Red Hat OpenShift Container Storage 4.6

Deploying OpenShift Container Storage in external mode

How to install and set up your environment

Red Hat OpenShift Container Storage 4.6 Deploying OpenShift Container Storage in external mode

How to install and set up your environment

Legal Notice

Abstract

Read this document for instructions on installing Red Hat OpenShift Container Storage 4.6 to use an external Red Hat Ceph Storage cluster.

Table of Contents

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 for worker nodes, 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 Storage on an OpenShift Container Platform with worker nodes 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 OPENSHIFT 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 (RHOCP) 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

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

2. Use **Filter by keyword** text box or the filter list to search for OpenShift Container Storage from the list of operators.

3. Click **OpenShift Container Storage**.

4. On the **OpenShift Container Storage operator** page, click Install.

5. On the **Install Operator** page, ensure the following options are selected by default::

   a. Update Channel as **stable-4.6**

   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. Select **Enable operator recommended cluster monitoring on this namespace** checkbox as this is required for cluster monitoring.

   e. 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 **Manual approval required** page, you can either click **Approve** or **View Installed Operators in namespace openshift-storage** to install the operator.

     > **IMPORTANT**
     >
     > Before you click either of the options, wait for a few minutes on the **Manual approval required** page until the install plan gets loaded in the window.

     > **IMPORTANT**
     >
     > If you choose to click **Approve**, you must review the install plan before you proceed.

     - If you click **Approve**.

       - Wait for a few minutes while the OpenShift Container Storage Operator is getting installed.

       - On the **Installed operator - ready for use** page, click **View Operator**.

       - Ensure the **Project** is **openshift-storage**. By default, the **Project** is **openshift-storage**.

       - Click **Operators → Installed Operators**

       - Wait for the **Status** of **OpenShift Container Storage** to change to **Succeeded**.

     - If you click **View Installed Operators in namespace openshift-storage**.

       - On the **Installed Operators** page, click **ocs-operator**.

       - On the **Subscription Details** page, click the **Install Plan** link.

- On the **InstallPlan Details** page, click **Preview Install Plan**.

- Review the install plan and click **Approve**.

- Wait for the **Status** of the **Components** to change from **Unknown** to either **Created** or **Present**.

- Click **Operators → Installed Operators**

- Ensure the **Project** is **openshift-storage**. By default, the **Project** is **openshift-storage**.

- Wait for the **Status** of **OpenShift Container Storage** to change to **Succeeded**.

Verification steps

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

- Click **View Installed Operators in namespace openshift-storage** link to verify that OpenShift Container Storage Operator shows the **Status** as **Succeeded** on the Installed Operators dashboard.

CHAPTER 4. CREATING AN OPENSHIFT 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 VMware vSphere or Bare metal infrastructures.

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 4.2z1 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 from a version lower than 4.1.1 to the latest 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 .

- Red Hat Ceph Storage must have Ceph Dashboard installed and configured, and must use port **9283** for Ceph Manager Prometheus exporter. For more information, see Ceph Dashboard installation and access.

- It is recommended that the PG Autoscaler option must be enabled for the external Red Hat Ceph Storage cluster. 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. It is recommended to use a separate pool for each OpenShift Container Storage cluster.

Procedure

1. Click **Operators → Installed Operators** to view all the installed operators.
   Ensure that the **Project** selected is **openshift-storage**.

