ensure you obtain the correct image versions.

Procedure

1. Edit the container preparation file. The default name for this file is usually **containers-prepare-parameter.yaml**.
2. Check the **tag** parameter is set to **16.1** for each rule set:

```
parameter_defaults:
  ContainerImagePrepare:
    - push_destination: true
      set:
        ...
        tag: '16.1'
        tag_from_label: '{version}-{release}'
```

NOTE

If you do not want to use a specific tag for the update, such as **16.1** or **16.1.2**, remove the **tag** key-value pair and specify **tag_from_label** only. This will use the installed Red Hat OpenStack Platform version when determining the value for the tag to use as part of the update process.

1. Save this file.

2.6. UPDATING YOUR SSL/TLS CONFIGURATION

Remove the **NodeTLSData** resource from the **resource_registry** to update your SSL/TLS configuration.

Procedure

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

```
$ source ~/stackrc
```

3. Edit your custom overcloud SSL/TLS public endpoint file, which is usually named **~/templates/enable-tls.yaml**.
4. Remove the **NodeTLSData** resource from the `resource_registry`:

```
resource_registry:
  OS::TripleO::NodeTLSData: /usr/share/openstack-tripleo-heat-
  templates/puppet/extraconfig/tls/tls-cert-inject.yaml
  ...
```

The overcloud deployment uses a new service in HAProxy to determine if SSL/TLS is enabled.



NOTE

If this is the only resource in the **resource_registry** section of the **enable-tls.yaml** file, remove the complete **resource_registry** section.

5. Save the SSL/TLS public endpoint file file.

2.7. DISABLING FENCING IN THE OVERCLOUD

Before you update the overcloud, ensure that fencing is disabled.

If fencing is deployed in your environment during the Controller nodes update process, the overcloud might detect certain nodes as disabled and attempt fencing operations, which can cause unintended results.

If you have enabled fencing in the overcloud, you must temporarily disable fencing for the duration of the update to avoid any unintended results.

Procedure

1. Log in to the undercloud as the **stack** user.
2. Source the **stackrc** file.

```
$ source ~/stackrc
```

3. Log in to a Controller node and run the Pacemaker command to disable fencing:

```
┆ $ ssh heat-admin@CONTROLLER_IP "sudo pcs property set stonith-enabled=false"
```

4. In the **fencing.yaml** environment file, set the **EnableFencing** parameter to **false** to ensure that fencing stays disabled during the update process.

Additional Resources

- [Fencing Controller nodes with STONITH](#)

CHAPTER 3. UPDATING THE UNDERCLOUD

This process updates the undercloud and its overcloud images to the latest **Red Hat OpenStack Platform 16.1** version.

3.1. PERFORMING A MINOR UPDATE OF A CONTAINERIZED UNDERCLOUD

Director provides commands to update the main 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 in to the director as the **stack** user.

2. Run **dnf** to upgrade the director main packages:

```
$ sudo dnf update -y python3-tripleoclient* tripleo-ansible 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-16.1.tar /usr/share/rhosp-director-images/ironic-python-agent-latest-16.1.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 you deploy overcloud nodes, ensure that the overcloud image version corresponds to the respective heat template version. For example, use only the RHOSP 16.1 images with the RHOSP 16.1 heat templates.
- If you deployed a connected environment that uses the Red Hat Customer Portal or Red Hat Satellite Server, the overcloud image and package repository versions might be out of sync. To ensure that the overcloud image and package repository versions match, you can use the **virt-customize** tool. For more information, see the Red Hat Knowledgebase solution [Modifying the Red Hat Linux OpenStack Platform Overcloud Image with virt-customize](#).
- 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 you are using a local set of core templates in your **stack** users home directory, ensure that you update the templates by 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.

CHAPTER 4. UPDATING THE OVERCLOUD

After you update the undercloud, you can update the overcloud by running the overcloud and container image preparation commands, updating your nodes, and running the **overcloud update converge** command. The control plane API is fully available during a minor update.

Prerequisites

- You have updated the undercloud to the latest version.

4.1. RUNNING THE OVERCLOUD UPDATE PREPARATION

To prepare the overcloud for the update process, run the **openstack overcloud update prepare** command, which performs the following tasks:

- Updates the overcloud plan to OpenStack Platform 16.1
- Prepares the nodes for the update

Prerequisites

- If you use a Ceph subscription and have configured director to use the **overcloud-minimal** image for Ceph storage nodes, you must ensure that in the **roles_data.yaml** role definition file, the **rhsm_enforce** parameter is set to **False**.
- If you rendered custom NIC templates, you must regenerate the templates with the updated version of the **openstack-tripleo-heat-templates** collection to avoid incompatibility with the overcloud version. For more information about custom NIC templates, see [Rendering default network interface templates for customization](#) in the *Advanced Overcloud Customization* guide.

Procedure

1. Source the **stackrc** file:

```
$ source ~/stackrc
```

2. Run the update preparation command:

```
$ openstack overcloud update prepare \
  --templates \
  --stack <stack_name> \
  -r <roles_data_file> \
  -n <network_data_file> \
  -e <environment_file> \
  -e <environment_file> \
  ...
```

Include the following options relevant to your environment:

- 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.
- If you use your own custom roles, include your custom roles (**<roles_data> file (-r)**).

- If you use custom networks, include your composable network (`_<network_data> file _ (-n)`).
 - If you deploy a high availability cluster, include the `--ntp-server` option in the update preparation command, or include the `NtpServer` parameter and value in your environment file.
 - Any custom configuration environment files (`-e`).
3. Wait until the update preparation completes.

4.2. RUNNING THE CONTAINER IMAGE PREPARATION

The overcloud requires the latest OpenStack Platform 16.1 container images before performing the update. This involves executing the `container_image_prepare` external update process. To execute this process, you must run the `openstack overcloud external-update run` command against tasks tagged with the `container_image_prepare` tag. These tasks perform the following actions:

- 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.



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 `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. OPTIONAL: UPDATING THE OVN-CONTROLLER CONTAINER ON ALL OVERCLOUD SERVERS

If you deployed your overcloud with the Modular Layer 2 Open Virtual Network mechanism driver (ML2/OVN), update the ovn-controller container to the latest RHOSP 16.1 version. The update occurs on every overcloud server that runs the ovn-controller container.



NOTE

If you are not using the default stack name, which is `overcloud`, set your stack name with the `--stack <stack_name>` option and replace `<stack_name>` with the name of your stack.

Procedure

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

```
$ source ~/stackrc
```

3. Run the `openstack overcloud external-update run` command against the tasks that have the `ovn` tag:

```
$ openstack overcloud external-update run --stack <stack_name> --tags ovn
```

4. Wait until the `ovn-controller` container update completes.

4.4. UPDATING ALL CONTROLLER NODES

This process updates all the Controller nodes to the latest OpenStack Platform 16.1 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. The control plane API is fully available during the minor update.



IMPORTANT

Until [BZ#1872404](#) is resolved, for nodes based on composable roles, you must update the **Database** role first, before you can update **Controller, Messaging, Compute, Ceph**, and other roles.



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
```

3. Wait until the Controller node update completes.

4.5. UPDATING ALL COMPUTE NODES

This process updates all Compute nodes to the latest OpenStack Platform 16.1 version. The process involves running the **openstack overcloud update run** command and including the **--limit Compute** option to restrict operations to the Compute nodes only.

Parallelization considerations

When you update a large number of Compute nodes, to improve performance, you can run the **openstack overcloud update run** command with the **--limit Compute** option in parallel on batches of 20 nodes. For example, if you have 80 Compute nodes in your deployment, you can run the following commands to update the Compute nodes in parallel:

```
$ openstack overcloud update run -y --limit 'Compute[0:19]' > update-compute-0-19.log 2>&1 &
$ openstack overcloud update run -y --limit 'Compute[20:39]' > update-compute-20-39.log 2>&1 &
$ openstack overcloud update run -y --limit 'Compute[40:59]' > update-compute-40-59.log 2>&1 &
$ openstack overcloud update run -y --limit 'Compute[60:79]' > update-compute-60-79.log 2>&1 &
```

The '**Compute[0:19]**', '**Compute[20:39]**', '**Compute[40:59]**', and '**Compute[60:79]**' way of partitioning the nodes space is random and you don't have control over which nodes are updated.

To update specific Compute nodes, list the nodes that you want to update in a batch separated by a comma:

```
$ openstack overcloud update run --limit <Compute0>,<Compute1>,<Compute2>,<Compute3>
```



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
```

3. Wait until the Compute node update completes.

4.6. 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 **--limit 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.

Prerequisites

- On a Ceph Monitor or Controller node that is running the **ceph-mon** service, check that the Red Hat Ceph Storage cluster status is healthy and the pg status is **active+clean**:

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

If the Ceph cluster is healthy, it returns a status of **HEALTH_OK**.

If the Ceph cluster status is unhealthy, it returns a status of **HEALTH_WARN** or **HEALTH_ERR**. For troubleshooting guidance, see the [Red Hat Ceph Storage 4 Troubleshooting Guide](#) .

Procedure

1. Source the **stackrc** file:

```
$ source ~/stackrc
```

2. Run the update command:

```
$ openstack overcloud update run --stack <stack_name> --limit ComputeHCI
```

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.

4.7. UPDATING ALL CEPH STORAGE NODES

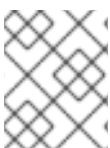
This process updates the Ceph Storage nodes. The process involves:

- Running the **openstack overcloud update run** command and including the **--limit 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 3 containers.



IMPORTANT

RHOSP 16.1 is supported on RHEL 8.2. However, hosts that are mapped to the Ceph Storage role update to the latest major RHEL release. For more information, see [Red Hat Ceph Storage: Supported configurations](#).



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.

Prerequisites

- On a Ceph Monitor or Controller node that is running the **ceph-mon** service, check that the Red Hat Ceph Storage cluster status is healthy and the pg status is **active+clean**:

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

If the Ceph cluster is healthy, it returns a status of **HEALTH_OK**.

If the Ceph cluster status is unhealthy, it returns a status of **HEALTH_WARN** or **HEALTH_ERR**. For troubleshooting guidance, see the [Red Hat Ceph Storage 4 Troubleshooting Guide](#) .

Procedure

1. Source the **stackrc** file:

```
$ source ~/stackrc
```

2. Update group nodes.

To update all nodes in a group:

```
$ openstack overcloud update run --limit <GROUP_NAME>
```

To update a single node in a group:

```
$ openstack overcloud update run --limit <GROUP_NAME> [NODE_INDEX]
```



NOTE

Ensure that you update all nodes if you choose to update nodes individually.

The index of the first node in a group is zero (0). For example, to update the first node in a group named **CephStorage**:

```
openstack overcloud update run --limit CephStorage[0]
```

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

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

5. Wait until the Ceph Storage container update completes.

4.8. 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 that use the **online_upgrade** tag:

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

4.9. 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 16.1 and allows you to perform standard **openstack overcloud deploy** functions in the future.

Procedure

1. Source the **stackrc** file:

```
$ source ~/stackrc
```

2. To re-enable fencing in your overcloud, in the **fencing.yaml** environment file, set the **EnableFencing** parameter to **true**.
3. Run the update finalization command:

```
$ openstack overcloud update converge \
  --templates \
  --stack <stack_name> \
  -r <roles_data_file> \
  -n <network_data_file> \
  -e <environment_file> \
  -e <environment_file> \
  ...
  ...
```

Include the following options relevant to your environment:

- The **fencing.yaml** environment file, with the **EnableFencing** parameter set to **true**.
 - 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.
 - 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**)
 - Any custom configuration environment files (**-e**).
4. 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.

Prerequisites

- On a Ceph Monitor or Controller node that is running the **ceph-mon** service, check that the Red Hat Ceph Storage cluster status is healthy and the pg status is **active+clean**:

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

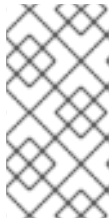
If the Ceph cluster is healthy, it returns a status of **HEALTH_OK**.

If the Ceph cluster status is unhealthy, it returns a status of **HEALTH_WARN** or **HEALTH_ERR**. For troubleshooting guidance, see the [Red Hat Ceph Storage 4 Troubleshooting Guide](#) .

Procedure

1. Log in to a Ceph Monitor or Controller node that is running the **ceph-mon** service, 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
```



NOTE

If you have a multistack or distributed compute node (DCN) architecture, you must specify the cluster name when you set the **noout** and **norebalance** flags. For example: **sudo podman exec -it ceph-mon-controller-0 ceph osd set noout --cluster <cluster_name>**

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

```
$ sudo reboot
```

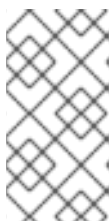
4. Wait until the node boots.
5. Log into the node and check the cluster status:

```
$ 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 in to a Ceph Monitor or Controller node that is running the **ceph-mon** service, 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
```



NOTE

If you have a multistack or distributed compute node (DCN) architecture, you must specify the cluster name when you unset the **noout** and **norebalance** flags. For example: **sudo podman exec -it ceph-mon-controller-0 ceph osd set noout --cluster <cluster_name>**

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.

Review the list of migration constraints that you might run into when migrating virtual machine instances between Compute nodes. For more information, see [Migration constraints](#) in *Configuring the Compute Service for Instance Creation*.

If you cannot 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 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

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
```

- List all instances on the Compute node:

```
(overcloud) $ openstack server list --host <hostname> --all-projects
```

- If you decide not to migrate instances, skip to [this step](#).
- 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.

- Wait until migration completes.
- Confirm that the migration was successful:

```
(overcloud) $ openstack server list --host <hostname> --all-projects
```

- Continue to migrate instances until none remain on the chosen Compute node.
- Log in to the Compute node and reboot the node:

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

- Wait until the node boots.
- Re-enable the Compute node:

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

- Check that the Compute node is enabled:

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

