



# Red Hat OpenShift Container Storage 4.8

## Deploying OpenShift Container Storage using Microsoft Azure and Azure Red Hat OpenShift

How to install and uninstall



# Red Hat OpenShift Container Storage 4.8 Deploying OpenShift Container Storage using Microsoft Azure and Azure Red Hat OpenShift

---

How to install and uninstall

## Legal Notice

Copyright © 2022 Red Hat, Inc.

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

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

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

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

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

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

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

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

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

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

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

All other trademarks are the property of their respective owners.

## Abstract

Read this document for instructions on installing and managing Red Hat OpenShift Container Storage on Microsoft Azure and Azure Red Hat OpenShift.

---

## Table of Contents

<b>MAKING OPEN SOURCE MORE INCLUSIVE</b> .....	<b>3</b>
<b>PROVIDING FEEDBACK ON RED HAT DOCUMENTATION</b> .....	<b>4</b>
<b>PREFACE</b> .....	<b>5</b>
<b>CHAPTER 1. PREPARING TO DEPLOY OPENSIFT CONTAINER STORAGE</b> .....	<b>6</b>
1.1. ENABLING KEY VALUE BACKEND PATH AND POLICY IN VAULT	6
<b>CHAPTER 2. DEPLOYING OPENSIFT CONTAINER STORAGE ON MICROSOFT AZURE</b> .....	<b>8</b>
2.1. INSTALLING RED HAT OPENSIFT CONTAINER STORAGE OPERATOR	8
2.2. CREATING AN OPENSIFT CONTAINER STORAGE CLUSTER SERVICE IN INTERNAL MODE	9
<b>CHAPTER 3. DEPLOYING OPENSIFT CONTAINER STORAGE ON AZURE RED HAT OPENSIFT</b> .....	<b>13</b>
3.1. GETTING A RED HAT PULL SECRET FOR NEW DEPLOYMENT OF AZURE RED HAT OPENSIFT	13
3.2. PREPARING A RED HAT PULL SECRET FOR EXISTING AZURE RED HAT OPENSIFT CLUSTERS	14
3.3. ADDING THE PULL SECRET TO THE CLUSTER	14
3.3.1. Modifying the configuration files to enable Red Hat operators	14
3.4. VALIDATING YOUR RED HAT PULL SECRET IS WORKING	14
3.5. INSTALLING RED HAT OPENSIFT CONTAINER STORAGE OPERATOR	15
3.6. CREATING AN OPENSIFT CONTAINER STORAGE CLUSTER SERVICE IN INTERNAL MODE	16
<b>CHAPTER 4. VERIFYING OPENSIFT CONTAINER STORAGE DEPLOYMENT</b> .....	<b>19</b>
4.1. VERIFYING THE STATE OF THE PODS	19
4.2. VERIFYING THE OPENSIFT CONTAINER STORAGE CLUSTER IS HEALTHY	20
4.3. VERIFYING THE MULTICLOUD OBJECT GATEWAY IS HEALTHY	21
4.4. VERIFYING THAT THE OPENSIFT CONTAINER STORAGE SPECIFIC STORAGE CLASSES EXIST	21
<b>CHAPTER 5. UNINSTALLING OPENSIFT CONTAINER STORAGE</b> .....	<b>22</b>
5.1. UNINSTALLING OPENSIFT CONTAINER STORAGE IN INTERNAL MODE	22
5.2. REMOVING MONITORING STACK FROM OPENSIFT CONTAINER STORAGE	28
5.3. REMOVING OPENSIFT CONTAINER PLATFORM REGISTRY FROM OPENSIFT CONTAINER STORAGE	31
5.4. REMOVING THE CLUSTER LOGGING OPERATOR FROM OPENSIFT CONTAINER STORAGE	32



## 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. Do let us know how we can make it better. To give feedback:

- For simple comments on specific passages:
  1. Make sure you are viewing the documentation in the *Multi-page HTML* format. In addition, ensure you see the **Feedback** button in the upper right corner of the document.
  2. Use your mouse cursor to highlight the part of text that you want to comment on.
  3. Click the **Add Feedback** pop-up that appears below the highlighted text.
  4. Follow the displayed instructions.
- For submitting more complex feedback, create a Bugzilla ticket:
  1. Go to the [Bugzilla](#) website.
  2. In the **Component** section, choose **documentation**.
  3. Fill in the **Description** field with your suggestion for improvement. Include a link to the relevant part(s) of documentation.
  4. Click **Submit Bug**.



---

## PREFACE

Red Hat OpenShift Container Storage 4.8 supports deployment on existing Red Hat OpenShift Container Platform (RHOCP) Azure clusters.



### NOTE

Only internal OpenShift Container Storage clusters are supported on Microsoft Azure. See [Planning your deployment](#) for more information about deployment requirements.

To deploy OpenShift Container Storage in internal mode, start with the requirements in [Preparing to deploy OpenShift Container Storage](#) chapter and then follow the deployment process [Deploying OpenShift Container Storage on Microsoft Azure](#)

# CHAPTER 1. PREPARING TO DEPLOY OPENSIFT CONTAINER STORAGE

Deploying OpenShift Container Storage on OpenShift Container Platform using dynamic storage devices provides you with the option to create internal cluster resources. This will result in the internal provisioning of the base services, which helps to make additional storage classes available to applications.

Before you begin the deployment of Red Hat OpenShift Container Storage, follow these steps:

1. Setup a chrony server. See [Configuring chrony time service](#) and use [knowledgebase solution](#) to create rules allowing all traffic.
2. Optional: If you want to enable cluster-wide encryption using an external Key Management System (KMS):
  - Ensure that a policy with a token exists and the key value backend path in Vault is enabled. See [Enabling the key value backend path and policy in Vault](#) .
  - Ensure that you are using signed certificates on your Vault servers.
3. Minimum starting node requirements [Technology Preview]  
An OpenShift Container Storage cluster will be deployed with minimum configuration when the standard deployment resource requirement is not met. See [Resource requirements](#) section in Planning guide.

## 1.1. ENABLING KEY VALUE BACKEND PATH AND POLICY IN VAULT

### Prerequisites

- Administrator access to Vault.
- Choose a unique path name as the backend **path** that follows the naming convention since it cannot be changed later.

