OpenShift Container Platform 4.3

Updating clusters

Updating OpenShift Container Platform clusters
Updating OpenShift Container Platform clusters
Abstract

This document provides instructions for updating, or upgrading, OpenShift Container Platform clusters. In version , updating your cluster is a simple process that does not require you to take your cluster offline.
CHAPTER 1. UPDATING A CLUSTER BETWEEN MINOR VERSIONS

You can update, or upgrade, an OpenShift Container Platform cluster between minor versions.

NOTE

Because of the difficulty of changing update channels by using `oc`, use the web console to change the update channel. It is recommended to complete the update process within the web console. You can follow the steps in Updating a cluster within a minor version by using the CLI to complete the update after you change to a 4.3 channel.

Prerequisites

- Have access to the cluster as a user with `admin` privileges. See Using RBAC to define and apply permissions.
- Have a recent `etcd backup` in case your upgrade fails and you must restore your cluster to a previous state.

IMPORTANT

If you are upgrading to this release from OpenShift Container Platform 4.2, you must restart all Pods after the upgrade is complete.

This is because the service CA is automatically rotated as of OpenShift Container Platform 4.3.5. The service CA is rotated during the upgrade and a restart is required afterward to ensure that all services are using the new service CA before the previous service CA expires.

After this one-time manual restart, subsequent upgrades and rotations will ensure restart before the service CA expires without requiring manual intervention.

1.1. ABOUT THE OPENSHIFT CONTAINER PLATFORM UPDATE SERVICE

The OpenShift Container Platform update service is the hosted service that provides over-the-air updates to both OpenShift Container Platform and Red Hat Enterprise Linux CoreOS (RHCOS). It provides a graph, or diagram that contains vertices and the edges that connect them, of component Operators. The edges in the graph show which versions you can safely update to, and the vertices are update payloads that specify the intended state of the managed cluster components.

The Cluster Version Operator (CVO) in your cluster checks with the OpenShift Container Platform update service to see the valid updates and update paths based on current component versions and information in the graph. When you request an update, the OpenShift Container Platform CVO uses the release image for that update to upgrade your cluster. The release artifacts are hosted in Quay as container images.

To allow the OpenShift Container Platform update service to provide only compatible updates, a release verification pipeline exists to drive automation. Each release artifact is verified for compatibility with supported cloud platforms and system architectures as well as other component packages. After the pipeline confirms the suitability of a release, the OpenShift Container Platform update service notifies you that it is available.
Because the update service displays all valid updates, you must not force an update to a version that the update service does not display.

During continuous update mode, two controllers run. One continuously updates the payload manifests, applies them to the cluster, and outputs the status of the controlled rollout of the Operators, whether they are available, upgrading, or failed. The second controller polls the OpenShift Container Platform update service to determine if updates are available.

Reverting your cluster to a previous version, or a rollback, is not supported. Only upgrading to a newer version is supported.

During the upgrade process, the Machine Config Operator (MCO) applies the new configuration to your cluster machines. It cordons the number of nodes that is specified by the `maxUnavailable` field on the machine configuration pool and marks them as unavailable. By default, this value is set to 1. It then applies the new configuration and reboots the machine. If you use Red Hat Enterprise Linux (RHEL) machines as workers, the MCO does not update the kubelet on these machines because you must update the OpenShift API on them first. Because the specification for the new version is applied to the old kubelet, the RHEL machine cannot return to the Ready state. You cannot complete the update until the machines are available. However, the maximum number of nodes that are unavailable is set to ensure that normal cluster operations are likely to continue with that number of machines out of service.

1.2. UNDERSTANDING OPENSOURCE CONTAINER PLATFORM UPGRADE CHANNELS

In OpenShift Container Platform 4.1, Red Hat introduced the concept of upgrade channels for recommending the appropriate upgrade versions to your cluster. Upgrade channels separate upgrade strategies and also are used to control the cadence of updates. Channels are tied to a minor version of OpenShift Container Platform. For instance, OpenShift Container Platform 4.3 channels will never include an upgrade to a 4.4 release. This ensures administrators make an explicit decision to upgrade to the next minor version of OpenShift Container Platform. Channels only control updates and have no impact on the version of the cluster you install; the `openshift-install` binary for a given patch level of OpenShift Container Platform always installs that patch level.

OpenShift Container Platform 4.3, which includes the upgrade from the previous 4.2 release, has three upgrade channels to choose from:

- candidate-4.3
- fast-4.3
- stable-4.3

The upgrade channels contain two types of updates:

1. General Availability Software (or GA) - These versions of OpenShift Container Platform are fully supported and are considered production quality. You may upgrade to the general availability release from either of the fast and stable channels.

2. Release Candidate Software (or RC) - These versions of OpenShift Container Platform are representative of the eventual general availability release and are available only in the candidate-4.3 channel. The release candidate will contain all the features of the product. You
are allowed to upgrade from a release candidate to another release candidate and to upgrade from a previous minor version of OpenShift Container Platform to the current release candidate. Release candidate builds are not supported by Red Hat and you will not be able to upgrade from a release candidate to the general availability release of OpenShift Container Platform. Candidates should be used to test feature acceptance and assist in qualifying the next version of OpenShift Container Platform in your infrastructure.

**NOTE**

Release candidates differ from the nightly builds found on try.openshift.com. You cannot upgrade nightly builds to nightly builds. Nightly builds are available for early access to features but are not upgradable or supported.

For GA versions, the fast and stable channels present a choice between receiving updates as soon as they are available or allowing Red Hat to control the rollout of those updates.

**fast-4.3**
The fast channel is updated with new 4.3 patch versions as soon as Red Hat declares they are generally available. Use this channel if you wish to receive updates as soon as they are available or for your pre-production environments when participating in the connected customer program. This channel will contain all z-stream (4.3.z) updates but will not suggest upgrades to the next minor release (4.4.z) when the next minor release is available.

**stable-4.3**
The stable channel will contain updates on a time delay as they are gradually rolled out to customers based on data from our SRE teams, support services, and pre-production and production environments that participate in our connected customer program, rather than being immediately available as they are in the fast channel. For patch and CVE fixes this can range from several hours to a day and allows an extra period of assessment in how the software performs. If issues are detected during rollout, upgrades to that version may be blocked in both the fast and stable channels, and a new version may be introduced that will be the new preferred upgrade target.

