Red Hat OpenShift Container Storage 4.7

Deploying OpenShift Container Storage using IBM Power Systems

How to install and set up your IBM Power Systems environment

Last Updated: 2021-05-19

Red Hat OpenShift Container Storage 4.7 Deploying OpenShift Container Storage using IBM Power Systems

How to install and set up your IBM Power Systems environment

Legal Notice

Abstract

Read this document for instructions on installing Red Hat OpenShift Container Storage 4.7 to use local storage on IBM Power Systems.

Table of Contents

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**.

PREFACE

Red Hat OpenShift Container Storage 4.7 supports deployment on existing Red Hat OpenShift Container Platform (RHOCP) IBM Power clusters in connected environments along with out-of-the-box support for proxy environments.

NOTE

Only internal Openshift Container Storage clusters are supported on IBM Power Systems. See Planning your deployment for more information about deployment requirements.

To deploy OpenShift Container Storage, follow the appropriate deployment process:

- Internal-Attached Devices mode
  - Deploy using local storage devices

CHAPTER 1. DEPLOYING USING LOCAL STORAGE DEVICES

Deploying OpenShift Container Storage on OpenShift Container Platform using local storage devices provided by IBM Power Systems enables you to create internal cluster resources. This approach internally provisions base services. Then, all applications can access additional storage classes.

NOTE

Only internal Openshift Container Storage clusters are supported on IBM Power Systems. See Planning your deployment for more information about deployment requirements.

1. Understand the requirements for installing OpenShift Container Storage using local storage devices.
2. Install the Red Hat OpenShift Container Storage Operator .
3. Install Local Storage Operator .
4. Creating OpenShift Container Storage cluster on IBM Power Systems .

1.1. REQUIREMENTS FOR INSTALLING OPENSHIFT CONTAINER STORAGE USING LOCAL STORAGE DEVICES

- The cluster must consist of at least three OpenShift Container Platform worker nodes in the cluster with locally attached storage devices on each of them.
  - Each of the three selected nodes must have at least one raw block device available to be used by OpenShift Container Storage.
  - The devices to be used must be empty, that is, there should be no persistent volumes (PVs), volume groups (VGs), or local volumes (LVs) remaining on the disks.
- You must have a minimum of three labeled nodes.
  - Each node that has local storage devices to be used by OpenShift Container Storage must have a specific label to deploy OpenShift Container Storage pods. To label the nodes, use the following command:

```
$ oc label nodes <NodeNames> cluster.ocs.openshift.io/openshift-storage=''
```

Minimum starting node requirements

See Resource requirements section in Planning guide.

Requirements for upgrading local storage devices

- You must upgrade to the OpenShift Container Platform 4.7 before deploying OpenShift Container Storage 4.7. For information, see Updating OpenShift Container Platform clusters guide.
- The Local Storage Operator version must match the Red Hat OpenShift Container Platform version in order to have the Local Storage Operator fully supported with Red Hat OpenShift Container Storage. Upgrading Red Hat OpenShift Container Platform does not upgrade the Local Storage Operator.

1.2. INSTALLING RED HAT OPENSHIFT CONTAINER STORAGE OPERATOR

You can install Red Hat OpenShift Container Storage Operator using the Red Hat OpenShift Container Platform Operator Hub. For information about the hardware and software requirements, see Planning your deployment.

Prerequisites

- Access to an OpenShift Container Platform cluster using an account with cluster-admin and Operator installation permissions.
- You must have at least three worker nodes in the RHOCP cluster.

NOTE

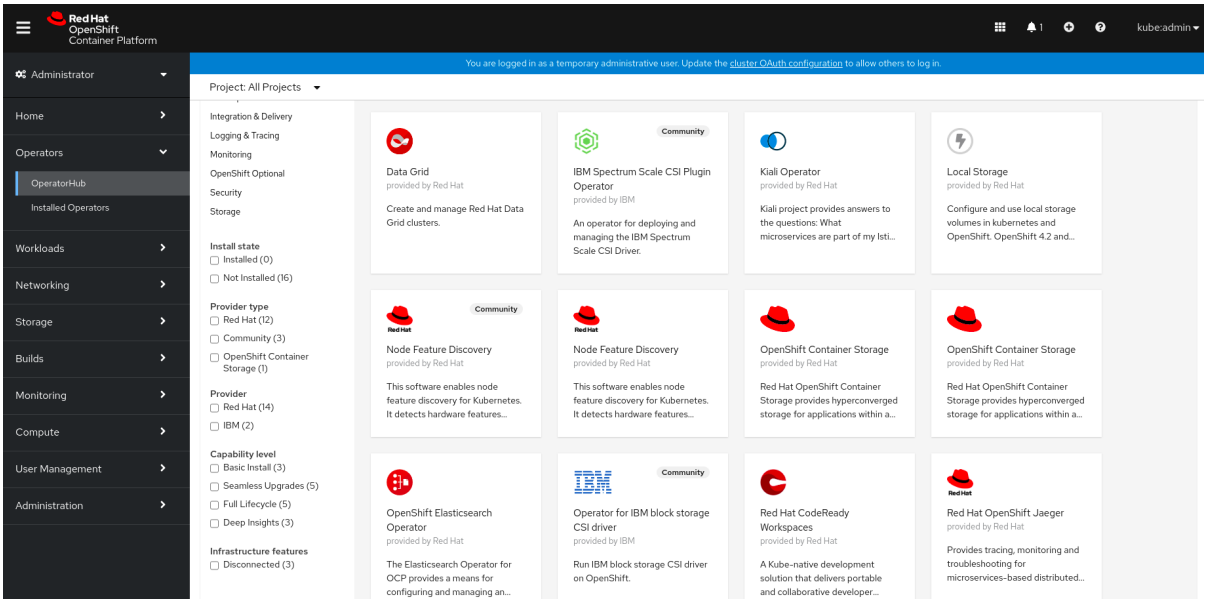
When you need to override the cluster-wide default node selector for OpenShift Container Storage, you can use the following command in command line interface to specify a blank node selector for the **openshift-storage** namespace:

