Deploying OpenShift Data Foundation in external mode

Instructions for deploying OpenShift Data Foundation to use an external Red Hat Ceph Storage cluster and IBM FlashSystem.
Instructions for deploying OpenShift Data Foundation to use an external Red Hat Ceph Storage cluster and IBM FlashSystem.
Abstract

Read this document for instructions on installing Red Hat OpenShift Data Foundation 4.14 to use an external Red Hat Ceph Storage cluster or IBM FlashSystem.
## Table of Contents

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

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

**CHAPTER 1. OVERVIEW OF DEPLOYING IN EXTERNAL MODE** .......................... 5
  1.1. DISASTER RECOVERY REQUIREMENTS .............................................. 5
  1.2. NETWORK PORTS REQUIRED BETWEEN OPENSHIFT CONTAINER PLATFORM AND CEPH WHEN USING EXTERNAL MODE DEPLOYMENT ........................................ 5

**CHAPTER 2. DEPLOY OPENShift DATA FOUNDATION USING RED HAT CEPH STORAGE** .......... 6
  2.1. INSTALLING RED HAT OPENShift DATA FOUNDATION OPERATOR .............. 6
  2.2. CREATING AN OPENSShift DATA FOUNDATION CLUSTER FOR EXTERNAL CEPH STORAGE SYSTEM ................................................................. 7
  2.3. VERIFYING YOUR OPENShift DATA FOUNDATION INSTALLATION FOR EXTERNAL CEPH STORAGE SYSTEM ......................................................... 12
    2.3.1. Verifying the state of the pods .................................................... 12
    2.3.2. Verifying that the OpenShift Data Foundation cluster is healthy .......... 13
    2.3.3. Verifying that the Multicloud Object Gateway is healthy .................. 14
    2.3.4. Verifying that the storage classes are created and listed ............... 14
    2.3.5. Verifying that Ceph cluster is connected ...................................... 14
    2.3.6. Verifying that storage cluster is ready ........................................ 15

**CHAPTER 3. DEPLOY OPENSShift DATA FOUNDATION USING IBM FLASHSYSTEM** ............. 16
  3.1. INSTALLING RED HAT OPENSShift DATA FOUNDATION OPERATOR .............. 16
  3.2. CREATING AN OPENSShift DATA FOUNDATION CLUSTER FOR EXTERNAL IBM FLASHSYSTEM STORAGE ................................................................. 17

**CHAPTER 4. UNINSTALLING OPENSShift DATA FOUNDATION FROM EXTERNAL STORAGE SYSTEM** . 23
  4.1. REMOVING MONITORING STACK FROM OPENSShift DATA FOUNDATION ......... 26
  4.2. REMOVING OPENSShift CONTAINER PLATFORM REGISTRY FROM OPENSShift DATA FOUNDATION ................................................................. 30
  4.3. REMOVING THE CLUSTER LOGGING OPERATOR FROM OPENSShift DATA FOUNDATION ................................................................. 31
  4.4. REMOVING EXTERNAL IBM FLASHSYSTEM SECRET .................................. 32
MAKING OPEN SOURCE MORE INCLUSIVE

Red Hat is committed to replacing problematic language in our code, documentation, and web properties. We are beginning with these four terms: master, slave, blacklist, and whitelist. Because of the enormity of this endeavor, these changes will be implemented gradually over several upcoming releases. For more details, see our CTO Chris Wright’s message.
PROVIDING FEEDBACK ON RED HAT DOCUMENTATION

We appreciate your input on our documentation. Do let us know how we can make it better.

To give feedback, create a Bugzilla ticket:

1. Go to the Bugzilla website.
2. In the Component section, choose documentation.
3. Fill in the Description field with your suggestion for improvement. Include a link to the relevant part(s) of documentation.
4. Click Submit Bug.
CHAPTER 1. OVERVIEW OF DEPLOYING IN EXTERNAL MODE

Red Hat OpenShift Data Foundation can make services from an external Red Hat Ceph Storage cluster available for consumption through OpenShift Container Platform clusters running on any platform. See Planning your deployment for more information.

For instructions regarding how to install a RHCS cluster, see the installation guide.

Follow these steps to deploy OpenShift Data Foundation in external mode:

- Deploy OpenShift Data Foundation using Red Hat Ceph Storage.
- Deploy OpenShift Data Foundation using IBM FlashSystem.

1.1. DISASTER RECOVERY REQUIREMENTS

Disaster Recovery features supported by Red Hat OpenShift Data Foundation require all of the following prerequisites to successfully implement a disaster recovery solution:

- A valid Red Hat OpenShift Data Foundation Advanced subscription
- A valid Red Hat Advanced Cluster Management for Kubernetes subscription

For more information, see the knowledgebase article on OpenShift Data Foundation subscriptions.

For detailed disaster recovery solution requirements, see Configuring OpenShift Data Foundation Disaster Recovery for OpenShift Workloads guide, and Requirements and recommendations section of the Install guide in Red Hat Advanced Cluster Management for Kubernetes documentation.

1.2. NETWORK PORTS REQUIRED BETWEEN OPENSHIFT CONTAINER PLATFORM AND CEPH WHEN USING EXTERNAL MODE DEPLOYMENT

List of TCP ports, source OpenShift Container Platform and destination RHCS

<table>
<thead>
<tr>
<th>TCP ports</th>
<th>To be used for</th>
</tr>
</thead>
<tbody>
<tr>
<td>6789, 3300</td>
<td>Ceph Monitor</td>
</tr>
<tr>
<td>6800 - 7300</td>
<td>Ceph OSD, MGR, MDS</td>
</tr>
<tr>
<td>9283</td>
<td>Ceph MGR Prometheus Exporter</td>
</tr>
</tbody>
</table>

For more information about why these ports are required, see Chapter 2. Ceph network configuration of RHCS Configuration Guide.
CHAPTER 2. DEPLOY OPENSSHIFT DATA FOUNDATION USING RED HAT CEPH STORAGE

Red Hat OpenShift Data Foundation can make services from an external Red Hat Ceph Storage cluster available for consumption through OpenShift Container Platform clusters. You need to install the OpenShift Data Foundation operator and then create an OpenShift Data Foundation cluster for the external Ceph storage system.