Customers can improve this process by configuring pre-production systems on the fast channel, production systems on the stable channel, and participating in Red Hat's connected customer program — this allows Red Hat to observe the impact of updates on your specific hardware and software configurations. Future releases may improve or alter the pace at which updates move from the fast to the stable channels.

If issues are discovered with an upgrade between patch levels, Red Hat may withdraw that suggested upgrade for affected versions. A newer patch would become available in the appropriate channels and be suggested for upgrade.

**Upgrade version paths**
OpenShift Container Platform maintains an upgrade recommendation service that understands the version of OpenShift Container Platform you have installed as well as the path to take within the channel you choose to get you to the next release. You can imagine seeing the following in the fast-4.3 channel:

- 4.3.0
- 4.3.1
- 4.3.3
- 4.3.4

The service only recommends upgrades that have been tested and have no known issues. If you are on
4.3.1 and OpenShift Container Platform is allowing you to select 4.3.4, then it is safe for you to go from .4.3.1 to .4.3.4. Likewise, the absence of 4.3.2 may be due to a CVE that was fixed in 4.3.3 and Red Hat no longer suggests upgrading to a known vulnerable version. If an issue is found that results in a new version being retracted from the recommendations, Red Hat will release a new version that is capable of upgrading from all necessary versions, including the retracted version.

Disconnected clusters
Customers which have chosen to not be connected to Red Hat and are curating their own OpenShift Container Platform container image content manually should consult the Red Hat errata associated with product releases and note any comments impacting upgrades. During upgrade the user interface may caution about switching between these versions and it is up to the customer to ensure they have correctly selected the appropriate version before bypassing those cautions.

Switching between channels
It is supported for customers to switch between the fast and stable channel at any time. Channels only offer suggested upgrades, and will never suggest a dangerous upgrade. If you switch to the candidate channel after installing from a GA version, you will see a warning the current version is not recognized, and you can safely switch back to a GA channel.

1.3. UPDATING A CLUSTER BY USING THE WEB CONSOLE

If updates are available, you can update your cluster from the web console.

You can find information about available OpenShift Container Platform advisories and updates in the errata section of the Customer Portal.

Prerequisites
- Have access to the web console as a user with admin privileges.

Procedure
1. From the web console, click Administration > Cluster Settings and review the contents of the Overview tab.

2. For production clusters, ensure that the CHANNEL is set to the correct channel for your current minor version, such as stable-4.3.

   **IMPORTANT**

   For production clusters, you must subscribe to a stable-* or fast-* channel.

   - If the UPDATE STATUS is not Updates Available, you cannot upgrade your cluster.
   - The DESIRED VERSION indicates the cluster version that your cluster is running or is updating to.

3. Click Updates Available, select the highest available version and click Update. The UPDATE STATUS changes to Updating, and you can review the progress of the Operator upgrades on the Cluster Operators tab.

4. If you are upgrading to this release from OpenShift Container Platform 4.2, you must restart all Pods after the upgrade is complete. You can do this using the following command, which requires the OpenShift CLI (oc):
$ for I in $(oc get ns -o jsonpath='{range .items[*]} {.metadata.name}{"\n"}{end}');
do oc delete pods --all -n $I;
sleep 1;
done

NOTE

Restarting all Pods is required because the service CA is automatically rotated as of OpenShift Container Platform 4.3.5. The service CA is rotated during the upgrade and a restart is required afterward to ensure that all services are using the new service CA before the previous service CA expires.

After this one-time manual restart, subsequent upgrades and rotations will ensure restart before the service CA expires without requiring manual intervention.

5. After the update completes and the Cluster Version Operator refreshes the available updates, check if more updates are available in your current channel.

- If updates are available, continue to perform updates in the current channel until you can no longer update.
- If no updates are available, change the CHANNEL to the stable-* or fast-* channel for the next minor version, and update to the version that you want in that channel.

You might need to perform several intermediate updates until you reach the version that you want.
CHAPTER 2. UPDATING A CLUSTER WITHIN A MINOR VERSION FROM THE WEB CONSOLE

You can update, or upgrade, an OpenShift Container Platform cluster by using the web console.

Prerequisites

- Have access to the cluster as a user with admin privileges. See Using RBAC to define and apply permissions.
- Have a recent etcd backup in case your upgrade fails and you must restore your cluster to a previous state.

IMPORTANT

If you are upgrading to this release from OpenShift Container Platform 4.3.3 or earlier, you must restart all Pods after the upgrade is complete.

This is because the service CA is automatically rotated as of OpenShift Container Platform 4.3.5. The service CA is rotated during the upgrade and a restart is required afterward to ensure that all services are using the new service CA before the previous service CA expires.

After this one-time manual restart, subsequent upgrades and rotations will ensure restart before the service CA expires without requiring manual intervention.

2.1. ABOUT THE OPENSHIFT CONTAINER PLATFORM UPDATE SERVICE

The OpenShift Container Platform update service is the hosted service that provides over-the-air updates to both OpenShift Container Platform and Red Hat Enterprise Linux CoreOS (RHCOS). It provides a graph, or diagram that contain vertices and the edges that connect them, of component Operators. The edges in the graph show which versions you can safely update to, and the vertices are update payloads that specify the intended state of the managed cluster components.

The Cluster Version Operator (CVO) in your cluster checks with the OpenShift Container Platform update service to see the valid updates and update paths based on current component versions and information in the graph. When you request an update, the OpenShift Container Platform CVO uses the release image for that update to upgrade your cluster. The release artifacts are hosted in Quay as container images.

To allow the OpenShift Container Platform update service to provide only compatible updates, a release verification pipeline exists to drive automation. Each release artifact is verified for compatibility with supported cloud platforms and system architectures as well as other component packages. After the pipeline confirms the suitability of a release, the OpenShift Container Platform update service notifies you that it is available.

IMPORTANT

Because the update service displays all valid updates, you must not force an update to a version that the update service does not display.

During continuous update mode, two controllers run. One continuously updates the payload manifests,
applies them to the cluster, and outputs the status of the controlled rollout of the Operators, whether they are available, upgrading, or failed. The second controller polls the OpenShift Container Platform update service to determine if updates are available.