### Procedure

1. Enable the Key/Value (KV) backend path in Vault.  
For Vault KV secret engine API, version 1:

```
$ vault secrets enable -path=ocs kv
```

For Vault KV secret engine API, version 2:

```
$ vault secrets enable -path=ocs kv-v2
```

2. Create a policy to restrict users to perform a write or delete operation on the secret using the following commands:

```
echo '
path "ocs/*" {
  capabilities = ["create", "read", "update", "delete", "list"]
}
```

```
path "sys/mounts" {  
  capabilities = ["read"]  
}| vault policy write ocs -
```

3. Create a token matching the above policy:

```
$ vault token create -policy=ocs -format json
```

## CHAPTER 2. DEPLOYING OPENSIFT CONTAINER STORAGE ON MICROSOFT AZURE

Deploying OpenShift Container Storage on OpenShift Container Platform using dynamic storage devices provided by Microsoft Azure installer-provisioned infrastructure (IPI) (type: **managed-premium**) enables you to create internal cluster resources. This results in internal provisioning of the base services, which helps to make additional storage classes available to applications.



### NOTE

Only internal OpenShift Container Storage clusters are supported on Microsoft Azure. See [Planning your deployment](#) for more information about deployment requirements.

Ensure that you have addressed the requirements in [Preparing to deploy OpenShift Container Storage](#) chapter before proceeding with the below steps for deploying using dynamic storage devices:

1. [Install the Red Hat OpenShift Container Storage Operator](#) .
2. [Create the OpenShift Container Storage Cluster Service](#)

### 2.1. INSTALLING RED HAT OPENSIFT CONTAINER STORAGE OPERATOR

You can install Red Hat OpenShift Container Storage Operator using the Red Hat OpenShift Container Platform Operator Hub.

#### Prerequisites

- Access to an OpenShift Container Platform cluster using an account with cluster-admin and operator installation permissions.
- You have at least three worker nodes in the Red Hat OpenShift Container Platform cluster.
- You have satisfied any additional requirements required. For more information, see [Planning your deployment](#).



### NOTE

- When you need to override the cluster-wide default node selector for OpenShift Container Storage, you can use the following command to specify a blank node selector for the **openshift-storage** namespace (create openshift-storage namespace in this case):

```
$ oc annotate namespace openshift-storage openshift.io/node-selector=
```

- Taint a node as **infra** to ensure only Red Hat OpenShift Container Storage resources are scheduled on that node. This helps you save on subscription costs. For more information, see [How to use dedicated worker nodes for Red Hat OpenShift Container Storage](#) chapter in Managing and Allocating Storage Resources guide.

#### Procedure

1. Log in to OpenShift Web Console.
2. Click **Operators** → **OperatorHub**.
3. Search for **OpenShift Container Storage** from the list of operators and click on it.
4. Click **Install**.
5. Set the following options on the **Install Operator** page:
  - a. Channel as **stable-4.8**.
  - b. Installation Mode as **A specific namespace on the cluster**
  - c. Installed Namespace as **Operator recommended namespace openshift-storage**. If Namespace **openshift-storage** does not exist, it will be created during the operator installation.
  - d. **Approval Strategy** as **Automatic** or **Manual**.
  - e. Click **Install**.

If you select **Automatic** updates, the Operator Lifecycle Manager (OLM) automatically upgrades the running instance of your operator without any intervention.

If you select **Manual** updates, the OLM creates an update request. As a cluster administrator, you must then manually approve that update request to have the operator updated to the new version.

### Verification step

- Verify that the **OpenShift Container Storage** Operator shows a green tick indicating successful installation.

## 2.2. CREATING AN OPENSIFT CONTAINER STORAGE CLUSTER SERVICE IN INTERNAL MODE

Use this procedure to create an OpenShift Container Storage Cluster Service after you install the OpenShift Container Storage operator.

### Prerequisites

- The OpenShift Container Storage operator must be installed from the Operator Hub. For more information, see [Installing OpenShift Container Storage Operator using the Operator Hub](#) .

### Procedure

1. Log into the OpenShift Web Console.
2. Click **Operators** → **Installed Operators** to view all the installed operators. Ensure that the **Project** selected is **openshift-storage**.
3. Click **OpenShift Container Storage** > **Create Instance** link of Storage Cluster.
4. **Select Mode** is set to **Internal** by default.
5. Select **Capacity and nodes**

- a. Select **Storage Class**. By default, it is set to **managed-premium**.
- b. Select **Requested Capacity** from the drop down list. It is set to **2 TiB** by default. You can use the drop down to modify the capacity value.

**NOTE**

Once you select the initial storage capacity, cluster expansion is performed only using the selected usable capacity (3 times of raw storage).

- c. In the **Select Nodes** section, select at least three available nodes.  
For cloud platforms with multiple availability zones, ensure that the Nodes are spread across different Locations/availability zones.

If the nodes selected do not match the OpenShift Container Storage cluster requirement of an aggregated 30 CPUs and 72 GiB of RAM, a minimal cluster will be deployed. For minimum starting node requirements, see [Resource requirements](#) section in Planning guide.

- d. Click **Next**.

6. (Optional) Set **Security and network** configuration

- a. Select the **Enable encryption** checkbox to encrypt block and file storage.
- b. Choose any one or both **Encryption level**:
  - **Cluster-wide encryption** to encrypt the entire cluster (block and file).
  - **Storage class encryption** to create encrypted persistent volume (block only) using encryption enabled storage class.
- c. Select the **Connect to an external key management service** checkbox. This is optional for cluster-wide encryption.
  - i. **Key Management Service Provider** is set to **Vault** by default.
  - ii. Enter Vault **Service Name**, host **Address** of Vault server ('https://<hostname or ip>'), **Port number** and **Token**.
  - iii. Expand **Advanced Settings** to enter additional settings and certificate details based on your Vault configuration:
    - A. Enter the Key Value secret path in **Backend Path** that is dedicated and unique to OpenShift Container Storage.
    - B. (Optional) Enter **TLS Server Name** and **Vault Enterprise Namespace**.
    - C. Provide **CA Certificate**, **Client Certificate** and **Client Private Key** by uploading the respective PEM encoded certificate file.
    - D. Click **Save**.

7. Select **Default** (SDN) if you are using a single network or **Custom** (Multus) Network if you plan on using multiple network interfaces.

- a. Select a **Public Network Interface** from drop down.

- b. Select a **Cluster Network Interface** from drop down.



#### NOTE

If only using one additional network interface select the single NetworkAttachementDefinition (i.e. ocs-public-cluster) for the Public Network Interface and leave the Cluster Network Interface blank.

