



# Red Hat OpenShift Data Foundation 4.12

## 以外部模式部署 OpenShift Data Foundation

部署 OpenShift Data Foundation 以使用外部 Red Hat Ceph Storage 集群和 IBM FlashSystem 的说明。



## Red Hat OpenShift Data Foundation 4.12 以外部模式部署 OpenShift Data Foundation

---

部署 OpenShift Data Foundation 以使用外部 Red Hat Ceph Storage 集群和 IBM FlashSystem 的说明。

## 法律通告

Copyright © 2023 Red Hat, Inc.

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

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

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

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

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

Linux<sup>®</sup> is the registered trademark of Linus Torvalds in the United States and other countries.

Java<sup>®</sup> is a registered trademark of Oracle and/or its affiliates.

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

MySQL<sup>®</sup> is a registered trademark of MySQL AB in the United States, the European Union and other countries.

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

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

All other trademarks are the property of their respective owners.

## 摘要

有关安装 Red Hat OpenShift Data Foundation 4.12 的说明，以使用外部 Red Hat Ceph Storage 集群或 IBM FlashSystem。

---

# 目录

使开源包含更多 .....	3
对红帽文档提供反馈 .....	4
<b>第 1 章 以外部模式部署概述 .....</b>	<b>5</b>
<b>第 2 章 使用 RED HAT CEPH STORAGE 部署 OPENSIFT DATA FOUNDATION .....</b>	<b>6</b>
2.1. 安装 RED HAT OPENSIFT DATA FOUNDATION OPERATOR .....	6
2.2. 为外部 CEPH 存储系统创建 OPENSIFT DATA FOUNDATION 集群 .....	7
2.3. 验证您的 OPENSIFT DATA FOUNDATION 安装用于外部 CEPH 存储系统 .....	12
<b>第 3 章 使用 IBM FLASHSYSTEM 部署 OPENSIFT DATA FOUNDATION .....</b>	<b>15</b>
3.1. 安装 RED HAT OPENSIFT DATA FOUNDATION OPERATOR .....	15
3.2. 为外部 IBM FLASHSYSTEM 存储创建 OPENSIFT DATA FOUNDATION 集群 .....	16
<b>第 4 章 从外部存储系统卸载 OPENSIFT DATA FOUNDATION .....</b>	<b>22</b>
4.1. 从 OPENSIFT DATA FOUNDATION 中删除监控堆栈 .....	25
4.2. 从 OPENSIFT DATA FOUNDATION 中删除 OPENSIFT CONTAINER PLATFORM REGISTRY .....	29
4.3. 从 OPENSIFT DATA FOUNDATION 中删除集群日志记录操作器 .....	30
4.4. 删除外部 IBM FLASHSYSTEM SECRET .....	30



---

## 使开源包含更多

红帽致力于替换我们的代码、文档和 Web 属性中存在问题的语言。我们从这四个术语开始：master、slave、黑名单和白名单。由于此项工作十分艰巨，这些更改将在即将推出的几个发行版本中逐步实施。详情请查看 [CTO Chris Wright 的信息](#)。

## 对红帽文档提供反馈

我们感谢您对文档提供反馈信息。请告诉我们如何让它更好。

要提供反馈，请创建一个 Bugzilla ticket：

1. 进入 [Bugzilla](#) 网站。
2. 在 **Component** 部分中，选择 **文档**。
3. 在 **Description** 中输入您要提供的信息。包括文档相关部分的链接。
4. 点 **Submit Bug**。



# 第 1 章 以外部模式部署概述

Red Hat OpenShift Data Foundation 可以从外部 Red Hat Ceph Storage 集群提供服务，或使用 IBM FlashSystems 通过在以下平台上运行的 OpenShift Container Platform 集群使用：

- VMware vSphere
- 裸机
- Red Hat OpenStack Platform (技术预览)
- IBM Power
- IBM Z 基础架构

如需更多信息，请参阅[规划部署](#)。

有关如何安装 RHCS 集群的说明，请参阅[安装指南](#)。

按照以下步骤，以外部模式部署 OpenShift Data Foundation：

1. 部署以下内容：
  - [使用 Red Hat Ceph Storage 部署 OpenShift Data Foundation](#)。
  - [使用 IBM FlashSystem 部署 OpenShift Data Foundation](#)。

## 实现灾难恢复的要求 [技术预览]

Red Hat OpenShift Data Foundation 支持的灾难恢复功能需要满足以下所有先决条件，才能成功实施灾难恢复解决方案：

- 有效的 Red Hat OpenShift Data Foundation 高级订阅
- 有效的 Red Hat Advanced Cluster Management for Kubernetes 订阅

如需更多信息，请参阅 [OpenShift Data Foundation 订阅中的知识库文章](#)。

如需更多与灾难恢复相关的信息，请参阅 [Configuring OpenShift Data Foundation Disaster Recovery for OpenShift Workloads](#) 指南，以及 Red Hat Advanced Cluster Management for Kubernetes 文档中的 [安装指南](#) 中的 *Requirements and recommendations* 部分。

## 第 2 章 使用 RED HAT CEPH STORAGE 部署 OPENSIFT DATA FOUNDATION

Red Hat OpenShift Data Foundation 可以从外部 Red Hat Ceph Storage 集群提供服务，以便通过 OpenShift Container Platform 集群使用。您需要安装 OpenShift Data Foundation 操作器，然后为外部 Ceph 存储系统创建 OpenShift Data Foundation 集群。

### 2.1. 安装 RED HAT OPENSIFT DATA FOUNDATION OPERATOR

您可以使用 Red Hat OpenShift Container Platform Operator Hub 安装 Red Hat OpenShift Data Foundation Operator。

#### 先决条件

- 使用具有 **cluster-admin** 和 operator 安装权限的账户访问 OpenShift Container Platform 集群。
- 有关其他资源要求，请参阅[规划您的部署指南](#)。



