



# Red Hat OpenShift Data Foundation 4.12

## 替换设备

安全替换正常操作或失败的设备的说明



安全替换正常操作或失败的设备的说明

## 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

本文档介绍如何安全地替换 Red Hat OpenShift Data Foundation 的存储设备。

---

## 目录

使开源包含更多 .....	3
对红帽文档提供反馈 .....	4
前言 .....	5
<b>第 1 章 在 AWS 上部署动态置备的 OPENSIFT DATA FOUNDATION .....</b>	<b>6</b>
1.1. 在 AWS 用户置备的基础架构中替换操作或失败的存储设备 .....	6
1.2. 在 AWS 安装程序置备的基础架构中替换操作或失败的存储设备 .....	6
<b>第 2 章 在 VMWARE 上部署动态置备的 OPENSIFT DATA FOUNDATION .....</b>	<b>7</b>
2.1. 替换 VMWARE 基础架构上的操作或失败存储设备 .....	7
<b>第 3 章 在 RED HAT VIRTUALIZATION 上部署的动态置备的 OPENSIFT DATA FOUNDATION .....</b>	<b>13</b>
3.1. 在 RED HAT VIRTUALIZATION 安装程序置备的基础架构中替换操作或失败的存储设备 .....	13
<b>第 4 章 在 MICROSOFT AZURE 上部署的动态置备的 OPENSIFT DATA FOUNDATION .....</b>	<b>19</b>
4.1. 在 AZURE 安装程序置备的基础架构中替换操作或失败的存储设备 .....	19
<b>第 5 章 使用本地存储设备部署的 OPENSIFT DATA FOUNDATION .....</b>	<b>20</b>
5.1. 在由本地存储设备支持的集群中替换操作或失败的存储设备 .....	20
5.2. 替换 IBM POWER 上的正常工作或失败的存储设备 .....	26
5.3. 在 IBM Z 或 LINUXONE 基础架构中替换操作或失败的存储设备 .....	36



## 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**。



## 前言

根据部署类型，您可以选择以下步骤之一来替换存储设备：

- 有关在 AWS 上部署的动态创建存储集群，请参阅：
  - [第 1.1 节 “在 AWS 用户置备的基础架构中替换操作或失败的存储设备”](#)。
  - [第 1.2 节 “在 AWS 安装程序置备的基础架构中替换操作或失败的存储设备”](#)。
- 有关 VMware 上部署的动态创建存储集群，请参阅 [第 2.1 节 “替换 VMware 基础架构上的操作或失败存储设备”](#)。
- 有关在 Red Hat Virtualization 上部署的动态创建存储集群，请参阅 [第 3.1 节 “在 Red Hat Virtualization 安装程序置备的基础架构中替换操作或失败的存储设备”](#)。
- 有关在 Microsoft Azure 上部署的动态创建存储集群，请参阅 [第 4.1 节 “在 Azure 安装程序置备的基础架构中替换操作或失败的存储设备”](#)。
- 有关使用本地存储设备部署的存储集群，请参阅：
  - [第 5.1 节 “在由本地存储设备支持的集群中替换操作或失败的存储设备”](#)。
  - [第 5.2 节 “替换 IBM Power 上的正常工作或失败的存储设备”](#)。
  - [第 5.3 节 “在 IBM Z 或 LinuxONE 基础架构中替换操作或失败的存储设备”](#)。



### 注意

OpenShift Data Foundation 不支持异构 OSD 大小。

# 第 1 章 在 AWS 上部署动态置备的 OPENSIFT DATA FOUNDATION

## 1.1. 在 AWS 用户置备的基础架构中替换操作或失败的存储设备

当您需要替换 AWS 用户置备的基础架构上动态创建的存储集群中的设备时，必须替换存储节点。有关如何替换节点的详情，请参考：

- [在用户置备的基础架构中替换可运行的 AWS 节点。](#)
- [在用户置备的基础架构中替换失败的 AWS 节点。](#)

## 1.2. 在 AWS 安装程序置备的基础架构中替换操作或失败的存储设备

当您需要替换 AWS 安装程序置备的基础架构上动态创建的存储集群中的设备时，必须替换存储节点。有关如何替换节点的详情，请参考：

- [在安装程序置备的基础架构中替换可正常工作的 AWS 节点。](#)
- [在安装程序置备的基础架构中替换失败的 AWS 节点。](#)

## 第 2 章 在 VMWARE 上部署动态置备的 OPENSIFT DATA FOUNDATION

### 2.1. 替换 VMWARE 基础架构上的操作或失败存储设备

在新卷上创建新的持久性卷声明(PVC)，并在 OpenShift Data Foundation 中需要替换一个或多个虚拟机磁盘(VMD)时删除旧对象存储设备(OSD)。

#### 先决条件

- 确保数据具有弹性。
  - 在 OpenShift Web 控制台中，点 **Storage** → **Data Foundation**。
  - 点 **Storage Systems** 选项卡，然后点 **ocs-storagecluster-storage-system**。
  - 在 **Block and File** 仪表板的 **Status** 卡中，在 **Overview** 选项卡中，验证 *Data Resiliency* 是否具有绿色勾号。

#### 流程

1. 确定需要替换的 OSD，以及在其上调度 OSD 的 OpenShift Container Platform 节点。

```
$ oc get -n openshift-storage pods -l app=rook-ceph-osd -o wide
```