8. Click **Next**.
9. Review the configuration details. To modify any configuration settings, click **Back** to go back to the previous configuration page.
10. Click **Create**.
11. Edit the configmap if Vault Key/Value (KV) secret engine API, version 2 is used for cluster-wide encryption with Key Management System (KMS).
  - a. On the OpenShift Web Console, navigate to **Workloads → ConfigMaps**.
  - b. To view the KMS connection details, click **ocs-kms-connection-details**.
  - c. Edit the configmap.
    - i. Click **Action menu ( ⋮ ) → Edit ConfigMap**
    - ii. Set the **VAULT\_BACKEND** parameter to **v2**.

```
kind: ConfigMap
apiVersion: v1
metadata:
  name: ocs-kms-connection-details
[...]
data:
  KMS_PROVIDER: vault
  KMS_SERVICE_NAME: vault
[...]
  VAULT_BACKEND: v2
[...]
```

- iii. Click **Save**.

#### Verification steps

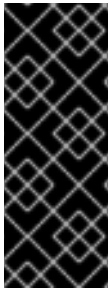
1. On the storage cluster details page, the storage cluster name displays a green tick next to it to indicate that the cluster was created successfully.
2. Verify that the final **Status** of the installed storage cluster shows as **Phase: Ready** with a green tick mark.
  - Click **Operators → Installed Operators → Storage Cluster** link to view the storage cluster installation status.
  - Alternatively, when you are on the Operator **Details** tab, you can click on the **Storage Cluster** tab to view the status.

3. To verify that all components for OpenShift Container Storage are successfully installed, see [Verifying your OpenShift Container Storage installation](#) .



## CHAPTER 3. DEPLOYING OPENSIFT CONTAINER STORAGE ON AZURE RED HAT OPENSIFT

The Azure Red Hat OpenShift service enables you to deploy fully managed OpenShift clusters. Red Hat OpenShift Container Storage can be deployed on Azure Red Hat OpenShift service.



### IMPORTANT

OpenShift Container storage on Azure Red Hat OpenShift is not a managed service offering. Red Hat OpenShift Container Storage subscriptions are required to have the installation supported by the Red Hat support team. Open support cases by choosing the product as **Red Hat OpenShift Container Storage** with the [Red Hat support](#) team (and not Microsoft) if you need any assistance for OpenShift Container Storage on Azure Red Hat OpenShift.

To install OpenShift Container Storage on Azure Red Hat OpenShift, follow sections:

1. [Getting a Red Hat pull secret for new deployment of Azure Red Hat OpenShift](#) .
2. [Preparing a Red Hat pull secret for existing Azure Red Hat OpenShift clusters](#) .
3. [Adding the pull secret to the cluster](#) .
4. [Validating your Red Hat pull secret is working](#) .
5. [Install the Red Hat OpenShift Container Storage Operator](#) .
6. [Create the OpenShift Container Storage Cluster Service](#) .

### 3.1. GETTING A RED HAT PULL SECRET FOR NEW DEPLOYMENT OF AZURE RED HAT OPENSIFT

A Red Hat pull secret enables the cluster to access Red Hat container registries along with additional content.

#### Prerequisites

- A Red Hat portal account.
- OpenShift Container Storage subscription.

#### Procedure

To get a Red Hat pull secret for a new deployment of Azure Red Hat OpenShift, follow the steps in the section [Get a Red Hat pull secret](#) in the official Microsoft Azure documentation.

Note that while creating the [Azure Red Hat OpenShift cluster](#), you may need larger worker nodes, controlled by `--worker-vm-size` or more worker nodes, controlled by `--worker-count`. The recommended `worker-vm-size` is **Standard\_D16s\_v3**. You can also use dedicated worker nodes, for more information, see [How to use dedicated worker nodes for Red Hat OpenShift Container Storage](#) in the *Managing and allocating storage resources* guide.

## 3.2. PREPARING A RED HAT PULL SECRET FOR EXISTING AZURE RED HAT OPENSIFT CLUSTERS

When you create an Azure Red Hat OpenShift cluster without adding a Red Hat pull secret, a pull secret is still created on the cluster automatically. However, this pull secret is not fully populated.

Use this section to update the automatically created pull secret with the additional values from the Red Hat pull secret.

### Prerequisites

- Existing Azure Red Hat OpenShift cluster without a Red Hat pull secret.

### Procedure

To prepare a Red Hat pull secret for existing an existing Azure Red Hat OpenShift clusters, follow the steps in the section [Prepare your pull secret](#) in the official Microsoft Azure documentation.

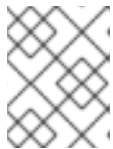
## 3.3. ADDING THE PULL SECRET TO THE CLUSTER

### Prerequisites

- A Red Hat pull secret.

### Procedure

- Run the following command to update your pull secret.



### NOTE

Running this command causes the cluster nodes to restart one by one as they are updated.

```
oc set data secret/pull-secret -n openshift-config --from-file=.dockerconfigjson=./pull-secret.json
```

After the secret is set, you can enable the Red Hat Certified Operators.

### 3.3.1. Modifying the configuration files to enable Red Hat operators

To modify the configuration files to enable Red Hat operators, follow the steps in the section [Modify the configuration files](#) in the official Microsoft Azure documentation.

## 3.4. VALIDATING YOUR RED HAT PULL SECRET IS WORKING

After you add the pull secret and modify the configuration files, the cluster can take several minutes to get updated.

To check if the cluster has been updated, run the following command to show the **Certified Operators** and **Red Hat Operators** sources available:

```
$ oc get catalogsource -A
```

NAMESPACE	NAME	DISPLAY
openshift-marketplace	redhat-operators	Red Hat Operators

TYPE	PUBLISHER	AGE
grpc	Red Hat	11s

If you do not see the Red Hat Operators, wait a few minutes and try again.

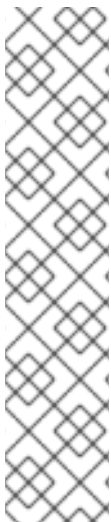
To ensure that your pull secret has been updated and is working correctly, open **Operator Hub** and check for any Red Hat verified Operator. For example, check if the OpenShift Container Storage Operator is available, and see if you have permissions to install it.

## 3.5. INSTALLING RED HAT OPENSIFT CONTAINER STORAGE OPERATOR

You can install Red Hat OpenShift Container Storage Operator using the Red Hat OpenShift Container Platform Operator Hub.

### Prerequisites

- Access to an OpenShift Container Platform cluster using an account with cluster-admin and operator installation permissions.
- You have at least three worker nodes in the Red Hat OpenShift Container Platform cluster.
- You have satisfied any additional requirements required. For more information, see [Planning your deployment](#).



### NOTE

- When you need to override the cluster-wide default node selector for OpenShift Container Storage, you can use the following command to specify a blank node selector for the **openshift-storage** namespace (create openshift-storage namespace in this case):

```
$ oc annotate namespace openshift-storage openshift.io/node-selector=
```

- Taint a node as **infra** to ensure only Red Hat OpenShift Container Storage resources are scheduled on that node. This helps you save on subscription costs. For more information, see [How to use dedicated worker nodes for Red Hat OpenShift Container Storage](#) chapter in *Managing and Allocating Storage Resources* guide.

