



Red Hat OpenShift Container Storage 4.5

Deploying OpenShift Container Storage in external mode

How to install and set up your environment

Red Hat OpenShift Container Storage 4.5 Deploying OpenShift Container Storage in external mode

How to install and set up your environment

Legal Notice

Copyright © 2021 Red Hat, Inc.

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

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

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

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

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

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

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

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

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

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

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

All other trademarks are the property of their respective owners.

Abstract

Read this document for instructions on installing Red Hat OpenShift Container Storage 4.5 to use an external Red Hat Ceph Storage cluster.

Table of Contents

CHAPTER 1. OVERVIEW OF DEPLOYING IN EXTERNAL MODE	3
CHAPTER 2. ENABLING FILE SYSTEM ACCESS FOR CONTAINERS ON RED HAT ENTERPRISE LINUX BASED NODES	4
CHAPTER 3. INSTALLING RED HAT OPENSIFT CONTAINER STORAGE OPERATOR	5
CHAPTER 4. CREATING AN OPENSIFT CONTAINER STORAGE CLUSTER SERVICE FOR EXTERNAL MODE	8
CHAPTER 5. VERIFYING YOUR OPENSIFT CONTAINER STORAGE INSTALLATION FOR EXTERNAL MODE	13
5.1. VERIFY THAT THE PODS ARE IN RUNNING STATE	13
5.2. VERIFY THAT THE OPENSIFT CONTAINER STORAGE CLUSTER IS HEALTHY	13
5.3. VERIFY THAT THE MULTICLOUD OBJECT GATEWAY IS HEALTHY	14
5.4. VERIFY THAT THE STORAGE CLASSES ARE CREATED AND LISTED	15
5.5. VERIFY THAT CEPH CLUSTER IS CONNECTED	16
5.6. VERIFY THAT STORAGE CLUSTER IS READY	16
CHAPTER 6. UNINSTALLING OPENSIFT CONTAINER STORAGE	17
6.1. UNINSTALLING OPENSIFT CONTAINER STORAGE ON EXTERNAL MODE	17
6.2. REMOVING MONITORING STACK FROM OPENSIFT CONTAINER STORAGE	20
6.3. REMOVING OPENSIFT CONTAINER PLATFORM REGISTRY FROM OPENSIFT CONTAINER STORAGE	24
6.4. REMOVING THE CLUSTER LOGGING OPERATOR FROM OPENSIFT CONTAINER STORAGE	25

CHAPTER 1. OVERVIEW OF DEPLOYING IN EXTERNAL MODE

Red Hat OpenShift Container Storage can use an externally hosted Red Hat Ceph Storage (RHCS) cluster as the storage provider. This deployment type is supported for bare metal and user-provisioned VMware environments. See [Planning your deployment](#) for more information.

For instructions regarding how to install a RHCS 4 cluster, see [Installation guide](#).

Follow these steps to deploy OpenShift Container Storage in external mode:

1. If you use Red Hat Enterprise Linux hosts, [Enable file system access for containers](#).
Skip this step if you use Red Hat Enterprise Linux CoreOS (RHCOS) hosts.
2. [Install the OpenShift Container Storage Operator](#).
3. [Create the OpenShift Container Storage Cluster Service](#).

CHAPTER 2. ENABLING FILE SYSTEM ACCESS FOR CONTAINERS ON RED HAT ENTERPRISE LINUX BASED NODES

Deploying OpenShift Container Platform on a Red Hat Enterprise Linux base in a user provisioned infrastructure (UPI) does not automatically provide container access to the underlying Ceph file system.



NOTE

This process is not necessary for hosts based on Red Hat Enterprise Linux CoreOS.

Procedure

Perform the following steps on each node in your cluster.

1. Log in to the Red Hat Enterprise Linux based node and open a terminal.
2. Verify that the node has access to the `rhel-7-server-extras-rpms` repository.

```
# subscription-manager repos --list-enabled | grep rhel-7-server
```

If you do not see both **rhel-7-server-rpms** and **rhel-7-server-extras-rpms** in the output, or if there is no output, run the following commands to enable each repository.

```
# subscription-manager repos --enable=rhel-7-server-rpms
# subscription-manager repos --enable=rhel-7-server-extras-rpms
```

3. Install the required packages.

```
# yum install -y policycoreutils container-selinux
```

4. Persistently enable container use of the Ceph file system in SELinux.

```
# setsebool -P container_use_cephfs on
```


CHAPTER 3. 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. For information about the hardware and software requirements, see [Planning your deployment](#).

Prerequisites

- You must be logged into the OpenShift Container Platform cluster.



NOTE

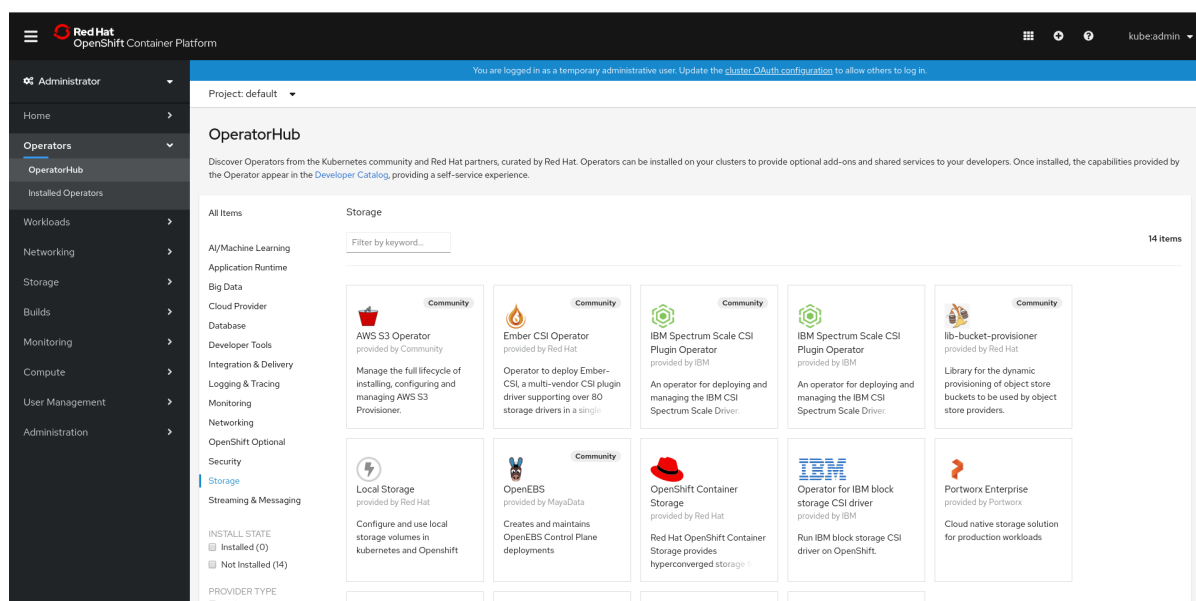
When you need to override the cluster-wide default node selector for OpenShift Container Storage, you can use the following command in command line interface to specify a blank node selector for the **openshift-storage** namespace:

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

Procedure

- Click **Operators** → **OperatorHub** in the left pane of the OpenShift Web Console.