#### 重要

- 当您需要覆盖 OpenShift Data Foundation 的集群范围默认节点选择器时，您可以使用以下命令为 **openshift-storage** 命名空间指定空白节点选择器（在这种情况下创建 **openshift-storage** 命名空间）：

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

#### 流程

1. 登录 OpenShift Web 控制台。
2. 点 **Operators** → **OperatorHub**。
3. 在 **Filter by keyword** 框中滚动或键入 **OpenShift Data Foundation**，以查找 **OpenShift Data Foundation Operator**。
4. 点 **Install**。
5. 在 **Install Operator** 页面中设置以下选项：
  - a. 更新频道为 **stable-4.12**。
  - b. 安装模式是 **A specific namespace on the cluster**。
  - c. Installed Namespace 为 **Operator recommended namespace openshift-storage**。如果命名空间 **openshift-storage** 不存在，它会在 Operator 安装过程中创建。
  - d. 将 Approval Strategy 选为 **Automatic** 或 **Manual**。  
如果选择 **Automatic** 更新，Operator Lifecycle Manager(OLM)将自动升级 Operator 的运行实例，而无需任何干预。  
  
如果选择 **手动** 更新，则 OLM 会创建一个更新请求。作为集群管理员，您必须手动批准该更新请求，才能将 Operator 更新至更新的版本。
  - e. 确保为 **Console 插件** 选择了 **Enable** 选项。

f. 点 **Install**。

## 验证步骤

- 成功安装 Operator 后，用户界面中会显示一个带有 **Web console update is available** 信息的弹出窗口。点这个弹出窗口中的 **Refresh web console** 来反映控制台的更改。
- 在 Web 控制台中：
  - 进入到 **Installed Operators**，再验证 **OpenShift Data Foundation Operator** 是否显示绿色勾号，指示安装成功。
  - 进入到 **Storage**，再验证 **Data Foundation** 仪表板是否可用。

## 2.2. 为外部 CEPH 存储系统创建 OPENSIFT DATA FOUNDATION 集群

在 VMware vSphere 或用户置备的裸机基础架构上部署的 OpenShift Container Platform 上安装 OpenShift Data Foundation operator 后，需要创建新的 OpenShift Data Foundation 集群。

### 先决条件

- 有效的 Red Hat OpenShift Data Foundation 高级订阅。要了解 OpenShift Data Foundation 订阅如何工作，请参阅[与 OpenShift Data Foundation 订阅相关的知识库文章](#)。
- 在部署 OpenShift Data Foundation 4.12 之前，请确保 OpenShift Container Platform 版本为 4.12 或更高版本。
- 必须安装 OpenShift Data Foundation 操作器。如需更多信息，请参阅[使用 Operator Hub 安装 OpenShift Data Foundation Operator](#)。
- 要以外部模式检查 Red Hat Ceph Storage (RHCS) 与 Red Hat OpenShift Data Foundation 的支持性和互操作性，请转至 lab [Red Hat OpenShift Data Foundation 支持性和互操作性检查器](#)。
  - 选择 **Service Type** 为 **ODF as Self-Managed Service**。
  - 从下拉菜单中选择适当的 **Version**。
  - 在 **Versions** 选项卡中，点 **支持的 RHCS 兼容性** 选项卡。
- 如果您已将红帽 Ceph 存储集群从低于 4.11 的版本更新为最新版本，且不是全新部署的集群，您必须在红帽 Ceph 存储集群中手动设置 CephFS 池的应用类型，以外部模式启用 CephFS PVC 创建。如需了解更多详细信息，请参阅[在外部模式中对 CephFS PVC 创建进行故障排除](#)。
- Red Hat Ceph Storage 必须安装并配置 Ceph 控制面板。如需更多信息，请参阅[Ceph 控制面板安装和访问](#)。
- 建议外部 Red Hat Ceph Storage 集群启用 PG Autoscaler。如需更多信息，请参阅 Red Hat Ceph Storage 文档中的[放置组自动扩展](#)部分。
- 外部 Ceph 集群应当预配置有一个现有的 RBD 池，供使用。如果不存在，请在进行 OpenShift Data Foundation 部署前，联系您的 Red Hat Ceph Storage 管理员创建一个。红帽建议为每个 OpenShift Data Foundation 集群使用单独的池。
- 可选：如果在默认的 zonegroup 之外创建了一个 zonegroup，则需要将主机名 **rook-ceph-rgw-ocs-external-storage-cephobjectstore.openshift-storage.svc** 添加到 zonegroup，因为

OpenShift Data Foundation 会将 S3 请求发送到此主机名的 RADOS 对象网关(RGWs)。有关更多信息，请参阅红帽知识库解决方案 [Ceph - 如何在 RGW zonegroup 中添加主机名？](#)

## 流程

1. 点 **Operators** → **Installed Operators** 查看所有已安装的 Operator。确保所选项目为 **openshift-storage**。
2. 单击 **OpenShift Data Foundation**，然后单击 **Create StorageSystem**。
3. 在 Backing storage 页面中，选择以下选项：
  - a. 为 **Deployment 类型** 选项选择 **Full Deployment**。
  - b. 从可用选项中选择 **Connect a external storage platform**。
  - c. 为 **Storage platform** 选择 **Red Hat Ceph Storage**。
  - d. 点 **Next**。
4. 在连接详情页面中，提供必要的信息：
  - a. 在 Connect to external cluster 部分中，点 **Download Script** 链接，以下载用于提取 Ceph 集群详细信息的 python 脚本。
  - b. 要提取 Red Hat Ceph Storage (RHCS) 集群详情，请联系 RHCS 管理员，以在带有 **admin** 密钥的 Red Hat Ceph Storage 节点上运行下载的 python 脚本。
    - i. 在 RHCS 节点上运行以下命令查看可用参数列表：

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



### 重要

如果在 Red Hat Enterprise Linux 7.x (RHEL 7.x) 集群中部署了 Red Hat Ceph Storage 4.x 集群，则使用 **python** 而不是 **python3**。