输出示例：

```
rook-ceph-osd-0-6d77d6c7c6-m8xj6 0/1 CrashLoopBackOff 0 24h 10.129.0.16
compute-2 <none> <none>
rook-ceph-osd-1-85d99fb95f-2svc7 1/1 Running 0 24h 10.128.2.24 compute-
0 <none> <none>
rook-ceph-osd-2-6c66cdb977-jp542 1/1 Running 0 24h 10.130.0.18 compute-
1 <none> <none>
```

在本例中，**rook-ceph-osd-0-6d77d6c7c6-m8xj6** 需要替换，**compute-2** 是调度 OSD 的 OpenShift Container Platform 节点。



#### 注意

如果要更换的 OSD 处于健康状态，则 Pod 的状态将为 **Running**。

2. 缩减 OSD 部署，以替换 OSD。  
每次您要替换 OSD 时，将 **osd\_id\_to\_remove** 参数更新为 OSD ID，再重复此步骤。

```
$ osd_id_to_remove=0
```

```
$ oc scale -n openshift-storage deployment rook-ceph-osd-${osd_id_to_remove} --replicas=0
```

其中，**osd\_id\_to\_remove** 是 pod 名称中紧接在 **rook-ceph-osd** 前缀后面的整数。在本例中，部署名称为 **rook-ceph-osd-0**。

输出示例：

```
deployment.extensions/rook-ceph-osd-0 scaled
```

3. 验证 **rook-ceph-osd** pod 是否已终止。

```
$ oc get -n openshift-storage pods -l ceph-osd-id=${osd_id_to_remove}
```

输出示例：

```
No resources found.
```

### 重要

如果 **rook-ceph-osd** pod 处于 **terminating** 状态，请使用 **force** 选项删除 pod。

```
$ oc delete pod rook-ceph-osd-0-6d77d6c7c6-m8xj6 --force --grace-period=0
```

输出示例：

```
warning: Immediate deletion does not wait for confirmation that the running
resource has been terminated. The resource may continue to run on the
cluster indefinitely.
```

```
pod "rook-ceph-osd-0-6d77d6c7c6-m8xj6" force deleted
```

4. 从集群中移除旧 OSD，以便您可以添加新 OSD。

- a. 删除所有旧的 **ocs-osd-removal** 任务。

```
$ oc delete -n openshift-storage job ocs-osd-removal-job
```

输出示例：

```
job.batch "ocs-osd-removal-job" deleted
```

- b. 进入 **openshift-storage** 项目。

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

- c. 从集群中移除旧 OSD。

```
$ oc process -n openshift-storage ocs-osd-removal -p
FAILED_OSD_IDS=${osd_id_to_remove} -p FORCE_OSD_REMOVAL=false |oc create
-n openshift-storage -f -
```

**FORCE\_OSD\_REMOVAL** 值必须在有三个 OSD 的集群中更改为"true"，或者有足够空间的集群在移除 OSD 后恢复所有这三个数据副本。

**警告**

这一步会导致 OSD 完全从集群中移除。确保提供了 **osd\_id\_to\_remove** 的正确值。

5. 通过检查 **ocs-osd-removal-job** pod 的状态，验证 OSD 是否已成功移除。状态为 **Completed**，确认 OSD 移除作业已成功。

```
$ oc get pod -l job-name=ocs-osd-removal-job -n openshift-storage
```

6. 确保 OSD 移除已完成。

```
$ oc logs -l job-name=ocs-osd-removal-job -n openshift-storage --tail=-1 | egrep -i 'completed removal'
```

输出示例：

```
2022-05-10 06:50:04.501511 I | cephosd: completed removal of OSD 0
```

**重要**

如果 **ocs-osd-removal-job** pod 失败且 pod 处于预期的 **Completed** 状态，请检查 pod 日志以进一步调试。

例如：

```
# oc logs -l job-name=ocs-osd-removal-job -n openshift-storage --tail=-1
```

7. 如果在安装时启用了加密，在从相应 OpenShift Data Foundation 节点中删除的 OSD 设备中删除 **dm-crypt** 关联的 **device-mapper** 映射。

- a. 从 **ocs-osd-removal-job** pod 日志中获取被替换 OSD 的 PVC 名称。

```
$ oc logs -l job-name=ocs-osd-removal-job -n openshift-storage --tail=-1 | egrep -i 'pvc|deviceset'
```

输出示例：

```
2021-05-12 14:31:34.666000 I | cephosd: removing the OSD PVC "ocs-deviceset-xxxx-xxx-xxx-xxx"
```

- b. 对于之前标识的每个节点，执行以下操作：

- i. 创建 **debug** pod 和 **chroot** 到存储节点上的主机。

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

```
<node name>
```

是节点的名称。

```
$ chroot /host
```

ii. 根据上一步中标识的 PVC 名称，查找相关的设备名称。

```
$ dmsetup ls| grep <pvc name>
```

**<pvc name>**

是 PVC 的名称。

输出示例：

```
ocs-deviceset-xxx-xxx-xxx-xxx-block-dmccrypt (253:0)
```

iii. 删除映射的设备。

```
$ cryptsetup luksClose --debug --verbose ocs-deviceset-xxx-xxx-xxx-xxx-block-dmccrypt
```