### Procedure

1. Log in to OpenShift Web Console.
2. Click **Operators** → **OperatorHub**.
3. Search for **OpenShift Container Storage** from the list of operators and click on it.
4. Click **Install**.
5. Set the following options on the **Install Operator** page:

- a. Channel as **stable-4.8**.
- b. Installation Mode as **A specific namespace on the cluster**
- c. Installed Namespace as **Operator recommended namespace openshift-storage**. If Namespace **openshift-storage** does not exist, it will be created during the operator installation.
- d. **Approval Strategy** as **Automatic** or **Manual**.
- e. Click **Install**.

If you select **Automatic** updates, the Operator Lifecycle Manager (OLM) automatically upgrades the running instance of your operator without any intervention.

If you select **Manual** updates, the OLM creates an update request. As a cluster administrator, you must then manually approve that update request to have the operator updated to the new version.

### Verification step

- Verify that the **OpenShift Container Storage** Operator shows a green tick indicating successful installation.

## 3.6. CREATING AN OPENSIFT CONTAINER STORAGE CLUSTER SERVICE IN INTERNAL MODE

Use this procedure to create an OpenShift Container Storage Cluster Service after you install the OpenShift Container Storage operator.

### Prerequisites

- The OpenShift Container Storage operator must be installed from the Operator Hub. For more information, see [Installing OpenShift Container Storage Operator using the Operator Hub](#) .

### Procedure

1. Log into the OpenShift Web Console.
2. Click **Operators** → **Installed Operators** to view all the installed operators. Ensure that the **Project** selected is **openshift-storage**.
3. Click **OpenShift Container Storage** > **Create Instance** link of Storage Cluster.
4. **Select Mode** is set to **Internal** by default.
5. Select **Capacity and nodes**
  - a. Select **Storage Class**. By default, it is set to **managed-premium**.
  - b. Select **Requested Capacity** from the drop down list. It is set to **2 TiB** by default. You can use the drop down to modify the capacity value.

**NOTE**

Once you select the initial storage capacity, cluster expansion is performed only using the selected usable capacity (3 times of raw storage).

- c. In the **Select Nodes** section, select at least three available nodes.  
For cloud platforms with multiple availability zones, ensure that the Nodes are spread across different Locations/availability zones.  
  
If the nodes selected do not match the OpenShift Container Storage cluster requirement of an aggregated 30 CPUs and 72 GiB of RAM, a minimal cluster will be deployed. For minimum starting node requirements, see [Resource requirements](#) section in Planning guide.
  - d. Click **Next**.
6. (Optional) Set **Security and network** configuration
    - a. Select the **Enable encryption** checkbox to encrypt block and file storage.
    - b. Choose any one or both **Encryption level**:
      - **Cluster-wide encryption** to encrypt the entire cluster (block and file).
      - **Storage class encryption** to create encrypted persistent volume (block only) using encryption enabled storage class.
    - c. Select the **Connect to an external key management service** checkbox. This is optional for cluster-wide encryption.
      - i. **Key Management Service Provider** is set to **Vault** by default.
      - ii. Enter Vault **Service Name**, host **Address** of Vault server ('https://<hostname or ip>'), **Port number** and **Token**.
      - iii. Expand **Advanced Settings** to enter additional settings and certificate details based on your Vault configuration:
        - A. Enter the Key Value secret path in **Backend Path** that is dedicated and unique to OpenShift Container Storage.
        - B. (Optional) Enter **TLS Server Name** and **Vault Enterprise Namespace**.
        - C. Provide **CA Certificate**, **Client Certificate** and **Client Private Key** by uploading the respective PEM encoded certificate file.
    - D. Click **Save**.
  7. Select **Default** (SDN) if you are using a single network or **Custom** (Multus) Network if you plan on using multiple network interfaces.
    - a. Select a **Public Network Interface** from drop down.
    - b. Select a **Cluster Network Interface** from drop down.

**NOTE**

If only using one additional network interface select the single NetworkAttachmentDefinition (i.e. ocs-public-cluster) for the Public Network Interface and leave the Cluster Network Interface blank.

8. Click **Next**.
9. Review the configuration details. To modify any configuration settings, click **Back** to go back to the previous configuration page.
10. Click **Create**.
11. Edit the configmap if Vault Key/Value (KV) secret engine API, version 2 is used for cluster-wide encryption with Key Management System (KMS).
  - a. On the OpenShift Web Console, navigate to **Workloads → ConfigMaps**
  - b. To view the KMS connection details, click **ocs-kms-connection-details**.
  - c. Edit the configmap.
    - i. Click **Action menu ( ⋮ ) → Edit ConfigMap**
    - ii. Set the **VAULT\_BACKEND** parameter to **v2**.

```
kind: ConfigMap
apiVersion: v1
metadata:
  name: ocs-kms-connection-details
  [...]
data:
  KMS_PROVIDER: vault
  KMS_SERVICE_NAME: vault
  [...]
  VAULT_BACKEND: v2
  [...]
```

- iii. Click **Save**.

**Verification steps**

1. On the storage cluster details page, the storage cluster name displays a green tick next to it to indicate that the cluster was created successfully.
2. Verify that the final **Status** of the installed storage cluster shows as **Phase: Ready** with a green tick mark.
  - Click **Operators → Installed Operators → Storage Cluster** link to view the storage cluster installation status.
  - Alternatively, when you are on the Operator **Details** tab, you can click on the **Storage Cluster** tab to view the status.
3. To verify that all components for OpenShift Container Storage are successfully installed, see [Verifying your OpenShift Container Storage installation](#) .

# CHAPTER 4. VERIFYING OPENSIFT CONTAINER STORAGE DEPLOYMENT

Use this section to verify that OpenShift Container Storage is deployed correctly.