Figure 3.1. List of operators in the Operator Hub



- Click on **OpenShift Container Storage**.
You can use the **Filter by keyword** text box or the filter list to search for OpenShift Container Storage from the list of operators.
- On the OpenShift Container Storage operator page, click **Install**.
- On the **Install Operator** page, ensure the following options are selected:
 - Update Channel as **stable-4.5**
 - Installation Mode as **A specific namespace on the cluster**

- c. Installed Namespace as **Operator recommended namespace PR openshift-storage**. If Namespace **openshift-storage** does not exist, it will be created during the operator installation.
- d. Select **Approval Strategy** as **Automatic** or **Manual**. Approval Strategy is set to **Automatic** by default.
 - **Approval Strategy as Automatic.**

**NOTE**

When you select the Approval Strategy as **Automatic**, approval is not required either during fresh installation or when updating to the latest version of OpenShift Container Storage.

- i. Click **Install**
 - ii. Wait for the install to initiate. This may take up to 20 minutes.
 - iii. Click **Operators → Installed Operators**
 - iv. Ensure the **Project** is **openshift-storage**. By default, the **Project** is **openshift-storage**.
 - v. Wait for the **Status** of **OpenShift Container Storage** to change to **Succeeded**.
- **Approval Strategy as Manual.**

**NOTE**

When you select the Approval Strategy as **Manual**, approval is required during fresh installation or when updating to the latest version of OpenShift Container Storage.

- i. Click **Install**.
- ii. On the **Installed Operators** page, click **ocs-operator**.
- iii. On the **Subscription Details** page, click the **Install Plan** link.
- iv. On the **InstallPlan Details** page, click **Preview Install Plan**.
- v. Review the install plan and click **Approve**.
- vi. Wait for the **Status** of the **Components** to change from **Unknown** to either **Created** or **Present**.
- vii. Click **Operators → Installed Operators**
- viii. Ensure the **Project** is **openshift-storage**. By default, the **Project** is **openshift-storage**.
- ix. Wait for the **Status** of **OpenShift Container Storage** to change to **Succeeded**.

Verification steps

- Verify that OpenShift Container Storage Operator shows the Status as **Succeeded** on the Installed Operators dashboard.

CHAPTER 4. CREATING AN OPENSIFT CONTAINER STORAGE CLUSTER SERVICE FOR EXTERNAL MODE

You need to create a new OpenShift Container Storage cluster service after you install OpenShift Container Storage operator on OpenShift Container Platform deployed on user provisioned infrastructures VMware vSphere or Bare metal platform.

Prerequisites

- You must be logged into the working OpenShift Container Platform version 4.5.4 or above.
- OpenShift Container Storage operator must be installed. For more information, see [Installing OpenShift Container Storage Operator using the Operator Hub](#).
- Red Hat Ceph Storage version 4.1.1 or later is required for the external cluster. For more information, see this [knowledge base article on Red Hat Ceph Storage releases and corresponding Ceph package versions](#).

If you have updated the Red Hat Ceph Storage cluster to version 4.1.1 or later from a previous release and is not a freshly deployed cluster, you must manually set the application type for CephFS pool on the Red Hat Ceph Storage cluster to enable CephFS PVC creation in external mode.

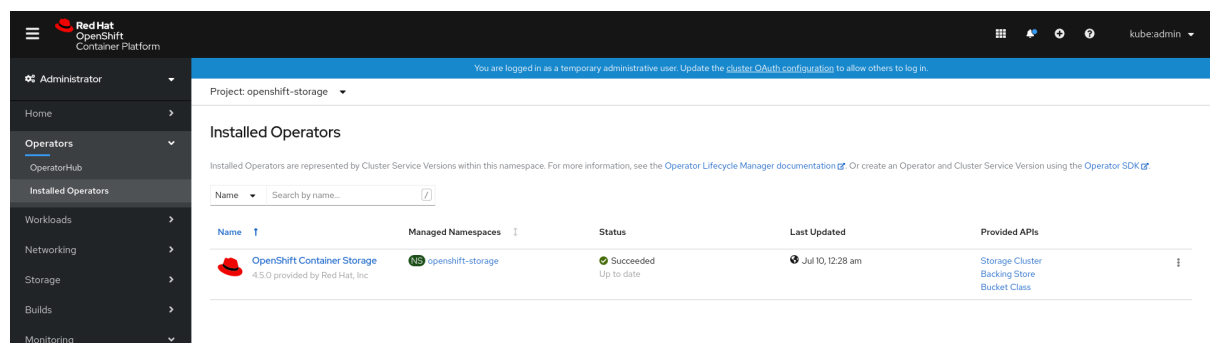
For more details, see [Troubleshooting CephFS PVC creation in external mode](#).

- It is recommended that the external Red Hat Ceph Storage cluster has the PG Autoscaler enabled, with a target_size_ratio of 0.49. For more information, see [The placement group autoscaler](#) section in the Red Hat Ceph Storage documentation.
- The external Ceph cluster should have an existing RBD pool pre-configured for use. If it does not exist, contact your Red Hat Ceph Storage administrator to create one before you move ahead with OpenShift Container Storage deployment.