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

Procedure

1. Navigate in the left pane of the OpenShift Web Console to click **Operators → OperatorHub**.

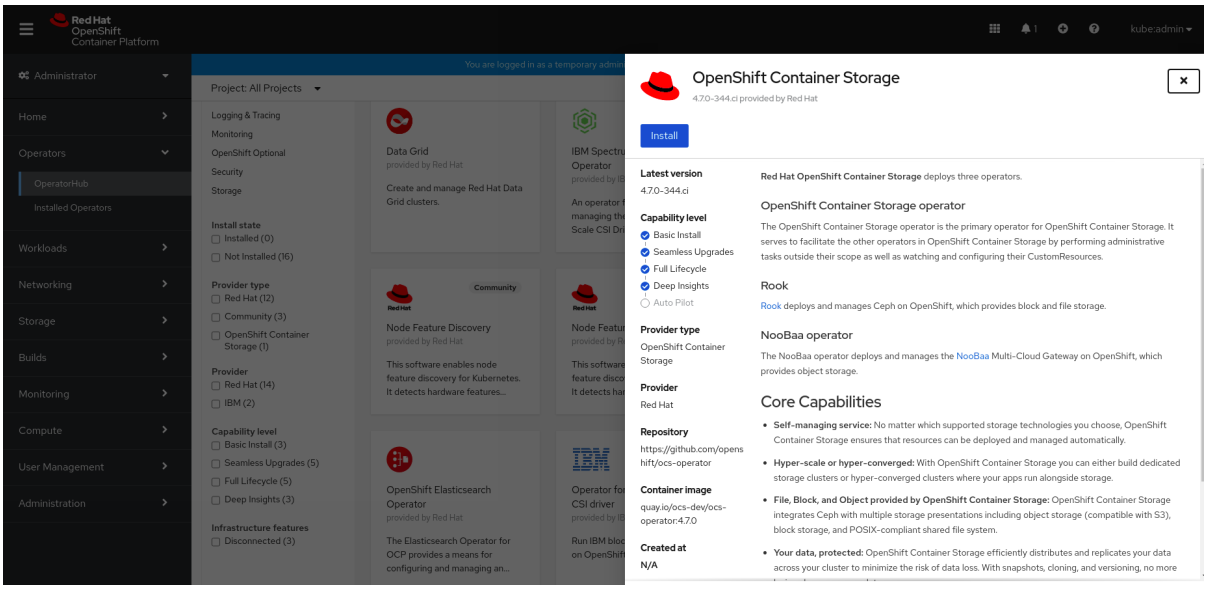
   Figure 1.1. List of operators in the Operator Hub



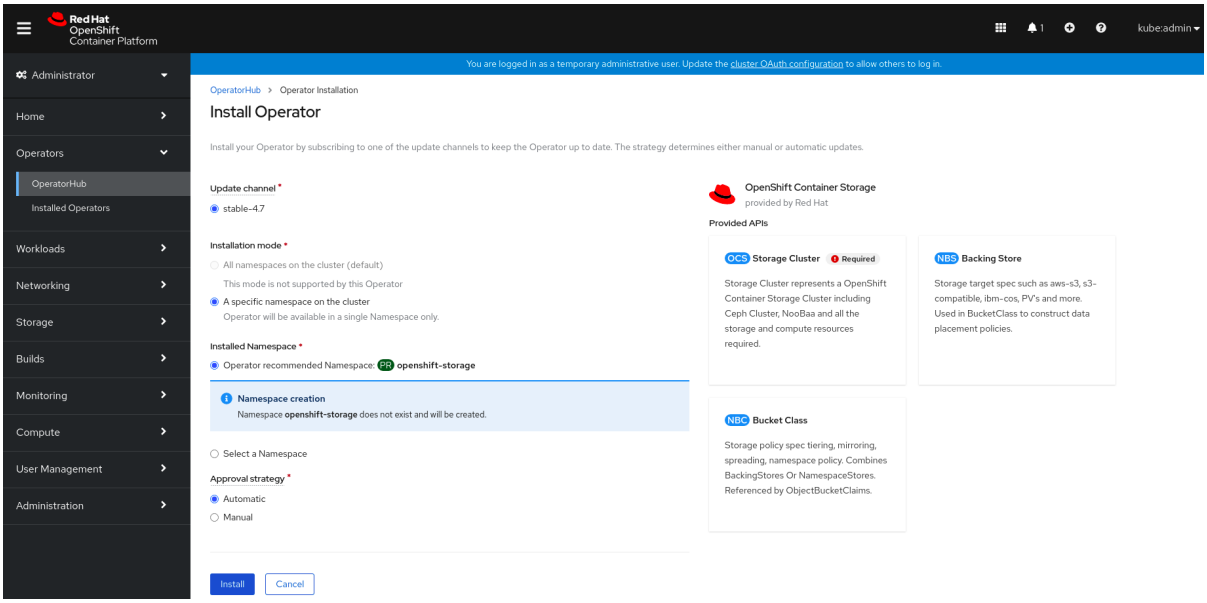
2. Click on **OpenShift Container Storage**.
   You can use the **Filter by keyword** text box or the filter list to search for OpenShift Container Storage from the list of operators.

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

Figure 1.2. Install Operator page



After Clicking on Install button, following page will appear.



4. On the **Install Operator** page, the following 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**.

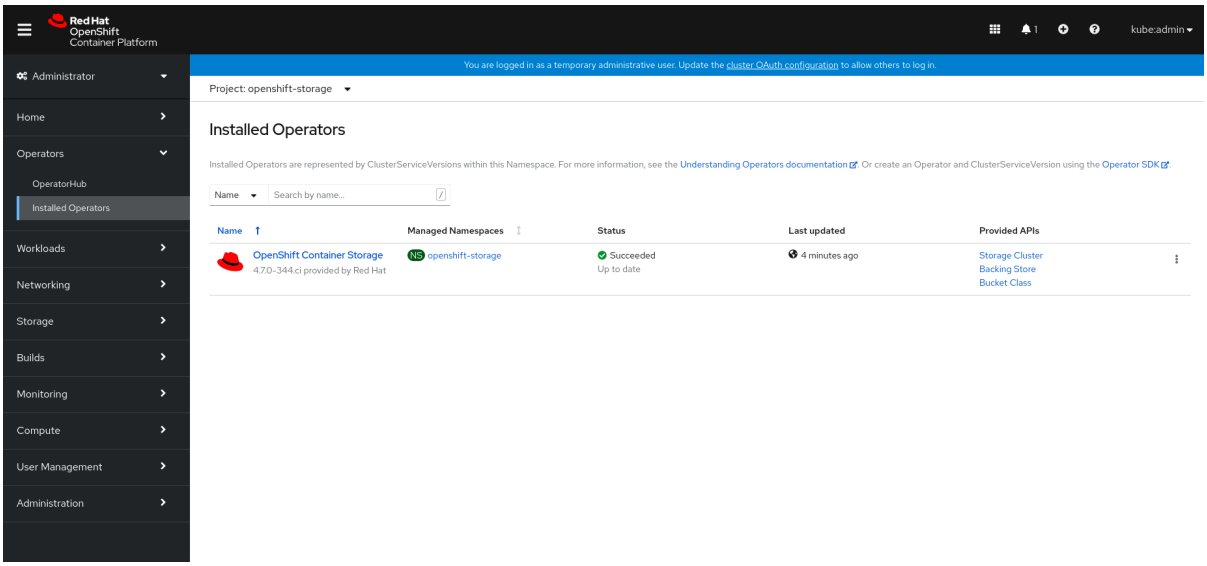
   e. Click **Install**.
   If you selected **Automatic** updates, then the Operator Lifecycle Manager (OLM) automatically upgrades the running instance of your Operator without human intervention.
   If you selected **Manual** updates, then the OLM creates an update request. As a cluster administrator, you must then manually approve that update request to have the Operator updated to the new version.