您也可以从 MON 容器（容器化部署）或 MON 节点（RPM 部署）运行脚本。



### 注意

使用 **yum install cephadm** 命令，然后运行 **cephadm** 命令，使用容器部署 RHCS 集群。您必须使用 **cephadm** 命令拉取 RHCS 容器镜像，而不是使用 **yum** 将 Ceph 软件包安装到节点上。如需更多信息，请参阅 [RHCS 产品文档](#)。

- ii. 要从 RHCS 集群检索外部集群详情，请运行以下命令：

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

例如：

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

在本例中，

rbd-data-pool-name	用于在 OpenShift Data Foundation 中提供块存储的必填参数。
rgw-endpoint	<p>(可选) 只有在 OpenShift Data Foundation 的 Ceph Rados 网关置备了对象存储时，才需要此参数。使用以下格式提供端点：<b>&lt;ip_address&gt;:&lt;port&gt;</b></p> <p> <b>注意</b></p> <p>完全限定域名 (FQDN) 也支持 <b>&lt;FQDN&gt;:&lt;PORT&gt;</b> 格式。</p>
monitoring-endpoint	(可选) 此参数接受从 OpenShift Container Platform 集群访问的活跃和待机 <b>mgrs</b> 的 IP 地址的逗号分隔列表。如果没有提供，则会自动填充该值。
monitoring-endpoint-port	(可选) 它是与 <b>--monitoring-endpoint</b> 指定的 <b>ceph-mgr</b> Prometheus exporter 关联的端口。如果没有提供，则会自动填充该值。
run-as-user	<p>(可选) 此参数用于为脚本创建的 Ceph 用户提供名称。如果没有指定此参数，则会创建一个默认的用户名 <b>client.healthchecker</b>。新用户的权限被设置为：</p> <ul style="list-style-type: none"> <li>● caps: [mgr] allow command config</li> <li>● caps: [mon] allow r, allow command quorum_status, allow command version</li> <li>● caps: [osd] allow rwx pool=<b>RGW_POOL_PREFIX.rgw.meta</b>, allow r pool=<b>rgw.root</b>, allow rw pool=<b>RGW_POOL_PREFIX.rgw.control</b>, allow rx pool=<b>RGW_POOL_PREFIX.rgw.log</b>, allow x pool=<b>RGW_POOL_PREFIX.rgw.buckets.index</b></li> </ul>

附加标记：

rgw-pool-prefix	(可选) RGW 池的前缀。如果未指定，默认的前缀为 <b>default</b> 。
rgw-tls-cert-path	(可选) RADOS 网关端点 TLS 证书的文件路径。
rgw-skip-tls	(可选) 此参数在提供自签名证书时会忽略 TLS 证书验证(不推荐)。

ceph-conf	(可选) Ceph 配置文件的名称。
cluster-name	(可选) Ceph 集群名称。
output	(可选) 需要存储输出的文件。
cephfs-metadata-pool-name	(可选) CephFS 元数据池的名称。
cephfs-data-pool-name	(可选) CephFS 数据池的名称。
cephfs-filesystem-name	(可选) CephFS 文件的名称。
rbd-metadata-ec-pool-name	(可选) 擦除代码 RBD 元数据池的名称。
dry-run	(可选) 此参数会打印已执行命令的输出而不实际运行它们。
restricted-auth-permission	<p>(可选) 此参数将 <b>cephCSIKeyrings</b> auth 权限限制为特定的池和集群。需要与此一起设置的强制标记是 <b>rbd-data-pool-name</b> 和 <b>cluster-name</b>。如果 CephFS 用户有限制，您可以通过 <b>cephfs-filesystem-name</b> 标志，以便权限仅限于特定的 CephFS 文件系统。</p> <p> <b>注意</b></p> <p>此参数必须只应用于新部署。要限制每个池和每个集群的 <b>csi-users</b>，您需要为这些 <b>csi-users</b> 创建新的 <b>csi-users</b> 和新的 <b>secret</b>。</p>

#### 带有受限身份验证权限的示例：

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

#### 使用 python 脚本生成的 JSON 输出示例：

```
[{"name": "rook-ceph-mon-endpoints", "kind": "ConfigMap", "data": {"data":
"xxx.xxx.xxx.xxx:xxxx", "maxMonId": "0", "mapping": "{}"}}, {"name": "rook-ceph-mon",
"kind": "Secret", "data": {"admin-secret": "admin-secret", "fsid": "<fs-id>", "mon-
```

```
secret": "mon-secret"}}, {"name": "rook-ceph-operator-creds", "kind": "Secret", "data":
{"userID": "<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>"}]}
```

- iii. 将 JSON 输出保存到带有 **.json** 扩展名的文件



### 注意

要使 OpenShift Data Foundation 无缝工作，请确保使用 JSON 文件上传的参数（RGW 端点、CephFS 详细信息和 RBD 池等）在创建存储集群后在 RHCS 外部集群上保持不变。

- iv. 在多租户部署中的 RHCS 集群已连接到带有较低版本的 OpenShift Data Foundation 部署时，运行命令。

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

- c. 单击 **Browse** 以选择并上传 JSON 文件。  
JSON 文件的内容填充并在文本框中显示。
- d. 点 **Next**  
只有上传 **.json** 文件后，**Next** 按钮才会启用。
5. 在 Review and create 页面中，检查所有详情是否正确：
- 若要修改任何配置设置，请单击 **Back** 以返回到上一配置页面。
6. 单击 **Create StorageSystem**。

## 验证步骤

验证已安装存储集群的最终状态：

- 在 OpenShift Web 控制台中，导航到 **Installed Operators** → **OpenShift Data Foundation** → **Storage System** → **ocs-external-storagecluster-storagesystem** → **Resources**。
- 验证 **StorageCluster** 的 **Status** 是否为 **Ready** 且具有绿色勾号。
- 要验证 OpenShift Data Foundation、Pod 和 StorageClass 是否已成功安装，请参阅[验证外部 Ceph 存储系统的 OpenShift Data Foundation 安装模式](#)。