IMPORTANT

Reverting your cluster to a previous version, or a rollback, is not supported. Only upgrading to a newer version is supported.

During the upgrade process, the Machine Config Operator (MCO) applies the new configuration to your cluster machines. It cordons the number of nodes that is specified by the maxUnavailable field on the machine configuration pool and marks them as unavailable. By default, this value is set to 1. It then applies the new configuration and reboots the machine. If you use Red Hat Enterprise Linux (RHEL) machines as workers, the MCO does not update the kubelet on these machines because you must update the OpenShift API on them first. Because the specification for the new version is applied to the old kubelet, the RHEL machine cannot return to the Ready state. You cannot complete the update until the machines are available. However, the maximum number of nodes that are unavailable is set to ensure that normal cluster operations are likely to continue with that number of machines out of service.

2.2. UNDERSTANDING OPENSIFT CONTAINER PLATFORM UPGRADE CHANNELS

In OpenShift Container Platform 4.1, Red Hat introduced the concept of upgrade channels for recommending the appropriate upgrade versions to your cluster. Upgrade channels separate upgrade strategies and also are used to control the cadence of updates. Channels are tied to a minor version of OpenShift Container Platform. For instance, OpenShift Container Platform 4.3 channels will never include an upgrade to a 4.4 release. This ensures administrators make an explicit decision to upgrade to the next minor version of OpenShift Container Platform. Channels only control updates and have no impact on the version of the cluster you install; the openshift-install binary for a given patch level of OpenShift Container Platform always installs that patch level.

OpenShift Container Platform 4.3, which includes the upgrade from the previous 4.2 release, has three upgrade channels to choose from:

- candidate-4.3
- fast-4.3
- stable-4.3

The upgrade channels contain two types of updates:

1. General Availability Software (or GA) - These versions of OpenShift Container Platform are fully supported and are considered production quality. You may upgrade to the general availability release from either of the fast and stable channels.

2. Release Candidate Software (or RC) - These versions of OpenShift Container Platform are representative of the eventual general availability release and are available only in the candidate-4.3 channel. The release candidate will contain all the features of the product. You are allowed to upgrade from a release candidate to another release candidate and to upgrade from a previous minor version of OpenShift Container Platform to the current release candidate. Release candidate builds are not supported by Red Hat and you will not be able to upgrade from a release candidate to the general availability release of OpenShift Container Platform. Candidates should be used to test feature acceptance and assist in qualifying the next version of OpenShift Container Platform in your infrastructure.
For GA versions, the fast and stable channels present a choice between receiving updates as soon as they are available or allowing Red Hat to control the rollout of those updates.

**fast-4.3**
The fast channel is updated with new 4.3 patch versions as soon as Red Hat declares they are generally available. Use this channel if you wish to receive updates as soon as they are available or for your pre-production environments when participating in the connected customer program. This channel will contain all z-stream (4.3.z) updates but will not suggest upgrades to the next minor release (4.4.z) when the next minor release is available.

**stable-4.3**
The stable channel will contain updates on a time delay as they are gradually rolled out to customers based on data from our SRE teams, support services, and pre-production and production environments that participate in our connected customer program, rather than being immediately available as they are in the fast channel. For patch and CVE fixes this can range from several hours to a day and allows an extra period of assessment in how the software performs. If issues are detected during rollout, upgrades to that version may be blocked in both the fast and stable channels, and a new version may be introduced that will be the new preferred upgrade target.

Customers can improve this process by configuring pre-production systems on the fast channel, production systems on the stable channel, and participating in Red Hat’s connected customer program - this allows Red Hat to observe the impact of updates on your specific hardware and software configurations. Future releases may improve or alter the pace at which updates move from the fast to the stable channels.

If issues are discovered with an upgrade between patch levels, Red Hat may withdraw that suggested upgrade for affected versions. A newer patch would become available in the appropriate channels and be suggested for upgrade.

**Upgrade version paths**
OpenShift Container Platform maintains an upgrade recommendation service that understands the version of OpenShift Container Platform you have installed as well as the path to take within the channel you choose to get you to the next release. You can imagine seeing the following in the fast-4.3 channel:

- 4.3.0
- 4.3.1
- 4.3.3
- 4.3.4

The service only recommends upgrades that have been tested and have no known issues. If you are on 4.3.1 and OpenShift Container Platform is allowing you to select 4.3.4, then it is safe for you to go from 4.3.1 to 4.3.4. Likewise, the absence of 4.3.2 may be due to a CVE that was fixed in 4.3.3 and Red Hat no longer suggests upgrading to a known vulnerable version. If an issue is found that results in a new version being retracted from the recommendations, Red Hat will release a new version that is capable of upgrading from all necessary versions, including the retracted version.

**Disconnected clusters**
Customers which have chosen to not be connected to Red Hat and are curating their own OpenShift Container Platform container image content manually should consult the Red Hat errata associated with product releases and note any comments impacting upgrades. During upgrade the user interface may caution about switching between these versions and it is up to the customer to ensure they have correctly selected the appropriate version before bypassing those cautions.

**Switching between channels**
It is supported for customers to switch between the fast and stable channel at any time. Channels only offer suggested upgrades, and will never suggest a dangerous upgrade. If you switch to the candidate channel after installing from a GA version, you will see a warning the current version is not recognized, and you can safely switch back to a GA channel.

### 2.3. UPDATING A CLUSTER BY USING THE WEB CONSOLE

If updates are available, you can update your cluster from the web console.

You can find information about available OpenShift Container Platform advisories and updates in the errata section of the Customer Portal.

**Prerequisites**
- Have access to the web console as a user with admin privileges.

**Procedure**

1. From the web console, click Administration > Cluster Settings and review the contents of the Overview tab.

2. For production clusters, ensure that theannel is set to the correct channel for the version that you want to update to, your current minor version, such as stable-4.3.

   **IMPORTANT**
   
   For production clusters, you must subscribe to a stable-* or fast-* channel.

   - If the UPDATE STATUS is not Updates Available, you cannot upgrade your cluster.

   - The DESIRED VERSION indicates the cluster version that your cluster is running or is updating to.

3. Click Updates Available, select a version to update to, the highest available version and click Update. The UPDATE STATUS changes to Updating, and you can review the progress of the Operator upgrades on the Cluster Operators tab.