Procedure

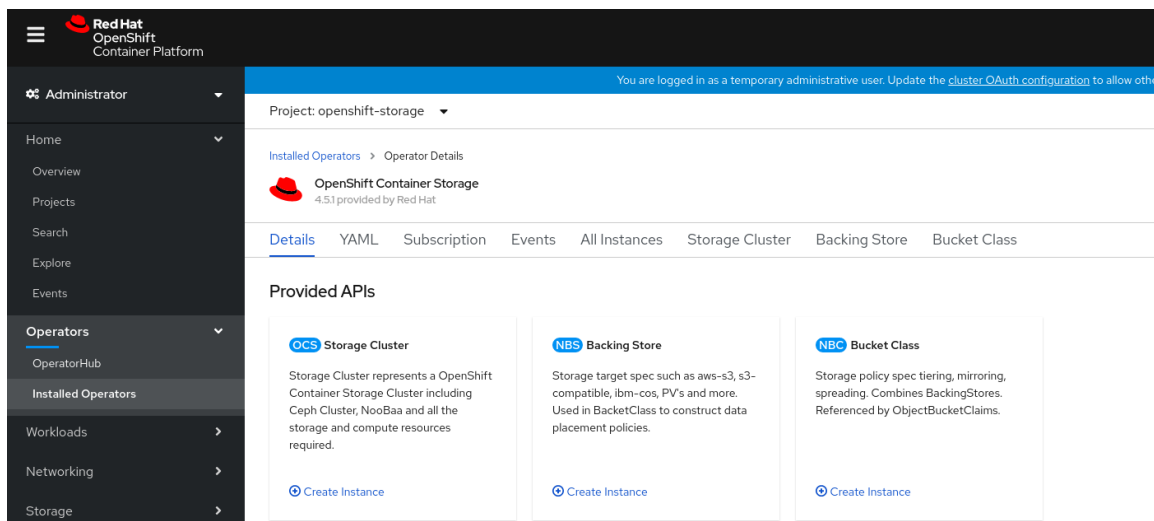
1. Click **Operators** → **Installed Operators** from the OpenShift Web Console to view the installed operators. Ensure that the **Project** selected is **openshift-storage**.
2. On the **Installed Operators** page, Click **Openshift Container Storage**.

Figure 4.1. OpenShift Container Storage Operator page



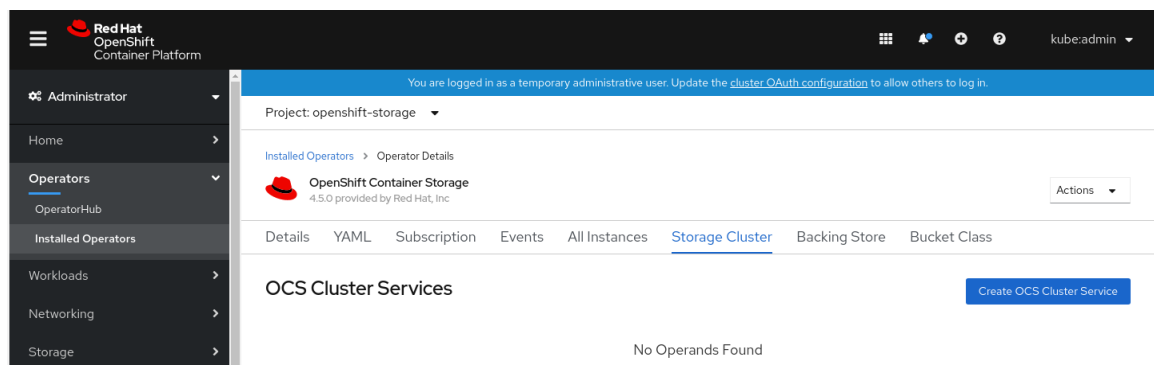
3. On the **Installed Operators** → **Operator Details** page, perform either of the following to create a Storage Cluster Service.
 - a. On the **Details tab** → **Provided APIs** → **OCS Storage Cluster**, click **Create Instance**.

Figure 4.2. Operator Details Page



- b. Alternatively, select the **Storage cluster** tab and click **Create OCS Cluster Service**.

Figure 4.3. Storage Cluster tab



4. On the **Create Storage Cluster** page, ensure that the following options are selected:

Figure 4.4. Connect to external cluster section on Create Storage Cluster form

Project: openshift-storage ▾

OpenShift Container Storage > Create OCS Cluster Service

Create Storage Cluster

OCS runs as a cloud-native service for optimal integration with applications in need of storage, and handles the scenes such as provisioning and management.

Select Mode

Internal

External

Connect to external cluster

Download [ceph-external-cluster-details-exporter.py](#) script and run on the RHCS cluster, then upload the results(JSON) in the External cluster metadata field. [Download Script](#)

A bucket will be created to provide the OCS Service.

External cluster metadata *

Upload Credentials file

- Select Mode as **External**. By default, Internal is selected as deployment mode.
- In the Connect to external cluster section, click on the **Download Script** link to download the python script for extracting Ceph cluster details.
- For extracting the Red Hat Ceph Storage (RHCS) cluster details, contact the RHCS admin to run the downloaded python script on a Red Hat Ceph Storage client node.
 - Run the following command on the RHCS client node to view the list of available arguments.

```
# python3 ceph-external-cluster-details-exporter.py --help
```



IMPORTANT

Use **python** instead of **python3** if the Red Hat Ceph Storage 4.x cluster is deployed on Red Hat Enterprise Linux 7.x (RHEL 7.x) cluster.



NOTE

If you do not have access to the RHCS client node, you can also run the script from inside a MON container (containerized deployment) or from a MON node (rpm deployment).

- To retrieve the external cluster details from the RHCS cluster, run the following command

```
# python3 ceph-external-cluster-details-exporter.py --rbd-data-pool-name <rbd block pool name> [optional arguments]
```

For example

```
# python3 ceph-external-cluster-details-exporter.py --rbd-data-pool-name ceph-rbd --rgw-endpoint xxx.xxx.xxx.xxx:xxxx --run-as-user client.ocs
```

In the above example,

- **--rbd-data-pool-name** is a mandatory parameter used for providing Block Storage in OpenShift Container Storage.
- **--rgw-endpoint** is optional. Provide this parameter if object storage is to be provisioned through Ceph Rados Gateway for OpenShift Container Storage.
- **--run-as-user** is an optional parameter used for providing a name for the Ceph user which is created by the script. If this parameter is not specified, a default user name **client.healthchecker** is created. The permissions for the new user is set as:
 - caps: [mon] allow r, allow command quorum_status
 - caps: [osd] allow rwx pool=**RGW_POOL_PREFIX.rgw.meta**, allow r pool=**rgw.root**, allow rw pool=**RGW_POOL_PREFIX.rgw.control**, allow x pool=**RGW_POOL_PREFIX.rgw.buckets.index**