   Figure 4.1. OpenShift Container Storage Operator page
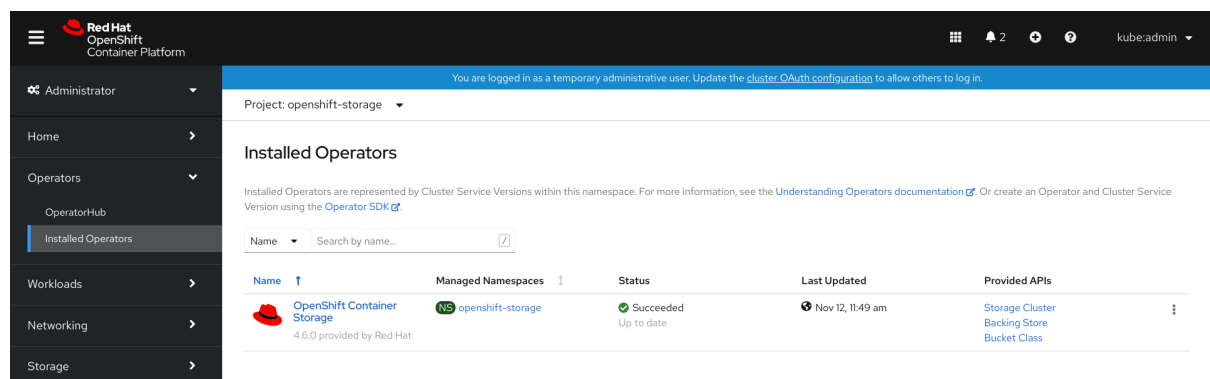
2. Click **OpenShift Container Storage**.

   Figure 4.2. Details tab of OpenShift Container Storage

3. Click **Create Instance** link of Storage Cluster.

4. Select Mode as **External**. By default, Internal is selected as deployment mode.

   Figure 4.3. Connect to external cluster section on Create Storage Cluster form

5. In the Connect to external cluster section, click on the **Download Script** link to download the python script for extracting Ceph cluster details.