4. If you are upgrading to this release from OpenShift Container Platform 4.3.3 or earlier, 4.2, you must restart all Pods after the upgrade is complete. You can do this using the following command, which requires the OpenShift CLI (oc):

   ```
   $ for I in $(oc get ns -o jsonpath='{range .items[*]} {.metadata.name}{"\n"} {end}');
   do oc delete pods --all -n $I; sleep 1; done
   ```
NOTE

Restarting all Pods is required because the service CA is automatically rotated as of OpenShift Container Platform 4.3.5. The service CA is rotated during the upgrade and a restart is required afterward to ensure that all services are using the new service CA before the previous service CA expires.

After this one-time manual restart, subsequent upgrades and rotations will ensure restart before the service CA expires without requiring manual intervention.

5. After the update completes and the Cluster Version Operator refreshes the available updates, check if more updates are available in your current channel.

   - If updates are available, continue to perform updates in the current channel until you can no longer update.
   
   - If no updates are available, change the CHANNEL to the stable-* or fast-* channel for the next minor version, and update to the version that you want in that channel.

You might need to perform several intermediate updates until you reach the version that you want.
You can update, or upgrade, an OpenShift Container Platform cluster within a minor version by using the OpenShift CLI (oc).

**Prerequisites**

- Have access to the cluster as a user with admin privileges. See Using RBAC to define and apply permissions.
- Have a recent etcd backup in case your upgrade fails and you must restore your cluster to a previous state.

**IMPORTANT**

If you are upgrading to this release from OpenShift Container Platform 4.3.3 or earlier, you must restart all Pods after the upgrade is complete.

This is because the service CA is automatically rotated as of OpenShift Container Platform 4.3.5. The service CA is rotated during the upgrade and a restart is required afterward to ensure that all services are using the new service CA before the previous service CA expires.

After this one-time manual restart, subsequent upgrades and rotations will ensure restart before the service CA expires without requiring manual intervention.

### 3.1. ABOUT THE OPENSHIFT CONTAINER PLATFORM UPDATE SERVICE

The OpenShift Container Platform update service is the hosted service that provides over-the-air updates to both OpenShift Container Platform and Red Hat Enterprise Linux CoreOS (RHCOS). It provides a graph, or diagram that contain vertices and the edges that connect them, of component Operators. The edges in the graph show which versions you can safely update to, and the vertices are update payloads that specify the intended state of the managed cluster components.

The Cluster Version Operator (CVO) in your cluster checks with the OpenShift Container Platform update service to see the valid updates and update paths based on current component versions and information in the graph. When you request an update, the OpenShift Container Platform CVO uses the release image for that update to upgrade your cluster. The release artifacts are hosted in Quay as container images.

To allow the OpenShift Container Platform update service to provide only compatible updates, a release verification pipeline exists to drive automation. Each release artifact is verified for compatibility with supported cloud platforms and system architectures as well as other component packages. After the pipeline confirms the suitability of a release, the OpenShift Container Platform update service notifies you that it is available.

**IMPORTANT**

Because the update service displays all valid updates, you must not force an update to a version that the update service does not display.
During continuous update mode, two controllers run. One continuously updates the payload manifests, applies them to the cluster, and outputs the status of the controlled rollout of the Operators, whether they are available, upgrading, or failed. The second controller polls the OpenShift Container Platform update service to determine if updates are available.

**IMPORTANT**

Reverting your cluster to a previous version, or a rollback, is not supported. Only upgrading to a newer version is supported.

During the upgrade process, the Machine Config Operator (MCO) applies the new configuration to your cluster machines. It cordons the number of nodes that is specified by the `maxUnavailable` field on the machine configuration pool and marks them as unavailable. By default, this value is set to 1. It then applies the new configuration and reboots the machine. If you use Red Hat Enterprise Linux (RHEL) machines as workers, the MCO does not update the kubelet on these machines because you must update the OpenShift API on them first. Because the specification for the new version is applied to the old kubelet, the RHEL machine cannot return to the **Ready** state. You cannot complete the update until the machines are available. However, the maximum number of nodes that are unavailable is set to ensure that normal cluster operations are likely to continue with that number of machines out of service.

### 3.2. UNDERSTANDING OPENSHIFT CONTAINER PLATFORM UPGRADE CHANNELS

In OpenShift Container Platform 4.1, Red Hat introduced the concept of upgrade channels for recommending the appropriate upgrade versions to your cluster. Upgrade channels separate upgrade strategies and also are used to control the cadence of updates. Channels are tied to a minor version of OpenShift Container Platform. For instance, OpenShift Container Platform 4.3 channels will never include an upgrade to a 4.4 release. This ensures administrators make an explicit decision to upgrade to the next minor version of OpenShift Container Platform. Channels only control updates and have no impact on the version of the cluster you install; the `openshift-install` binary for a given patch level of OpenShift Container Platform always installs that patch level.

OpenShift Container Platform 4.3, which includes the upgrade from the previous 4.2 release, has three upgrade channels to choose from:

- candidate-4.3
- fast-4.3
- stable-4.3

The upgrade channels contain two types of updates:

1. **General Availability Software (or GA)** - These versions of OpenShift Container Platform are fully supported and are considered production quality. You may upgrade to the general availability release from either of the fast and stable channels.

2. **Release Candidate Software (or RC)** - These versions of OpenShift Container Platform are representative of the eventual general availability release and are available only in the candidate-4.3 channel. The release candidate will contain all the features of the product. You are allowed to upgrade from a release candidate to another release candidate and to upgrade from a previous minor version of OpenShift Container Platform to the current release candidate. Release candidate builds are not supported by Red Hat and you will not be able to
For GA versions, the fast and stable channels present a choice between receiving updates as soon as they are available or allowing Red Hat to control the rollout of those updates.

**fast-4.3**
The fast channel is updated with new 4.3 patch versions as soon as Red Hat declares they are generally available. Use this channel if you wish to receive updates as soon as they are available or for your pre-production environments when participating in the connected customer program. This channel will contain all z-stream (4.3.z) updates but will not suggest upgrades to the next minor release (4.4.z) when the next minor release is available.

