Red Hat OpenShift Container Storage

4.7

Deploying OpenShift Container Storage using Microsoft Azure

How to install and manage

Last Updated: 2021-06-30
Red Hat OpenShift Container Storage 4.7 Deploying OpenShift Container Storage using Microsoft Azure

How to install and manage
Abstract

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

MAKING OPEN SOURCE MORE INCLUSIVE ......................................................... 3

PROVIDING FEEDBACK ON RED HAT DOCUMENTATION ................................. 4

PREFACE ........................................................................................................ 5

CHAPTER 1. PREPARING TO DEPLOY OPENSHIFT CONTAINER STORAGE .......... 6
1.1. ENABLING KEY VALUE BACKEND PATH AND POLICY IN VAULT ............. 6

CHAPTER 2. DEPLOYING OPENSHIFT CONTAINER STORAGE ON MICROSOFT AZURE ........................... 8
2.1. INSTALLING RED HAT OPENSHIFT CONTAINER STORAGE OPERATOR ........ 8
2.2. CREATING AN OPENSHIFT CONTAINER STORAGE CLUSTER SERVICE IN INTERNAL MODE ........ 9

CHAPTER 3. VERIFYING OPENSHIFT CONTAINER STORAGE DEPLOYMENT ........ 13
3.1. VERIFYING THE STATE OF THE PODS ..................................................... 13
3.2. VERIFYING THE OPENSHIFT CONTAINER STORAGE CLUSTER IS HEALTHY 14
3.3. VERIFYING THE MULTICLOUD OBJECT GATEWAY IS HEALTHY .............. 15
3.4. VERIFYING THAT THE OPENSHIFT CONTAINER STORAGE SPECIFIC STORAGE CLASSES EXIST 16

CHAPTER 4. UNINSTALLING OPENSHIFT CONTAINER STORAGE ..................... 17
4.1. UNINSTALLING OPENSHIFT CONTAINER STORAGE IN INTERNAL MODE ...... 17
4.2. REMOVING MONITORING STACK FROM OPENSHIFT CONTAINER STORAGE 22
4.3. REMOVING OPENSHIFT CONTAINER PLATFORM REGISTRY FROM OPENSHIFT CONTAINER STORAGE 24
4.4. REMOVING THE CLUSTER LOGGING OPERATOR FROM OPENSHIFT CONTAINER STORAGE 25
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. As the Component, use 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.
Red Hat OpenShift Container Storage 4.7 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 OPENSOURCE
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. On the external key management system (KMS),
   - Ensure that a policy with a token exists and the key value backend path in Vault is enabled. See enabled the key value backend path and policy in Vault.
   - Ensure that you are using signed certificates on your Vault servers.

2. 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.
- Carefully, 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:

```bash
$ vault token create -policy=ocs -format=json
```
CHAPTER 2. DEPLOYING OPENSOURCES 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 OPENSOURCES CONTAINER STORAGE OPERATOR


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 RHOCP cluster.
- For additional resource requirements, 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 in command line interface 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. Navigate in the web console to the click Operators → OperatorHub.
2. Scroll or type a keyword into the Filter by keyword box to search for OpenShift Container Storage Operator.

3. Click Install on the OpenShift Container Storage operator page.

4. On the Install Operator page, the following required options are selected by default:
   a. Update Channel as stable-4.7.
   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 Approval Strategy as Automatic or Manual.
   e. Click Install.

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

   If you selected Manual updates, then 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 steps

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

Next steps

- Create OpenShift Container Storage cluster.
  For information, see Creating an OpenShift Container Storage Cluster Service in internal mode.

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. In **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) Security 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.

   **IMPORTANT**
   Storage class encryption is a Technology Preview feature available only for RBD PVs. Technology Preview features are not supported with Red Hat production service level agreements (SLAs) and might not be functionally complete. Red Hat does not recommend using them in production. These features provide early access to upcoming product features, enabling customers to test functionality and provide feedback during the development process.

   For more information, see **Technology Preview Features Support Scope**.

   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. 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**.

d. Click **Next**.