## 2.3. 验证您的 OPENSIFT DATA FOUNDATION 安装用于外部 CEPH 存储系统

使用本节验证 OpenShift Data Foundation 是否已正确部署。

### 2.3.1. 验证 pod 的状态

1. 从 OpenShift Web 控制台左侧窗格中，点击 **Workloads → Pods**。
2. 从 **Project** 下拉列表中选择 **openshift-storage**。



#### 注意

如果禁用 **Show default projects** 选项，请使用切换按钮列出所有默认项目。

有关每个组件预期的 pod 数量及其变化取决于节点数量的更多信息，请参阅 [表 2.1 “对应 OpenShift Data Foundation 组件的 Pod”](#)

3. 验证以下 pod 是否处于运行状态：

**表 2.1. 对应 OpenShift Data Foundation 组件的 Pod**

组件	对应的 pod
OpenShift Data Foundation Operator	<ul style="list-style-type: none"> <li>● <b>OCS-operator-*</b> (在任何 worker 节点上有 1 个 pod)</li> <li>● <b>ocS-metrics-exporter-*</b> (任何 worker 节点上 1 个 pod)</li> <li>● <b>odF-operator-controller-manager-*</b> (任何 worker 节点上 1 个 pod)</li> <li>● <b>odf-console-*</b> (任何 worker 节点上 1 个 pod)</li> <li>● <b>csi-addons-controller-manager-*</b> (任何 worker 节点上 1 个 pod)</li> </ul>
Rook-ceph Operator	<p><b>rook-ceph-operator-*</b></p> <p>(任何 worker 节点上有 1 个 pod)</p>
多云对象网关	<ul style="list-style-type: none"> <li>● <b>noobaa-operator-*</b> (任何 worker 节点上 1 个 pod)</li> <li>● <b>noobaa-core-*</b> (任何 worker 节点上有 1 个 pod)</li> <li>● <b>noobaa-db-pg-*</b> (任何 worker 节点上有 1 个 pod)</li> <li>● <b>noobaa-endpoint-*</b> (任何 worker 节点上有 1 个 pod)</li> </ul>



组件	对应的 pod
CSI	<ul style="list-style-type: none"> <li>● <b>cephfs</b> <ul style="list-style-type: none"> <li>○ <b>csi-cephfsplugin-*</b> (每个 worker 节点上 1 个 pod)</li> <li>○ <b>csi-cephfsplugin-provisioner-*</b> (2 个 pod 在不同的 worker 节点上分布)</li> </ul> </li> </ul> <div style="display: flex; align-items: center; margin: 10px 0;">  <div> <p><b>注意</b></p> <p>如果没有在外部集群中部署 MDS，则不会创建 csi-cephfs 插件 pod。</p> </div> </div> <ul style="list-style-type: none"> <li>● <b>rbd</b> <ul style="list-style-type: none"> <li>○ <b>csi-rbdplugin-*</b> (每个 worker 节点上 1 个 pod)</li> <li>○ <b>csi-rbdplugin-provisioner-*</b> (2 个 pod 在不同的 worker 节点上分布)</li> </ul> </li> </ul>

### 2.3.2. 验证 OpenShift Data Foundation 集群是否健康

1. 在 OpenShift Web 控制台中，点 **Storage → Data Foundation**。
2. 在 **Overview** 选项卡的 **Status** 卡中，点 **Storage System**，然后点弹出框中的存储系统链接。
3. 在 **Block and File** 选项卡的 **Status** 卡中，验证 *Storage Cluster* 是否具有绿色勾号。
4. 在 **Details** 卡中，验证是否显示集群信息。

有关使用 Block and File 仪表板的 OpenShift Data Foundation 健康状况的更多信息，请参阅[监控 OpenShift Data Foundation](#)。

### 2.3.3. 验证 Multicloud 对象网关是否健康

1. 在 OpenShift Web 控制台中，点 **Storage → Data Foundation**。
2. 在 **Overview** 选项卡的 **Status** 卡中，点 **Storage System**，然后点弹出框中的存储系统链接。
  - a. 在 **Object** 选项卡的 **Status** 卡中，验证 *Object Service* 和 *数据弹性* 都具有绿色勾号。
  - b. 在 **Details** 卡中，验证 Multicloud Object Gateway(MCG)信息是否已显示。



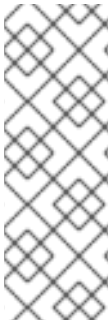
#### 注意

RADOS 对象网关仅列出在以外部模式部署 OpenShift Data Foundation 时包含 RADOS 对象网关端点详细信息的情况。

有关使用对象仪表板的 OpenShift Data Foundation 健康状况的更多信息，请参阅[监控 OpenShift Data Foundation](#)。

### 2.3.4. 验证存储类是否已创建并列出

1. 从 OpenShift Web 控制台左侧窗格中，点击 **Storage → Storage Classes**。
2. 验证是否在创建 OpenShift Data Foundation 集群时创建了以下存储类：
  - **ocs-external-storagecluster-ceph-rbd**
  - **ocs-external-storagecluster-ceph-rgw**
  - **ocs-external-storagecluster-cephfs**
  - **openshift-storage.noobaa.io**



### 注意

- 如果没有在外部集群中部署 MDS，则不会创建 **ocs-external-storagecluster-cephfs** 存储类。
- 如果没有在外部集群中部署 RGW，则不会创建 **ocs-external-storagecluster-ceph-rgw** 存储类。

如需有关 MDS 和 RGW 的更多信息，请参阅 [Red Hat Ceph Storage 文档](#)

### 2.3.5. 验证 Ceph 集群是否已连接

运行以下命令，以验证 OpenShift Data Foundation 集群是否已连接到外部 Red Hat Ceph Storage 集群。

```
$ 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. 验证存储集群是否已就绪

运行以下命令，以验证存储集群是否已就绪，并且 **External** 选项设为 **true**。