### 重要

如果上述命令因为权限不足而卡住，请运行以下命令：

- 按 **CTRL+Z** 退出上述命令。
- 查找阻塞的进程的 PID。

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

- 使用 **kill** 命令终止进程。

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

**<PID>**

是进程 ID。

- 验证设备名称是否已移除。

```
$ dmsetup ls
```

8. 删除 **ocs-osd-removal** 任务。

```
$ oc delete -n openshift-storage job ocs-osd-removal-job
```

输出示例：

```
job.batch "ocs-osd-removal-job" deleted
```



## 注意

使用带有数据加密的外部密钥管理系统(KMS)时，可以从 Vault 服务器中删除旧的 OSD 加密密钥，因为它现在是孤立的密钥。

## 验证步骤

1. 验证是否有新的 OSD 正在运行。

```
$ oc get -n openshift-storage pods -l app=rook-ceph-osd
```

输出示例：

```
rook-ceph-osd-0-5f7f4747d4-snshw      1/1   Running   0      4m47s
rook-ceph-osd-1-85d99fb95f-2svc7     1/1   Running   0      1d20h
rook-ceph-osd-2-6c66cdb977-jp542    1/1   Running   0      1d20h
```

2. 验证是否创建了处于 **Bound** 状态的新 PVC。

```
$ oc get -n openshift-storage pvc
```

输出示例：

```
NAME                                STATUS VOLUME                                CAPACITY ACCESS
MODES STORAGECLASS AGE
ocs-deviceset-0-0-2s6w4 Bound  pvc-7c9bcaf7-de68-40e1-95f9-0b0d7c0ae2fc 512Gi
RWO thin 5m
ocs-deviceset-1-0-q8fwh Bound  pvc-9e7e00cb-6b33-402e-9dc5-b8df4fd9010f 512Gi
RWO thin 1d20h
ocs-deviceset-2-0-9v8lq Bound  pvc-38cdfcee-ea7e-42a5-a6e1-aaa6d4924291 512Gi
RWO thin 1d20h
```

3. 可选：如果在集群中启用了集群范围的加密，请验证新 OSD 设备是否已加密。

- a. 识别运行新 OSD pod 的节点。

```
$ oc get -n openshift-storage -o=custom-columns=NODE:.spec.nodeName pod/<OSD-
pod-name>
```

**<OSD-pod-name>**

是 OSD pod 的名称。

例如：

```
$ oc get -n openshift-storage -o=custom-columns=NODE:.spec.nodeName pod/rook-
ceph-osd-0-544db49d7f-qrgqm
```

输出示例：

```
NODE
compute-1
```

- b. 对于上一步中确定的每个节点，请执行以下操作：

- i. 创建调试 pod，并为所选主机打开 chroot 环境。

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

**<node name>**

是节点的名称。

```
$ chroot /host
```

- ii. 检查 **ocs-deviceset** 名称旁边的 **crypt** 关键字。

```
$ lsblk
```

4. 登录 OpenShift Web 控制台并查看存储仪表板。



## 第 3 章 在 RED HAT VIRTUALIZATION 上部署的动态置备的 OPENSIFT DATA FOUNDATION

### 3.1. 在 RED HAT VIRTUALIZATION 安装程序置备的基础架构中替换操作或失败的存储设备

在新卷上创建一个新的持久性卷声明(PVC)，再删除旧的对象存储设备(OSD)。

#### 先决条件

- 确保数据具有弹性。
  - 在 OpenShift Web 控制台中，点 **Storage** → **Data Foundation**。
  - 点 **Storage Systems** 选项卡，然后点 **ocs-storagecluster-storagesystem**。
  - 在 **Block and File** 仪表板的 **Status** 卡中，在 **Overview** 选项卡中，验证 *Data Resiliency* 是否具有绿色勾号。

#### 流程

1. 确定需要替换的 OSD，以及在其上调度 OSD 的 OpenShift Container Platform 节点。

```
$ oc get -n openshift-storage pods -l app=rook-ceph-osd -o wide
```

输出示例：

```
rook-ceph-osd-0-6d77d6c7c6-m8xj6 0/1 CrashLoopBackOff 0 24h 10.129.0.16
compute-2 <none> <none>
rook-ceph-osd-1-85d99fb95f-2svc7 1/1 Running 0 24h 10.128.2.24 compute-
0 <none> <none>
rook-ceph-osd-2-6c66cdb977-jp542 1/1 Running 0 24h 10.130.0.18 compute-
1 <none> <none>
```

在本例中，**rook-ceph-osd-0-6d77d6c7c6-m8xj6** 需要替换，**compute-2** 是调度 OSD 的 OpenShift Container Platform 节点。



#### 注意

如果要更换的 OSD 处于健康状态，则 Pod 的状态将为 **Running**。

2. 缩减 OSD 部署，以替换 OSD。  
每次您要替换 OSD 时，将 **osd\_id\_to\_remove** 参数更新为 OSD ID，再重复此步骤。

```
$ osd_id_to_remove=0
```

```
$ oc scale -n openshift-storage deployment rook-ceph-osd-${osd_id_to_remove} --replicas=0
```

其中，**osd\_id\_to\_remove** 是 pod 名称中紧接在 **rook-ceph-osd** 前缀后面的整数。在本例中，部署名称为 **rook-ceph-osd-0**。