2.1. INSTALLING RED HAT OPENSHEET DATA FOUNDATION OPERATOR

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

Prerequisites

- Access to an OpenShift Container Platform cluster using an account with `cluster-admin` and operator installation permissions.
- For additional resource requirements, see the Planning your deployment guide.

IMPORTANT

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

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

Procedure

1. Log in to the OpenShift Web Console.
2. Click Operators → OperatorHub.
3. Scroll or type OpenShift Data Foundation into the Filter by keyword box to find the OpenShift Data Foundation Operator.
4. Click Install.
5. Set the following options on the Install Operator page:
   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 is created during the operator installation.
   d. Select Approval Strategy as Automatic or Manual. If you select Automatic updates, then the Operator Lifecycle Manager (OLM) automatically upgrades the running instance of your Operator without any intervention.
If you select Manual updates, then the OLM creates an update request. As a cluster administrator, you must then manually approve that update request to update the Operator to a newer version.

e. Ensure that the Enable option is selected for the Console plugin.

f. Click Install.

Verification steps

- After the operator is successfully installed, a pop-up with a message, Web console update is available appears on the user interface. Click Refresh web console from this pop-up for the console changes to reflect.

- In the Web Console:
  - Navigate to Installed Operators and verify that the OpenShift Data Foundation Operator shows a green tick indicating successful installation.
  - Navigate to Storage and verify if the Data Foundation dashboard is available.

2.2. CREATING AN OPENSHIFT DATA FOUNDATION CLUSTER FOR EXTERNAL CEPH STORAGE SYSTEM

You need to create a new OpenShift Data Foundation cluster after you install OpenShift Data Foundation operator on OpenShift Container Platform deployed on VMware vSphere or user-provisioned bare metal infrastructures.

Prerequisites

- A valid Red Hat OpenShift Data Foundation Advanced subscription. To know how subscriptions for OpenShift Data Foundation work, see knowledgebase article on OpenShift Data Foundation subscriptions.

- Ensure the OpenShift Container Platform version is 4.14 or above before deploying OpenShift Data Foundation 4.14.

- OpenShift Data Foundation operator must be installed. For more information, see Installing OpenShift Data Foundation Operator using the Operator Hub.

- To check the supportability and interoperability of Red Hat Ceph Storage (RHCS) with Red Hat OpenShift Data Foundation in external mode, go to the lab Red Hat OpenShift Data Foundation Supportability and Interoperability Checker.
  - Select Service Type as ODF as Self-Managed Service.
  - Select appropriate Version from the drop down.
  - On the Versions tab, click the Supported RHCS Compatibility tab.

- 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 the 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. For more information, see Ceph Dashboard installation and access.

It is recommended that the external Red Hat Ceph Storage cluster has the PG Autoscaler enabled. 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 Data Foundation deployment. Red Hat recommends to use a separate pool for each OpenShift Data Foundation cluster.

Optional: If there is a zonegroup created apart from the default zonegroup, you need to add the hostname, `rook-ceph-rgw-ocs-external-storagecluster-cephobjectstore.openshift-storage.svc` to the zonegroup as OpenShift Data Foundation sends S3 requests to the RADOS Object Gateways (RGWs) with this hostname. For more information, see the Red Hat Knowledgebase solution Ceph - How to add hostnames in RGW zonegroup?

### Procedure

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

2. Click **OpenShift Data Foundation** and then click **Create StorageSystem**.

3. In the Backing storage page, select the following options:
   a. Select **Full deployment** for the **Deployment type** option.
   b. Select **Connect an external storage platform** from the available options.
   c. Select **Red Hat Ceph Storage** for **Storage platform**.
   d. Click **Next**.

4. In the Connection details page, provide the necessary information:
   a. Click on the **Download Script** link to download the python script for extracting Ceph cluster details.
   b. 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 the **admin key**.
      i. Run the following command on the RHCS node to view the list of available arguments:

      ```bash
      # 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.

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

Use the `yum install cephadm` command and then the `cephadm` command to deploy your RHCS cluster using containers. You must pull the RHCS container images using the `cephadm` command, rather than using `yum` for installing the Ceph packages onto nodes. For more information, see RHCS product documentation.

ii. 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:

```
```

In this example,

<table>
<thead>
<tr>
<th>rbd-data-pool-name</th>
<th>A mandatory parameter that is used for providing block storage in OpenShift Data Foundation.</th>
</tr>
</thead>
<tbody>
<tr>
<td>rgw-endpoint</td>
<td>(Optional) This parameter is required only if the object storage is to be provisioned through Ceph Rados Gateway for OpenShift Data Foundation. Provide the endpoint in the following format: <code>&lt;ip_address&gt;:</code>:&lt;port&gt;</td>
</tr>
<tr>
<td></td>
<td><strong>NOTE</strong> A fully-qualified domain name (FQDN) is also supported in the format <code>&lt;FQDN&gt;:</code>:&lt;PORT&gt;.</td>
</tr>
<tr>
<td>monitoring-endpoint</td>
<td>(Optional) This parameter accepts comma-separated list of IP addresses of active and standby mgrs reachable from the OpenShift Container Platform cluster. If not provided, the value is automatically populated.</td>
</tr>
<tr>
<td>monitoring-endpoint-port</td>
<td>(Optional) It is the port associated with the ceph-mgr Prometheus exporter specified by <code>--monitoring-endpoint</code>. If not provided, the value is automatically populated.</td>
</tr>
</tbody>
</table>
run-as-user  | (Optional) This parameter is used for providing 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

<table>
<thead>
<tr>
<th>Additional flags:</th>
</tr>
</thead>
<tbody>
<tr>
<td>rgw-pool-prefix</td>
</tr>
<tr>
<td>rgw-tls-cert-path</td>
</tr>
<tr>
<td>rgw-skip-tls</td>
</tr>
<tr>
<td>ceph-conf</td>
</tr>
<tr>
<td>cluster-name</td>
</tr>
<tr>
<td>output</td>
</tr>
<tr>
<td>cephfs-metadata-pool-name</td>
</tr>
<tr>
<td>cephfs-data-pool-name</td>
</tr>
<tr>
<td>cephfs-filesystem-name</td>
</tr>
<tr>
<td>rbd-metadata-ec-pool-name</td>
</tr>
</tbody>
</table>
(Optional) This parameter helps to print the executed commands without running them.