## 4.1. VERIFYING THE STATE OF THE PODS

To verify that the pods of OpenShift Containers Storage are in running state, follow the below procedure:

### Procedure

1. Log in to OpenShift Web Console.
2. Click **Workloads** → **Pods** from the left pane of the OpenShift Web Console.
3. Select **openshift-storage** from the **Project** drop down list.  
For more information on the expected number of pods for each component and how it varies depending on the number of nodes, see [Table 4.1, "Pods corresponding to OpenShift Container storage cluster"](#).
4. Click on the **Running** and **Completed** tabs to verify that the pods are running and in a completed state:

**Table 4.1. Pods corresponding to OpenShift Container storage cluster**

Component	Corresponding pods
OpenShift Container Storage Operator	<ul style="list-style-type: none"> <li>● <b>ocs-operator-*</b> (1 pod on any worker node)</li> <li>● <b>ocs-metrics-exporter-*</b></li> </ul>
Rook-ceph Operator	<b>rook-ceph-operator-*</b> (1 pod on any worker node)
Multicloud Object Gateway	<ul style="list-style-type: none"> <li>● <b>noobaa-operator-*</b> (1 pod on any worker node)</li> <li>● <b>noobaa-core-*</b> (1 pod on any storage node)</li> <li>● <b>noobaa-db-pg-*</b> (1 pod on any storage node)</li> <li>● <b>noobaa-endpoint-*</b> (1 pod on any storage node)</li> </ul>
MON	<b>rook-ceph-mon-*</b> (3 pods distributed across storage nodes)

Component	Corresponding pods
MGR	<b>rook-ceph-mgr-*</b> (1 pod on any storage node)
MDS	<b>rook-ceph-mds-ocs-storagecluster-cephfilesystem-*</b> (2 pods distributed across storage nodes)
CSI	<ul style="list-style-type: none"> <li>● <b>cephfs</b> <ul style="list-style-type: none"> <li>○ <b>csi-cephfsplugin-*</b> (1 pod on each worker node)</li> <li>○ <b>csi-cephfsplugin-provisioner-*</b> (2 pods distributed across worker nodes)</li> </ul> </li> <li>● <b>rbd</b> <ul style="list-style-type: none"> <li>○ <b>csi-rbdplugin-*</b> (1 pod on each worker node)</li> <li>○ <b>csi-rbdplugin-provisioner-*</b> (2 pods distributed across worker nodes)</li> </ul> </li> </ul>
rook-ceph-crashcollector	<b>rook-ceph-crashcollector-*</b> (1 pod on each storage node)
OSD	<ul style="list-style-type: none"> <li>● <b>rook-ceph-osd-*</b> (1 pod for each device)</li> <li>● <b>rook-ceph-osd-prepare-ocs-deviceset-*</b> (1 pod for each device)</li> </ul>

## 4.2. VERIFYING THE OPENSIFT CONTAINER STORAGE CLUSTER IS HEALTHY

To verify that the cluster of OpenShift Container Storage is healthy, follow the steps in the procedure.

### Procedure

1. Click **Storage** → **Overview** and click the **Block and File** tab.
2. In the **Status card**, verify that *Storage Cluster* and *Data Resiliency* has a green tick mark.
3. In the **Details card**, verify that the cluster information is displayed.

For more information on the health of the OpenShift Container Storage clusters using the Block and File dashboard, see [Monitoring OpenShift Container Storage](#).



## 4.3. VERIFYING THE MULTICLOUD OBJECT GATEWAY IS HEALTHY

To verify that the OpenShift Container Storage Multicloud Object Gateway is healthy, follow the steps in the procedure.

### Procedure

1. Click **Storage** → **Overview** from the OpenShift Web Console and click the **Object** tab.
2. In the **Status card**, verify that both *Object Service* and *Data Resiliency* are in **Ready** state (green tick).
3. In the **Details card**, verify that the Multicloud Object Gateway information is displayed.

For more information on the health of the OpenShift Container Storage cluster using the object service dashboard, see [Monitoring OpenShift Container Storage](#).

## 4.4. VERIFYING THAT THE OPENSIFT CONTAINER STORAGE SPECIFIC STORAGE CLASSES EXIST

To verify the storage classes exists in the cluster, follow the steps in the procedure.

### Procedure

1. Click **Storage** → **Storage Classes** from the OpenShift Web Console.
2. Verify that the following storage classes are created with the OpenShift Container Storage cluster creation:
  - **ocs-storagecluster-ceph-rbd**
  - **ocs-storagecluster-cephfs**
  - **openshift-storage.noobaa.io**

## CHAPTER 5. UNINSTALLING OPENSIFT CONTAINER STORAGE

### 5.1. UNINSTALLING OPENSIFT CONTAINER STORAGE IN INTERNAL MODE

Use the steps in this section to uninstall OpenShift Container Storage.

#### Uninstall Annotations

Annotations on the Storage Cluster are used to change the behavior of the uninstall process. To define the uninstall behavior, the following two annotations have been introduced in the storage cluster:

- **uninstall.ocs.openshift.io/cleanup-policy: delete**
- **uninstall.ocs.openshift.io/mode: graceful**

The below table provides information on the different values that can be used with these annotations:

**Table 5.1. uninstall.ocs.openshift.io uninstall annotations descriptions**

Annotation	Value	Default	Behavior
cleanup-policy	delete	Yes	Rook cleans up the physical drives and the <b>DataDirHostPath</b>
cleanup-policy	retain	No	Rook does <b>not</b> clean up the physical drives and the <b>DataDirHostPath</b>
mode	graceful	Yes	Rook and NooBaa <b>pauses</b> the uninstall process until the PVCs and the OBCs are removed by the administrator/user
mode	forced	No	Rook and NooBaa proceeds with uninstall even if PVCs/OBCs provisioned using Rook and NooBaa exist respectively.

You can change the cleanup policy or the uninstall mode by editing the value of the annotation by using the following commands:

```
$ oc annotate storagecluster -n openshift-storage ocs-storagecluster
uninstall.ocs.openshift.io/cleanup-policy="retain" --overwrite
storagecluster.ocs.openshift.io/ocs-storagecluster annotated
```

```
$ oc annotate storagecluster -n openshift-storage ocs-storagecluster
uninstall.ocs.openshift.io/mode="forced" --overwrite
storagecluster.ocs.openshift.io/ocs-storagecluster annotated
```

## Prerequisites

- Ensure that the OpenShift Container Storage cluster is in a healthy state. The uninstall process can fail when some of the pods are not terminated successfully due to insufficient resources or nodes. In case the cluster is in an unhealthy state, contact Red Hat Customer Support before uninstalling OpenShift Container Storage.
- Ensure that applications are not consuming persistent volume claims (PVCs) or object bucket claims (OBCs) using the storage classes provided by OpenShift Container Storage.
- If any custom resources (such as custom storage classes, cephblockpools) were created by the admin, they must be deleted by the admin after removing the resources which consumed them.

## Procedure

1. Delete the volume snapshots that are using OpenShift Container Storage.

- a. List the volume snapshots from all the namespaces.

```
$ oc get volumesnapshot --all-namespaces
```

- b. From the output of the previous command, identify and delete the volume snapshots that are using OpenShift Container Storage.

```
$ oc delete volumesnapshot <VOLUME-SNAPSHOT-NAME> -n <NAMESPACE>
```

2. Delete PVCs and OBCs that are using OpenShift Container Storage.

In the default uninstall mode (graceful), the uninstaller waits till all the PVCs and OBCs that use OpenShift Container Storage are deleted.