Verification steps

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

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

Figure 1.3. Installed Operators dashboard



1.3. INSTALLING LOCAL STORAGE OPERATOR

Use this procedure to install the Local Storage Operator from the Operator Hub before creating OpenShift Container Storage clusters on local storage devices.

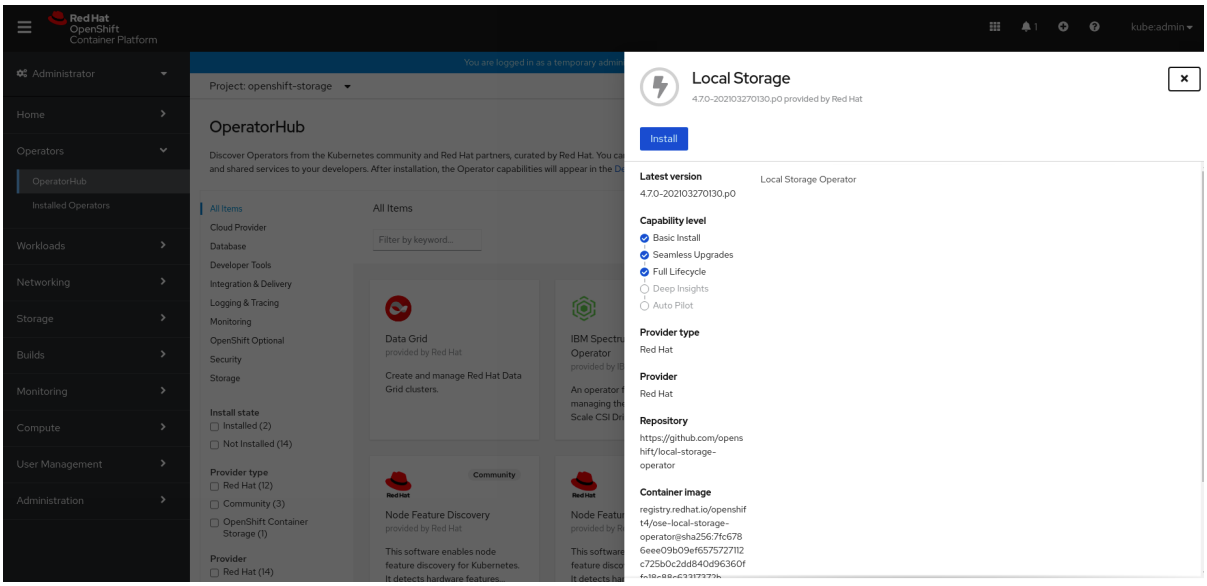
Procedure

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

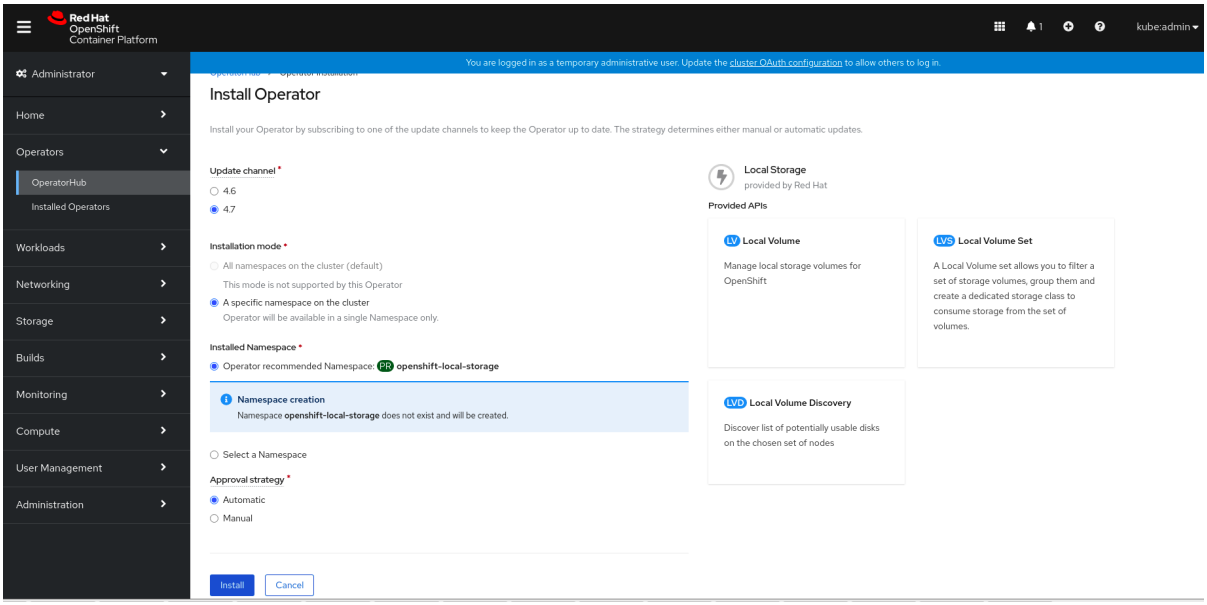
2. Search for **Local Storage Operator** from the list of operators and click on it.