(Optional) This parameter restricts cephCSIKeyrings auth permissions to specific pools and clusters. Mandatory flags that need to be set with this are rbd-data-pool-name and cluster-name. You can also pass the cephfs-filesystem-name flag if there is CephFS user restriction so that permission is restricted to a particular CephFS filesystem.

NOTE
This parameter must be applied only for the new deployments. To restrict csi-users per pool and per cluster, you need to create new csi-users and new secrets for those csi-users.

Example with restricted auth permission:

```
# python3 /etc/ceph/create-external-cluster-resources.py --cephfs-filesystem-name myfs --rbd-data-pool-name replicapool --cluster-name rookStorage --restricted-auth-permission true
```

Example of JSON output generated using the python script:

```json
[{
"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-ids>", "mon-secret": "mon-secret"},
"name": "rook-ceph-operator-creds", "kind": "Secret", "data": {
"userID": "<user-id>", "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": "<pool>"},
"name": "monitoring-endpoint", "kind": "CephCluster", "data": {
"MonitoringEndpoint": "xxx.xxx.xxx.xxx", "MonitoringPort": "xxxx"},
"name": "rook-ceph-dashboard-link", "kind": "Secret", "data": {
"userID": "ceph-dashboard-link", "userKey": "<user-key>"},
"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"},
"name": "rgw-admin-ops-user", "kind": "Secret", "data": {
"accessKey": "<access-key>"},
"secretKey": "<secret-key>"}]}
```

Save the JSON output to a file with .json extension

For OpenShift Data Foundation to work seamlessly, ensure that the parameters (RGW endpoint, CephFS details, RBD pool, and so on) to be uploaded using the JSON file remain unchanged on the RHCS external cluster after the storage cluster creation.

**NOTE**
For OpenShift Data Foundation to work seamlessly, ensure that the parameters (RGW endpoint, CephFS details, RBD pool, and so on) to be uploaded using the JSON file remain unchanged on the RHCS external cluster after the storage cluster creation.
iv. Run the command when there is a multi-tenant deployment in which the RHCS cluster is already connected to OpenShift Data Foundation deployment with a lower version.

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

c. Click **Browse** to select and upload the JSON file. The content of the JSON file is populated and displayed in the text box.

d. Click **Next**
   The **Next** button is enabled only after you upload the `.json` file.

5. In the Review and create page, review if all the details are correct:
   - To modify any configuration settings, click **Back** to go back to the previous configuration page.

6. Click **Create StorageSystem**

**Verification steps**

To verify the final Status of the installed storage cluster:

1. In the OpenShift Web Console, navigate to **Installed Operators** → **OpenShift Data Foundation** → **Storage System** → **ocs-external-storagecluster-storagesystem** → **Resources**.

2. Verify that **Status** of **StorageCluster** is **Ready** and has a green tick.

3. To verify that OpenShift Data Foundation, pods and StorageClass are successfully installed, see **Verifying your external mode OpenShift Data Foundation installation for external Ceph storage system**.

### 2.3. VERIFYING YOUR OPENSHEET DATA FOUNDATION INSTALLATION FOR EXTERNAL CEPH STORAGE SYSTEM

Use this section to verify that OpenShift Data Foundation is deployed correctly.

#### 2.3.1. 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.

   **NOTE**

   If the **Show default projects** option is disabled, use the toggle button to list all the default projects.

   For more information on the expected number of pods for each component and how it varies depending on the number of nodes, see Table 2.1, “Pods corresponding to OpenShift Data Foundation components”

3. Verify that the following pods are in running state:

   **Table 2.1. Pods corresponding to OpenShift Data Foundation components**
<table>
<thead>
<tr>
<th>Component</th>
<th>Corresponding pods</th>
</tr>
</thead>
<tbody>
<tr>
<td>OpenShift Data Foundation Operator</td>
<td><a href="#">ocs-operator-*</a> (1 pod on any worker node)</td>
</tr>
<tr>
<td></td>
<td><a href="#">ocs-metrics-exporter-*</a> (1 pod on any worker node)</td>
</tr>
<tr>
<td></td>
<td><a href="#">odf-operator-controller-manager-*</a> (1 pod on any worker node)</td>
</tr>
<tr>
<td></td>
<td><a href="#">odf-console-*</a> (1 pod on any worker node)</td>
</tr>
<tr>
<td></td>
<td><a href="#">csi-addons-controller-manager-*</a> (1 pod on any worker node)</td>
</tr>
<tr>
<td>Rook-ceph Operator</td>
<td><a href="#">rook-ceph-operator-*</a></td>
</tr>
<tr>
<td></td>
<td>(1 pod on any worker node)</td>
</tr>
<tr>
<td>Multicloud Object Gateway</td>
<td><a href="#">noobaa-operator-*</a> (1 pod on any worker node)</td>
</tr>
<tr>
<td></td>
<td><a href="#">noobaa-core-*</a> (1 pod on any worker node)</td>
</tr>
<tr>
<td></td>
<td><a href="#">noobaa-db-pg-*</a> (1 pod on any worker node)</td>
</tr>
<tr>
<td></td>
<td><a href="#">noobaa-endpoint-*</a> (1 pod on any worker node)</td>
</tr>
<tr>
<td>CSI</td>
<td><a href="#">cephfs</a></td>
</tr>
<tr>
<td></td>
<td><a href="#">csi-cephfsplugin-*</a> (1 pod on each worker node)</td>
</tr>
<tr>
<td></td>
<td><a href="#">csi-cephfsplugin-provisioner-*</a> (2 pods distributed across worker nodes)</td>
</tr>
<tr>
<td></td>
<td>NOTE</td>
</tr>
<tr>
<td></td>
<td>If an MDS is not deployed in the external cluster, the csi-cephfsplugin pods will not be created.</td>
</tr>
<tr>
<td></td>
<td><a href="#">rbd</a></td>
</tr>
<tr>
<td></td>
<td><a href="#">csi-rbdplugin-*</a> (1 pod on each worker node)</td>
</tr>
<tr>
<td></td>
<td><a href="#">csi-rbdplugin-provisioner-*</a> (2 pods distributed across worker nodes)</td>
</tr>
</tbody>
</table>