If you wish to delete the Storage Cluster without deleting the PVCs beforehand, you may set the uninstall mode annotation to **forced** and skip this step. Doing this results in orphan PVCs and OBCs in the system.

- a. Delete OpenShift Container Platform monitoring stack PVCs using OpenShift Container Storage.

For more information, see [Section 5.2, "Removing monitoring stack from OpenShift Container Storage"](#).

- b. Delete OpenShift Container Platform Registry PVCs using OpenShift Container Storage.

For more information, see [Section 5.3, "Removing OpenShift Container Platform registry from OpenShift Container Storage"](#).

- c. Delete OpenShift Container Platform logging PVCs using OpenShift Container Storage.

For more information, see [Section 5.4, "Removing the cluster logging operator from OpenShift Container Storage"](#).

- d. Delete other PVCs and OBCs provisioned using OpenShift Container Storage.

- Following script is sample script to identify the PVCs and OBCs provisioned using OpenShift Container Storage. The script ignores the PVCs that are used internally by

Openshift Container Storage.

```
#!/bin/bash

RBD_PROVISIONER="openshift-storage.rbd.csi.ceph.com"
CEPHFS_PROVISIONER="openshift-storage.cephfs.csi.ceph.com"
NOOBAA_PROVISIONER="openshift-storage.noobaa.io/obc"
RGW_PROVISIONER="openshift-storage.ceph.rook.io/bucket"

NOOBAA_DB_PVC="noobaa-db"
NOOBAA_BACKINGSTORE_PVC="noobaa-default-backing-store-noobaa-pvc"

# Find all the OCS StorageClasses
OCS_STORAGECLASSES=$(oc get storageclasses | grep -e
"$RBD_PROVISIONER" -e "$CEPHFS_PROVISIONER" -e
"$NOOBAA_PROVISIONER" -e "$RGW_PROVISIONER" | awk '{print $1}')

# List PVCs in each of the StorageClasses
for SC in $OCS_STORAGECLASSES
do
    echo
    "=====
=="
    echo "$SC StorageClass PVCs and OBCs"
    echo
    "=====
=="
    oc get pvc --all-namespaces --no-headers 2>/dev/null | grep $SC | grep -v -e
"$NOOBAA_DB_PVC" -e "$NOOBAA_BACKINGSTORE_PVC"
    oc get obc --all-namespaces --no-headers 2>/dev/null | grep $SC
    echo
done
```



#### NOTE

Omit **RGW\_PROVISIONER** for cloud platforms.

- Delete the OBCs.

```
$ oc delete obc <obc name> -n <project name>
```

- Delete the PVCs.

```
$ oc delete pvc <pvc name> -n <project-name>
```



#### NOTE

Ensure that you have removed any custom backing stores, bucket classes, etc., created in the cluster.

3. Delete the Storage Cluster object and wait for the removal of the associated resources.

```
$ oc delete -n openshift-storage storagecluster --all --wait=true
```

4. Check for cleanup pods if the **uninstall.ocs.openshift.io/cleanup-policy** was set to **delete** (default) and ensure that their status is **Completed**.

```
$ oc get pods -n openshift-storage | grep -i cleanup
NAME                READY STATUS  RESTARTS AGE
cluster-cleanup-job-

```

5. Confirm that the directory **/var/lib/rook** is now empty. This directory will be empty only if the **uninstall.ocs.openshift.io/cleanup-policy** annotation was set to **delete** (default).

```
$ for i in $(oc get node -l cluster.ocs.openshift.io/openshift-storage= -o jsonpath='{.items[*].metadata.name}'); do oc debug node/${i} -- chroot /host ls -l /var/lib/rook; done
```

6. If encryption was enabled at the time of install, remove **dm-crypt** managed **device-mapper** mapping from OSD devices on all the OpenShift Container Storage nodes.

- a. Create a **debug** pod and **chroot** to the host on the storage node.

```
$ oc debug node/<node name>
$ chroot /host
```

- b. Get Device names and make note of the OpenShift Container Storage devices.

```
$ dmsetup ls
ocs-deviceset-0-data-0-57snx-block-dmccrypt (253:1)
```

- c. Remove the mapped device.

```
$ cryptsetup luksClose --debug --verbose ocs-deviceset-0-data-0-57snx-block-dmccrypt
```



## NOTE

If the above command gets stuck due to insufficient privileges, run the following commands:

- Press **CTRL+Z** to exit the above command.
- Find PID of the process which was stuck.

```
$ ps -ef | grep crypt
```

- Terminate the process using **kill** command.

```
$ kill -9 <PID>
```

- Verify that the device name is removed.

```
$ dmsetup ls
```

7. Delete the namespace and wait till the deletion is complete. You need to switch to another project if **openshift-storage** is the active project.

For example:

```
$ oc project default
$ oc delete project openshift-storage --wait=true --timeout=5m
```

The project is deleted if the following command returns a NotFound error.

```
$ oc get project openshift-storage
```



#### NOTE

While uninstalling OpenShift Container Storage, if **namespace** is not deleted completely and remains in **Terminating** state, perform the steps in [Troubleshooting and deleting remaining resources during Uninstall](#) to identify objects that are blocking the namespace from being terminated.

8. Unlabel the storage nodes.

```
$ oc label nodes --all cluster.ocs.openshift.io/openshift-storage-
$ oc label nodes --all topology.rook.io/rack-
```

9. Remove the OpenShift Container Storage taint if the nodes were tainted.

```
$ oc adm taint nodes --all node.ocs.openshift.io/storage-
```

10. Confirm all PVs provisioned using OpenShift Container Storage are deleted. If there is any PV left in the **Released** state, delete it.

```
$ oc get pv
$ oc delete pv <pv name>
```

11. Delete the Multicloud Object Gateway storageclass.

```
$ oc delete storageclass openshift-storage.noobaa.io --wait=true --timeout=5m
```

12. Remove **CustomResourceDefinitions**.

```
$ oc delete crd backingstores.noobaa.io bucketclasses.noobaa.io
cephblockpools.ceph.rook.io cephclusters.ceph.rook.io cephfilesystems.ceph.rook.io
cephnfses.ceph.rook.io cephobjectstores.ceph.rook.io cephobjectstoreusers.ceph.rook.io
noobaas.noobaa.io ocsinitializations.ocs.openshift.io storageclusters.ocs.openshift.io
cephclients.ceph.rook.io cephobjectrealms.ceph.rook.io cephobjectzonegroups.ceph.rook.io
cephobjectzones.ceph.rook.io cephrbdmirrors.ceph.rook.io --wait=true --timeout=5m
```

13. Optional: To ensure that the vault keys are deleted permanently you need to manually delete the metadata associated with the vault key.