3. Click **Install**.

Figure 1.4. Install Operator page

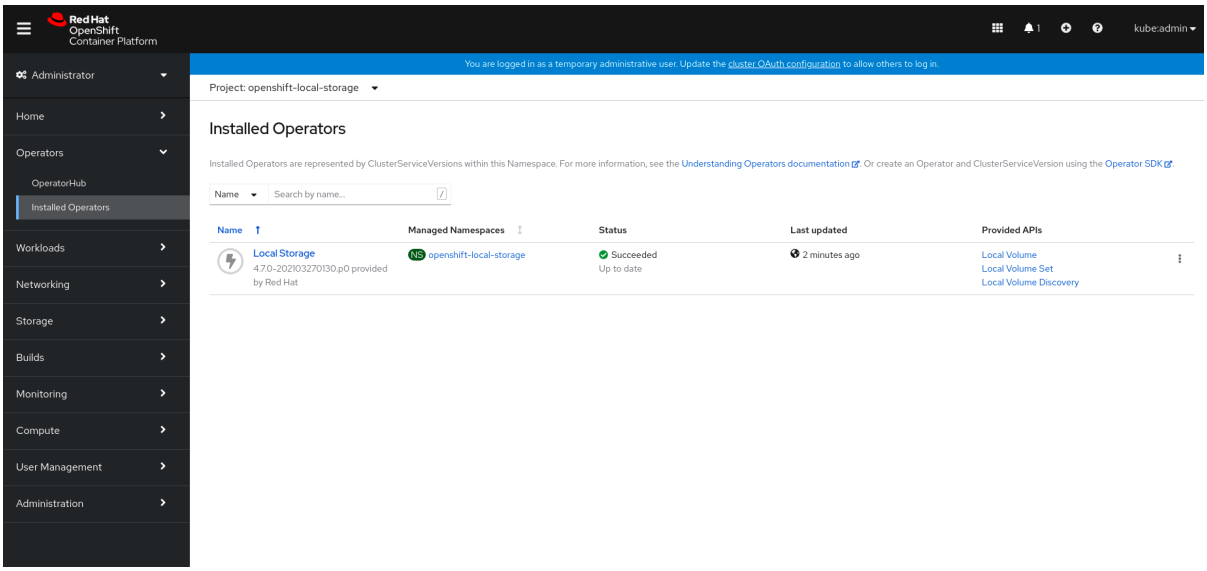


After Clicking on Install button, following page will appear.



4. Set the following options on the **Install Operator** page:
    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-local-storage**.
    d. Approval Strategy as **Automatic**
5. Click **Install**.

Figure 1.5. Installed Operators dashboard



Verification steps

- Verify that the Local Storage Operator shows the Status as **Succeeded**.

1.4. CREATING OPENSHIFT CONTAINER STORAGE CLUSTER ON IBM POWER SYSTEMS

Prerequisites

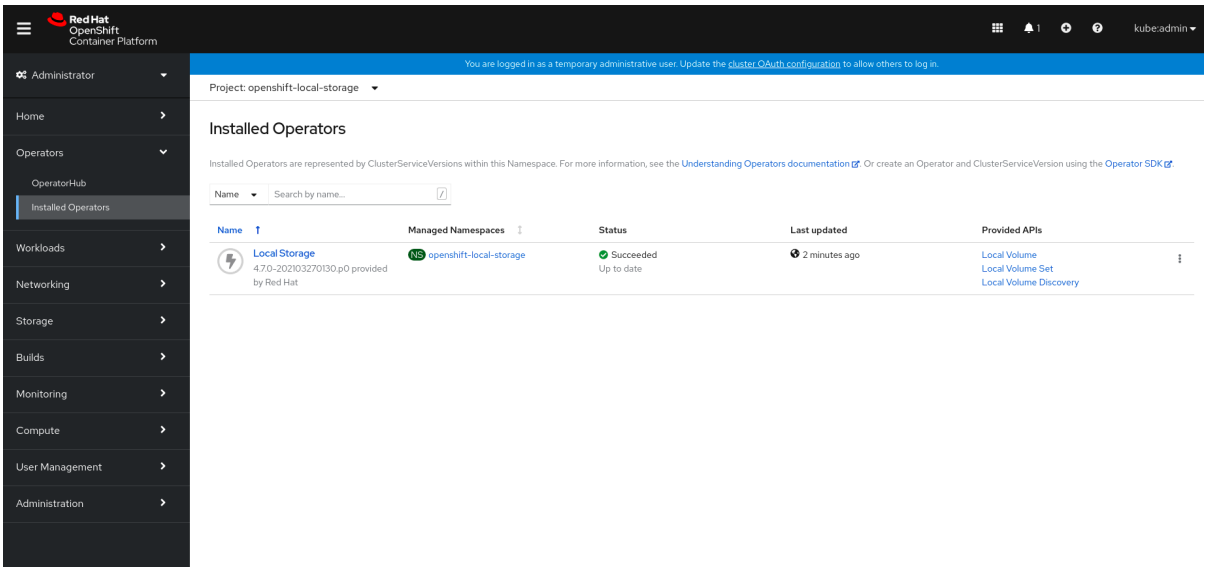
- Ensure that all the requirements in the Requirements for installing OpenShift Container Storage using local storage devices section are met.
- You must have a minimum of three worker nodes with the same storage type and size attached to each node (for example, 200 GB SSD) to use local storage devices on IBM Power Systems.
- Verify your OpenShift Container Platform worker nodes are labeled for OpenShift Container Storage:

```
oc get nodes -l cluster.ocs.openshift.io/openshift-storage -o jsonpath='{range .items[*]}
{.metadata.name}{"\n"}'
```

Procedure

1. Log into the OpenShift Web Console.
2. In **openshift-local-storage** namespace Click **Operators → Installed Operators** from the left pane of the OpenShift Web Console to view the installed operators.