2.3.2. Verifying that the OpenShift Data Foundation cluster is healthy

1. In the OpenShift Web Console, click **Storage → Data Foundation**.
2. In the **Status** card of the **Overview** tab, click **Storage System** and then click the storage system link from the pop up that appears.
3. In the **Status** card of the **Block and File** tab, verify that the **Storage Cluster** has a green tick.
4. In the **Details** card, verify that the cluster information is displayed.
For more information on the health of OpenShift Data Foundation cluster using the Block and File dashboard, see Monitoring OpenShift Data Foundation.

2.3.3. Verifying that the Multicloud Object Gateway is healthy

1. In the OpenShift Web Console, click **Storage → Data Foundation**.

2. In the **Status** card of the **Overview** tab, click **Storage System** and then click the storage system link from the pop up that appears.

   a. In the **Status** card of the **Object** tab, verify that both **Object Service** and **Data Resiliency** have a green tick.

   b. In the **Details** card, verify that the Multicloud Object Gateway (MCG) information is displayed.

   **NOTE**
   The RADOS Object Gateway is only listed in case RADOS Object Gateway endpoint details are included while deploying OpenShift Data Foundation in external mode.

For more information on the health of OpenShift Data Foundation cluster using the object dashboard, see Monitoring OpenShift Data Foundation.

2.3.4. Verifying that the storage classes are created and listed

1. Click **Storage → Storage Classes** from the left pane of the OpenShift Web Console.

2. Verify that the following storage classes are created with the OpenShift Data Foundation 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 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.

2.3.5. Verifying that Ceph cluster is connected

Run the following command to verify if the OpenShift Data Foundation cluster is connected to the external Red Hat Ceph Storage cluster.
$ oc get cephcluster -n openshift-storage
NAME                                      DATADIRHOSTPATH   MONCOUNT   AGE   PHASE       MESSAGE
HEALTH      EXTERNAL
ocs-external-storagecluster-cephcluster                                30m   Connected   Cluster connected successfully   HEALTH_OK   true

2.3.6. 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  30m   30m Ready true   2021-11-17T09:09:52Z  4.14.0
CHAPTER 3. DEPLOY OPENSHELL DATA FOUNDATION USING IBM FLASHSYSTEM

OpenShift Data Foundation can use IBM FlashSystem storage available for consumption through OpenShift Container Platform clusters. You need to install the OpenShift Data Foundation operator and then create an OpenShift Data Foundation cluster for IBM FlashSystem storage.

3.1. INSTALLING RED HAT OPENSHELL DATA FOUNDATION OPERATOR

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

Prerequisites

- Access to an OpenShift Container Platform cluster using an account with cluster-admin and operator installation permissions.
- For additional resource requirements, see the Planning your deployment guide.

IMPORTANT

- When you need to override the cluster-wide default node selector for OpenShift Data Foundation, you can use the following command to specify a blank node selector for the openshift-storage namespace (create openshift-storage namespace in this case):
  
  ```
  $ oc annotate namespace openshift-storage openshift.io/node-selector=
  ```

Procedure

1. Log in to the OpenShift Web Console.

2. Click Operators → OperatorHub.

3. Scroll or type OpenShift Data Foundation into the Filter by keyword box to find the OpenShift Data Foundation Operator.

4. Click Install.

5. Set the following options on the Install Operator page:
   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 is created during the operator installation.
   d. Select Approval Strategy as Automatic or Manual.

   If you select Automatic updates, then the Operator Lifecycle Manager (OLM) automatically upgrades the running instance of your Operator without any intervention.
If you select Manual updates, then the OLM creates an update request. As a cluster administrator, you must then manually approve that update request to update the Operator to a newer version.

e. Ensure that the Enable option is selected for the Console plugin.

f. Click Install.

Verification steps

- After the operator is successfully installed, a pop-up with a message, Web console update is available appears on the user interface. Click Refresh web console from this pop-up for the console changes to reflect.

- In the Web Console:
  - Navigate to Installed Operators and verify that the OpenShift Data Foundation Operator shows a green tick indicating successful installation.
  - Navigate to Storage and verify if the Data Foundation dashboard is available.

3.2. CREATING AN OPENSOURCE DATA FOUNDATION CLUSTER FOR EXTERNAL IBM FLASHSYSTEM STORAGE

You need to create a new OpenShift Data Foundation cluster after you install the OpenShift Data Foundation operator on the OpenShift Container Platform.

Prerequisites

- A valid Red Hat OpenShift Data Foundation Advanced subscription. For more information, see the knowledgebase article on OpenShift Data Foundation subscriptions.

- For Red Hat Enterprise Linux® operating system, ensure that there is iSCSI connectivity and then configure Linux multipath devices on the host.