```
$ oc get storagecluster -n openshift-storage
NAME                                AGE  PHASE  EXTERNAL  CREATED AT          VERSION
ocs-external-storagecluster  30m  Ready  true      2021-11-17T09:09:52Z  4.12.0
```

## 第 3 章 使用 IBM FLASHSYSTEM 部署 OPENSIFT DATA FOUNDATION

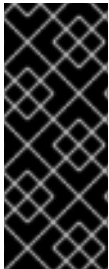
OpenShift Data Foundation 可以使用可通过 OpenShift Container Platform 集群使用的 IBM FlashSystem 存储。您需要安装 OpenShift Data Foundation 操作器，然后为 IBM FlashSystem 存储创建 OpenShift Data Foundation 集群。

### 3.1. 安装 RED HAT OPENSIFT DATA FOUNDATION OPERATOR

您可以使用 Red Hat OpenShift Container Platform Operator Hub 安装 Red Hat OpenShift Data Foundation Operator。

#### 先决条件

- 使用具有 **cluster-admin** 和 operator 安装权限的账户访问 OpenShift Container Platform 集群。
- 有关其他资源要求，请参阅[规划您的部署指南](#)。



#### 重要

- 当您需要覆盖 OpenShift Data Foundation 的集群范围默认节点选择器时，您可以使用以下命令为 **openshift-storage** 命名空间指定空白节点选择器（在这种情况下创建 **openshift-storage** 命名空间）：

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

#### 流程

1. 登录 OpenShift Web 控制台。
2. 点 **Operators** → **OperatorHub**。
3. 在 **Filter by keyword** 框中滚动或键入 **OpenShift Data Foundation**，以查找 **OpenShift Data Foundation Operator**。
4. 点 **Install**。
5. 在 **Install Operator** 页面中设置以下选项：
  - a. 更新频道为 **stable-4.12**。
  - b. 安装模式是 **A specific namespace on the cluster**。
  - c. Installed Namespace 为 **Operator recommended namespace openshift-storage**。如果命名空间 **openshift-storage** 不存在，它会在 Operator 安装过程中创建。
  - d. 将 Approval Strategy 选为 **Automatic** 或 **Manual**。  
如果选择 **Automatic** 更新，Operator Lifecycle Manager(OLM)将自动升级 Operator 的运行实例，而无需任何干预。  
  
如果选择 **手动** 更新，则 OLM 会创建一个更新请求。作为集群管理员，您必须手动批准该更新请求，才能将 Operator 更新至更新的版本。
  - e. 确保为 **Console 插件** 选择了 **Enable** 选项。

- f. 点 **Install**。

### 验证步骤

- 成功安装 Operator 后，用户界面中会显示一个带有 **Web console update is available** 信息的弹出窗口。点这个弹出窗口中的 **Refresh web console** 来反映控制台的更改。
- 在 Web 控制台中：
  - 进入到 **Installed Operators**，再验证 **OpenShift Data Foundation Operator** 是否显示绿色勾号，指示安装成功。
  - 进入到 **Storage**，再验证 **Data Foundation** 仪表板是否可用。

## 3.2. 为外部 IBM FLASHSYSTEM 存储创建 OPENSIFT DATA FOUNDATION 集群

在 OpenShift Container Platform 上安装 OpenShift Data Foundation operator 后，需要创建新的 OpenShift Data Foundation 集群。

### 先决条件

- 有效的 Red Hat OpenShift Data Foundation 高级订阅。如需更多信息，请参阅 [OpenShift Data Foundation 订阅中的知识库文章](#)。
- 对于 Red Hat Enterprise Linux® 操作系统，确保 iSCSI 的连接性，然后在主机上配置 Linux 多路径设备。
- 对于 Red Hat Enterprise Linux CoreOS 或已经安装的软件包，请在主机上配置 Linux 多路径设备。
- 根据您的存储系统说明，确保为每个 worker 配置存储连接。有关最新支持的 FlashSystem 存储系统和版本，请参阅 [ODF FlashSystem 驱动程序文档](#)。

### 流程

1. 在 OpenShift Web 控制台中，点 **Operators** → **Installed Operators** 查看所有已安装的 Operator。  
确保所选项目为 **openshift-storage**。
2. 单击 OpenShift Data Foundation，然后单击 **Create StorageSystem**。
3. 在 Backing storage 页面中，选择以下选项：
  - a. 为 **Deployment 类型**选项选择 **Full Deployment**。
  - b. 从可用选项中选择 **Connect a external storage platform**。
  - c. 从 **Storage platform** 列表中选择 **IBM FlashSystem Storage**。
  - d. 点 **Next**。
4. 在 Create Storage class 页面中，提供以下信息：
  - a. 为存储类输入一个名称。

在创建块存储持久性卷时，请选择存储类 <storage\_class\_name> 以获得最佳性能。存储类允许到 FlashSystem 的直接 I/O 路径。

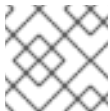
- b. 输入以下 IBM FlashSystem 连接详情：
    - IP 地址
    - 用户名
    - 密码
    - 池名称
  - c. 在 **卷模式** 中，选择 **thick** 或 **thin**。
  - d. 点 **Next**。
5. 在 Capacity and nodes 页面中，提供所需的详情：
- a. 为 Requested capacity 选择一个值。  
可用的选项有 **0.5 TiB**、**2 TiB** 和 **4 TiB**。请求的容量在基础架构存储类上动态分配。
  - b. 在三个不同的区域中至少选择三个节点。  