**stable-4.3**
The stable channel will contain updates on a time delay as they are gradually rolled out to customers based on data from our SRE teams, support services, and pre-production and production environments that participate in our connected customer program, rather than being immediately available as they are in the fast channel. For patch and CVE fixes this can range from several hours to a day and allows an extra period of assessment in how the software performs. If issues are detected during rollout, upgrades to that version may be blocked in both the fast and stable channels, and a new version may be introduced that will be the new preferred upgrade target.

Customers can improve this process by configuring pre-production systems on the fast channel, production systems on the stable channel, and participating in Red Hat’s connected customer program - this allows Red Hat to observe the impact of updates on your specific hardware and software configurations. Future releases may improve or alter the pace at which updates move from the fast to the stable channels.

If issues are discovered with an upgrade between patch levels, Red Hat may withdraw that suggested upgrade for affected versions. A newer patch would become available in the appropriate channels and be suggested for upgrade.

**Upgrade version paths**
OpenShift Container Platform maintains an upgrade recommendation service that understands the version of OpenShift Container Platform you have installed as well as the path to take within the channel you choose to get you to the next release. You can imagine seeing the following in the fast-4.3 channel:

- 4.3.0
- 4.3.1
- 4.3.3
- 4.3.4

The service only recommends upgrades that have been tested and have no known issues. If you are on 4.3.1 and OpenShift Container Platform is allowing you to select 4.3.4, then it is safe for you to go from 4.3.1 to 4.3.4. Likewise, the absence of 4.3.2 may be due to a CVE that was fixed in 4.3.3 and Red Hat
no longer suggests upgrading to a known vulnerable version. If an issue is found that results in a new version being retracted from the recommendations, Red Hat will release a new version that is capable of upgrading from all necessary versions, including the retracted version.

Disconnected clusters
Customers which have chosen to not be connected to Red Hat and are curating their own OpenShift Container Platform container image content manually should consult the Red Hat errata associated with product releases and note any comments impacting upgrades. During upgrade the user interface may caution about switching between these versions and it is up to the customer to ensure they have correctly selected the appropriate version before bypassing those cautions.

Switching between channels
It is supported for customers to switch between the fast and stable channel at any time. Channels only offer suggested upgrades, and will never suggest a dangerous upgrade. If you switch to the candidate channel after installing from a GA version, you will see a warning the current version is not recognized, and you can safely switch back to a GA channel.

3.3. UPDATING A CLUSTER BY USING THE CLI

If updates are available, you can update your cluster by using the OpenShift CLI (oc).

You can find information about available OpenShift Container Platform advisories and updates in the errata section of the Customer Portal.

Prerequisites

- Install the version of the OpenShift Command-line Interface (CLI), commonly known as oc, that matches the version for your updated version.
- Log in to the cluster as user with cluster-admin privileges.
- Install the jq package.

Procedure

1. Ensure that your cluster is available:

   
   $ oc get clusterversion

   NAME VERSION AVAILABLE PROGRESSING SINCE STATUS
   version 4.3.0 True False 158m Cluster version is 4.3.0

2. Review the current update channel information and confirm that your channel is set to stable-4.3:

   
   $ oc get clusterversion -o json|jq ".items[0].spec"

   
   {
   "channel": "stable-4.3",
   "clusterID": "990f7ab8-109b-4c95-8480-2bd1deec55ff",
   "upstream": "https://api.openshift.com/api/upgrades_info/v1/graph"
   }
For production clusters, you must subscribe to a stable-* channel.

3. View the available updates and note the version number of the update that you want to apply:

$ oc adm upgrade

Cluster version is 4.1.0

Updates:

**VERSION IMAGE**

<table>
<thead>
<tr>
<th>VERSION</th>
<th>IMAGE</th>
<th>SHA256</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1.2</td>
<td>quay.io/openshift-release-dev/ocp-release@sha256:9c5f0df8b192a0d7b46cd5f6a4da2289c155fd5302dec7954f8f06c878160b8b</td>
<td></td>
</tr>
</tbody>
</table>

4. Apply an update:

- To update to the latest version:
  
  $ oc adm upgrade --to-latest=true

- To update to a specific version:
  
  $ oc adm upgrade --to=<version>

**<version>** is the update version that you obtained from the output of the previous command.

5. Review the status of the Cluster Version Operator:

$ oc get clusterversion -o json | jq ".items[0].spec"

```json
{
  "channel": "stable-4.3",
  "clusterID": "990f7ab8-109b-4c95-8480-2bd1dec55ff",
  "desiredUpdate": {
    "force": false,
    "image": "quay.io/openshift-release-dev/ocp-release@sha256:9c5f0df8b192a0d7b46cd5f6a4da2289c155fd5302dec7954f8f06c878160b8b",
    "version": "4.3.1"
  },
  "upstream": "https://api.openshift.com/api/upgrades_info/v1/graph"
}
```

1 If the **version** number in the **desiredUpdate** stanza matches the value that you specified, the update is in progress.

6. Review the cluster version status history to monitor the status of the update. It might take some time for all the objects to finish updating.
The history contains a list of the most recent versions applied to the cluster. This value is updated when the CVO applies an update. The list is ordered by date, where the newest update is first in the list. Updates in the history have state **Completed** if the rollout completed and **Partial** if the update failed or did not complete.

**IMPORTANT**

If an upgrade fails, the Operator stops and reports the status of the failing component. Rolling your cluster back to a previous version is not supported. If your upgrade fails, contact Red Hat support.

7. After the update completes, you can confirm that the cluster version has updated to the new version:

```
$ oc get clusterversion
```

<table>
<thead>
<tr>
<th>NAME</th>
<th>VERSION</th>
<th>AVAILABLE</th>
<th>PROGRESSING</th>
<th>SINCE</th>
<th>STATUS</th>
</tr>
</thead>
<tbody>
<tr>
<td>version</td>
<td>4.1.2</td>
<td>True</td>
<td>False</td>
<td>2m</td>
<td>Cluster version is 4.1.2</td>
</tr>
</tbody>
</table>

8. If you are upgrading to this release from OpenShift Container Platform 4.3.3 or earlier, you must restart all Pods after the upgrade is complete. You can do this using the following command:

```
$ for I in $(oc get ns -o jsonpath='{range .items[*]} {.metadata.name}{"\n"}{end}'); do oc delete pods --all -n $I; sleep 1; done
```
NOTE

Restarting all Pods is required because the service CA is automatically rotated as of OpenShift Container Platform 4.3.5. The service CA is rotated during the upgrade and a restart is required afterward to ensure that all services are using the new service CA before the previous service CA expires.