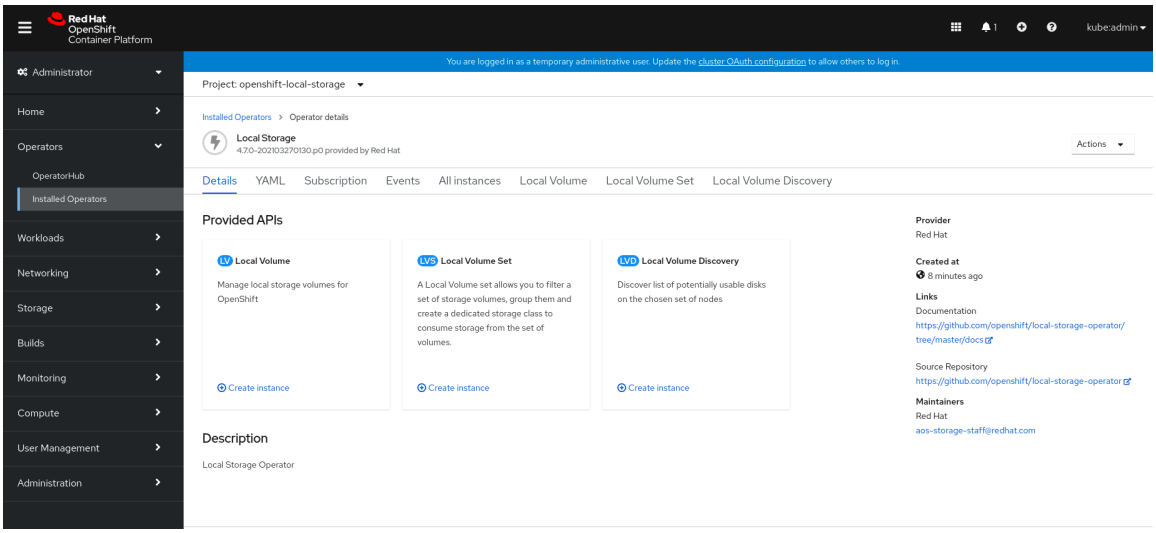
Figure 1.6. Local Storage Operator page



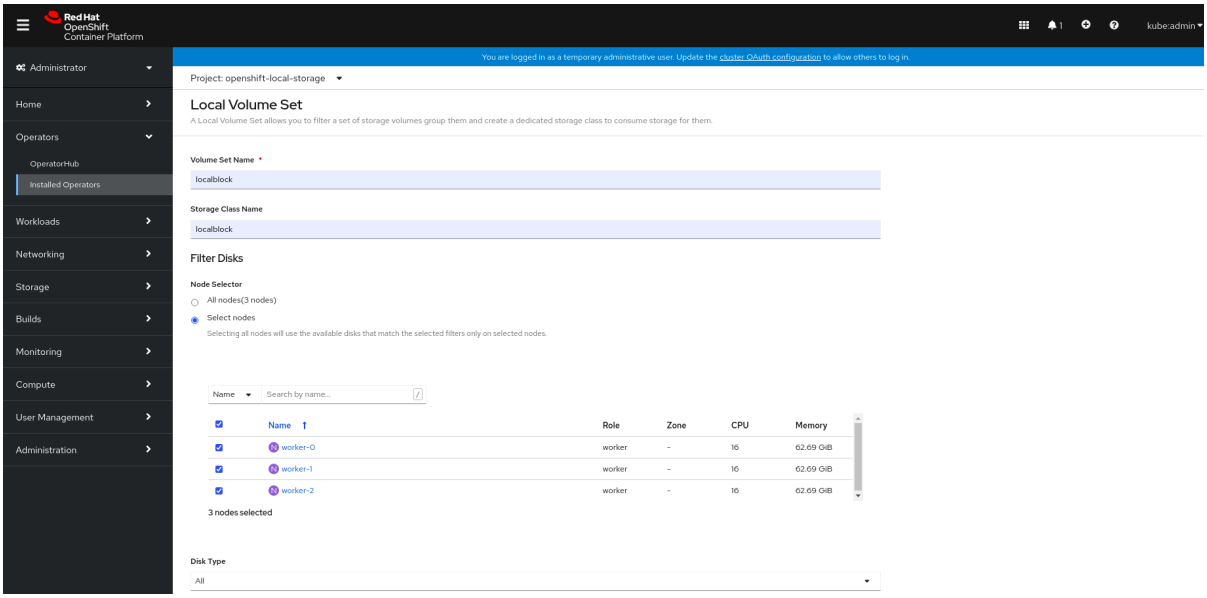
a. Click the **Local Storage** installed operator.

b. On the **Operator Details** page, click the **Local Volume Set** link.

Figure 1.7. Local Volume Set tab



3. Click **Create Local Volume Set**



a. Enter the Volume Set name. By default, Storage Class name appears for the Volume Set name.

b. To discover available disks, you can choose one of the following:

- **All nodes** to discover disks in all the nodes.
- **Select nodes** to choose a subset of nodes from a list of nodes
  To find specific worker nodes in the cluster, you can filter nodes on the basis of Name or Label. Name allows you to search by name of the node and Label allows you to search by selecting the predefined label.

c. Select the **Disk type**. The following options are available:

All	Selects all types of disks present on the nodes. By default, this option is selected.
SSD/NVME	Selects only SSD NVME type of disks.
HDD	Selects only HDD type of disks.

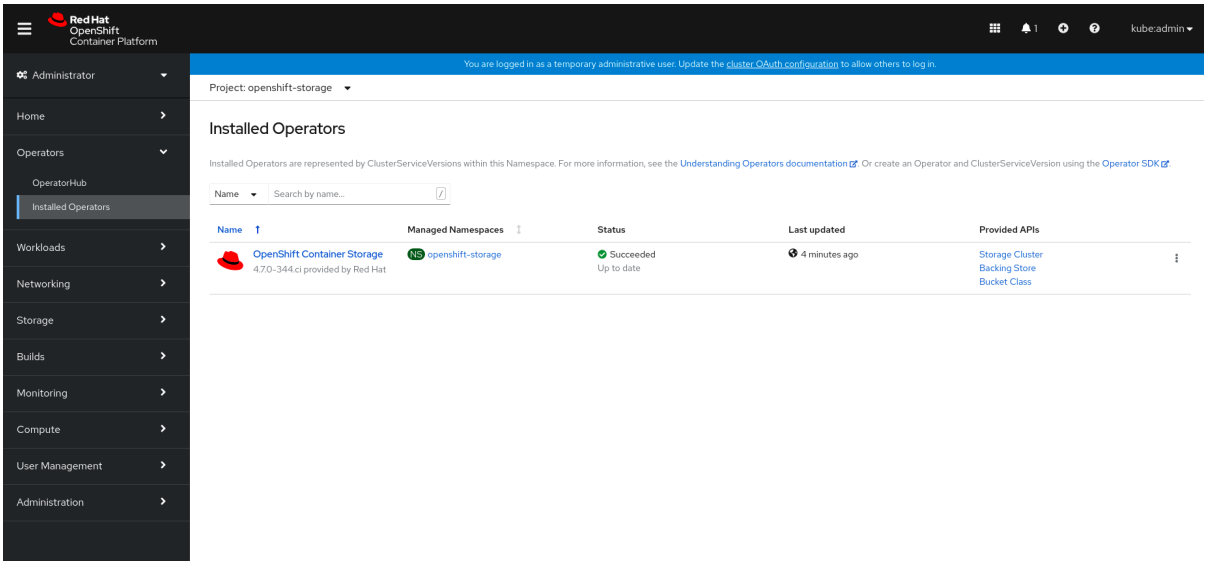
d. In the **Advanced** section, you can set the following:

Volume Mode	Block is selected by default.
Disk Size	Minimum and maximum avaialable size of the device that needs to be included. **NOTE** Choose the minimum disk size equivalent to the size of additional attached disk.
Max Disk Limit	This indicates the maximum number of PVs that can be created on a node. If this field is left empty, then PVs are created for all the available disks on the matching nodes.

e. Click **Create**.
The **Create** button is enabled only after you select a minimum of three nodes. Local Volume Set is created with one volume per worker node with the available disks.

4. In **openshift-storage** namespace Click **Operators → Installed Operators** from the left pane of the OpenShift Web Console to view the installed operators.