Example of JSON output generated using the python script:

```
[{"name": "rook-ceph-mon-endpoints", "kind": "ConfigMap", "data": {"data": {"ceph-mon-node=xxx.xxx.xxx.xxx:xxxx", "maxMonId": "0", "mapping": "{}"}, {"name": "rook-ceph-mon", "kind": "Secret", "data": {"admin-secret": "<admin-secret>", "cluster-name": "openshift-storage", "fsid": "<fs-id>", "mon-secret": "<mon-secret>"}}, {"name": "rook-ceph-operator-creds", "kind": "Secret", "data": {"userID": "client.healthchecker", "userKey": "<user-key>"}}, {"name": "rook-csi-rbd-node", "kind": "Secret", "data": {"userID": "csi-rbd-node", "userKey": "<user-key>"}}, {"name": "rook-csi-rbd-provisioner", "kind": "Secret", "data": {"userID": "csi-rbd-provisioner", "userKey": "<user-key>"}}, {"name": "rook-csi-cephfs-node", "kind": "Secret", "data": {"adminID": "csi-cephfs-node", "adminKey": "<admin-key>"}}, {"name": "rook-csi-cephfs-provisioner", "kind": "Secret", "data": {"adminID": "csi-cephfs-provisioner", "adminKey": "<admin-key>"}}, {"name": "ceph-rbd", "kind": "StorageClass", "data": {"pool": "ceph-rbd"}}, {"name": "cephfs", "kind": "StorageClass", "data": {"fsName": "cephfs", "pool": "cephfs_data"}}, {"name": "ceph-rgw", "kind": "StorageClass", "data": {"endpoint": "xxx.xxx.xxx.xxx:xxxx"}}]
```

- Save the JSON output to a file with **.json** extension

**NOTE**

For OpenShift Container Storage to work seamlessly, ensure that the parameters (RGW endpoint, CephFS details, RBD pool, etc.) to be uploaded using the JSON file remains unchanged on the RHCS external cluster, post storage cluster creation.

- d. Click **External cluster metadata** → **Browse** to select and upload the json file. The json file content will be populated and displayed in the text box.

Figure 4.5. Json file content

External cluster metadata *

Upload Credentials file Browse

```
ROOK CEPH MON ENDPOINTS = {"maxMonId":"0","data":{"ceph-mon-node=[xxx.xxx.xxx.xxx:xxxx],"mapping":{"{}}"}
ROOK CEPH MON = {"mon-secret":"mon-secret","fsid":"<fs-id>","cluster-name":"openshift-storage","admin-secret":"
<admin-secret>"}
ROOK CEPH OPERATOR CREDS = {"userKey":"<user-Key>","userID":"client.healthchecker"}
ROOK CSI RBD NODE = {"userKey":"<user-Key>","userID":"csi-rbd-node"}
ROOK CSI RBD PROVISIONER = {"userKey":"<user-Key>","userID":"csi-rbd-provisioner"}
ROOK CSI CEPHFS NODE = {"adminID":"csi-cephfs-node","adminKey":"<admin-Key>"}
ROOK CSI CEPHFS PROVISIONER = {"adminID":"csi-cephfs-provisioner","adminKey":"<admin-Key>"}
CEPH RBD = {"pool":"blockpoolname"}
CEPHFS = {"pool":"cephfs_data","fsName":"cephfs"}
```

5. Click **Create**.

**NOTE**

The Create button is enabled only after you upload the .json file.

Verification steps

- Verify that OpenShift Container Storage Cluster Service shows the Status as **Ready**.

Figure 4.6. Storage Cluster tab showing storage cluster service.

Project: openshift-storage ▾

Installed Operators > Operator Details

OpenShift Container Storage
4.5.0 provided by Red Hat, Inc Actions ▾

Details Storage Cluster Backing Store Bucket Class

OCS Cluster Services Create OCS Cluster Service

Name ▾ Search by name...

Name ↑	Kind ↓	Status ↓	Labels ↓	Last Updated ↓
ocs-external-storagecluster	StorageCluster	Phase: ● Ready	No labels	🕒 Jul 30, 2:13 am

- To verify that OpenShift Container Storage, pods and StorageClass are successfully installed, see [Verifying your external mode OpenShift Container Storage installation](#) .

CHAPTER 5. VERIFYING YOUR OPENSIFT CONTAINER STORAGE INSTALLATION FOR EXTERNAL MODE

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

5.1. VERIFY THAT THE PODS ARE IN RUNNING STATE

1. Click **Workloads** → **Pods** from the left pane of the OpenShift Web Console.
2. 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 5.1, "Pods corresponding to OpenShift Container Storage components"](#)
3. Verify that the following pods are in running state:

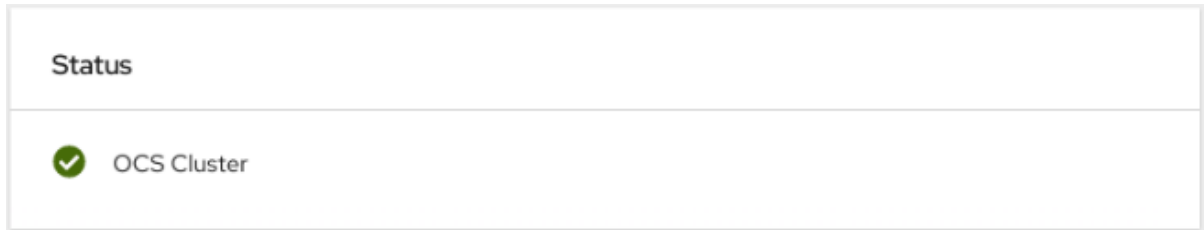
Table 5.1. Pods corresponding to OpenShift Container Storage components

Component	Corresponding pods
OpenShift Container Storage Operator	<p>ocs-operator-*</p> <p>(1 pod on any worker node)</p>
Rook-ceph Operator	<p>rook-ceph-operator-*</p> <p>(1 pod on any worker node)</p>
Multicloud Object Gateway	<ul style="list-style-type: none"> ● noobaa-operator-* (1 pod on any worker node) ● noobaa-core-* (1 pod on any worker node) ● nooba-db-* (1 pod on any worker node) ● noobaa-endpoint-* (1 pod on any worker node)
CSI	<ul style="list-style-type: none"> ● cephfs <ul style="list-style-type: none"> ○ csi-cephfsplugin-* (1 pod on each worker node) ○ csi-cephfsplugin-provisioner-* (2 pods distributed across worker nodes) ● rbd <ul style="list-style-type: none"> ○ csi-rbdplugin-* (1 pod on each worker node) ○ csi-rbdplugin-provisioner-* (2 pods distributed across worker nodes)