After this one-time manual restart, subsequent upgrades and rotations will ensure restart before the service CA expires without requiring manual intervention.
CHAPTER 4. UPDATING A CLUSTER THAT INCLUDES RHEL COMPUTE MACHINES

You can update, or upgrade, an OpenShift Container Platform cluster. If your cluster contains Red Hat Enterprise Linux (RHEL) machines, you must perform more steps to update those machines.

Prerequisites

- Have access to the cluster as a user with admin privileges. See Using RBAC to define and apply permissions.
- Have a recent etcd backup in case your upgrade fails and you must restore your cluster to a previous state.

IMPORTANT

If you are upgrading to this release from OpenShift Container Platform 4.2, you must restart all Pods after the upgrade is complete.

This is because the service CA is automatically rotated as of OpenShift Container Platform 4.3.5. The service CA is rotated during the upgrade and a restart is required afterward to ensure that all services are using the new service CA before the previous service CA expires.

After this one-time manual restart, subsequent upgrades and rotations will ensure restart before the service CA expires without requiring manual intervention.

4.1. ABOUT THE OPENSHIFT CONTAINER PLATFORM UPDATE SERVICE

The OpenShift Container Platform update service is the hosted service that provides over-the-air updates to both OpenShift Container Platform and Red Hat Enterprise Linux CoreOS (RHCOS). It provides a graph, or diagram that contain vertices and the edges that connect them, of component Operators. The edges in the graph show which versions you can safely update to, and the vertices are update payloads that specify the intended state of the managed cluster components.

The Cluster Version Operator (CVO) in your cluster checks with the OpenShift Container Platform update service to see the valid updates and update paths based on current component versions and information in the graph. When you request an update, the OpenShift Container Platform CVO uses the release image for that update to upgrade your cluster. The release artifacts are hosted in Quay as container images.

To allow the OpenShift Container Platform update service to provide only compatible updates, a release verification pipeline exists to drive automation. Each release artifact is verified for compatibility with supported cloud platforms and system architectures as well as other component packages. After the pipeline confirms the suitability of a release, the OpenShift Container Platform update service notifies you that it is available.

IMPORTANT

Because the update service displays all valid updates, you must not force an update to a version that the update service does not display.
During continuous update mode, two controllers run. One continuously updates the payload manifests, applies them to the cluster, and outputs the status of the controlled rollout of the Operators, whether they are available, upgrading, or failed. The second controller polls the OpenShift Container Platform update service to determine if updates are available.

**IMPORTANT**

Reverting your cluster to a previous version, or a rollback, is not supported. Only upgrading to a newer version is supported.

During the upgrade process, the Machine Config Operator (MCO) applies the new configuration to your cluster machines. It cordons the number of nodes that is specified by the `maxUnavailable` field on the machine configuration pool and marks them as unavailable. By default, this value is set to 1. It then applies the new configuration and reboots the machine. If you use Red Hat Enterprise Linux (RHEL) machines as workers, the MCO does not update the kubelet on these machines because you must update the OpenShift API on them first. Because the specification for the new version is applied to the old kubelet, the RHEL machine cannot return to the `Ready` state. You cannot complete the update until the machines are available. However, the maximum number of nodes that are unavailable is set to ensure that normal cluster operations are likely to continue with that number of machines out of service.

### 4.2. UNDERSTANDING OPENSShift CONTAINER PLATFORM UPGRADE CHANNELS

In OpenShift Container Platform 4.1, Red Hat introduced the concept of upgrade channels for recommending the appropriate upgrade versions to your cluster. Upgrade channels separate upgrade strategies and also are used to control the cadence of updates. Channels are tied to a minor version of OpenShift Container Platform. For instance, OpenShift Container Platform 4.3 channels will never include an upgrade to a 4.4 release. This ensures administrators make an explicit decision to upgrade to the next minor version of OpenShift Container Platform. Channels only control updates and have no impact on the version of the cluster you install; the `openshift-install` binary for a given patch level of OpenShift Container Platform always installs that patch level.

OpenShift Container Platform 4.3, which includes the upgrade from the previous 4.2 release, has three upgrade channels to choose from:

- candidate-4.3
- fast-4.3
- stable-4.3

The upgrade channels contain two types of updates:

1. **General Availability Software (or GA)** - These versions of OpenShift Container Platform are fully supported and are considered production quality. You may upgrade to the general availability release from either of the fast and stable channels.

2. **Release Candidate Software (or RC)** - These versions of OpenShift Container Platform are representative of the eventual general availability release and are available only in the candidate-4.3 channel. The release candidate will contain all the features of the product. You are allowed to upgrade from a release candidate to another release candidate and to upgrade from a previous minor version of OpenShift Container Platform to the current release candidate. Release candidate builds are not supported by Red Hat and you will not be able to
upgrade from a release candidate to the general availability release of OpenShift Container Platform. Candidates should be used to test feature acceptance and assist in qualifying the next version of OpenShift Container Platform in your infrastructure.

**NOTE**

Release candidates differ from the nightly builds found on try.openshift.com. You cannot upgrade nightly builds to nightly builds. Nightly builds are available for early access to features but are not upgradable or supported.

For GA versions, the fast and stable channels present a choice between receiving updates as soon as they are available or allowing Red Hat to control the rollout of those updates.

**fast-4.3**

The fast channel is updated with new 4.3 patch versions as soon as Red Hat declares they are generally available. Use this channel if you wish to receive updates as soon as they are available or for your pre-production environments when participating in the connected customer program. This channel will contain all z-stream (4.3.z) updates but will not suggest upgrades to the next minor release (4.4.z) when the next minor release is available.

**stable-4.3**

The stable channel will contain updates on a time delay as they are gradually rolled out to customers based on data from our SRE teams, support services, and pre-production and production environments that participate in our connected customer program, rather than being immediately available as they are in the fast channel. For patch and CVE fixes this can range from several hours to a day and allows an extra period of assessment in how the software performs. If issues are detected during rollout, upgrades to that version may be blocked in both the fast and stable channels, and a new version may be introduced that will be the new preferred upgrade target.