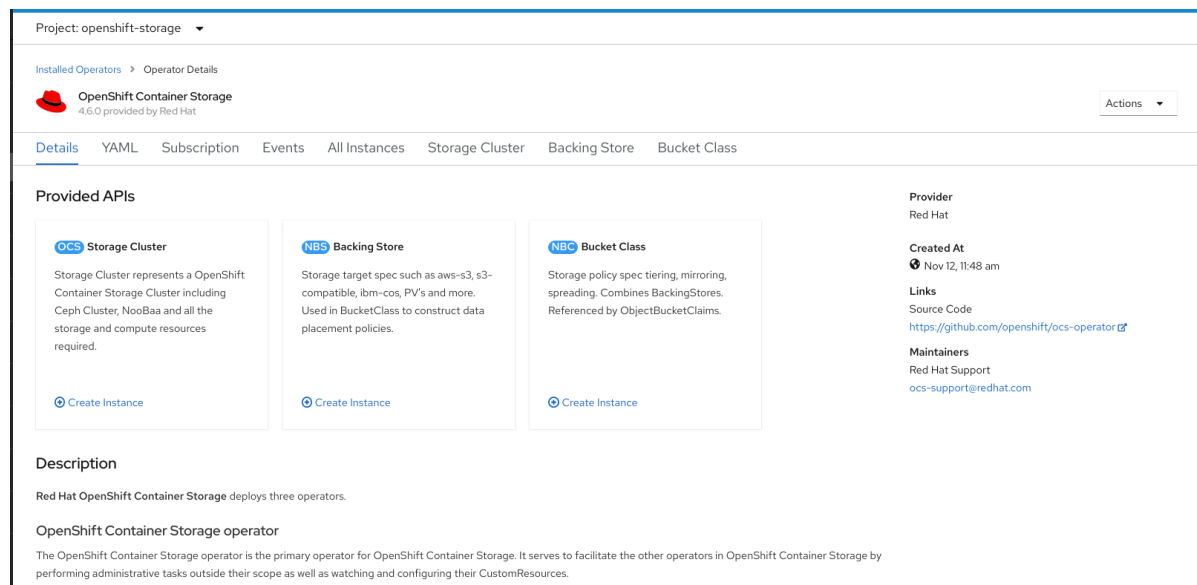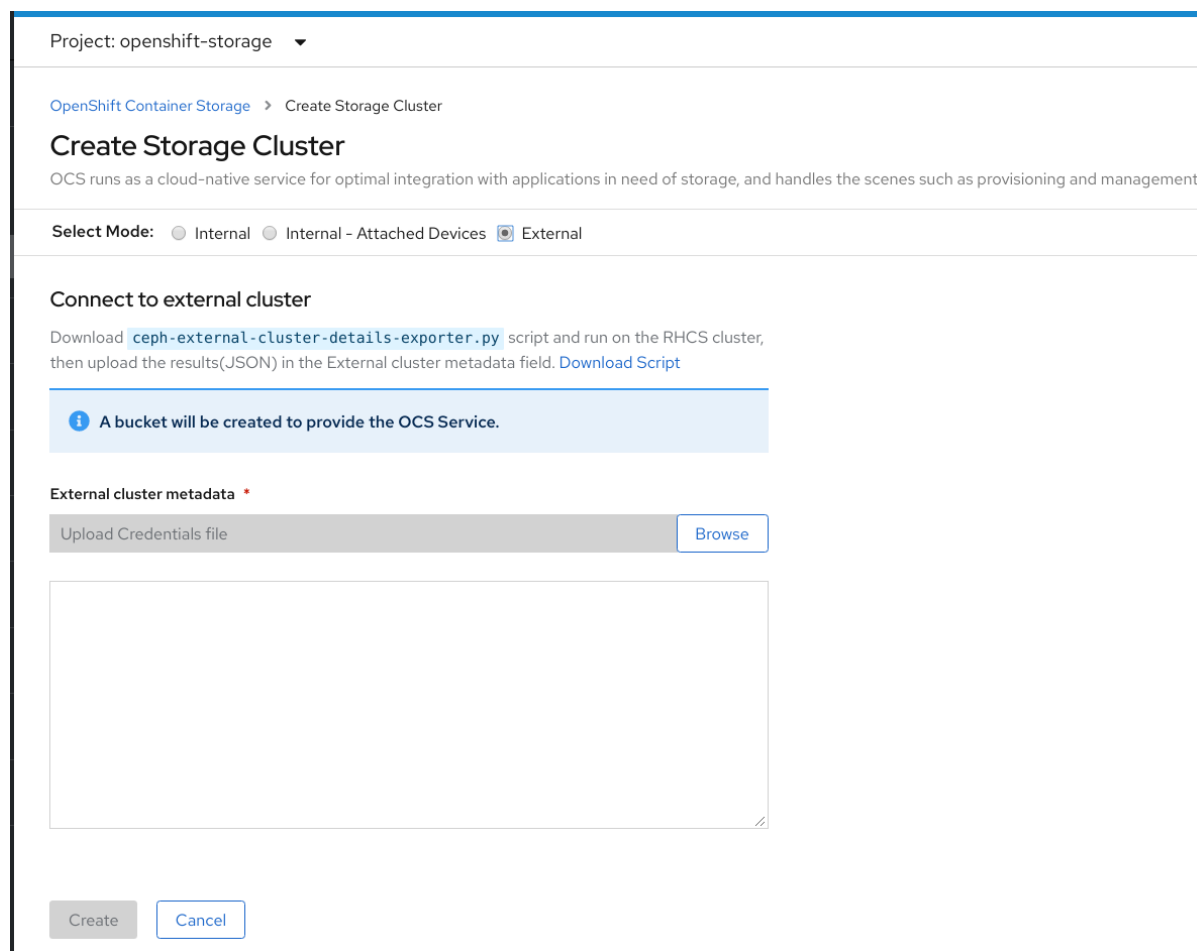6. For extracting the Red Hat Ceph Storage (RHCS) cluster details, contact the RHCS administrator to run the downloaded python script on a Red Hat Ceph Storage node with **admin key**.

   a. Run the following command on the RHCS 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

   You can also run the script from inside a MON container (containerized deployment) or from a MON node (rpm deployment).

   b. 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 --
   monitoring-endpoint xxx.xxx.xxx.xxx --monitoring-endpoint-port 9283 --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. Provide the endpoint in the following format: **<ip_address>:<port>**

   - **--monitoring-endpoint** is optional. It is the IP address of the active **ceph-mgr** reachable from the OpenShift Container Platform cluster. If not provided, the value is automatically populated.