7. Review the configuration details. To modify any configuration settings, click **Back** to go back to the previous configuration page.

8. Click **Create**.

9. 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**.

         ```yaml
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. VERIFYING OPENSHIFT CONTAINER STORAGE DEPLOYMENT

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

3.1. VERIFYING THE STATE OF THE PODS

To determine if OpenShift Container storage is deployed successfully, you can verify that the pods are in Running state.

Procedure

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 3.1, “Pods corresponding to OpenShift Container storage cluster”.
3. Verify that the following pods are in running and completed state by clicking on the Running and the Completed tabs:

   Table 3.1. Pods corresponding to OpenShift Container storage cluster

<table>
<thead>
<tr>
<th>Component</th>
<th>Corresponding pods</th>
</tr>
</thead>
<tbody>
<tr>
<td>OpenShift Container Storage Operator</td>
<td>• ocs-operator-* (1 pod on any worker node)</td>
</tr>
<tr>
<td></td>
<td>• ocs-metrics-exporter-*</td>
</tr>
<tr>
<td>Rook-ceph Operator</td>
<td>rook-ceph-operator-*</td>
</tr>
<tr>
<td></td>
<td>(1 pod on any worker node)</td>
</tr>
<tr>
<td>Multicloud Object Gateway</td>
<td>• noobaa-operator-* (1 pod on any worker node)</td>
</tr>
<tr>
<td></td>
<td>• noobaa-core-* (1 pod on any storage node)</td>
</tr>
<tr>
<td></td>
<td>• nooba-db-* (1 pod on any storage node)</td>
</tr>
<tr>
<td></td>
<td>• noobaa-endpoint-* (1 pod on any storage node)</td>
</tr>
<tr>
<td>MON</td>
<td>rook-ceph-mon-*</td>
</tr>
<tr>
<td></td>
<td>(3 pods distributed across storage nodes)</td>
</tr>
<tr>
<td>Component</td>
<td>Corresponding pods</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>-------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>MGR</td>
<td>rook-ceph-mgr-*</td>
</tr>
<tr>
<td></td>
<td>(1 pod on any storage node)</td>
</tr>
<tr>
<td>MDS</td>
<td>rook-ceph-mds-ocs-storagecluster-cephfilesystem-*</td>
</tr>
<tr>
<td></td>
<td>(2 pods distributed across storage nodes)</td>
</tr>
<tr>
<td>CSI</td>
<td></td>
</tr>
<tr>
<td></td>
<td>● cephfs</td>
</tr>
<tr>
<td></td>
<td>○ csi-cephfsplugin-*  (1 pod on each worker node)</td>
</tr>
<tr>
<td></td>
<td>○ csi-cephfsplugin-provisioner-*  (2 pods distributed across storage nodes)</td>
</tr>
<tr>
<td></td>
<td>● rbd</td>
</tr>
<tr>
<td></td>
<td>○ csi-rbdplugin-*  (1 pod on each worker node)</td>
</tr>
<tr>
<td></td>
<td>○ csi-rbdplugin-provisioner-*  (2 pods distributed across storage nodes)</td>
</tr>
<tr>
<td>rook-ceph-crashcollector</td>
<td>(rook-ceph-crashcollector-*</td>
</tr>
<tr>
<td></td>
<td>(1 pod on each storage node)</td>
</tr>
<tr>
<td>OSD</td>
<td></td>
</tr>
<tr>
<td></td>
<td>● rook-ceph-osd-*  (1 pod for each device)</td>
</tr>
<tr>
<td></td>
<td>● rook-ceph-osd-prepare-ocs-deviceset-*  (1 pod for each device)</td>
</tr>
</tbody>
</table>

### 3.2. VERIFYING 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 and Data Resiliency has a green tick mark as shown in the following image:
Figure 3.1. Health status card in Persistent Storage Overview Dashboard

<table>
<thead>
<tr>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>✅ OCS Cluster</td>
</tr>
<tr>
<td>✅ Data Resiliency</td>
</tr>
</tbody>
</table>