- For Red Hat Enterprise Linux CoreOS or when the packages are already installed, configure Linux multipath devices on the host.

- Ensure to configure each worker with storage connectivity according to your storage system instructions. For the latest supported FlashSystem storage systems and versions, see IBM ODF FlashSystem driver documentation.

Procedure

1. In the OpenShift Web Console, click Operators → Installed Operators to view all the installed operators.
   Ensure that the Project selected is openshift-storage.

2. Click OpenShift Data Foundation and then click Create StorageSystem.

3. In the Backing storage page, select the following options:
   a. Select Full deployment for the Deployment type option.
   b. Select Connect an external storage platform from the available options.
c. Select **IBM FlashSystem Storage** from the **Storage platform** list.

d. Click **Next**.

4. In the Create storage class page, provide the following information:

a. Enter a name for the storage class.
   When creating block storage persistent volumes, select the storage class `<storage_class_name>` for best performance. The storage class allows direct I/O path to the FlashSystem.

b. Enter the following details of IBM FlashSystem connection:
   - IP address
   - User name
   - Password
   - Pool name

c. Select **thick** or **thin** for the **Volume mode**.

d. Click **Next**.

5. In the Capacity and nodes page, provide the necessary details:

a. Select a value for Requested capacity.
   The available options are **0.5 TiB**, **2 TiB**, and **4 TiB**. The requested capacity is dynamically allocated on the infrastructure storage class.

b. Select at least three nodes in three different zones.
   It is recommended to start with at least 14 CPUs and 34 GiB of RAM per node. If the nodes selected do not match the OpenShift Data Foundation cluster requirement of an aggregated 30 CPUs and 72 GiB of RAM, a minimal cluster will be deployed. For minimum starting node requirements, see the **Resource requirements** section in the Planning guide.

c. Click **Next**.

6. Optional: In the Security and network page, provide the necessary details:

a. To enable encryption, select **Enable data encryption for block and file storage**
   i. Choose any one or both Encryption level:
      - **Cluster-wide encryption** to encrypt the entire cluster (block and file).
      - **StorageClass encryption** to create encrypted persistent volume (block only) using encryption enabled storage class.
   ii. Select the **Connect to an external key management service** checkbox. This is optional for cluster-wide encryption.
      A. Key Management Service Provider is set to Vault by default.
      B. 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 the Backend Path that is dedicated and unique to OpenShift Data Foundation.

B. Optional: Enter TLS Server Name and Vault Enterprise Namespace.

C. Provide CA Certificate, Client Certificate, and Client Private Key by uploading the respective PEM encoded certificate file.

iv. Click Save.

v. Select Default (SDN) if you are using a single network or Custom (Multus) if you are using multiple network interfaces.

A. Select a Public Network Interface from the dropdown.

B. Select a Cluster Network Interface from the dropdown. NOTE: If you are using only one additional network interface, select the single NetworkAttachmentDefinition, that is, ocs-public-cluster for the Public Network Interface, and leave the Cluster Network Interface blank.

vi. Click Next.

b. To enable in-transit encryption, select In-transit encryption.

i. Select a Network.

ii. Click Next.

7. In the Review and create page, review if all the details are correct:

● To modify any configuration settings, click Back to go back to the previous configuration page.

8. Click Create StorageSystem.

Verification Steps

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.

   NOTE

   If the Show default projects option is disabled, use the toggle button to list all the default projects.

Table 3.1. Pods corresponding to OpenShift Data Foundation components
<table>
<thead>
<tr>
<th>Component</th>
<th>Corresponding pods</th>
</tr>
</thead>
<tbody>
<tr>
<td>OpenShift Data Foundation Operator</td>
<td>• ocs-operator-* (1 pod on any worker node)</td>
</tr>
<tr>
<td></td>
<td>• ocs-metrics-exporter-* (1 pod on any worker node)</td>
</tr>
<tr>
<td></td>
<td>• odf-operator-controller-manager-* (1 pod on any worker node)</td>
</tr>
<tr>
<td></td>
<td>• odf-console-* (1 pod on any worker node)</td>
</tr>
<tr>
<td></td>
<td>• csi-addons-controller-manager-* (1 pod on any worker node)</td>
</tr>
<tr>
<td>ibm-storage-odf-operator</td>
<td>• ibm-storage-odf-operator-* (2 pods on any worker nodes)</td>
</tr>
<tr>
<td></td>
<td>• ibm-odf-console-*</td>
</tr>
<tr>
<td>ibm-flashsystem-storage</td>
<td>ibm-flashsystem-storage-* (1 pod on any worker node)</td>
</tr>
<tr>
<td>rook-ceph Operator</td>
<td>rook-ceph-operator-* (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 worker node)</td>
</tr>
<tr>
<td></td>
<td>• noobaa-db-pg-* (1 pod on any worker node)</td>
</tr>
<tr>
<td></td>
<td>• noobaa-endpoint-* (1 pod on any worker node)</td>
</tr>
<tr>
<td>CSI</td>
<td>• ibm-block-csi-* (1 pod on any worker node)</td>
</tr>
</tbody>
</table>

**Verifying that the OpenShift Data Foundation cluster is healthy**

1. In the Web Console, click **Storage → Data Foundation**.
2. In the Status card of the **Overview** tab, verify that the **Storage System** has a green tick mark.
3. In the **Details** card, verify that the cluster information is displayed.

For more information on the health of the OpenShift Data Foundation cluster using the Block and File dashboard, see **Monitoring OpenShift Data Foundation**.

**Verifying that the Multicloud Object Gateway is healthy**

1. In the Web Console, click **Storage → Data Foundation**.
2. In the Status card of the Overview tab, click Storage System and then click the storage system link from the pop up.

3. In the Status card of the Object tab, verify that both Object Service and Data Resiliency have a green tick.

4. In the Details card, verify that the MCG information is displayed.

For more information on the health of the OpenShift Data Foundation cluster using the object dashboard, see Monitoring OpenShift Data Foundation.