**NOTE**

Execute this step only if Vault Key/Value (KV) secret engine API, version 2 is used for cluster-wide encryption with Key Management System (KMS) since the vault keys are marked as deleted and not permanently deleted during the uninstallation of OpenShift Container Storage. You can always restore it later if required.

- a. List the keys in the vault.

```
$ vault kv list <backend_path>
```

**<backend\_path>**

Is the path in the vault where the encryption keys are stored.  
For example:

```
$ vault kv list kv-v2
```

Example output:

```
Keys
-----
NOOBAA_ROOT_SECRET_PATH/
rook-ceph-osd-encryption-key-ocs-deviceset-thin-0-data-0m27q8
rook-ceph-osd-encryption-key-ocs-deviceset-thin-1-data-0sq227
rook-ceph-osd-encryption-key-ocs-deviceset-thin-2-data-0xzszb
```

- b. List the metadata associated with the vault key.

```
$ vault kv get kv-v2/<key>
```

For the Multicloud Object Gateway (MCG) key:

```
$ vault kv get kv-v2/NOOBAA_ROOT_SECRET_PATH/<key>
```

**<key>**

Is the encryption key.  
For Example:

```
$ vault kv get kv-v2/rook-ceph-osd-encryption-key-ocs-deviceset-thin-0-data-0m27q8
```

Example output:

```
=====  
=====  
Key      Value  
---      -  
created_time  2021-06-23T10:06:30.650103555Z  
deletion_time 2021-06-23T11:46:35.045328495Z  
destroyed     false  
version       1
```

- c. Delete the metadata.

```
$ vault kv metadata delete kv-v2/<key>
```

For the MCG key:

```
$ vault kv metadata delete kv-v2/NOOBAA_ROOT_SECRET_PATH/<key>
```

**<key>**

Is the encryption key.

For Example:

```
$ vault kv metadata delete kv-v2/rook-ceph-osd-encryption-key-ocs-deviceset-thin-0-
data-0m27q8
```

Example output:

```
Success! Data deleted (if it existed) at: kv-v2/metadata/rook-ceph-osd-encryption-key-
ocs-deviceset-thin-0-data-0m27q8
```

- d. Repeat these steps to delete the metadata associated with all the vault keys.
14. To ensure that OpenShift Container Storage is uninstalled completely, on the OpenShift Container Platform Web Console,
    - a. Click **Storage**.
    - b. Verify that **Overview** no longer appears under Storage.

## 5.2. REMOVING MONITORING STACK FROM OPENSIFT CONTAINER STORAGE

Use this section to clean up the monitoring stack from OpenShift Container Storage.

The PVCs that are created as a part of configuring the monitoring stack are in the **openshift-monitoring** namespace.

### Prerequisites

- PVCs are configured to use OpenShift Container Platform monitoring stack. For information, see [configuring monitoring stack](#).

### Procedure

1. List the pods and PVCs that are currently running in the **openshift-monitoring** namespace.

```
$ oc get pod,pvc -n openshift-monitoring
NAME                READY STATUS  RESTARTS  AGE
pod/alertmanager-main-0    3/3   Running  0         8d
pod/alertmanager-main-1    3/3   Running  0         8d
pod/alertmanager-main-2    3/3   Running  0         8d
pod/cluster-monitoring-
```



```

operator-84457656d-pkrxm      1/1  Running 0      8d
pod/grafana-79ccf6689f-2ll28  2/2  Running 0      8d
pod/kube-state-metrics-
7d86fb966-rvd9w              3/3  Running 0      8d
pod/node-exporter-25894      2/2  Running 0      8d
pod/node-exporter-4dsd7      2/2  Running 0      8d
pod/node-exporter-6p4zc      2/2  Running 0      8d
pod/node-exporter-jbjvg      2/2  Running 0      8d
pod/node-exporter-jj4t5      2/2  Running 0     6d18h
pod/node-exporter-k856s      2/2  Running 0     6d18h
pod/node-exporter-rf8gn      2/2  Running 0      8d
pod/node-exporter-rmb5m      2/2  Running 0     6d18h
pod/node-exporter-zj7kx      2/2  Running 0      8d
pod/openshift-state-metrics-
59dbd4f654-4clng            3/3  Running 0      8d
pod/prometheus-adapter-
5df5865596-k8dzn            1/1  Running 0     7d23h
pod/prometheus-adapter-
5df5865596-n2gj9            1/1  Running 0     7d23h
pod/prometheus-k8s-0          6/6  Running 1      8d
pod/prometheus-k8s-1          6/6  Running 1      8d
pod/prometheus-operator-
55cfb858c9-c4zd9            1/1  Running 0     6d21h
pod/telemeter-client-
78fc8fc97d-2rgfp            3/3  Running 0      8d

```

NAME	CAPACITY	ACCESS MODES	STATUS	VOLUME	STORAGECLASS	AGE
persistentvolumeclaim/my-alertmanager-claim-alertmanager-main-0	40Gi	RWO	Bound	pvc-0d519c4f-15a5-11ea-baa0-026d231574aa	ocs-storagecluster-ceph-rbd	8d
persistentvolumeclaim/my-alertmanager-claim-alertmanager-main-1	40Gi	RWO	Bound	pvc-0d5a9825-15a5-11ea-baa0-026d231574aa	ocs-storagecluster-ceph-rbd	8d
persistentvolumeclaim/my-alertmanager-claim-alertmanager-main-2	40Gi	RWO	Bound	pvc-0d6413dc-15a5-11ea-baa0-026d231574aa	ocs-storagecluster-ceph-rbd	8d
persistentvolumeclaim/my-prometheus-claim-prometheus-k8s-0	40Gi	RWO	Bound	pvc-0b7c19b0-15a5-11ea-baa0-026d231574aa	ocs-storagecluster-ceph-rbd	8d
persistentvolumeclaim/my-prometheus-claim-prometheus-k8s-1	40Gi	RWO	Bound	pvc-0b8aed3f-15a5-11ea-baa0-026d231574aa	ocs-storagecluster-ceph-rbd	8d

2. Edit the monitoring **configmap**.

```
$ oc -n openshift-monitoring edit configmap cluster-monitoring-config
```

3. Remove any **config** sections that reference the OpenShift Container Storage storage classes as shown in the following example and save it.