输出示例：

```
deployment.extensions/rook-ceph-osd-0 scaled
```

3. 验证 **rook-ceph-osd** pod 是否已终止。

```
$ oc get -n openshift-storage pods -l ceph-osd-id=${osd_id_to_remove}
```

输出示例：

```
No resources found.
```

### 重要

如果 **rook-ceph-osd** pod 处于 **terminating** 状态，请使用 **force** 选项删除 pod。

```
$ oc delete pod rook-ceph-osd-0-6d77d6c7c6-m8xj6 --force --grace-period=0
```

输出示例：

```
warning: Immediate deletion does not wait for confirmation that the running
resource has been terminated. The resource may continue to run on the
cluster indefinitely.
```

```
pod "rook-ceph-osd-0-6d77d6c7c6-m8xj6" force deleted
```

4. 从集群中移除旧 OSD，以便您可以添加新 OSD。

- a. 删除所有旧的 **ocs-osd-removal** 任务。

```
$ oc delete -n openshift-storage job ocs-osd-removal-job
```

输出示例：

```
job.batch "ocs-osd-removal-job"
```

- b. 进入 **openshift-storage** 项目。

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

- c. 从集群中移除旧 OSD。

```
$ oc process -n openshift-storage ocs-osd-removal -p
FAILED_OSD_IDS=${osd_id_to_remove} -p FORCE_OSD_REMOVAL=false |oc create
-n openshift-storage -f -
```

**FORCE\_OSD\_REMOVAL** 值必须在有三个 OSD 的集群中更改为"true"，或者有足够空间的集群在移除 OSD 后恢复所有这三个数据副本。

**警告**

这一步会导致 OSD 完全从集群中移除。确保提供了 **osd\_id\_to\_remove** 的正确值。

5. 通过检查 **ocs-osd-removal-job** pod 的状态，验证 OSD 是否已成功移除。状态为 **Completed**，确认 OSD 移除作业已成功。

```
$ oc get pod -l job-name=ocs-osd-removal-job -n openshift-storage
```

6. 确保 OSD 移除已完成。

```
$ oc logs -l job-name=ocs-osd-removal-job -n openshift-storage --tail=-1 | egrep -i 'completed removal'
```

输出示例：

```
2022-05-10 06:50:04.501511 I | cephosd: completed removal of OSD 0
```

**重要**

如果 **ocs-osd-removal-job** 失败并且 pod 不在预期的 **Completed** 状态，请检查 pod 日志来进一步调试。

例如：

```
# oc logs -l job-name=ocs-osd-removal-job -n openshift-storage --tail=-1'
```

7. 如果在安装时启用了加密，在从相应 OpenShift Data Foundation 节点中删除的 OSD 设备中删除 **dm-crypt** 关联的 **device-mapper** 映射。

- a. 从 **ocs-osd-removal-job** pod 日志中获取被替换 OSD 的 PVC 名称。

```
$ oc logs -l job-name=ocs-osd-removal-job -n openshift-storage --tail=-1 | egrep -i 'pvc|deviceset'
```

输出示例：

```
2021-05-12 14:31:34.666000 I | cephosd: removing the OSD PVC "ocs-deviceset-xxxx-xxx-xxx-xxx"
```

- b. 对于之前标识的每个节点，执行以下操作：

- i. 创建 **debug** pod 和 **chroot** 到存储节点上的主机。

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

```
<node name>
```

是节点的名称。

```
$ chroot /host
```

ii. 根据上一步中标识的 PVC 名称，查找相关的设备名称。

```
$ dmsetup ls| grep <pvc name>
```

**<pvc name>**

是 PVC 的名称。

输出示例：

```
ocs-deviceset-xxx-xxx-xxx-xxx-block-dmccrypt (253:0)
```

iii. 删除映射的设备。

```
$ cryptsetup luksClose --debug --verbose ocs-deviceset-xxx-xxx-xxx-xxx-block-dmccrypt
```

### 重要

如果上述命令因为权限不足而卡住，请运行以下命令：

- 按 **CTRL+Z** 退出上述命令。
- 查找阻塞的进程的 PID。

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

- 使用 **kill** 命令终止进程。

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

**<PID>**

是进程 ID。

- 验证设备名称是否已移除。

```
$ dmsetup ls
```

8. 删除 **ocs-osd-removal** 任务。

```
$ oc delete -n openshift-storage job ocs-osd-removal-job
```

输出示例：

```
job.batch "ocs-osd-removal-job" deleted
```



## 注意

使用带有数据加密的外部密钥管理系统(KMS)时，可以从 Vault 服务器中删除旧的 OSD 加密密钥，因为它现在是孤立的密钥。

## 验证步骤

1. 验证是否有新的 OSD 正在运行。

```
$ oc get -n openshift-storage pods -l app=rook-ceph-osd
```

输出示例：

```
rook-ceph-osd-0-5f7f4747d4-snshw          1/1   Running   0       4m47s
rook-ceph-osd-1-85d99fb95f-2svc7        1/1   Running   0       1d20h
rook-ceph-osd-2-6c66cdb977-jp542        1/1   Running   0       1d20h
```

2. 验证是否创建了处于 **Bound** 状态的新 PVC。

```
$ oc get -n openshift-storage pvc
```

3. 可选：如果在集群中启用了集群范围的加密，请验证新 OSD 设备是否已加密。

- a. 识别运行新 OSD pod 的节点。

```
$ oc get -n openshift-storage -o=custom-columns=NODE:.spec.nodeName pod/<OSD-  
pod-name>
```