5.2. VERIFY THAT THE OPENSIFT CONTAINER STORAGE CLUSTER IS HEALTHY

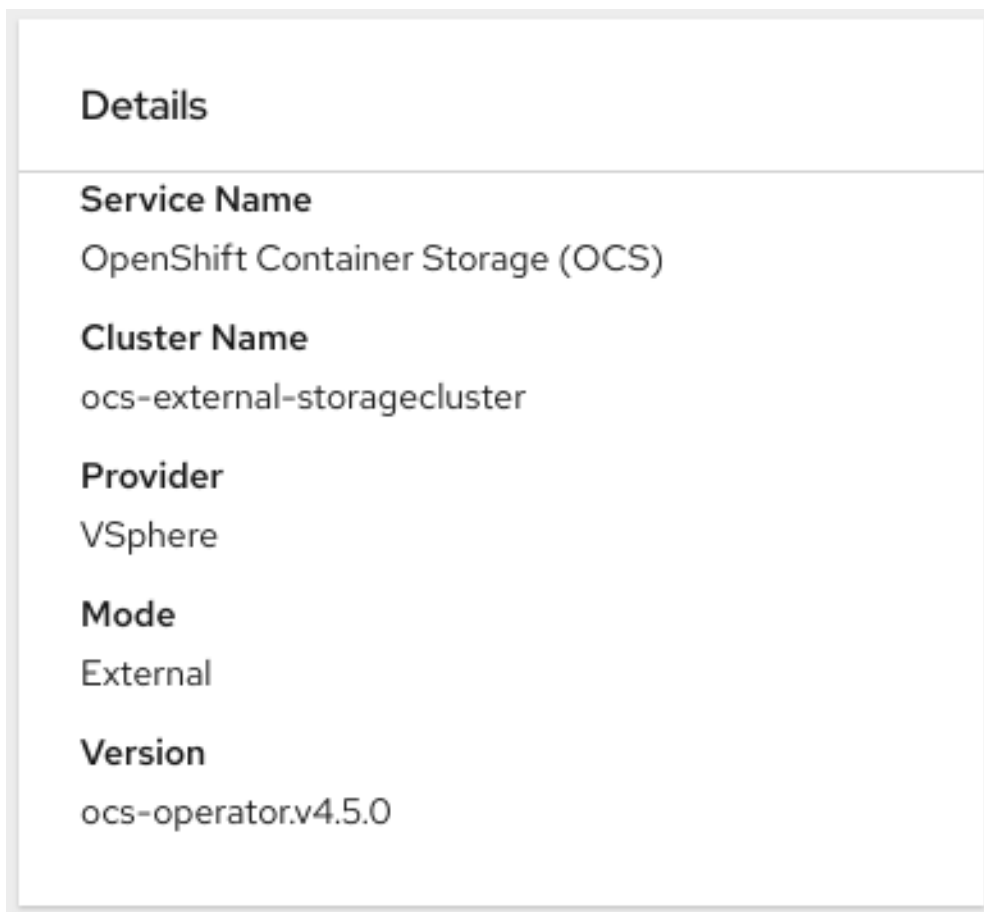
- Click **Home** → **Overview** from the left pane of the OpenShift Web Console and click **Persistent Storage** tab.
- In the **Status card**, verify that *OCS Cluster* has a green tick mark as shown in the following image:

Figure 5.1. Health status card in Persistent Storage Overview Dashboard



- In the **Details card**, verify that the cluster information is displayed appropriately as follows:

Figure 5.2. Details card in Persistent Storage Overview Dashboard

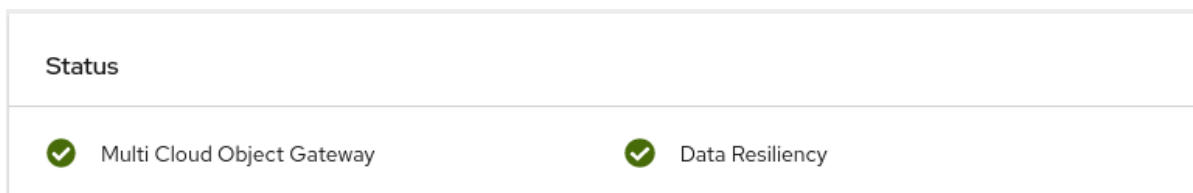


For more information on verifying the health of OpenShift Container Storage cluster using the persistent storage dashboard, see [Monitoring OpenShift Container Storage](#) .

5.3. VERIFY THAT THE MULTICLOUD OBJECT GATEWAY IS HEALTHY

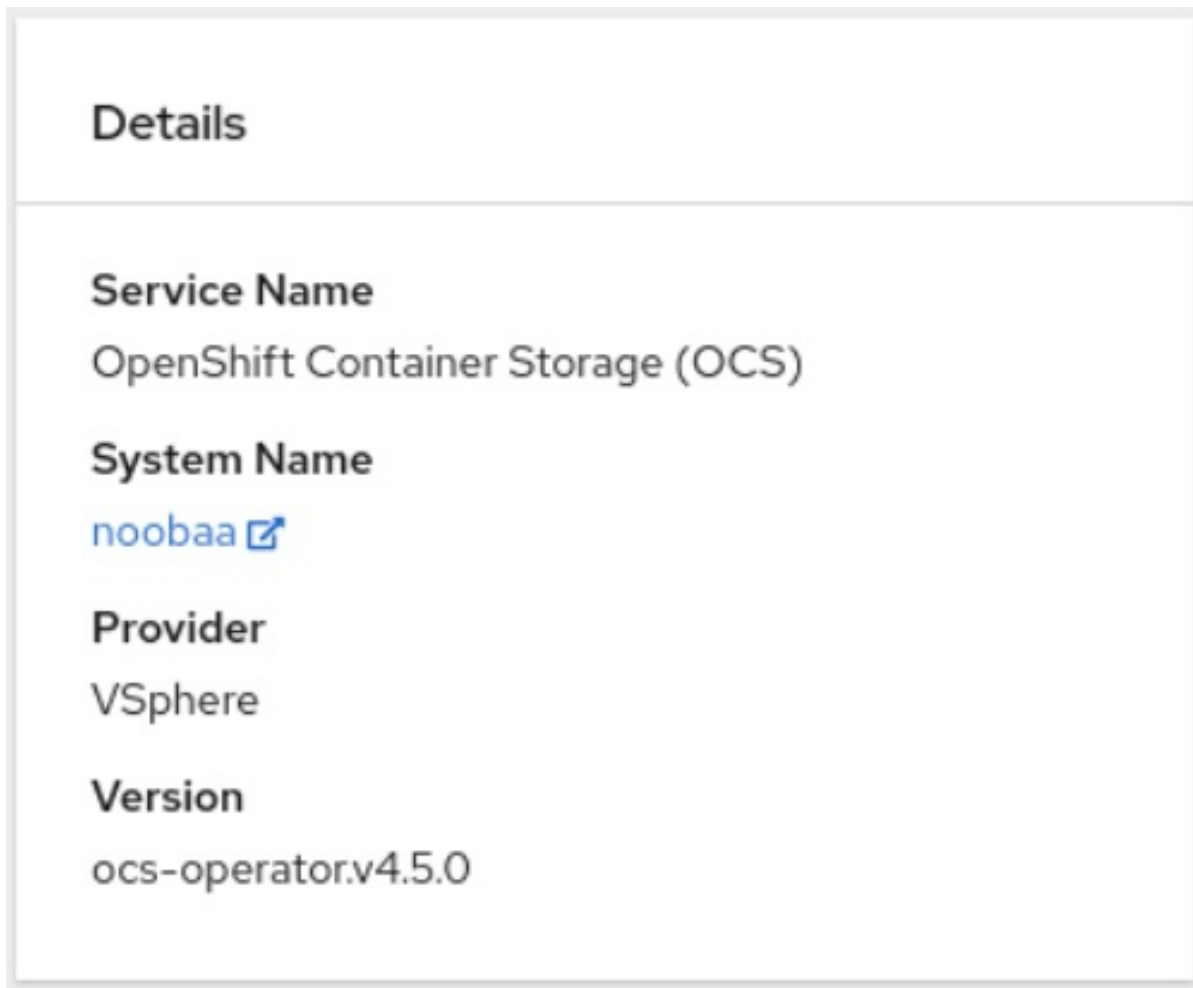
- Click **Home** → **Overview** from the left pane of the OpenShift Web Console and click the **Object Service** tab.
- In the **Status card**, verify that the Multicloud Object Gateway (MCG) storage displays a green tick icon as shown in following image:

Figure 5.3. Health status card in Object Service Overview Dashboard



- In the **Details card**, verify that the MCG information is displayed appropriately as follows:

Figure 5.4. Details card in Object Service Overview Dashboard



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

5.4. VERIFY THAT THE STORAGE CLASSES ARE CREATED AND LISTED

- Click **Storage** → **Storage Classes** from the left pane of the OpenShift Web Console.
- Verify that the following storage classes are created with the OpenShift Container Storage cluster creation:
 - **ocs-external-storagecluster-ceph-rbd**
 - **ocs-external-storagecluster-ceph-rgw**
 - **ocs-external-storagecluster-cephfs**

- **openshift-storage.noobaa.io**

**NOTE**

- If an MDS is not deployed in the external cluster, **ocs-external-storagecluster-cephfs** storage class will not be created.
- If an RGW is not deployed in the external cluster, the **ocs-external-storagecluster-ceph-rgw** storage class will not be created.

For more information regarding MDS and RGW, see [Red Hat Ceph Storage documentation](#)

5.5. VERIFY THAT CEPH CLUSTER IS CONNECTED

Run the following command to verify if the OpenShift Container Storage cluster is connected to Ceph cluster.

```
$ oc get cephcluster -n openshift-storage
```

NAME	DATADIRHOSTPATH	MONCOUNT	AGE	PHASE
MESSAGE	HEALTH			
ocs-external-storagecluster-cephcluster		31m15s	Connected	Cluster
connected successfully	HEALTH_OK			

5.6. VERIFY THAT STORAGE CLUSTER IS READY

Run the following command to verify if the storage cluster is ready and the **External** option is set to true.

```
$ oc get storagecluster -n openshift-storage
```

NAME	AGE	PHASE	EXTERNAL	CREATED AT	VERSION
ocs-external-storagecluster	31m15s	Ready	true	2020-07-29T20:43:04Z	4.5.0

CHAPTER 6. UNINSTALLING OPENSIFT CONTAINER STORAGE

6.1. UNINSTALLING OPENSIFT CONTAINER STORAGE ON EXTERNAL MODE

Use the steps in this section to uninstall OpenShift Container Storage instead of the Uninstall option from the user interface. Uninstalling OpenShift Container Storage will neither remove the RBD pool from the external cluster nor uninstall the external RedHat Ceph Storage cluster.

Prerequisites

- Make sure that the OpenShift Container Storage cluster is in a healthy state. The deletion might fail if some of the pods are not terminated successfully due to insufficient resources or nodes. In case the cluster is in an unhealthy state, you should contact Red Hat Customer Support before uninstalling OpenShift Container Storage.
- Make sure that applications are not consuming persistent volume claims (PVCs) or object bucket claims (OBCs) using the storage classes provided by OpenShift Container Storage. PVCs and OBCs will be deleted during the uninstall process.

Procedure

1. Query for PVCs and OBCs that use the OpenShift Container Storage based storage class provisioners.

For example :

```
$ oc get pvc -o=jsonpath='{range .items[?(@.spec.storageClassName=="ocs-external-storagecluster-ceph-rbd")]}{"Name: "}{@.metadata.name}" Namespace: "}{@.metadata.namespace}" Labels: "}{@.metadata.labels}"\n"}{end}' --all-namespaces|awk '! ( /Namespace: openshift-storage/ && /app:noobaa/ ) | grep -v noobaa-default-backing-store-noobaa-pvc
```



NOTE

If the external RedHat Ceph Storage cluster is not configured for CephFS, you can ignore the following query command for **ocs-external-storagecluster-cephfs**.

```
$ oc get pvc -o=jsonpath='{range .items[?(@.spec.storageClassName=="ocs-external-storagecluster-cephfs")]}{"Name: "}{@.metadata.name}" Namespace: "}{@.metadata.namespace}"\n"}{end}' --all-namespaces
```



NOTE

If the external RedHat Ceph Storage cluster is not configured for Object Storage, you can ignore the following query command for **ocs-external-storagecluster-ceph-rgw**.

```
$ oc get obc -o=jsonpath='{range .items[?(@.spec.storageClassName=="ocs-external-storagecluster-ceph-rgw")]}{"Name: "}{@.metadata.name}{ " Namespace: "}{@.metadata.namespace}{ "\n"}{end}' --all-namespaces
```

```
$ oc get obc -o=jsonpath='{range .items[?(@.spec.storageClassName=="openshift-storage.noobaa.io")]}{"Name: "}{@.metadata.name}{ " Namespace: "}{@.metadata.namespace}{ "\n"}{end}' --all-namespaces
```