Verifying that IBM FlashSystem is connected and the storage cluster is ready

- Run the following command to verify if the OpenShift Data Foundation cluster is connected to the external IBM FlashSystem.

```
$ oc get flashsystemclusters.odf.ibm.com
NAME                     AGE   PHASE   CREATED AT
ibm-flashsystemcluster   35s           2021-09-23T07:44:52Z
```

Verifying the StorageSystem of the storage

- Run the following command to verify the storageSystem of IBM FlashSystem storage cluster.

```
$ oc get storagesystems.odf.openshift.io
NAME                                   STORAGE-SYSTEM-KIND                       STORAGE-SYSTEM-NAME
ibm-flashsystemcluster-storagesystem   flashsystemcluster.odf.ibm.com/v1alpha1 ibm-flashsystemcluster
ocs-storagecluster-storagesystem       storagecluster.ocs.openshift.io/v1        ocs-storagecluster
```

Verifying the subscription of the IBM operator

- Run the following command to verify the subscription:

```
$ oc get subscriptions.operators.coreos.com
NAME                                                                      PACKAGE                    SOURCE          CHANNEL
ibm-storage-odf-operator                                                   ibm-storage-odf-operator odf-catalogsource alpha
noobaa-operator-alpha-odf-catalogsource-openshift-storage                noobaa-operator
odf-catalogsource alpha
ocs-operator-alpha-odf-catalogsource-openshift-storage                    ocs-operator               odf-catalogsource alpha
odf-operator                                                             odf-operator               odf-catalogsource alpha
```

Verifying the CSVs

- Run the following command to verify that the CSVs are in the succeeded state.

```
$ oc get csv
NAME             DISPLAY  VERSION  REPLACES
```

CHAPTER 3. DEPLOY OPENShift DATA FOUNDATION USING IBM FLASHSYSTEM

21
### PHASE

- **ibm-block-csi-operator.v1.6.0**  Operator for IBM block storage CSI driver  1.6.0  ibm-block-csi-operator.v1.5.0  Succeeded
- **ibm-storage-odf-operator.v0.2.1**  IBM Storage ODF operator  0.2.1
  - Installing  
  - Succeeded
- **noobaa-operator.v5.9.0**  NooBaa Operator  5.9.0  Succeeded
- **ocs-operator.v4.14.0**  OpenShift Container Storage  4.14.0  Succeeded
- **odf-operator.v4.14.0**  OpenShift Data Foundation  4.14.0  Succeeded

#### Verifying the IBM operator and CSI pods

- Run the following command to verify the IBM operator and CSI pods:

  ```
  $ oc get pods
  NAME                             READY   STATUS              RESTARTS AGE
  5cb2b16ec2b11bf63dbe691d44a63535dc026bb5315d5075dc6c398b3c58l94   0/1     Completed 10m
  7c806f6568f85cf10d72508261a2535c220429b54dbcf87349b9b4b9838fgctg 0/1   Completed 8m47s
  c4b05566c04876677a22d39fc9c02512401d0962109610e85c8fb900d3jd7k2 0/1   Completed 10m
  c5d1376974666727b02bf25b3a4828241612186744ef417a668b4bc1759rzts 0/1   Completed 10m
  ibm-block-csi-operator-7b656d6cc8-bqnwp                          1/1     Running   8m3s
  ibm-odf-console-97cb7c84c-r52dq                                   0/1     ContainerCreating 8m4s
  ibm-storage-odf-operator-57b8bc47df-mgkc7                         1/2     ImagePullBackOff 94s
  noobaa-operator-7698579d56-x2zqs                                  1/1     Running   9m37s
  ocs-metrics-exporter-94b57d764-zq2g2                              1/1     Running   9m32s
  ocs-operator-5d96d778f6-vx1q5                                     1/1     Running   9m33s
  odf-catalogsource-j7q72                                          1/1     Running 10m
  odf-console-8987868c8-m7v29                                       1/1     Running   9m35s
  odf-operator-controller-manager-5dbf785564-rwsgq                  2/2     Running   9m35s
  rook-ceph-operator-68b4b976d8-dlc6w                                1/1     Running   9m32s
  ```

---

Red Hat OpenShift Data Foundation 4.14 Deploying OpenShift Data Foundation in external mode
CHAPTER 4. UNINSTALLING OPENSHIFT DATA FOUNDATION FROM EXTERNAL STORAGE SYSTEM

Use the steps in this section to uninstall OpenShift Data Foundation. Uninstalling OpenShift Data Foundation 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>
<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 DataDirHostPath</td>
</tr>
<tr>
<td>cleanup-policy</td>
<td>retain</td>
<td>No</td>
<td>Rook does not clean up the physical drives and the DataDirHostPath</td>
</tr>
<tr>
<td>mode</td>
<td>graceful</td>
<td>Yes</td>
<td>Rook and NooBaa pauses 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 uninstall mode by editing the value of the annotation by using the following commands:
$ oc annotate storagecluster ocs-external-storagecluster -n openshift-storage uninstall.ocs.openshift.io/mode="forced" --overwrite storagecluster.ocs.openshift.io/ocs-external-storagecluster annotated

Prerequisites

- Ensure that the OpenShift Data Foundation 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 Data Foundation.

- Ensure that applications are not consuming persistent volume claims (PVCs) or object bucket claims (OBCs) using the storage classes provided by OpenShift Data Foundation.

Procedure

1. Delete the volume snapshots that are using OpenShift Data Foundation.
   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 Data Foundation.
      