**<OSD-pod-name>**

是 OSD pod 的名称。

例如：

```
$ oc get -n openshift-storage -o=custom-columns=NODE:.spec.nodeName pod/rook-  
ceph-osd-0-544db49d7f-qrgqm
```

输出示例：

```
NODE  
compute-1
```

- b. 对于之前标识的每个节点，执行以下操作：

- i. 创建调试 pod，并为所选主机打开 chroot 环境。

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

**<node name>**

是节点的名称。

```
$ chroot /host
```

- ii. 检查 **ocs-deviceset** 名称旁边的 **crypt** 关键字。

```
█ $ lsblk
```

4. 登录 OpenShift Web 控制台并查看存储仪表板。

## 第 4 章 在 MICROSOFT AZURE 上部署的动态置备的 OPENSIFT DATA FOUNDATION

### 4.1. 在 AZURE 安装程序置备的基础架构中替换操作或失败的存储设备

当您需要替换 Azure 安装程序置备的基础架构上动态创建的存储集群中的设备时，必须替换存储节点。有关如何替换节点的详情，请参考：

- [替换 Azure 安装程序置备的基础架构上的操作节点。](#)
- [在 Azure 安装程序置备的基础架构中替换失败的节点。](#)

## 第 5 章 使用本地存储设备部署的 OPENSIFT DATA FOUNDATION

### 5.1. 在由本地存储设备支持的集群中替换操作或失败的存储设备

您可以使用以下基础架构中的本地存储设备替换 OpenShift Data Foundation 中部署的对象存储设备 (OSD)：

- 裸机
- VMware
- Red Hat Virtualization



#### 注意

可能需要替换一个或多个底层存储设备。

#### 先决条件

- 红帽建议为替换设备配置类似的基础架构和资源，以用于被替换的设备。
- 确保数据具有弹性。
  - 在 OpenShift Web 控制台中，点 **Storage** → **Data Foundation**。
  - 点 **Storage Systems** 选项卡，然后点 **ocs-storagecluster-storagesystem**。
  - 在 **Block and File** 仪表板的 **Status** 卡中，在 **Overview** 选项卡中，验证 *Data Resiliency* 是否具有绿色勾号。

#### 流程

1. 从相关的 worker 节点中删除底层存储设备。
2. 验证相关的 OSD Pod 已移到 `CrashLoopBackOff` 状态。  
确定需要替换的 OSD，以及在其上调度 OSD 的 OpenShift Container Platform 节点。

```
$ oc get -n openshift-storage pods -l app=rook-ceph-osd -o wide
```

输出示例：

```
rook-ceph-osd-0-6d77d6c7c6-m8xj6 0/1 CrashLoopBackOff 0 24h 10.129.0.16
compute-2 <none> <none>
rook-ceph-osd-1-85d99fb95f-2svc7 1/1 Running 0 24h 10.128.2.24 compute-
0 <none> <none>
rook-ceph-osd-2-6c66cdb977-jp542 1/1 Running 0 24h 10.130.0.18 compute-
1 <none> <none>
```

在本例中，**rook-ceph-osd-0-6d77d6c7c6-m8xj6** 需要替换，**compute-2** 是调度 OSD 的 OpenShift Container Platform 节点。

3. 缩减 OSD 部署，以替换 OSD。



```
$ osd_id_to_remove=0
```

```
$ oc scale -n openshift-storage deployment rook-ceph-osd- $\{osd\_id\_to\_remove\}$  --replicas=0
```

其中，**osd\_id\_to\_remove** 是 pod 名称中紧接在 **rook-ceph-osd** 前缀后面的整数。在本例中，部署名称为 **rook-ceph-osd-0**。

输出示例：

```
deployment.extensions/rook-ceph-osd-0 scaled
```

#### 4. 验证 **rook-ceph-osd** pod 是否已终止。

```
$ oc get -n openshift-storage pods -l ceph-osd-id= $\{osd\_id\_to\_remove\}$ 
```

输出示例：

```
No resources found in openshift-storage namespace.
```

### 重要

如果 **rook-ceph-osd** pod 处于 **terminating** 状态超过几分钟，请使用 **force** 选项删除 pod。

```
$ oc delete -n openshift-storage pod rook-ceph-osd-0-6d77d6c7c6-m8xj6 --
  grace-period=0 --force
```

输出示例：

```
warning: Immediate deletion does not wait for confirmation that the running
  resource has been terminated. The resource may continue to run on the
  cluster indefinitely.
  pod "rook-ceph-osd-0-6d77d6c7c6-m8xj6" force deleted
```

#### 5. 从集群中移除旧 OSD，以便您可以添加新 OSD。

##### a. 删除所有旧的 **ocs-osd-removal** 任务。

```
$ oc delete -n openshift-storage job ocs-osd-removal-job
```

输出示例：

```
job.batch "ocs-osd-removal-job" deleted
```

##### b. 进入 **openshift-storage** 项目。

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

##### c. 从集群中移除旧 OSD。

```
$ oc process -n openshift-storage ocs-osd-removal -p
FAILED_OSD_IDS=${osd_id_to_remove} -p FORCE_OSD_REMOVAL=false |oc create
-n openshift-storage -f -
```