2. Follow these instructions to ensure the PVCs and OBCs listed in the previous step are deleted. If you have created PVCs as a part of configuring the monitoring stack, cluster logging operator, or image registry, then you must perform the clean up steps provided in the following sections as required:

- [Section 6.2, "Removing monitoring stack from OpenShift Container Storage"](#)
 - [Section 6.3, "Removing OpenShift Container Platform registry from OpenShift Container Storage"](#)
 - [Section 6.4, "Removing the cluster logging operator from OpenShift Container Storage"](#)
- For each of the remaining PVCs or OBCs, follow the steps mentioned below:

- a. Determine the pod that is consuming the PVC or OBC.
- b. Identify the controlling API object such as a **Deployment**, **StatefulSet**, **DaemonSet**, **Job**, or a custom controller. Each API object has a metadata field known as **OwnerReference**. This is a list of associated objects. The **OwnerReference** with the **controller** field set to true will point to controlling objects such as **ReplicaSet**, **StatefulSet**, **DaemonSet** and so on.
- c. Ensure that the API object is not consuming PVC or OBC provided by OpenShift Container Storage. Either the object should be deleted or the storage should be replaced. Ask the owner of the project to make sure that it is safe to delete or modify the object.



NOTE

You can ignore the **noobaa** pods.

- d. Delete the OBCs.

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

- e. Delete any custom Bucket Class you have created.

```
$ oc get bucketclass -A | grep -v noobaa-default-bucket-class
```

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

- f. If you have created any custom Multi Cloud Gateway backingstores, delete each of them.
 - i. List and note the backingstores.

```
for bs in $(oc get backingstore -o name -n openshift-storage | grep -v noobaa-
default-backing-store); do echo "Found backingstore $bs"; echo "Its has the
following pods running :"; echo "$(oc get pods -o name -n openshift-storage |
grep $(echo ${bs} | cut -f2 -d/))"; done
```

- ii. Delete each of the backingstores listed above and confirm that the corresponding pods and PVCs are deleted.

```
for bs in $(oc get backingstore -o name -n openshift-storage | grep -v noobaa-
default-backing-store); do echo "Deleting Backingstore $bs"; oc delete -n
openshift-storage $bs; done
```

- iii. If any of the backingstores listed above were based on the pv-pool, ensure that the corresponding pod and PVC are also deleted.

```
$ oc get pods -n openshift-storage | grep noobaa-pod | grep -v noobaa-default-
backing-store-noobaa-pod
```

```
$ oc get pvc -n openshift-storage --no-headers | grep -v noobaa-db | grep -v
noobaa-default-backing-store-noobaa-pvc
```

- g. Delete the remaining PVCs listed in Step 1.

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

3. Delete the **StorageCluster** object and wait for the removal of the associated resources.

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

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

- a. Switch to another namespace if openshift-storage is the active namespace.

For example :

```
$ oc project default
```

- b. Delete the **openshift-storage** namespace.

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

- c. Wait for approximately five minutes and confirm if the project is deleted successfully.

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

Output:

```
Error from server (NotFound): namespaces "openshift-storage" not found
```

**NOTE**

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

5. Delete the **openshift-storage.noobaa.io** storage class.

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

6. Confirm all PVs are deleted. If there is any PV left in the Released state, delete it.

```
# oc get pv|egrep 'ocs-external-storagecluster-ceph-rbd|ocs-external-storagecluster-cephfs'
```

```
# oc delete pv <pv name>
```

7. 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
storageclusterinitializations.ocs.openshift.io storageclusters.ocs.openshift.io
cephclients.ceph.rook.io --wait=true --timeout=5m
```

8. To ensure that OpenShift Container Storage is uninstalled completely, on the OpenShift Container Platform Web Console,

- a. Click **Home** → **Overview** to access the dashboard.
- b. Verify that the **Persistent Storage** and **Object Service** tabs no longer appear next to the **Cluster** tab.

6.2. REMOVING MONITORING STACK FROM OPENSIFT CONTAINER STORAGE

Use this section to clean up 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                                STATUS  VOLUME
CAPACITY ACCESS MODES STORAGECLASS  AGE
persistentvolumeclaim/my-alertmanager-claim-alertmanager-main-0 Bound  pvc-0d519c4f-
15a5-11ea-baa0-026d231574aa 40Gi  RWO          ocs-external-storagecluster-ceph-rbd
8d
persistentvolumeclaim/my-alertmanager-claim-alertmanager-main-1 Bound  pvc-
0d5a9825-15a5-11ea-baa0-026d231574aa 40Gi  RWO          ocs-external-
storagecluster-ceph-rbd 8d
persistentvolumeclaim/my-alertmanager-claim-alertmanager-main-2 Bound  pvc-
0d6413dc-15a5-11ea-baa0-026d231574aa 40Gi  RWO          ocs-external-
storagecluster-ceph-rbd 8d
persistentvolumeclaim/my-prometheus-claim-prometheus-k8s-0 Bound  pvc-0b7c19b0-
15a5-11ea-baa0-026d231574aa 40Gi  RWO          ocs-external-storagecluster-ceph-rbd
8d
persistentvolumeclaim/my-prometheus-claim-prometheus-k8s-1 Bound  pvc-0b8aed3f-
15a5-11ea-baa0-026d231574aa 40Gi  RWO          ocs-external-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-external-storagecluster-ceph-rbd
  prometheusK8s:
    volumeClaimTemplate:
      metadata:
        name: my-prometheus-claim
      spec:
        resources:
          requests:
            storage: 40Gi
        storageClassName: ocs-external-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. List the pods consuming the PVC.

In this example, the **alertmanagerMain** and **prometheusK8s** pods that were consuming the PVCs are in the **Terminating** state. You can delete the PVCs once these pods are no longer using OpenShift Container Storage PVC.