   - **--monitoring-endpoint-port** is optional. It is the port associated with the **ceph-mgr** Prometheus exporter specified by **--monitoring-endpoint**. If not provided, the value is automatically populated. Only port **9283** is supported in OpenShift Container Storage 4.6.

   - **-- 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: [mgr] allow command config

     - caps: [mon] allow r, allow command quorum_status, allow command version

- caps: [osd] allow rwx pool=**RGW_POOL_PREFIX.rgw.meta**, allow r pool=**.rgw.root**, allow rw pool=**RGW_POOL_PREFIX.rgw.control**, allow rx pool=**RGW_POOL_PREFIX.rgw.log**, 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": "xxx.xxx.xxx.xxx:xxxx", "maxMonId": "0", "mapping": "{}"}}, {"name": "rook-ceph-mon", "kind": "Secret", "data": {"admin-secret": "admin-secret", "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": "ceph-rbd", "kind": "StorageClass", "data": {"pool": "ceph-rbd"}}, {"name": "monitoring-endpoint", "kind": "CephCluster", "data": {"MonitoringEndpoint": "xxx.xxx.xxx.xxx", "MonitoringPort": "xxxx"}}, {"name": "rook-csi-rbd-provisioner", "kind": "Secret", "data": {"userID": "csi-rbd-provisioner", "userKey": "<user-key>"}}, {"name": "rook-csi-cephfs-provisioner", "kind": "Secret", "data": {"adminID": "csi-cephfs-provisioner", "adminKey": "<admin-key>"}}, {"name": "rook-csi-cephfs-node", "kind": "Secret", "data": {"adminID": "csi-cephfs-node", "adminKey": "<admin-key>"}}, {"name": "cephfs", "kind": "StorageClass", "data": {"fsName": "cephfs", "pool": "cephfs_data"}}, {"name": "ceph-rgw", "kind": "StorageClass", "data": {"endpoint": "xxx.xxx.xxx.xxx:xxxx", "poolPrefix": "default"}}]
  ```

c. 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, and so on) to be uploaded using the JSON file remains unchanged on the RHCS external cluster after the storage cluster creation.

7. Click **External cluster metadata → Browse** to select and upload the JSON file. The content of the JSON file is populated and displayed in the text box.

   Figure 4.4. Json file content
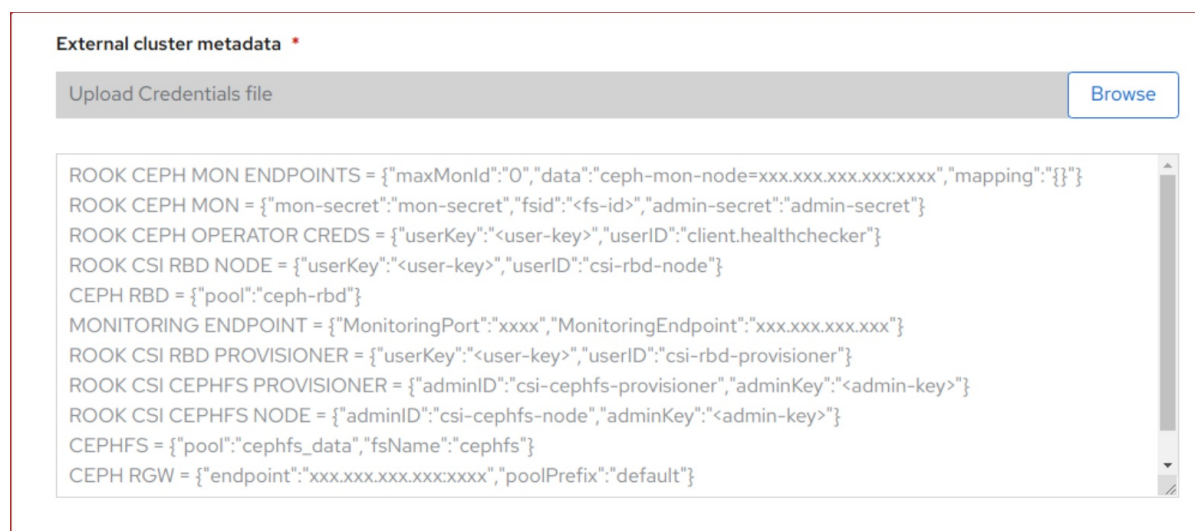
8. Click **Create**.
   The Create button is enabled only after you upload the **.json** file.

Verification steps

1. 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.

2. 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 OPENSHIFT CONTAINER STORAGE INSTALLATION FOR EXTERNAL MODE

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

5.1. VERIFYING THE STATE OF THE PODS