      $ oc delete volumesnapshot <VOLUME-SNAPSHOT-NAME> -n <NAMESPACE>

2. Delete PVCs and OBCs that are using OpenShift Data Foundation.
   In the default uninstall mode (graceful), the uninstaller waits till all the PVCs and OBCs that use OpenShift Data Foundation 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 Data Foundation.
      See Removing monitoring stack from OpenShift Data Foundation

   b. Delete OpenShift Container Platform Registry PVCs using OpenShift Data Foundation.
      Removing OpenShift Container Platform registry from OpenShift Data Foundation

   c. Delete OpenShift Container Platform logging PVCs using OpenShift Data Foundation.
      Removing the cluster logging operator from OpenShift Data Foundation

   d. Delete other PVCs and OBCs provisioned using OpenShift Data Foundation.
      
      - Given below is a sample script to identify the PVCs and OBCs provisioned using OpenShift Data Foundation. The script ignores the PVCs and OBCs that are used internally by OpenShift Data Foundation.

      ```bash
      #!/bin/bash
      RBD_PROVISIONER="openshift-storage.rbd.csi.ceph.com"
      CEPHFS_PROVISIONER="openshift-storage.cephfs.csi.ceph.com"
      ```
NOOBAA_PROVISIONER="openshift-storage.noobaa.io/ocb"
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, and so on
that are created in the cluster.

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

  $ oc delete -n openshift-storage storagesystem --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 Data Foundation, 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 Data Foundation are deleted. If there is any PV left in the **Released** state, delete it.

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

6. Remove **CustomResourceDefinitions**.

   ```bash
   ```

7. To ensure that OpenShift Data Foundation is uninstalled completely:

   a. In the OpenShift Container Platform Web Console, click **Storage**.

   b. Verify that **OpenShift Data Foundation** no longer appears under Storage.

### 4.1. REMOVING MONITORING STACK FROM OPENSHIFT DATA FOUNDATION

Use this section to clean up the monitoring stack from OpenShift Data Foundation.