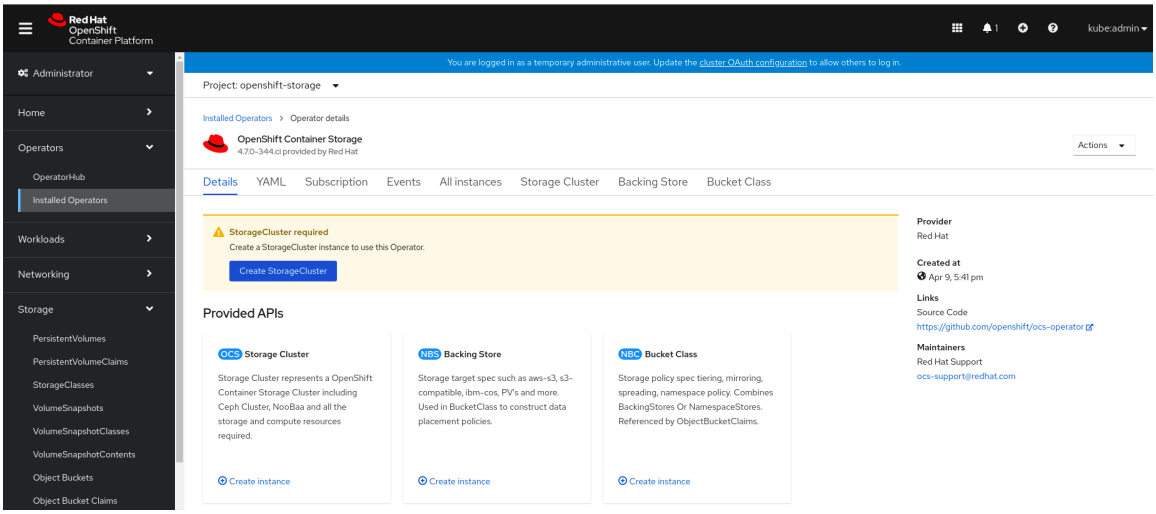
Figure 1.8. OpenShift Container Storage Operator page



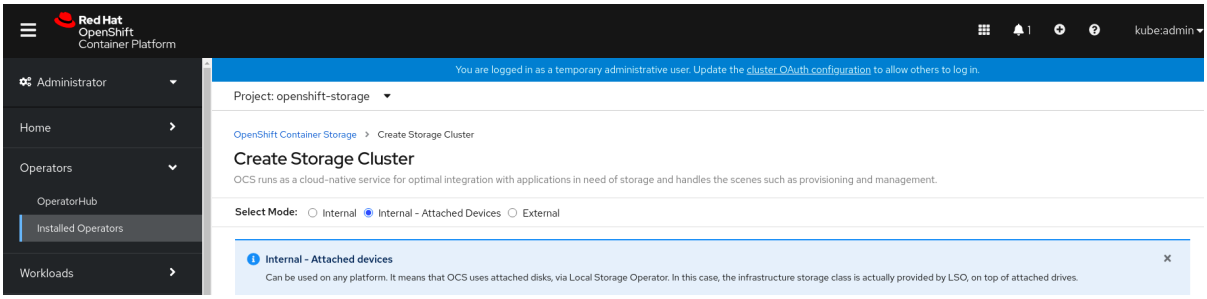
a. Click the **OpenShift Container Storage** installed operator.

b. On the **Operator Details** page, click the **Storage Cluster** link.