Customers can improve this process by configuring pre-production systems on the fast channel, production systems on the stable channel, and participating in Red Hat’s connected customer program - this allows Red Hat to observe the impact of updates on your specific hardware and software configurations. Future releases may improve or alter the pace at which updates move from the fast to the stable channels.

If issues are discovered with an upgrade between patch levels, Red Hat may withdraw that suggested upgrade for affected versions. A newer patch would become available in the appropriate channels and be suggested for upgrade.

**Upgrade version paths**

OpenShift Container Platform maintains an upgrade recommendation service that understands the version of OpenShift Container Platform you have installed as well as the path to take within the channel you choose to get you to the next release. You can imagine seeing the following in the fast-4.3 channel:

- 4.3.0
- 4.3.1
- 4.3.3
- 4.3.4

The service only recommends upgrades that have been tested and have no known issues. If you are on 4.3.1 and OpenShift Container Platform is allowing you to select 4.3.4, then it is safe for you to go from 4.3.1 to 4.3.4. Likewise, the absence of 4.3.2 may be due to a CVE that was fixed in 4.3.3 and Red Hat
no longer suggests upgrading to a known vulnerable version. If an issue is found that results in a new version being retracted from the recommendations, Red Hat will release a new version that is capable of upgrading from all necessary versions, including the retracted version.

Disconnected clusters
Customers which have chosen to not be connected to Red Hat and are curating their own OpenShift Container Platform container image content manually should consult the Red Hat errata associated with product releases and note any comments impacting upgrades. During upgrade the user interface may caution about switching between these versions and it is up to the customer to ensure they have correctly selected the appropriate version before bypassing those cautions.

Switching between channels
It is supported for customers to switch between the fast and stable channel at any time. Channels only offer suggested upgrades, and will never suggest a dangerous upgrade. If you switch to the candidate channel after installing from a GA version, you will see a warning the current version is not recognized, and you can safely switch back to a GA channel.

4.3. UPDATING A CLUSTER BY USING THE WEB CONSOLE

If updates are available, you can update your cluster from the web console.

You can find information about available OpenShift Container Platform advisories and updates in the errata section of the Customer Portal.

Prerequisites

- Have access to the web console as a user with admin privileges.

Procedure

1. From the web console, click Administration > Cluster Settings and review the contents of the Overview tab.

2. For production clusters, ensure that the CHANNEL is set to the correct channel for the version that you want to update to, your current minor version, such as stable-4.3.

   **IMPORTANT**
   
   For production clusters, you must subscribe to a stable-* or fast-* channel.

3. Click Updates Available, select a version to update to, the highest available version and click Update. The UPDATE STATUS changes to Updating, and you can review the progress of the Operator upgrades on the Cluster Operators tab.

4. If you are upgrading to this release from OpenShift Container Platform 4.3.3 or earlier, 4.2, you must restart all Pods after the upgrade is complete. You can do this using the following command, which requires the OpenShift CLI (oc):

   ```bash
   $ for I in $(oc get ns -o jsonpath='{range .items[*]} {.metadata.name}{"\n"} {end}'); \
   do oc delete pods --all -n $I; \
   ```
sleep 1; \
done

NOTE

Restarting all Pods is required because the service CA is automatically rotated as of OpenShift Container Platform 4.3.5. 4.3.5. The service CA is rotated during the upgrade and a restart is required afterward to ensure that all services are using the new service CA before the previous service CA expires.

After this one-time manual restart, subsequent upgrades and rotations will ensure restart before the service CA expires without requiring manual intervention.

5. After the update completes and the Cluster Version Operator refreshes the available updates, check if more updates are available in your current channel.

- If updates are available, continue to perform updates in the current channel until you can no longer update.
- If no updates are available, change the CHANNEL to the stable-* or fast-* channel for the next minor version, and update to the version that you want in that channel.

You might need to perform several intermediate updates until you reach the version that you want.

NOTE

When you update a cluster that contains Red Hat Enterprise Linux (RHEL) worker machines, those workers temporarily become unavailable during the update process. You must run the upgrade playbook against each RHEL machine as it enters the NotReady state for the cluster to finish updating.

4.4. (OPTIONAL) ADDING HOOKS TO PERFORM ANSIBLE TASKS ON RHEL MACHINES

You can use hooks to run Ansible tasks on the RHEL compute machines during the OpenShift Container Platform update.

4.4.1. About Ansible hooks for upgrades

When you update OpenShift Container Platform, you can run custom tasks on your Red Hat Enterprise Linux (RHEL) nodes during specific operations by using hooks. Hooks allow you to provide files that define tasks to run before or after specific update tasks. You can use hooks to validate or modify custom infrastructure when you update the RHEL compute nodes in your OpenShift Container Platform cluster.

Because when a hook fails, the operation fails, you must design hooks that are idempotent, or can run multiple times and provide the same results.

Hooks have the following important limitations: - Hooks do not have a defined or versioned interface. They can use internal openshift-ansible variables, but it is possible that the variables will be modified or removed in future OpenShift Container Platform releases. - Hooks do not have error handling, so an
error in a hook halts the update process. If you get an error, you must address the problem and then start the upgrade again.

4.4.2. Configuring the Ansible inventory file to use hooks

You define the hooks to use when you update the Red Hat Enterprise Linux (RHEL) compute machines, which are also known as worker machines, in the **hosts** inventory file under the **all:vars** section.

**Prerequisites**

- You have access to the machine that you used to add the RHEL compute machines cluster. You must have access to the **hosts** Ansible inventory file that defines your RHEL machines.

**Procedure**

1. After you design the hook, create a YAML file that defines the Ansible tasks for it. This file must be a set of tasks and cannot be a playbook, as shown in the following example:

   ```yaml
   ---
   # Trivial example forcing an operator to acknowledge the start of an upgrade
   # file=/home/user/openshift-ansible/hooks/pre_compute.yml
   
   - name: note the start of a compute machine update
     debug:
       msg: "Compute machine upgrade of {{ inventory_hostname }} is about to start"

   - name: require the user agree to start an upgrade
     pause:
       prompt: "Press Enter to start the compute machine update"
   ```

2. Modify the **hosts** Ansible inventory file to specify the hook files. The hook files are specified as parameter values in the **[all:vars]** section, as shown:

   **Example hook definitions in an inventory file**

   ```yaml
   [all:vars]
   openshift_node_pre_upgrade_hook=/home/user/openshift-ansible/hooks/pre_node.yml
   openshift_node_post_upgrade_hook=/home/user/openshift-ansible/hooks/post_node.yml
   ```

   To avoid ambiguity in the paths to the hook, use absolute paths instead of a relative paths in their definitions.