建议从每个节点至少 14 个 CPU 和 34 GiB RAM 开始。如果选择的节点与 OpenShift Data Foundation 的一个聚合的 30 个 CPU 和 72 GiB RAM 的要求不匹配，则会部署一个最小的集群。如需最低起始节点要求，请参阅规划指南中的[资源要求](#)部分。
  - c. 点 **Next**。
6. 可选：在 Security 和 network 页面中，提供必要详情：
- a. 若要启用加密，可选择为块存储和文件存储启用数据加密。
  - b. 选择任意一个或两个加密级别：
    - **Cluster-wide encryption** 来加密整个集群（块存储和文件存储）。
    - **Storage class encryption** 以使用加密启用的存储类创建加密的持久性卷（仅限块）。
  - c. 选择 **Connect to an external key management service** 复选框。这是集群范围加密的可选选项。
    - i. 默认情况下，Key Management Service Provider 设置为 Vault。
    - ii. 输入 Vault Service Name、Vault 服务器的主机地址 ('https://<hostname 或 ip>')、端口号和 Token。
  - d. 展开 Advanced Settings，以根据您的 Vault 配置输入其他设置和证书详情：
    - i. 在后端路径中输入为 OpenShift Data Foundation 专用且唯一的 Key Value secret 路径。
    - ii. 可选：输入 TLS Server Name 和 Vault Enterprise Namespace。
    - iii. 通过上传相应的 PEM 编码证书文件提供 CA 证书、客户端证书和客户端私钥。
  - e. 点 **Save**。
  - f. 如果使用一个单一网络，选择 **Default (SDN)**，如果使用多网络接口，选择 **Custom (Multus)**。

- i. 从下拉菜单中选择公共网络接口。
  - ii. 从下拉菜单中选择集群网络接口。备注：如果您只使用一个额外网络接口，请选择单个 **NetworkAttachmentDefinition**，即 **ocs-public-cluster** 用于公共网络接口，并将 Cluster Network Interface 留空。
- g. 点 Next。
7. 在 Review and create 页面中，检查所有详情是否正确：
    - 若要修改任何配置设置，请单击 **Back** 以返回到上一配置页面。
  8. 单击 **Create StorageSystem**。

## 验证步骤

### 验证 pod 的状态

1. 从 OpenShift Web 控制台左侧窗格中，单击 **Workloads → Pods**。
2. 从 **Project** 下拉列表中选择 **openshift-storage**。



### 注意

如果禁用 **Show default projects** 选项，请使用切换按钮列出所有默认项目。

表 3.1. 对应 OpenShift Data Foundation 组件的 Pod

组件	对应的 pod
OpenShift Data Foundation Operator	<ul style="list-style-type: none"> <li>• <b>OCS-operator-*</b> (在任何 worker 节点上有 1 个 pod)</li> <li>• <b>ocS-metrics-exporter-*</b> (任何 worker 节点上 1 个 pod)</li> <li>• <b>odF-operator-controller-manager-*</b> (任何 worker 节点上 1 个 pod)</li> <li>• <b>odf-console-*</b> (任何 worker 节点上 1 个 pod)</li> <li>• <b>csi-addons-controller-manager-*</b> (任何 worker 节点上 1 个 pod)</li> </ul>
<b>ibm-storage-odf-operator</b>	<ul style="list-style-type: none"> <li>• <b>ibm-storage-odf-operator-*</b> (任何 worker 节点上 2 个 pod)</li> <li>• <b>ibm-odf-console-*</b></li> </ul>
<b>ibm-flashsystem-storage</b>	<b>ibm-flashsystem-storage-*</b> (任何 worker 节点上 1 个 pod)
<b>rook-ceph Operator</b>	<b>rook-ceph-operator-*</b> (任何 worker 节点上 1 个 pod)

组件	对应的 pod
多云对象网关	<ul style="list-style-type: none"> <li>● <b>noobaa-operator-*</b> (任何 worker 节点上 1 个 pod)</li> <li>● <b>noobaa-core-*</b> (任何 worker 节点上有 1 个 pod)</li> <li>● <b>noobaa-db-pg-*</b> (任何 worker 节点上有 1 个 pod)</li> <li>● <b>noobaa-endpoint-*</b> (任何 worker 节点上有 1 个 pod)</li> </ul>
CSI	<ul style="list-style-type: none"> <li>● <b>ibm-block-csi-*</b> (任何 worker 节点上 1 个 pod)</li> </ul>

### 验证 OpenShift Data Foundation 集群是否健康

1. 在 Web 控制台中，点 **Storage → Data Foundation**。
2. 在 **Overview** 选项卡的 Status 卡中，验证 *Storage System* 有一个绿色勾号标记。
3. 在 **Details** 卡中，验证是否显示集群信息。

有关使用 Block and File 仪表板的 OpenShift Data Foundation 健康状况的更多信息，请参阅[监控 OpenShift Data Foundation](#)。

### 证明 Multicloud 对象网关处于健康状态

1. 在 Web 控制台中，点 **Storage → Data Foundation**。
2. 在 **Overview** 选项卡的 Status 卡中，单击 **Storage System**，然后单击弹出中的 storage 系统链接。
3. 在 **Object** 选项卡的 Status 卡中，验证 *Object Service* 和 *数据弹性* 都具有绿色勾号。
4. 在 **Details** 卡中，验证是否显示了 MCG 信息。

有关使用对象仪表板的 OpenShift Data Foundation 健康状况的更多信息，请参阅[监控 OpenShift Data Foundation](#)。

### 验证 IBM FlashSystem 是否已连接并且存储集群已就绪

- 运行以下命令，以验证 OpenShift 数据基础集群是否已连接到外部 IBM FlashSystem。

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

### 验证存储系统的存储系统

- 运行以下命令，以验证 IBM FlashSystem 存储集群的 storageSystem。