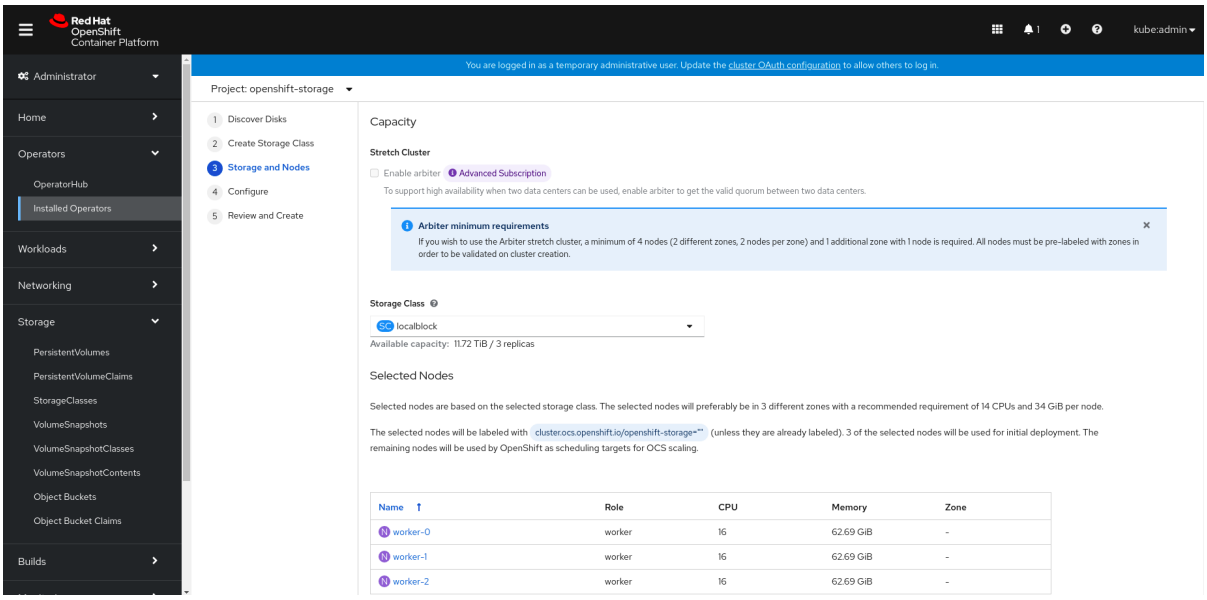
Figure 1.9. Storage Cluster tab



5. Click **Create Storage Cluster**.



6. Select **Internal-Attached devices** for the **Select Mode**.



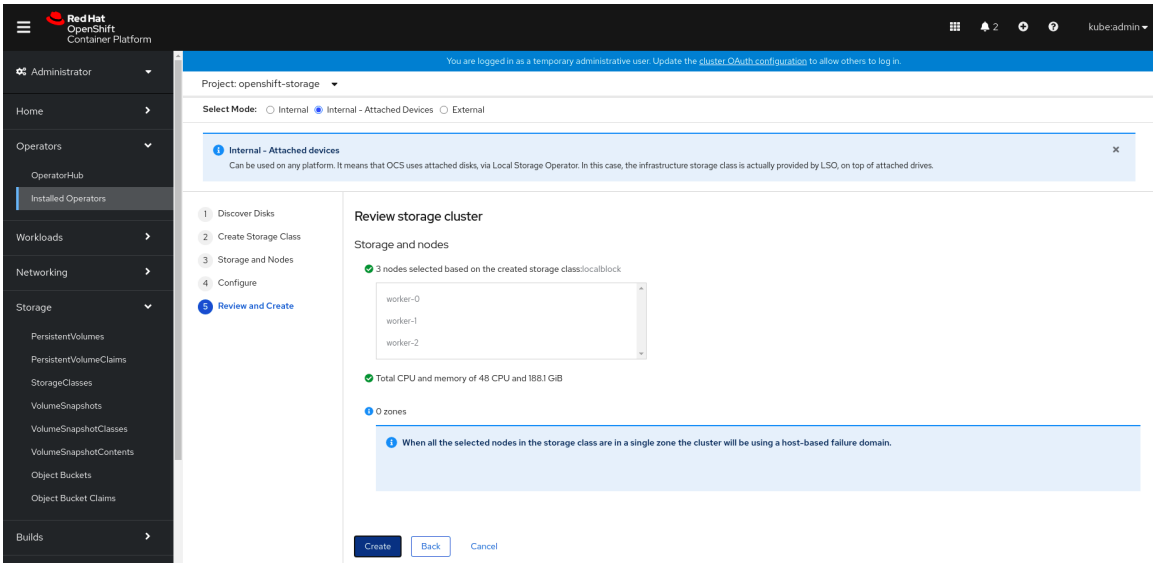
   a. Select the required storage class.

   b. The nodes corresponding to the storage class are displayed based on the storage class that you selected from the drop down.

   c. Click **Next**.

7. Again Click **Next** and then you will be redirected to **Review and Create** page.

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



8. Click **Create**.

Verification steps

See Verifying your OpenShift Container Storage installation .

Additional resources

To expand the capacity of the initial cluster, see the Scaling Storage guide.

CHAPTER 2. VERIFYING OPENSHIFT CONTAINER STORAGE DEPLOYMENT FOR INTERNAL MODE

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

2.1. 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 2.1, "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 2.1. Pods corresponding to OpenShift Container storage cluster

Component	Corresponding pods
OpenShift Container Storage Operator	• **ocs-operator-*** (1 pod on any worker node) • **ocs-metrics-exporter-***
Rook-ceph Operator	**rook-ceph-operator-*** (1 pod on any worker node)
Multicloud Object Gateway	• **noobaa-operator-*** (1 pod on any worker node) • **noobaa-core-*** (1 pod on any storage node) • **nooba-db-*** (1 pod on any storage node) • **noobaa-endpoint-*** (1 pod on any storage node)
MON	**rook-ceph-mon-*** (3 pods on each storage node)
MGR	**rook-ceph-mgr-*** (1 pod on any storage node)

Component	Corresponding pods
MDS	**rook-ceph-mds-ocs-storagecluster-cephfilesystem-*** (2 pods distributed across storage node)
RGW	**rook-ceph-rgw-ocs-storagecluster-cephobjectstore-*** (1 pod on any storage node)

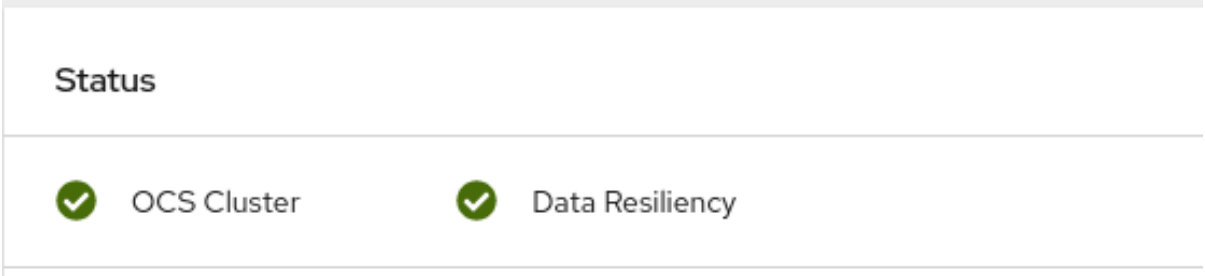
CSI	• **cephfs**
	  ◦ **csi-cephfsplugin-*** (1 pod on each worker node)
	  ◦ **csi-cephfsplugin-provisioner-*** (2 pods distributed across storage nodes)
	• **rbd**
	  ◦ **csi-rbdplugin-*** (1 pod on each worker node)
	  ◦ **csi-rbdplugin-provisioner-*** (2 pods distributed across storage nodes)

rook-ceph-crashcollector	**rook-ceph-crashcollector-*** (1 pod on each storage node)
OSD	• **rook-ceph-osd-*** (1 pod for each device) • **rook-ceph-osd-prepare-ocs-deviceset-*** (1 pod for each device)

2.2. VERIFYING THE OPENSHIFT CONTAINER STORAGE CLUSTER IS HEALTHY

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

Figure 2.1. Health status card in Persistent Storage Overview Dashboard



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

  Service Name

  OpenShift Container Storage

  Cluster Name

  ocs-storagecluster-cephcluster

  Provider

  None

  Mode

  Internal

  Version

  ocs-operator:v4.7.0

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

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

To verify the storage classes exists 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**
  - **ocs-storagecluster-ceph-rgw**

CHAPTER 3. UNINSTALLING OPENSHIFT CONTAINER STORAGE

3.1. 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 used with these annotations:

Table 3.1. uninstall.ocs.openshift.io uninstall annotations descriptions

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

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

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

```
$ oc -n openshift-storage 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) 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 that are using OpenShift Container Storage.
   In the default uninstall mode (graceful), the uninstaller waits till all the PVCs 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 in the system.

   a. Delete OpenShift Container Platform monitoring stack PVCs using OpenShift Container Storage.
      See Section 3.2, “Removing monitoring stack from OpenShift Container Storage”

   b. Delete OpenShift Container Platform Registry PVCs using OpenShift Container Storage.
      See Section 3.3, “Removing OpenShift Container Platform registry from OpenShift Container Storage”

   c. Delete OpenShift Container Platform logging PVCs using OpenShift Container Storage.
      See Section 3.4, “Removing the cluster logging operator from OpenShift Container Storage”

   d. Delete other PVCs provisioned using OpenShift Container Storage.

      - Given below is a sample script to identify the PVCs provisioned using OpenShift Container Storage. The script ignores the PVCs that are used internally by Openshift Container Storage.

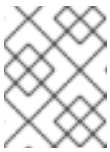
```
#!/bin/bash

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

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

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

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

NOTE

Omit **RGW_PROVISIONER** for cloud platforms.

- Delete the PVCs.

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



NOTE

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

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

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

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

```
$ oc get pods -n openshift-storage | grep -i cleanup
NAME                          READY   STATUS      RESTARTS   AGE
cluster-cleanup-job-<xx>       0/1     Completed   0          8m35s
```

```
cluster-cleanup-job-<yy>      0/1     Completed   0          8m35s
cluster-cleanup-job-<zz>      0/1     Completed   0          8m35s
```

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

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

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

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

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

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

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

   c. Remove the mapped device.

```
$ 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.

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

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

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

- Verify that the device name is removed.