FORCE\_OSD\_REMOVAL 值必须在有三个 OSD 的集群中更改为"true", 或者有足够空间的集群在移除 OSD 后恢复所有这三个数据副本。



### 警告

这一步会导致 OSD 完全从集群中移除。确保提供了 **osd\_id\_to\_remove** 的正确值。

- 通过检查 **ocs-osd-removal-job** pod 的状态, 验证 OSD 是否已成功移除。状态为 **Completed**, 确认 OSD 移除作业已成功。

```
$ oc get pod -l job-name=ocs-osd-removal-job -n openshift-storage
```

- 确保 OSD 移除已完成。

```
$ oc logs -l job-name=ocs-osd-removal-job -n openshift-storage --tail=-1 | egrep -i 'completed
removal'
```

输出示例 :

```
2022-05-10 06:50:04.501511 I | cephosd: completed removal of OSD 0
```

### 重要

如果 **ocs-osd-removal-job** 失败并且 pod 不在预期的 **Completed** 状态, 请检查 pod 日志来进一步调试。

例如 :

```
# oc logs -l job-name=ocs-osd-removal-job -n openshift-storage --tail=-1
```

- 如果在安装时启用了加密, 在从相应 OpenShift Data Foundation 节点中删除的 OSD 设备中删除 **dm-crypt** 关联的 **device-mapper** 映射。

- 从 **ocs-osd-removal-job** pod 的日志中获取已替换 OSD 的持久性卷声明(PVC)名称。

```
$ oc logs -l job-name=ocs-osd-removal-job -n openshift-storage --tail=-1 |egrep -i
'pvc|deviceset'
```

输出示例 :

```
2021-05-12 14:31:34.666000 I | cephosd: removing the OSD PVC "ocs-deviceset-xxxx-
xxx-xxx-xxx"
```

b. 对于之前标识的每个节点，执行以下操作：

i. 创建 **debug** pod 和 **chroot** 到存储节点上的主机。

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

**<node name>**

是节点的名称。

```
$ chroot /host
```

ii. 根据上一步中标识的 PVC 名称，查找相关的设备名称。

```
$ dmsetup ls | grep <pvc name>
```

**<pvc name>**

是 PVC 的名称。

输出示例：

```
ocs-deviceset-xxx-xxx-xxx-xxx-block-dmccrypt (253:0)
```

iii. 删除映射的设备。

```
$ cryptsetup luksClose --debug --verbose <ocs-deviceset-name>
```

**<ocs-deviceset-name>**

是基于上一步中标识的 PVC 名称的相关设备名称。

### 重要

如果上述命令因为权限不足而卡住，请运行以下命令：

- 按 **CTRL+Z** 退出上述命令。
- 查找阻塞的进程的 PID。

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

- 使用 **kill** 命令终止进程。

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

**<PID>**

是进程 ID。

- 验证设备名称是否已移除。

```
$ dmsetup ls
```

9. 查找需要删除的持久性卷(PV)。

```
$ oc get pv -L kubernetes.io/hostname | grep <storageclass-name> | grep Released
```

输出示例：

```
local-pv-d6bf175b      1490Gi    RWO      Delete   Released   openshift-
storage/ocs-deviceset-0-data-0-6c5pw  localblock  2d22h    compute-1
```

10. 删除 PV。

```
$ oc delete pv <pv_name>
```

11. 向节点物理地添加新设备。

12. 跟踪与 **deviceInclusionSpec** 匹配的设备的 PV 置备。置备 PV 可能需要几分钟时间。

```
$ oc -n openshift-local-storage describe localvolumeset localblock
```

输出示例：

```
[...]
Status:
Conditions:
  Last Transition Time:      2020-11-17T05:03:32Z
  Message:                  DiskMaker: Available, LocalProvisioner: Available
  Status:                   True
  Type:                     DaemonSetsAvailable
  Last Transition Time:      2020-11-17T05:03:34Z
  Message:                  Operator reconciled successfully.
  Status:                   True
  Type:                     Available
Observed Generation:        1
Total Provisioned Device Count: 4
Events:
Type Reason Age From Message
---- -
Normal Discovered 2m30s (x4 localvolumeset- node.example.com -
NewDevice over 2m30s) symlink-controller found possible
matching disk,
waiting 1m to claim
Normal FoundMatch 89s (x4 localvolumeset- node.example.com -
ingDisk over 89s) symlink-controller symlinking matching
disk
```

调配 PV 后，为 PV 自动创建新的 OSD pod。

13. 删除 **ocs-osd-removal** 任务。

```
$ oc delete -n openshift-storage job ocs-osd-removal-job
```

输出示例：

```
job.batch "ocs-osd-removal-job" deleted
```



### 注意

使用带有数据加密的外部密钥管理系统(KMS)时，可以从 Vault 服务器中删除旧的 OSD 加密密钥，因为它现在是孤立的密钥。

### 验证步骤

1. 验证是否有新的 OSD 正在运行。

```
$ oc get -n openshift-storage pods -l app=rook-ceph-osd
```

输出示例：

```
rook-ceph-osd-0-5f7f4747d4-snshw 1/1 Running 0 4m47s
rook-ceph-osd-1-85d99fb95f-2svc7 1/1 Running 0 1d20h
rook-ceph-osd-2-6c66cdb977-jp542 1/1 Running 0 1d20h
```