```
$ 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
```

### 验证 IBM operator 的订阅

- 运行以下命令验证订阅：

```
$ oc get subscriptions.operators.coreos.com
NAME                                PACKAGE                                SOURCE
CHANNEL
ibm-block-csi-operator-stable-certified-operators-openshift-marketplace ibm-block-csi-operator
certified-operators stable
ibm-storage-odf-operator            ibm-storage-odf-operator odf-
catalogsource stable-v1
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
```

### 验证 CSV

- 运行以下命令，以验证 CSV 是否处于 succeeded 状态。

```
$ oc get csv
NAME                                DISPLAY                                VERSION REPLACES
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
noobaa-operator.v5.9.0 NooBaa Operator 5.9.0
Succeeded
ocs-operator.v4.12.0 OpenShift Container Storage 4.12.0
Succeeded
odf-operator.v4.12.0 OpenShift Data Foundation 4.12.0
Succeeded
```

### 验证 IBM operator 和 CSI pod

- 运行以下命令以验证 IBM operator 和 CSI pod:

```
$ oc get pods
NAME                                READY STATUS RESTARTS AGE
5cb2b16ec2b11bf63dbe691d44a63535dc026bb5315d5075dc6c398b3c58194 0/1 Completed
0 10m
7c806f6568f85cf10d72508261a2535c220429b54dbcf87349b9b4b9838fctg 0/1 Completed
0 8m47s
c4b05566c04876677a22d39fc9c02512401d0962109610e85c8fb900d3jd7k2 0/1 Completed
0 10m
c5d1376974666727b02bf25b3a4828241612186744ef417a668b4bc1759rzts 0/1 Completed
0 10m
```



ibm-block-csi-operator-7b656d6cc8-bqnwp	1/1	Running	0	8m3s
ibm-odf-console-97cb7c84c-r52dq	0/1	ContainerCreating	0	8m4s
ibm-storage-odf-operator-57b8bc47df-mgkc7	1/2	ImagePullBackOff	0	94s
noobaa-operator-7698579d56-x2zqs	1/1	Running	0	9m37s
ocs-metrics-exporter-94b57d764-zq2g2	1/1	Running	0	9m32s
ocs-operator-5d96d778f6-vxlq5	1/1	Running	0	9m33s
odf-catalogsource-j7q72	1/1	Running	0	10m
odf-console-8987868cd-m7v29	1/1	Running	0	9m35s
odf-operator-controller-manager-5dbf785564-rwsgq	2/2	Running	0	9m35s
rook-ceph-operator-68b4b976d8-dlc6w	1/1	Running	0	9m32s

## 第 4 章 从外部存储系统卸载 OPENSIFT DATA FOUNDATION

使用本节中的步骤卸载 OpenShift Data Foundation。卸载 OpenShift Data Foundation 不会从外部集群中删除 RBD 池，或卸载外部 Red Hat Ceph Storage 集群。

### 卸载注解

Storage Cluster 上的注解用于更改卸载过程的行为。要定义卸载行为，在存储集群中引入了以下两个注解：

- `uninstall.ocs.openshift.io/cleanup-policy: delete`
- `uninstall.ocs.openshift.io/mode: graceful`



#### 注意

`uninstall.ocs.openshift.io/cleanup-policy` 不适用于外部模式。

下表提供了有关可用于这些注解的不同值的信息：

表 4.1. `uninstall.ocs.openshift.io` 卸载注解描述

注解	值	默认值	行为
<code>cleanup-policy</code>	<code>delete</code>	是	Rook 清理物理驱动器和 <b>DataDirHostPath</b>
<code>cleanup-policy</code>	<code>retain</code>	否	Rook <b>不会</b> 清理物理驱动器和 <b>DataDirHostPath</b>
模式	<code>graceful</code>	是	Rook 和 NooBaa <b>暂停</b> 卸载过程，直到管理员/用户移除 PVC 和 OBC
模式	<code>forced</code>	否	rook 和 NooBaa 会在卸载过程中继续进行，即使 PVC/OBCs 使用 Rook 和 NooBaa 置备

您可以通过使用以下命令编辑注解值来更改卸载模式：

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

### 先决条件

- 确保 OpenShift Data Foundation 集群处于健康状态。当因为资源或节点不足而导致部分 pod 无法成功终止时，卸载过程可能会失败。如果集群处于不健康状态，请在卸载 OpenShift Data Foundation 前联系红帽客户支持。
- 使用 OpenShift Data Foundation 提供的存储类，确保应用程序不使用持久性卷声明 (PVC) 或对象存储桶声明 (OBC)。

## 流程

1. 删除使用 OpenShift Data Foundation 的卷快照。

- a. 列出所有命名空间中的卷快照

```
$ oc get volumesnapshot --all-namespaces
```

- b. 从上一命令的输出中，识别和删除使用 OpenShift Data Foundation 的卷快照。

```
$ oc delete volumesnapshot <VOLUME-SNAPSHOT-NAME> -n <NAMESPACE>
```

2. 删除使用 OpenShift Data Foundation 的 PVC 和 OBC。

在默认的卸载模式 (graceful) 中，卸载程序会等待所有使用 OpenShift Data Foundation 的 PVC 和 OBC 被删除。

如果要事先删除 PVC 来删除存储集群，您可以将卸载模式注解设置为“强制”并跳过此步骤。这样做会导致系统中出现孤立 PVC 和 OBC。

- a. 使用 OpenShift Data Foundation 删除 OpenShift Container Platform 监控堆栈 PVC。  
请参阅[从 OpenShift Data Foundation 中删除监控堆栈](#)
- b. 使用 OpenShift Data Foundation 删除 OpenShift Container Platform Registry PVC。  
从[OpenShift Data Foundation 中删除 OpenShift Container Platform registry](#)
- c. 使用 OpenShift Data Foundation 删除 OpenShift Container Platform 日志 PVC。  
从[OpenShift Data Foundation 中删除集群日志记录操作器](#)
- d. 删除使用 OpenShift Data Foundation 置备的其他 PVC 和 OBC。

- 下面是一个示例脚本，用于标识使用 OpenShift Data Foundation 置备的 PVC 和 OBC。该脚本将忽略 OpenShift Data Foundation 内部使用的 PVC 和 OBC。

```
#!/bin/bash

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

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

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

# List PVCs in each of the StorageClasses
for SC in $OCS_STORAGECLASSES
do
    echo
    "=====
=="
    echo "$SC StorageClass PVCs and OBCs"
    echo
```