```
$ dmsetup ls
```

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

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

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

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

NOTE

While uninstalling OpenShift Container Storage, if **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. Delete local storage operator configurations if you have deployed OpenShift Container Storage using local storage devices. See Removing local storage operator configurations .

9. Unlabel the storage nodes.

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

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

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

11. 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>
```

12. Delete the Multicloud Object Gateway storageclass.

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

13. Remove **CustomResourceDefinitions**.

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

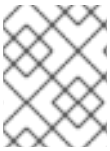
14. To ensure that OpenShift Container Storage is uninstalled completely, on the OpenShift Container Platform Web Console,

    a. Click **Home → Overview** to access the dashboard.

    b. Verify that the Persistent Storage tab no longer appear next to the **Cluster** tab.

3.1.1. Removing local storage operator configurations

Use the instructions in this section only if you have deployed OpenShift Container Storage using local storage devices.



NOTE

For OpenShift Container Storage deployments only using **localvolume** resources, go directly to step 8.

Procedure

1. Identify the **LocalVolumeSet** and the corresponding **StorageClassName** being used by OpenShift Container Storage.
2. Set the variable SC to the **StorageClass** providing the **LocalVolumeSet**.

```
$ export SC="<StorageClassName>"
```

3. Delete the **LocalVolumeSet**.

```
$ oc delete localvolumesets.local.storage.openshift.io <name-of-volumeset> -n openshift-local-storage
```

4. Delete the local storage PVs for the given **StorageClassName**.

```
$ oc get pv | grep $SC | awk '{print $1}'| xargs oc delete pv
```

5. Delete the **StorageClassName**.

```
$ oc delete sc $SC
```

6. Delete the symlinks created by the **LocalVolumeSet**.

```
[[ ! -z $SC ]] && 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 rm -rfv /mnt/local-storage/${SC}/; done
```

7. Delete **LocalVolumeDiscovery**.

```
$ oc delete localvolumediscovery.local.storage.openshift.io/auto-discover-devices -n openshift-local-storage
```

8. Removing **LocalVolume** resources (if any).
   Use the following steps to remove the **LocalVolume** resources that were used to provision PVs in the current or previous OpenShift Container Storage version. Also, ensure that these resources are not being used by other tenants on the cluster.

   For each of the local volumes, do the following:

   a. Identify the **LocalVolume** and the corresponding **StorageClassName** being used by OpenShift Container Storage.

   b. Set the variable LV to the name of the LocalVolume and variable SC to the name of the StorageClass
   For example:

```
$ LV=local-block
$ SC=localblock
```

c. Delete the local volume resource.

```
$ oc delete localvolume -n local-storage --wait=true $LV
```

d. Delete the remaining PVs and StorageClasses if they exist.

```
$ oc delete pv -l storage.openshift.com/local-volume-owner-name=${LV} --wait --
timeout=5m
$ oc delete storageclass $SC --wait --timeout=5m
```

e. Clean up the artifacts from the storage nodes for that resource.

```
$ [[ ! -z $SC ]] && 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 rm -rfv
/mnt/local-storage/${SC}/; done
```

Example output:

```
Starting pod/node-xxx-debug ...
To use host binaries, run `chroot /host`
removed '/mnt/local-storage/localblock/nvme2n1'
removed directory '/mnt/local-storage/localblock'

Removing debug pod ...
Starting pod/node-yyy-debug ...
To use host binaries, run `chroot /host`
removed '/mnt/local-storage/localblock/nvme2n1'
removed directory '/mnt/local-storage/localblock'

Removing debug pod ...
Starting pod/node-zzz-debug ...
To use host binaries, run `chroot /host`
removed '/mnt/local-storage/localblock/nvme2n1'
removed directory '/mnt/local-storage/localblock'

Removing debug pod ...
```

3.2. REMOVING MONITORING STACK FROM OPENSHIFT CONTAINER STORAGE

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

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

Prerequisites

- PVCs are configured to use OpenShift Container Platform monitoring stack.
  For information, see configuring monitoring stack.

Procedure

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

```
$ oc get pod,pvc -n openshift-monitoring
NAME                          READY   STATUS    RESTARTS   AGE
pod/alertmanager-main-0       3/3     Running   0          8d
pod/alertmanager-main-1       3/3     Running   0          8d
pod/alertmanager-main-2       3/3     Running   0          8d
pod/cluster-monitoring-
operator-84457656d-pkrxm      1/1     Running   0          8d
pod/grafana-79ccf6689f-2ll28  2/2     Running   0          8d
pod/kube-state-metrics-
7d86fb966-rvd9w               3/3     Running   0          8d
pod/node-exporter-25894       2/2     Running   0          8d
pod/node-exporter-4dsd7       2/2     Running   0          8d
pod/node-exporter-6p4zc       2/2     Running   0          8d
pod/node-exporter-jbjvg       2/2     Running   0          8d
pod/node-exporter-jj4t5       2/2     Running   0          6d18h
pod/node-exporter-k856s       2/2     Running   0          6d18h
pod/node-exporter-rf8gn       2/2     Running   0          8d
pod/node-exporter-rmb5m       2/2     Running   0          6d18h
pod/node-exporter-zj7kx       2/2     Running   0          8d
pod/openshift-state-metrics-
59dbd4f654-4clng              3/3     Running   0          8d
pod/prometheus-adapter-
5df5865596-k8dzn              1/1     Running   0          7d23h
pod/prometheus-adapter-
5df5865596-n2gj9              1/1     Running   0          7d23h
pod/prometheus-k8s-0          6/6     Running   1          8d
pod/prometheus-k8s-1          6/6     Running   1          8d
pod/prometheus-operator-
55cfb858c9-c4zd9              1/1     Running   0          6d21h
pod/telemeter-client-
78fc8fc97d-2rgfp              3/3     Running   0          8d

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

2. Edit the monitoring **configmap**.

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

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

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

After editing

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

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

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

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

3.3. REMOVING OPENSHIFT CONTAINER PLATFORM REGISTRY FROM OPENSHIFT CONTAINER STORAGE

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

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

Prerequisites

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

Procedure

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

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

Before editing

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

After editing

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

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

2. Delete the PVC.

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

3.4. REMOVING THE CLUSTER LOGGING OPERATOR FROM OPENSHIFT CONTAINER STORAGE

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

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

Prerequisites

- The cluster logging instance should have been configured to use OpenShift Container Storage PVCs.

Procedure

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

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

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

2. Delete PVCs.

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