**Before editing**

```
.  
. .  
apiVersion: v1  
data:  
  config.yaml: |  
    alertmanagerMain:  
      volumeClaimTemplate:  
        metadata:  
          name: my-alertmanager-claim  
        spec:  
          resources:  
            requests:  
              storage: 40Gi  
          storageClassName: ocs-storagecluster-ceph-rbd  
  prometheusK8s:  
    volumeClaimTemplate:  
      metadata:  
        name: my-prometheus-claim  
      spec:  
        resources:  
          requests:  
            storage: 40Gi  
        storageClassName: ocs-storagecluster-ceph-rbd  
kind: ConfigMap  
metadata:  
  creationTimestamp: "2019-12-02T07:47:29Z"  
  name: cluster-monitoring-config  
  namespace: openshift-monitoring  
  resourceVersion: "22110"  
  selfLink: /api/v1/namespaces/openshift-monitoring/configmaps/cluster-monitoring-config  
  uid: fd6d988b-14d7-11ea-84ff-066035b9efa8  
. . .
```

After editing

```

.
.
.
apiVersion: v1
data:
  config.yaml: |
kind: ConfigMap
metadata:
  creationTimestamp: "2019-11-21T13:07:05Z"
  name: cluster-monitoring-config
  namespace: openshift-monitoring
  resourceVersion: "404352"
  selfLink: /api/v1/namespaces/openshift-monitoring/configmaps/cluster-monitoring-config
  uid: d12c796a-0c5f-11ea-9832-063cd735b81c
.
.
.

```

In this example, **alertmanagerMain** and **prometheusK8s** monitoring components are using the OpenShift Container Storage PVCs.

4. Delete relevant PVCs. Make sure you delete all the PVCs that are consuming the storage classes.

```
$ oc delete -n openshift-monitoring pvc <pvc-name> --wait=true --timeout=5m
```

### 5.3. REMOVING OPENSIFT CONTAINER PLATFORM REGISTRY FROM OPENSIFT CONTAINER STORAGE

To clean the OpenShift Container Platform registry from OpenShift Container Storage, follow the steps in the procedure.

If you want to configure an alternative storage, see [image registry](#)

The PVCs created as a part of configuring OpenShift Container Platform registry are in the **openshift-image-registry** namespace.

#### Prerequisites

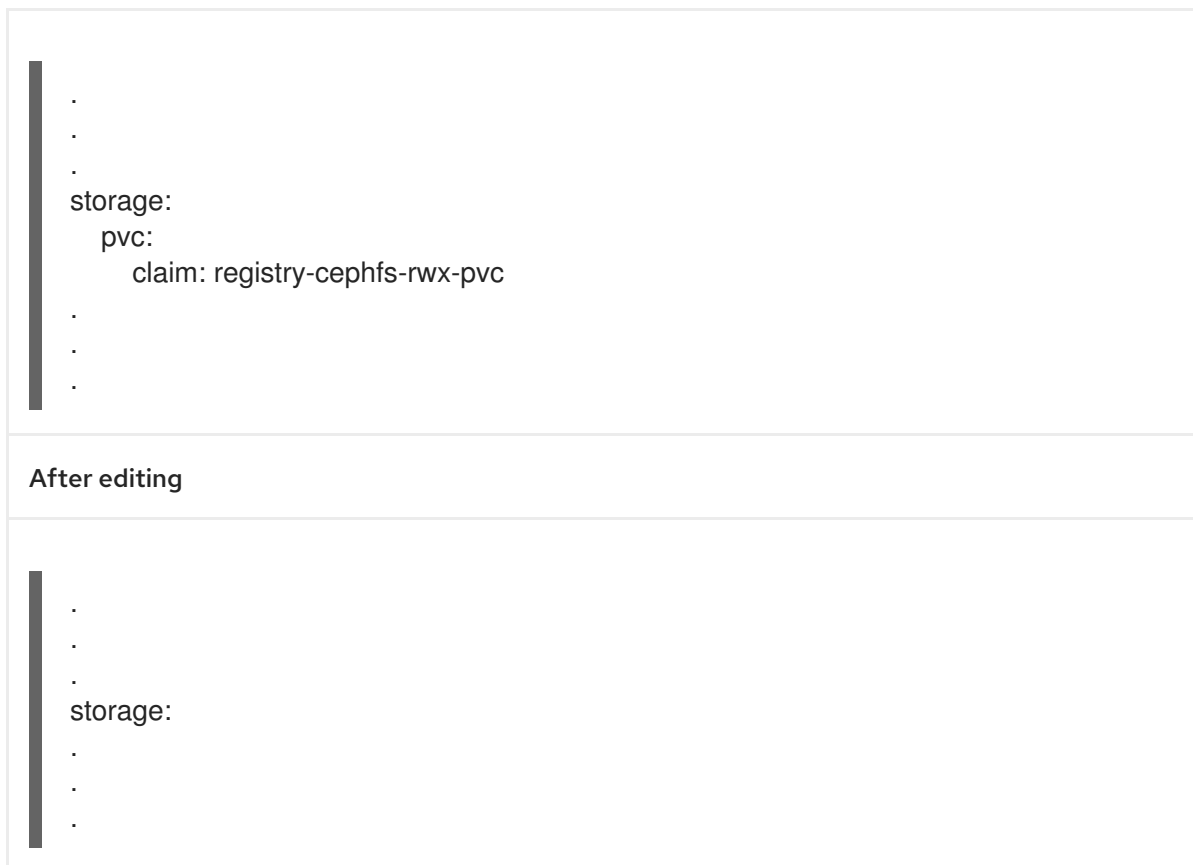
- The image registry must be configured to use an OpenShift Container Storage PVC.

#### Procedure

1. Edit the **configs.imageregistry.operator.openshift.io** object and remove the content in the **storage** section.

```
$ oc edit configs.imageregistry.operator.openshift.io
```

Before editing



In this example, the PVC is called **registry-cephfs-rwx-pvc**, which is now safe to delete.

2. Delete the PVC.

```
$ oc delete pvc <pvc-name> -n openshift-image-registry --wait=true --timeout=5m
```

## 5.4. REMOVING THE CLUSTER LOGGING OPERATOR FROM OPENSIFT CONTAINER STORAGE

To clean the cluster logging operator from the OpenShift Container Storage, follow the steps in the procedure.

The PVCs created as a part of configuring cluster logging operator are in the **openshift-logging** namespace.

### Prerequisites

- The cluster logging instance must be configured to use OpenShift Container Storage PVCs.

### Procedure

1. Remove the **ClusterLogging** instance in the namespace.

```
$ oc delete clusterlogging instance -n openshift-logging --wait=true --timeout=5m
```

The PVCs in the **openshift-logging** namespace are now safe to delete.

2. Delete PVCs.

```
$ oc delete pvc <pvc-name> -n openshift-logging --wait=true --timeout=5m
```