### 重要

如果新 OSD 在几分钟后没有显示 **Running**，请重启 **rook-ceph-operator** pod 来强制协调。

```
$ oc delete pod -n openshift-storage -l app=rook-ceph-operator
```

输出示例：

```
pod "rook-ceph-operator-6f74fb5bff-2d982" deleted
```

2. 验证是否创建了新 PVC。

```
$ oc get -n openshift-storage pvc | grep localblock
```

输出示例：

```
ocs-deviceset-0-0-c2mqb Bound local-pv-b481410 1490Gi RWO localblock
5m
ocs-deviceset-1-0-959rp Bound local-pv-414755e0 1490Gi RWO localblock
1d20h
ocs-deviceset-2-0-79j94 Bound local-pv-3e8964d3 1490Gi RWO localblock
1d20h
```

3. 可选：如果在集群中启用了集群范围的加密，请验证新 OSD 设备是否已加密。

- a. 识别运行新 OSD pod 的节点。

```
$ oc get -n openshift-storage -o=custom-columns=NODE:.spec.nodeName pod/<OSD-
pod-name>
```

**<OSD-pod-name>**

是 OSD pod 的名称。

例如：

```
$ oc get -n openshift-storage -o=custom-columns=NODE:.spec.nodeName pod/rook-ceph-osd-0-544db49d7f-qrgqm
```

输出示例：

```
NODE
compute-1
```

b. 对于上一步中确定的每个节点，请执行以下操作：

i. 创建调试 pod，并为所选主机打开 chroot 环境。

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

**<node name>**

是节点的名称。

```
$ chroot /host
```

ii. 检查 **ocs-deviceset** 名称旁边的 **crypt** 关键字。

```
$ lsblk
```

4. 登录 OpenShift Web 控制台，再检查存储控制面板上的 OSD 状态。

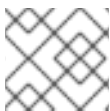


### 注意

根据正在恢复的数据量，完整数据恢复可能需要更长的时间。

## 5.2. 替换 IBM POWER 上的正常工作或失败的存储设备

您可以使用 IBM Power 上的本地存储设备替换 OpenShift Data Foundation 中部署的对象存储设备 (OSD)。



### 注意

可能需要替换一个或多个底层存储设备。

### 先决条件

- 红帽建议为替换设备配置类似的基础架构和资源，以用于被替换的设备。
- 确保数据具有弹性。
  - 在 OpenShift Web 控制台中，点 **Storage** → **Data Foundation**。
  - 点 **Storage Systems** 选项卡，然后点 **ocs-storagecluster-storagesystem**。

- 在 **Block and File** 仪表板的 **Status** 卡中，在 **Overview** 选项卡中，验证 *Data Resiliency* 是否具有绿色勾号。

## 流程

1. 确定需要替换的 OSD，以及在其上调度 OSD 的 OpenShift Container Platform 节点。

```
$ oc get -n openshift-storage pods -l app=rook-ceph-osd -o wide
```

输出示例：

```
rook-ceph-osd-0-86bf8cdc8-4nb5t 0/1  crashLoopBackOff 0 24h 10.129.2.26
worker-0 <none> <none>
rook-ceph-osd-1-7c99657cfb-jdzvz 1/1  Running 0 24h 10.128.2.46 worker-1
<none> <none>
rook-ceph-osd-2-5f9f6dfb5b-2mnw9 1/1  Running 0 24h 10.131.0.33 worker-2
<none> <none>
```

在本例中，需要替换 **rook-ceph-osd-0-86bf8cdc8-4nb5t**，**worker-0** 是调度 OSD 的 RHOC 节点。



### 注意

如果要更换的 OSD 处于健康状态，则 Pod 的状态将为 **Running**。

2. 缩减 OSD 部署，以替换 OSD。

```
$ osd_id_to_remove=0
```

```
$ oc scale -n openshift-storage deployment rook-ceph-osd-${osd_id_to_remove} --replicas=0
```

其中，**osd\_id\_to\_remove** 是 pod 名称中紧接在 **rook-ceph-osd** 前缀后面的整数。在本例中，部署名称为 **rook-ceph-osd-0**。

输出示例：

```
deployment.extensions/rook-ceph-osd-0 scaled
```

3. 验证 **rook-ceph-osd** pod 是否已终止。

```
$ oc get -n openshift-storage pods -l ceph-osd-id=${osd_id_to_remove}
```

输出示例：

```
No resources found in openshift-storage namespace.
```



## 重要

如果 **rook-ceph-osd** pod 处于 **terminating** 状态超过几分钟，请使用 **force** 选项删除 pod。

```
$ oc delete -n openshift-storage pod rook-ceph-osd-0-86bf8cdc8-4nb5t --
  grace-period=0 --force
```

输出示例：

```
warning: Immediate deletion does not wait for confirmation that the running
  resource has been terminated. The resource may continue to run on the
  cluster indefinitely.
  pod "rook-ceph-osd-0-86bf8cdc8-4nb5t" force deleted
```

4. 从集群中移除旧 OSD，以便您可以添加新 OSD。

a. 识别与要替换的 OSD 关联的 **DeviceSet**。

```
$ oc get -n openshift-storage -o yaml deployment rook-ceph-osd-${osd_id_to_remove} |
  grep ceph.rook.io/pvc
```