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	**ocs-operator-*** (1 pod on any worker node) **ocs-metrics-exporter-***
Rook-ceph Operator	**rook-ceph-operator-*** (1 pod on any worker node)
Multicloud Object Gateway	**noobaa-operator-*** (1 pod on any worker node) **noobaa-core-*** (1 pod on any worker node) **noobaa-db-*** (1 pod on any worker node) **noobaa-endpoint-*** (1 pod on any worker node)

Component	Corresponding pods
CSI	**cephfs** **csi-cephfsplugin-*** (1 pod on each worker node) **csi-cephfsplugin-provisioner-*** (2 pods distributed across worker nodes)

> **NOTE**
>
> If an MDS is not deployed in the external cluster, the csi-cephfsplugin pods will not be created.

- **rbd**
  - **csi-rbdplugin-*** (1 pod on each worker node)
  - **csi-rbdplugin-provisioner-*** (2 pods distributed across worker nodes)

5.2. VERIFYING THAT THE OPENSHIFT 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
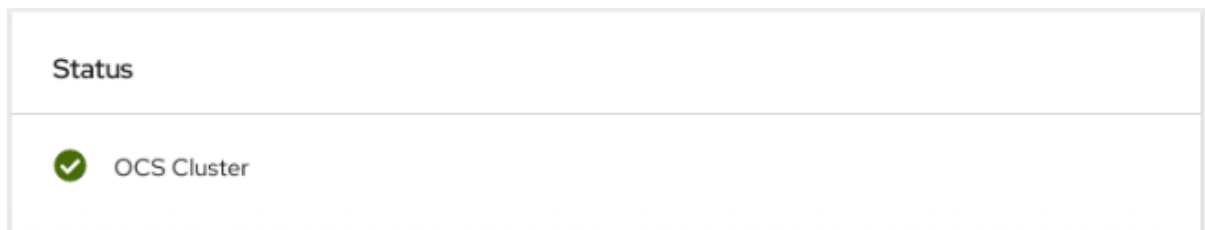
- In the **Details card**, verify that the cluster information is displayed as follows:

Service Name

OpenShift Container Storage

Cluster Name

ocs-external-storagecluster

Provider

VSphere

Mode

External

Version

ocs-operator-4.6.0

For more information on the health of OpenShift Container Storage cluster using the persistent storage dashboard, see Monitoring OpenShift Container Storage .

5.3. VERIFYING 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 both *Object Service* and *Data Resiliency* are in **Ready** state (green tick).

  Figure 5.2. Health status card in Object Service Overview Dashboard
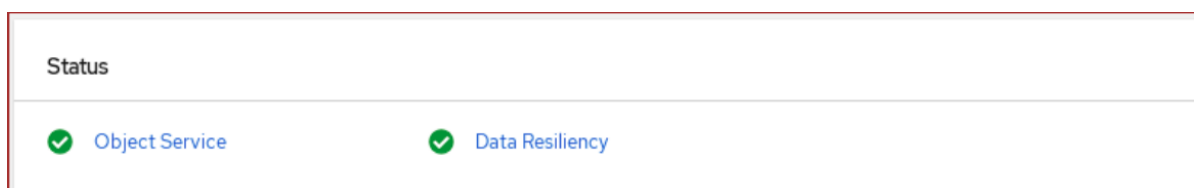
- In the **Details card**, verify that the MCG information is displayed appropriately as follows:

  Service Name

  OpenShift Container Storage

  System Name

  Multicloud Object Gateway
  RADOS Object Gateway

  Provider

  VSphere

  Version

  ocs-operator-4.6.0

> **NOTE**
>
> The RADOS Object Gateway is only listed in case RADOS Object Gateway endpoint details were included while deploying OpenShift Container Storage in external mode.

For more information on the health of OpenShift Container Storage cluster using the object service dashboard, see Monitoring OpenShift Container Storage .

5.4. VERIFYING 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. VERIFYING THAT CEPH CLUSTER IS CONNECTED

Run the following command to verify if the OpenShift Container Storage cluster is connected to the external Red Hat Ceph Storage 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. VERIFYING 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   4.6.0
```