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 the 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
   NAME                           READY STATUS    AGE
   pod/alertmanager-main-0        3/3     Running   8d
   pod/alertmanager-main-1        3/3     Running   8d
   pod/alertmanager-main-2        3/3     Running   8d
   pod/cluster-monitoring-operator-84457656d-pkrxm 1/1     Running   8d
   pod/grafana-79ccf6689f-2li28   2/2     Running   8d
   ```
<table>
<thead>
<tr>
<th>NAME</th>
<th>STATUS</th>
<th>VOLUME</th>
</tr>
</thead>
<tbody>
<tr>
<td>pod/kube-state-metrics-7d86fb966-rvd9w</td>
<td>3/3</td>
<td>Running 0 8d</td>
</tr>
<tr>
<td>pod/node-exporter-25894</td>
<td>2/2</td>
<td>Running 0 8d</td>
</tr>
<tr>
<td>pod/node-exporter-4dsd7</td>
<td>2/2</td>
<td>Running 0 8d</td>
</tr>
<tr>
<td>pod/node-exporter-6p4zc</td>
<td>2/2</td>
<td>Running 0 8d</td>
</tr>
<tr>
<td>pod/node-exporter-jbjvg</td>
<td>2/2</td>
<td>Running 0 8d</td>
</tr>
<tr>
<td>pod/node-exporter-jj4t5</td>
<td>2/2</td>
<td>Running 0 6d18h</td>
</tr>
<tr>
<td>pod/node-exporter-k856s</td>
<td>2/2</td>
<td>Running 0 6d18h</td>
</tr>
<tr>
<td>pod/node-exporter-rf8gn</td>
<td>2/2</td>
<td>Running 0 8d</td>
</tr>
<tr>
<td>pod/node-exporter-rmb5m</td>
<td>2/2</td>
<td>Running 0 6d18h</td>
</tr>
<tr>
<td>pod/node-exporter-zj7kx</td>
<td>2/2</td>
<td>Running 0 8d</td>
</tr>
<tr>
<td>pod/openshift-state-metrics-59dbd41654-4cilng</td>
<td>3/3</td>
<td>Running 0 8d</td>
</tr>
<tr>
<td>pod/prometheus-adapter-5df5865596-k8dzn</td>
<td>1/1</td>
<td>Running 0 7d23h</td>
</tr>
<tr>
<td>pod/prometheus-adapter-5df5865596-n2gj9</td>
<td>1/1</td>
<td>Running 0 7d23h</td>
</tr>
<tr>
<td>pod/prometheus-k8s-0</td>
<td>6/6</td>
<td>Running 1 8d</td>
</tr>
<tr>
<td>pod/prometheus-k8s-1</td>
<td>6/6</td>
<td>Running 1 8d</td>
</tr>
<tr>
<td>pod/prometheus-operator-55cf8858c9-c4zd9</td>
<td>1/1</td>
<td>Running 0 6d21h</td>
</tr>
<tr>
<td>pod/telemeter-client-78fc8fc97d-2rgfp</td>
<td>3/3</td>
<td>Running 0 8d</td>
</tr>
</tbody>
</table>

2. Edit the monitoring configmap.

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

Remove any config sections that reference the OpenShift Data Foundation 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
In this example, `alertmanagerMain` and `prometheusK8s` monitoring components are using the OpenShift Data Foundation PVCs.

3. 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 Data Foundation PVC.

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

<table>
<thead>
<tr>
<th>NAME</th>
<th>STATUS</th>
<th>VOLUME</th>
</tr>
</thead>
<tbody>
<tr>
<td>pod/alertmanager-main-0</td>
<td>Terminating</td>
<td></td>
</tr>
<tr>
<td>pod/alertmanager-main-1</td>
<td>Terminating</td>
<td></td>
</tr>
<tr>
<td>pod/alertmanager-main-2</td>
<td>Terminating</td>
<td></td>
</tr>
<tr>
<td>pod/cluster-monitoring-operator-84cd9df668-zhjfn</td>
<td>1/1</td>
<td>Running</td>
</tr>
<tr>
<td>pod/grafana-5db6fd97f8-pmtbf</td>
<td>2/2</td>
<td>Running</td>
</tr>
<tr>
<td>pod/kube-state-metrics-895899678-z2r9q</td>
<td>3/3</td>
<td>Running</td>
</tr>
<tr>
<td>pod/node-exporter-4njxv</td>
<td>2/2</td>
<td>Running</td>
</tr>
<tr>
<td>pod/node-exporter-b8ckz</td>
<td>2/2</td>
<td>Running</td>
</tr>
<tr>
<td>pod/node-exporter-c2vp5</td>
<td>2/2</td>
<td>Running</td>
</tr>
<tr>
<td>pod/node-exporter-cq65n</td>
<td>2/2</td>
<td>Running</td>
</tr>
<tr>
<td>pod/node-exporter-f5sm7</td>
<td>2/2</td>
<td>Running</td>
</tr>
<tr>
<td>pod/node-exporter-f852c</td>
<td>2/2</td>
<td>Running</td>
</tr>
<tr>
<td>pod/node-exporter-l9zn7</td>
<td>2/2</td>
<td>Running</td>
</tr>
<tr>
<td>pod/node-exporter-ngbs8</td>
<td>2/2</td>
<td>Running</td>
</tr>
<tr>
<td>pod/node-exporter-rv4v9</td>
<td>2/2</td>
<td>Running</td>
</tr>
<tr>
<td>pod/openshift-state-metrics-77d5f699d8-69q5x</td>
<td>3/3</td>
<td>Running</td>
</tr>
<tr>
<td>pod/prometheus-adapter-765465b56-4tbxx</td>
<td>1/1</td>
<td>Running</td>
</tr>
<tr>
<td>pod/prometheus-adapter-765465b56-s2qqg2</td>
<td>1/1</td>
<td>Running</td>
</tr>
<tr>
<td>pod/prometheus-k8s-0</td>
<td>6/6</td>
<td>Terminating</td>
</tr>
<tr>
<td>pod/prometheus-k8s-1</td>
<td>6/6</td>
<td>Terminating</td>
</tr>
<tr>
<td>pod/prometheus-operator-cbf89f9-ldnwc</td>
<td>1/1</td>
<td>Running</td>
</tr>
<tr>
<td>pod/telemeter-client-7b5dd4489-2xfpz</td>
<td>3/3</td>
<td>Running</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>NAME</th>
<th>STATUS</th>
<th>VOLUME</th>
<th>CAPACITY</th>
<th>ACCESS MODES</th>
<th>STORAGECLASS</th>
<th>AGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>pod/alertmanager-main-0</td>
<td>Terminating</td>
<td></td>
<td>10h</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pod/alertmanager-main-1</td>
<td>Terminating</td>
<td></td>
<td>10h</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pod/alertmanager-main-2</td>
<td>Terminating</td>
<td></td>
<td>10h</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pod/cluster-monitoring-operator-84cd9df668-zhjfn</td>
<td>1/1</td>
<td>Running</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pod/grafana-5db6fd97f8-pmtbf</td>
<td>2/2</td>
<td>Running</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pod/kube-state-metrics-895899678-z2r9q</td>
<td>3/3</td>
<td>Running</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pod/node-exporter-4njxv</td>
<td>2/2</td>
<td>Running</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pod/node-exporter-b8ckz</td>
<td>2/2</td>
<td>Running</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pod/node-exporter-c2vp5</td>
<td>2/2</td>
<td>Running</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pod/node-exporter-cq65n</td>
<td>2/2</td>
<td>Running</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pod/node-exporter-f5sm7</td>
<td>2/2</td>
<td>Running</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pod/node-exporter-f852c</td>
<td>2/2</td>
<td>Running</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pod/node-exporter-l9zn7</td>
<td>2/2</td>
<td>Running</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pod/node-exporter-ngbs8</td>
<td>2/2</td>
<td>Running</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pod/node-exporter-rv4v9</td>
<td>2/2</td>
<td>Running</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pod/openshift-state-metrics-77d5f699d8-69q5x</td>
<td>3/3</td>
<td>Running</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pod/prometheus-adapter-765465b56-4tbxx</td>
<td>1/1</td>
<td>Running</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pod/prometheus-adapter-765465b56-s2qqg2</td>
<td>1/1</td>
<td>Running</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pod/prometheus-k8s-0</td>
<td>6/6</td>
<td>Terminating</td>
<td>9m47s</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pod/prometheus-k8s-1</td>
<td>6/6</td>
<td>Terminating</td>
<td>9m47s</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pod/prometheus-operator-cbf89f9-ldnwc</td>
<td>1/1</td>
<td>Running</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pod/telemeter-client-7b5dd4489-2xfpz</td>
<td>3/3</td>
<td>Running</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
4. Delete relevant PVCs. Make sure you delete all the PVCs that are consuming the storage classes.

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

4.2. REMOVING OPENSOURCES CONTAINER PLATFORM REGISTRY FROM OPENSOURCES DATA FOUNDATION

Use this section to clean up the OpenShift Container Platform registry from OpenShift Data Foundation. 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 Data Foundation 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.

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

### 4.3. REMOVING THE CLUSTER LOGGING OPERATOR FROM OPENSHIFT DATA FOUNDATION

Use this section to clean up the cluster logging operator from OpenShift Data Foundation.

The Persistent Volume Claims (PVCs) that are created as a part of configuring the cluster logging operator are in the `openshift-logging` namespace.

**Prerequisites**

- The cluster logging instance should have been configured to use the OpenShift Data Foundation 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 the PVCs.
4.4. REMOVING EXTERNAL IBM FLASHSYSTEM SECRET

You need to clean up the FlashSystem secret from OpenShift Data Foundation while uninstalling. This secret is created when you configure the external IBM FlashSystem Storage. For more information, see Creating an OpenShift Data Foundation Cluster for external IBM FlashSystem storage.

Procedure

- Remove the IBM FlashSystem secret by using the following command:

  $ oc delete secret -n openshift-storage ibm-flashsystem-storage