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

  **Service Name**
  OpenShift Container Storage

  **Cluster Name**
  ocs-storagecluster

  **Provider**
  Azure

  **Mode**
  Internal

  **Version**
  ocs-operator-4.7.0

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

### 3.3. VERIFYING 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 3.2. Health status card in Object Service Overview Dashboard

<table>
<thead>
<tr>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>✅ Object Service</td>
</tr>
<tr>
<td>✅ Data Resiliency</td>
</tr>
</tbody>
</table>

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

  **Service Name**
  OpenShift Container Storage

  **System Name**
  Multicloud Object Gateway

  **Provider**
  Azure

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

3.4. VERIFYING THAT THE OPENSHIFT CONTAINER STORAGE SPECIFIC STORAGE CLASSES EXIST

To verify the storage classes exist in the cluster:

- 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-storagecluster-ceph-rbd
  - ocs-storagecluster-cephfs
  - openshift-storage.noobaa.io
CHAPTER 4. UNINSTALLING OPENSHIFT CONTAINER STORAGE

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

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 ocs-storagecluster uninstall.ocs.openshift.io/cleanup-policy="retain" --overwrite
storagecluster.ocs.openshift.io/ocs-storagecluster annotated
```
$ oc annotate storagecluster 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 so will result in orphan PVCs and OBCs in the system.
      See Section 4.2, “Removing monitoring stack from OpenShift Container Storage”
   
   b. Delete OpenShift Container Platform Registry PVCs using OpenShift Container Storage.
      See Section 4.3, “Removing OpenShift Container Platform registry from OpenShift Container Storage”
   
   c. Delete OpenShift Container Platform logging PVCs using OpenShift Container Storage.
      See Section 4.4, “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 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"
NOOBAAPROVISIONER="openshift-storage.noobaa.io/obc"
RGW_PROVISIONER="openshift-storage.ceph.rook.io/bucket"

NOOBAADBPVC="noobaa-db"
NOOBAABACKINGSTOREPVC="noobaa-default-backing-store-noobaa-pvc"

# Find all the OCS StorageClasses
OCS_STORAGECLASSES=$(oc get storageclasses | grep -e $RBD_PROVISIONER -e $CEPHFS_PROVISIONER -e $NOOBAAPROVISIONER -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 $NOOBAADBPVC -e $NOOBAABACKINGSTOREPVC
  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.
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).

   ```bash
   $ 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.

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

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

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

   c. Remove the mapped device.

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

   **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.
     ```bash
     $ ps -ef | grep crypt
     ```
   - Terminate the process using **kill** command.
     ```bash
     $ kill -9 <PID>
     ```
   - Verify that the device name is removed.
     ```bash
     $ dmsetup ls
     ```

7. 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.

   For example:

   ```bash
   $ oc get pods -n openshift-storage | grep -i cleanup
   NAME                          READY STATUS    RESTARTS AGE
   cluster-cleanup-job-<xx>     0/1  Completed 0  8m35s
   cluster-cleanup-job-<yy>     0/1  Completed 0  8m35s
   cluster-cleanup-job-<zz>     0/1  Completed 0  8m35s
   ```
$ 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 cephobjectrealmss.ceph.rook.io cephobjectzonegroups.ceph.rook.io
    cephobjectzones.ceph.rook.io cephobjectmirrors.ceph.rook.io --wait=true --timeout=5m

13. 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.
4.2. REMOVING MONITORING STACK FROM OPENSHEET 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.