CHAPTER 6. UNINSTALLING OPENSHIFT CONTAINER STORAGE

6.1. UNINSTALLING OPENSHIFT CONTAINER STORAGE IN EXTERNAL MODE

Use the steps in this section to uninstall OpenShift Container Storage. Uninstalling OpenShift Container Storage does not remove the RBD pool from the external cluster, or uninstall the external Red Hat Ceph Storage cluster.

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**

> **NOTE**
>
> The **uninstall.ocs.openshift.io/cleanup-policy** is not applicable for external mode.

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

Table 6.1. uninstall.ocs.openshift.io uninstall annotations descriptions

Annotation	Value	Default	Behavior
cleanup-policy	delete	Yes	Rook cleans up the physical drives and the **DataDirHostPath**
cleanup-policy	retain	No	Rook does **not** clean up the physical drives and the **DataDirHostPath**
mode	graceful	Yes	Rook and NooBaa **pauses** 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 uninstall mode by editing the value of the annotation by using the following commands:

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

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 so will result in orphan PVCs and OBCs in the system.

   a. Delete OpenShift Container Platform monitoring stack PVCs using OpenShift Container Storage.
      See Section 6.2, "Removing monitoring stack from OpenShift Container Storage"

   b. Delete OpenShift Container Platform Registry PVCs using OpenShift Container Storage.
      See Section 6.3, "Removing OpenShift Container Platform registry from OpenShift Container Storage"

   c. Delete OpenShift Container Platform logging PVCs using OpenShift Container Storage.
      See Section 6.4, "Removing the cluster logging operator from OpenShift Container Storage"

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

      - Given below is a sample script to identify the PVCs and OBCs provisioned using OpenShift Container Storage. The script ignores the PVCs and OBCs 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
```

- Delete the OBCs.

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

- Delete the PVCs.

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

  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
```
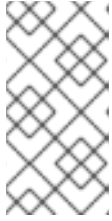
4. Delete the namespace and wait until the deletion is complete. You will need to switch to another project if **openshift-storage** is the active project.
   For example:

```
$ 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 the 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.

5. 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>
```

6. Delete the Multicloud Object Gateway storageclass.

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

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 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
```

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 OPENSHIFT 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                                      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-
2ebeee54-1fed-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 OPENSHIFT CONTAINER PLATFORM REGISTRY FROM OPENSHIFT 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**

```
.
.
.
storage:
  pvc:
    claim: registry-cephfs-rwx-pvc
.
.
.
```

After editing

```
.
.
.
storage:
  emptyDir: {}
.
.
.
```

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 OPENSHIFT 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 the **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
```