4.4.3. Available hooks for RHEL compute machines

You can use the following hooks when you update the Red Hat Enterprise Linux (RHEL) compute machines in your OpenShift Container Platform cluster.

<table>
<thead>
<tr>
<th>Hook name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>Hook name</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>openshift_node_pre_cordon_hook</td>
<td>• Runs before each node is cordoned.</td>
</tr>
<tr>
<td></td>
<td>• This hook runs against each node in serial.</td>
</tr>
<tr>
<td></td>
<td>• If a task must run against a different host, the task must use delegate_to or local_action.</td>
</tr>
<tr>
<td>openshift_node_pre_upgrade_hook</td>
<td>• Runs after each node is cordoned but before it is updated.</td>
</tr>
<tr>
<td></td>
<td>• This hook runs against each node in serial.</td>
</tr>
<tr>
<td></td>
<td>• If a task must run against a different host, the task must use delegate_to or local_action.</td>
</tr>
<tr>
<td>openshift_node_pre_uncordon_hook</td>
<td>• Runs after each node is updated but before it is uncordoned.</td>
</tr>
<tr>
<td></td>
<td>• This hook runs against each node in serial.</td>
</tr>
<tr>
<td></td>
<td>• If a task must run against a different host, they task must use delegate_to or local_action.</td>
</tr>
<tr>
<td>openshift_node_post_upgrade_hook</td>
<td>• Runs after each node uncordoned. It is the last node update action.</td>
</tr>
<tr>
<td></td>
<td>• This hook runs against each node in serial.</td>
</tr>
<tr>
<td></td>
<td>• If a task must run against a different host, the task must use delegate_to or local_action.</td>
</tr>
</tbody>
</table>

### 4.5. UPDATING RHEL COMPUTE MACHINES IN YOUR CLUSTER

After you update your cluster, you must update the Red Hat Enterprise Linux (RHEL) compute machines in your cluster.

**Prerequisites**

- You updated your cluster.
IMPORTANT

Because the RHEL machines require assets that are generated by the cluster to complete the update process, you must update the cluster before you update the RHEL compute machines in it.

- You have access to the machine that you used to add the RHEL compute machines cluster. You must have access to the `hosts` Ansible inventory file that defines your RHEL machines and the `upgrade` playbook.

Procedure

1. Stop and disable firewalld on the host:

   ```
   # systemctl disable --now firewalld.service
   ```

   **NOTE**

   You must not enable firewalld later. If you do, you cannot access OpenShift Container Platform logs on the worker.

2. Enable the repositories that are required for OpenShift Container Platform 4.3:

   a. On the machine that you run the Ansible playbooks, update the required repositories:

      ```
      # subscription-manager repos --disable=rhel-7-server-ansible-2.7-rpms \ 
      --disable=rhel-7-server-ose-4.2-rpms \ 
      --enable=rhel-7-server-ansible-2.8-rpms \ 
      --enable=rhel-7-server-ose-4.3-rpms
      ```

   b. On the machine that you run the Ansible playbooks, update the required packages, including `openshift-ansible`:

      ```
      # yum update openshift-ansible openshift-clients
      ```

   c. On each RHEL compute node, update the required repositories:

      ```
      # subscription-manager repos --disable=rhel-7-server-ose-4.2-rpms \ 
      --enable=rhel-7-server-ose-4.3-rpms
      ```

3. Update a RHEL worker machine:

   a. Review the current node status to determine which RHEL worker to update:

      ```
      # oc get node
      NAME                        STATUS                        ROLES    AGE    VERSION
      mycluster-control-plane-0   Ready                         master   145m   v1.16.2
      mycluster-control-plane-1   Ready                         master   145m   v1.16.2
      mycluster-control-plane-2   Ready                         master   145m   v1.16.2
      mycluster-rhel7-0           NotReady,SchedulingDisabled   worker   98m    v1.14.6+97c81d00e
      ```
mycluster-rhel7-1  Ready  worker  98m  v1.14.6+97c81d00e
mycluster-rhel7-2  Ready  worker  98m  v1.14.6+97c81d00e
mycluster-rhel7-3  Ready  worker  98m  v1.14.6+97c81d00e

Note which machine has the **NotReady, SchedulingDisabled** status.

b. Review your Ansible inventory file at `/<path>/inventory/hosts` and update its contents so that only the machine with the **NotReady, SchedulingDisabled** status is listed in the `[workers]` section, as shown in the following example:

```yaml
[all:vars]
  ansible_user=root
  #ansible_become=True

  openshift_kubeconfig_path="~/.kube/config"

[workers]
  mycluster-rhel7-0.example.com
```

c. Change to the `openshift-ansible` directory and run the `upgrade` playbook:

```bash
$ cd /usr/share/ansible/openshift-ansible
$ ansible-playbook -i /<path>/inventory/hosts playbooks/upgrade.yml
```

For `<path>`, specify the path to the Ansible inventory file that you created.

4. Follow the process in the previous step to update each RHEL worker machine in your cluster.

5. After you update all of the workers, confirm that all of your cluster nodes have updated to the new version:

```bash
# oc get node
NAME                        STATUS                        ROLES    AGE    VERSION
mycluster-control-plane-0   Ready                         master   145m   v1.16.2
mycluster-control-plane-1   Ready                         master   145m   v1.16.2
mycluster-control-plane-2   Ready                         master   145m   v1.16.2
mycluster-rhel7-0           NotReady,SchedulingDisabled   worker   98m    v1.16.2
mycluster-rhel7-1           Ready                         worker   98m    v1.16.2
mycluster-rhel7-2           Ready                         worker   98m    v1.16.2
mycluster-rhel7-3           Ready                         worker   98m    v1.16.2
```