   ```bash
   $ oc get pod,pvc -n openshift-monitoring
   ```

<table>
<thead>
<tr>
<th>NAME</th>
<th>READY</th>
<th>STATUS</th>
<th>RESTARTS</th>
<th>AGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>pod/alertmanager-main-0</td>
<td>3/3</td>
<td>Running</td>
<td>0</td>
<td>8d</td>
</tr>
<tr>
<td>pod/alertmanager-main-1</td>
<td>3/3</td>
<td>Running</td>
<td>0</td>
<td>8d</td>
</tr>
<tr>
<td>pod/alertmanager-main-2</td>
<td>3/3</td>
<td>Running</td>
<td>0</td>
<td>8d</td>
</tr>
<tr>
<td>pod/cluster-monitoring-operator-84457656d-pkrxm</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>8d</td>
</tr>
<tr>
<td>pod/grafana-79ccf6689f-2li28</td>
<td>2/2</td>
<td>Running</td>
<td>0</td>
<td>8d</td>
</tr>
<tr>
<td>pod/kube-state-metrics-7d86fb966-rvd9w</td>
<td>3/3</td>
<td>Running</td>
<td>0</td>
<td>8d</td>
</tr>
<tr>
<td>pod/node-exporter-25894</td>
<td>2/2</td>
<td>Running</td>
<td>0</td>
<td>8d</td>
</tr>
<tr>
<td>pod/node-exporter-4dsd7</td>
<td>2/2</td>
<td>Running</td>
<td>0</td>
<td>8d</td>
</tr>
<tr>
<td>pod/node-exporter-6p4zc</td>
<td>2/2</td>
<td>Running</td>
<td>0</td>
<td>8d</td>
</tr>
<tr>
<td>pod/node-exporter-jbjvg</td>
<td>2/2</td>
<td>Running</td>
<td>0</td>
<td>8d</td>
</tr>
<tr>
<td>pod/node-exporter-jj4t5</td>
<td>2/2</td>
<td>Running</td>
<td>0</td>
<td>6d18h</td>
</tr>
<tr>
<td>pod/node-exporter-k856s</td>
<td>2/2</td>
<td>Running</td>
<td>0</td>
<td>6d18h</td>
</tr>
<tr>
<td>pod/node-exporter-rf8gn</td>
<td>2/2</td>
<td>Running</td>
<td>0</td>
<td>8d</td>
</tr>
<tr>
<td>pod/node-exporter-rmb5m</td>
<td>2/2</td>
<td>Running</td>
<td>0</td>
<td>6d18h</td>
</tr>
<tr>
<td>pod/node-exporter-zj7kx</td>
<td>2/2</td>
<td>Running</td>
<td>0</td>
<td>8d</td>
</tr>
<tr>
<td>pod/openshift-state-metrics-59dbd4f654-4clng</td>
<td>3/3</td>
<td>Running</td>
<td>0</td>
<td>8d</td>
</tr>
<tr>
<td>pod/prometheus-adapter-5df5865596-k8dzn</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>7d23h</td>
</tr>
<tr>
<td>pod/prometheus-adapter-5df5865596-n2gj9</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>7d23h</td>
</tr>
<tr>
<td>pod/prometheus-k8s-0</td>
<td>6/6</td>
<td>Running</td>
<td>1</td>
<td>8d</td>
</tr>
<tr>
<td>pod/prometheus-k8s-1</td>
<td>6/6</td>
<td>Running</td>
<td>1</td>
<td>8d</td>
</tr>
<tr>
<td>pod/prometheus-operator-55c8b858c9-c4zd9</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>6d21h</td>
</tr>
<tr>
<td>pod/telemeter-client-78fc8fc97d-2rgfp</td>
<td>3/3</td>
<td>Running</td>
<td>0</td>
<td>8d</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>NAME</th>
<th>CAPACITY</th>
<th>ACCESS MODES</th>
<th>STATUSCLASS</th>
<th>VOLUME</th>
<th>AGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>persistentvolumeclaim/my-alertmanager-claim-alertmanager-main-0</td>
<td>Bound</td>
<td>pvc-0d519c4f-15a5-11ea-baa0-026d231574aa 40Gi RWO ocs-storagecluster-ceph-rbd 8d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>persistentvolumeclaim/my-alertmanager-claim-alertmanager-main-1</td>
<td>Bound</td>
<td>pvc-0d5a9825-15a5-11ea-baa0-026d231574aa 40Gi RWO ocs-storagecluster-ceph-</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
2. Edit the monitoring **configmap**.

   ```bash
   $ 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**

   ```yaml
   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

```yaml
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.

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

4.3. 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.

   ```bash
   $ 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.

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

### 4.4. 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.

```bash
$ 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