```

=====
==
    oc get pvc --all-namespaces --no-headers 2>/dev/null | grep $SC | grep -v -e
"$NOOBAA_DB_PVC" -e "$NOOBAA_BACKINGSTORE_PVC"
    oc get obc --all-namespaces --no-headers 2>/dev/null | grep $SC
    echo
done

```

- 删除 OBC。

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

- 删除 PVC。

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

确保您已删除了集群中创建的自定义后备存储、存储桶类等。

3. 删除 Storage Cluster 对象并等待相关资源被删除。

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

4. 删除命名空间并等待删除完成。如果 **openshift-storage** 是活跃的项目，则需要切换到另一个项目。  
例如：

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

如果以下命令返回 **NotFound** 错误，则项目被删除。

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



### 注意

卸载 OpenShift Data Foundation 时，如果没有完全删除命名空间并处于 **Terminating** 状态，请执行 [故障排除和删除 Uninstall 过程中剩余的资源](#) 的步骤，以识别阻塞命名空间的对象。

5. 确认已删除使用 OpenShift Data Foundation 置备的所有 PV。如果有任何 PV 处于 **Released** 状态，请将其删除。

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

6. 删除 **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 storagesystems.odf.openshift.io --
wait=true --timeout=5m
```

7. 确保完全卸载 OpenShift Data Foundation :
  - a. 在 OpenShift Container Platform Web 控制台点击 **Storage**。
  - b. 验证 **OpenShift Data Foundation** 是否不再出现在 Storage 下。

## 4.1. 从 OPENSIFT DATA FOUNDATION 中删除监控堆栈

使用本节清理 OpenShift Data Foundation 的监控堆栈。

在配置监控堆栈时创建的 PVC 位于 **openshift-monitoring** 命名空间中。

### 先决条件

- PVC 被配置为使用 OpenShift Container Platform 监控堆栈。  
如需更多信息，请参阅[配置监控堆栈](#)。

### 流程

1. 列出当前在 **openshift-monitoring** 命名空间中运行的 pod 和 PVC。

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

```

78fc8fc97d-2rgfp          3/3   Running 0      8d

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

```

## 2. 编辑监控 **configmap**。

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

删除引用 OpenShift Data Foundation 存储类的所有 **config** 部分，如下例所示并保存。

Before editing

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

After editing

```

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

```

在本例中，**alertmanagerMain** 和 **prometheusK8s** 监控组件使用 OpenShift Data Foundation PVC。

### 3. 列出使用 PVC 的 pod。

在本例中，消耗 PVC 的 **alertmanagerMain** 和 **prometheusK8s** pod 处于 **Terminating** 状态。这些 pod 不再使用 OpenShift Data Foundation PVC 后，您可以删除 PVC。

```

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

NAME                                STATUS VOLUME
CAPACITY ACCESS MODES STORAGECLASS AGE
persistentvolumeclaim/ocs-alertmanager-claim-alertmanager-main-0 Bound pvc-

```



```

2eb79797-1fed-11ea-93e1-0a88476a6a64 40Gi RWO ocs-external-
storagecluster-ceph-rbd 19h
persistentvolumeclaim/ocs-alertmanager-claim-alertmanager-main-1 Bound pvc-
2ebeee54-1fed-11ea-93e1-0a88476a6a64 40Gi RWO ocs-external-
storagecluster-ceph-rbd 19h
persistentvolumeclaim/ocs-alertmanager-claim-alertmanager-main-2 Bound pvc-2ec6a9cf-
1fed-11ea-93e1-0a88476a6a64 40Gi RWO ocs-external-storagecluster-ceph-rbd
19h
persistentvolumeclaim/ocs-prometheus-claim-prometheus-k8s-0 Bound pvc-3162a80c-
1fed-11ea-93e1-0a88476a6a64 40Gi RWO ocs-external-storagecluster-ceph-rbd
19h
persistentvolumeclaim/ocs-prometheus-claim-prometheus-k8s-1 Bound pvc-
316e99e2-1fed-11ea-93e1-0a88476a6a64 40Gi RWO ocs-external-
storagecluster-ceph-rbd 19h

```

- 删除相关的 PVC。请确定删除所有消耗存储类的 PVC。

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

## 4.2. 从 OPENSIFT DATA FOUNDATION 中删除 OPENSIFT CONTAINER PLATFORM REGISTRY

使用这个部分从 OpenShift Data Foundation 清理 OpenShift Container Platform registry。如果要配置其他存储，请参阅 [镜像 registry](#)

作为配置 OpenShift Container Platform registry 的一部分创建的 PVC 位于 **openshift-image-registry** 命名空间中。

### 先决条件

- 镜像 registry 应配置为使用 OpenShift Data Foundation PVC。

### 流程

- 编辑 **configs.imageregistry.operator.openshift.io** 对象，并删除 **storage** 部分中的内容。

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

#### 编辑前

```

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

```

编辑后

```

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

```

在本例中，PVC 称为 **registry-cephfs-rwx-pvc**，现在可以安全地删除。

## 2. 删除 PVC。

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

## 4.3. 从 OPENSIFT DATA FOUNDATION 中删除集群日志记录操作器

使用本节从 OpenShift Data Foundation 清理集群日志记录 Operator。

在配置集群日志记录 Operator 时创建的 PVC 位于 **openshift-logging** 命名空间中。

### 先决条件

- 集群日志记录实例应该已配置为使用 OpenShift Data Foundation PVC。

### 流程

#### 1. 删除命名空间中的 **ClusterLogging** 实例。

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

**openshift-logging** 命名空间中的 PVC 现在可以安全地删除。

#### 2. 删除 PVC。

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

**<pvc-name>**

是 PVC 的名称

## 4.4. 删除外部 IBM FLASHSYSTEM SECRET

卸载时，您需要从 OpenShift Data Foundation 清理 FlashSystem 机密。配置外部 IBM FlashSystem 存储时会创建此 secret。请参阅[为外部 IBM FlashSystem 存储创建 OpenShift Data Foundation 集群](#)。

### 流程

- 使用以下命令删除 IBM FlashSystem secret：

-

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