```

$ oc get pod,pvc -n openshift-monitoring
NAME                                READY STATUS   RESTARTS AGE
pod/alertmanager-main-0             3/3 Terminating 0    10h
pod/alertmanager-main-1             3/3 Terminating 0    10h
pod/alertmanager-main-2             3/3 Terminating 0    10h
pod/cluster-monitoring-operator-84cd9df668-zhjfn 1/1 Running    0    18h
pod/grafana-5db6fd97f8-pmtbf        2/2 Running    0    10h
pod/kube-state-metrics-895899678-z2r9q 3/3 Running    0    10h
pod/node-exporter-4njxv             2/2 Running    0    18h
pod/node-exporter-b8ckz             2/2 Running    0    11h
pod/node-exporter-c2vp5             2/2 Running    0    18h
pod/node-exporter-cq65n             2/2 Running    0    18h
pod/node-exporter-f5sm7             2/2 Running    0    11h
pod/node-exporter-f852c             2/2 Running    0    18h
pod/node-exporter-l9zn7             2/2 Running    0    11h
pod/node-exporter-ngbs8             2/2 Running    0    18h
pod/node-exporter-rv4v9             2/2 Running    0    18h
pod/openshift-state-metrics-77d5f699d8-69q5x 3/3 Running    0    10h
pod/prometheus-adapter-765465b56-4tbxx 1/1 Running    0    10h
pod/prometheus-adapter-765465b56-s2qg2 1/1 Running    0    10h
pod/prometheus-k8s-0                6/6 Terminating 1    9m47s
pod/prometheus-k8s-1                6/6 Terminating 1    9m47s
pod/prometheus-operator-cbfd89f9-ldnwc 1/1 Running    0    43m
pod/telemeter-client-7b5ddb4489-2xfpz 3/3 Running    0    10h

```

```

NAME                                STATUS VOLUME

```

CAPACITY	ACCESS MODES	STORAGECLASS	AGE
persistentvolumeclaim/ocs-alertmanager-claim-alertmanager-main-0	Bound	pvc-2eb79797-1fed-11ea-93e1-0a88476a6a64	40Gi RWO ocs-external-storagecluster-ceph-rbd 19h
persistentvolumeclaim/ocs-alertmanager-claim-alertmanager-main-1	Bound	pvc-2eb79797-1fed-11ea-93e1-0a88476a6a64	40Gi RWO ocs-external-storagecluster-ceph-rbd 19h
persistentvolumeclaim/ocs-alertmanager-claim-alertmanager-main-2	Bound	pvc-2ec6a9cf-1fed-11ea-93e1-0a88476a6a64	40Gi RWO ocs-external-storagecluster-ceph-rbd 19h
persistentvolumeclaim/ocs-prometheus-claim-prometheus-k8s-0	Bound	pvc-3162a80c-1fed-11ea-93e1-0a88476a6a64	40Gi RWO ocs-external-storagecluster-ceph-rbd 19h
persistentvolumeclaim/ocs-prometheus-claim-prometheus-k8s-1	Bound	pvc-316e99e2-1fed-11ea-93e1-0a88476a6a64	40Gi RWO ocs-external-storagecluster-ceph-rbd 19h

5. 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
```

6.3. REMOVING OPENSIFT CONTAINER PLATFORM REGISTRY FROM OPENSIFT CONTAINER STORAGE

Use this section to clean up OpenShift Container Platform registry from OpenShift Container Storage. If you want to configure an alternative storage, see [image registry](#)

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

Prerequisites

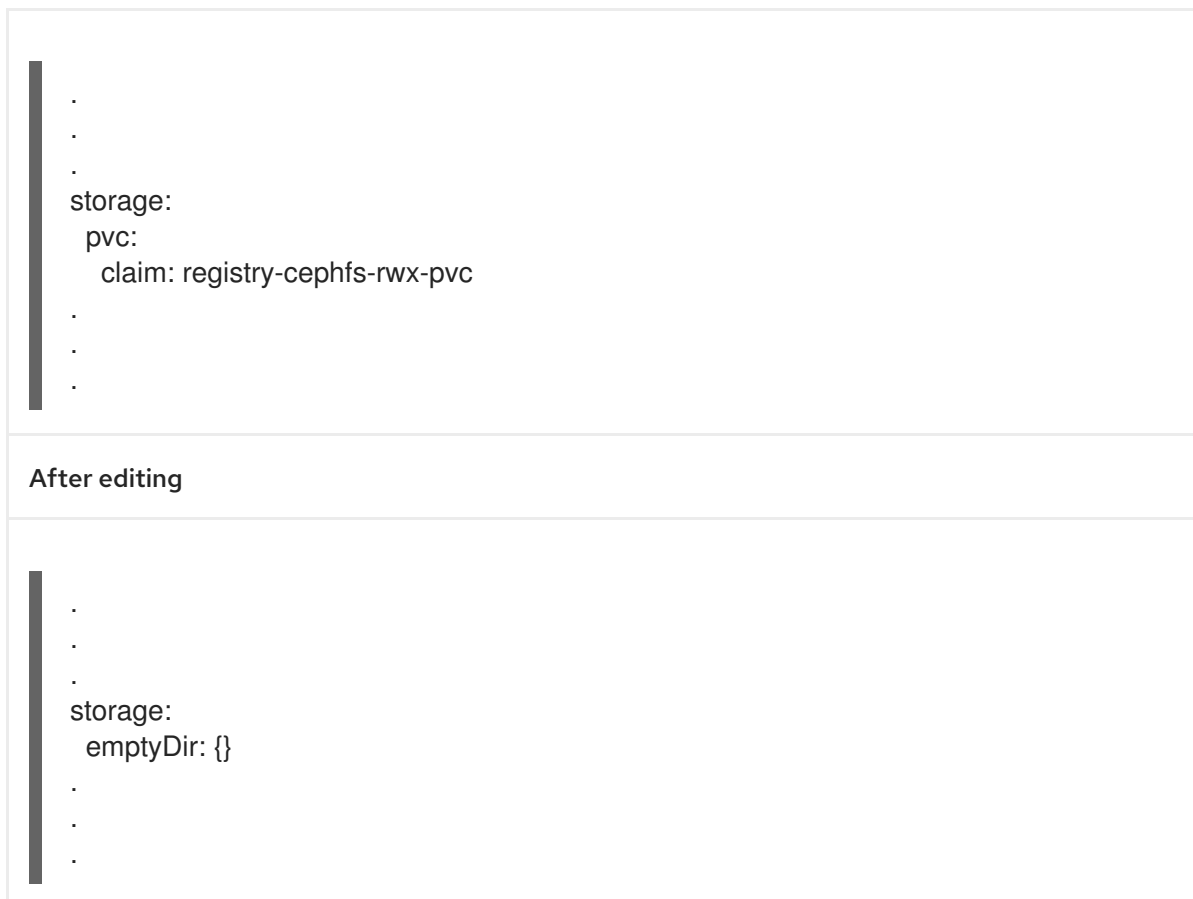
- The image registry should have been configured to use an OpenShift Container Storage PVC.

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
```

6.4. REMOVING THE CLUSTER LOGGING OPERATOR FROM OPENSIFT CONTAINER STORAGE

Use this section to clean up the cluster logging operator from OpenShift Container Storage.

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

Prerequisites

- The cluster logging instance should have been 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
```