输出示例：

```
ceph.rook.io/pvc: ocs-deviceset-localblock-0-data-0-64xjl
  ceph.rook.io/pvc: ocs-deviceset-localblock-0-data-0-64xjl
```

在本例中，持久性卷声明(PVC)名称是 **ocs-deviceset-localblock-0-data-0-64xjl**。

b. 识别与 PVC 关联的持久性卷(PV)。

```
$ oc get -n openshift-storage pvc ocs-deviceset-<x>-<y>-<pvc-suffix>
```

其中 **x**、**y** 和 **pvc-suffix** 是前面步骤中标识的 **DeviceSet** 中的值。

输出示例：

NAME	STATUS	VOLUME	CAPACITY	ACCESS MODES
ocs-deviceset-localblock-0-data-0-64xjl	Bound	local-pv-8137c873	256Gi	RWO
localblock	24h			

在本例中，关联的 PV 是 **local-pv-8137c873**。

c. 确定要替换的设备的名称。

```
$ oc get pv local-pv-<pv-suffix> -o yaml | grep path
```

其中，**pv-suffix** 是前面步骤中标识的 PV 名称中的值。

输出示例：

```
path: /mnt/local-storage/localblock/vdc
```



在本例中，设备名称为 **vdc**。

- d. 找到与要被替换的 OSD 关联的 **prepare-pod**。

```
$ oc describe -n openshift-storage pvc ocs-deviceset-<x>-<y>-<pvc-suffix> | grep Used
```

其中 **x**、**y** 和 **pvc-suffix** 是前面步骤中标识的 **DeviceSet** 中的值。

输出示例：

```
Used By: rook-ceph-osd-prepare-ocs-deviceset-localblock-0-data-0-64knzkc
```

在本例中，**prepare-pod** 名称为 **rook-ceph-osd-prepare-ocs-deviceset-localblock-0-data-0-64knzkc**。

- e. 删除所有旧的 **ocs-osd-removal** 任务。

```
$ oc delete -n openshift-storage job ocs-osd-removal-job
```

输出示例：

```
job.batch "ocs-osd-removal-job" deleted
```

- f. 更改到 **openshift-storage** 项目。

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

- g. 从集群中移除旧 OSD。

```
$ oc process -n openshift-storage ocs-osd-removal -p
FAILED_OSD_IDS=${osd_id_to_remove} -p FORCE_OSD_REMOVAL=false |oc create
-n openshift-storage -f -
```

**FORCE\_OSD\_REMOVAL** 值必须在有三个 OSD 的集群中更改为 "true"，或者有足够空间的集群在移除 OSD 后恢复所有这三个数据副本。



### 警告

这一步会导致 OSD 完全从集群中移除。确保提供了 **osd\_id\_to\_remove** 的正确值。

5. 通过检查 **ocs-osd-removal-job** pod 的状态，验证 OSD 是否已成功移除。状态为 **Completed**，确认 OSD 移除作业已成功。

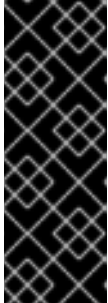
```
$ oc get pod -l job-name=ocs-osd-removal-job -n openshift-storage
```

6. 确保 OSD 移除已完成。

```
$ oc logs -l job-name=ocs-osd-removal-job -n openshift-storage --tail=-1 | egrep -i 'completed removal'
```

输出示例：

```
2022-05-10 06:50:04.501511 I | cephosd: completed removal of OSD 0
```



### 重要

如果 **ocs-osd-removal-job** 失败并且 pod 不在预期的 **Completed** 状态，请检查 pod 日志来进一步调试。

例如：

```
# oc logs -l job-name=ocs-osd-removal-job -n openshift-storage --tail=-1
```

7. 如果在安装时启用了加密，在从相应 OpenShift Data Foundation 节点中删除的 OSD 设备中删除 **dm-crypt** 关联的 **device-mapper** 映射。

- a. 从 **ocs-osd-removal-job** pod 日志中获取被替换 OSD 的 PVC 名称。

```
$ oc logs -l job-name=ocs-osd-removal-job -n openshift-storage --tail=-1 | egrep -i 'pvc|deviceset'
```

输出示例：

```
2021-05-12 14:31:34.666000 I | cephosd: removing the OSD PVC "ocs-deviceset-xxxx-xxx-xxx-xxx"
```

- b. 对于之前标识的每个节点，执行以下操作：

- i. 创建 **debug** pod 和 **chroot** 到存储节点上的主机。

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

**<node name>**

是节点的名称。

```
$ chroot /host
```

- ii. 根据上一步中标识的 PVC 名称，查找相关的设备名称。

```
$ dmsetup ls | grep <pvc name>
```

**<pvc name>**

是 PVC 的名称。

输出示例：

```
ocs-deviceset-xxx-xxx-xxx-xxx-block-dmccrypt (253:0)
```

iii. 删除映射的设备。

```
$ cryptsetup luksClose --debug --verbose ocs-deviceset-xxx-xxx-xxx-xxx-block-
dmccrypt
```

### 重要

如果上述命令因为权限不足而卡住，请运行以下命令：

- 按 **CTRL+Z** 退出上述命令。

- 查找阻塞的进程的 PID。

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

- 使用 **kill** 命令终止进程。

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

**<PID>**

是进程 ID。

- 验证设备名称是否已移除。

```
$ dmsetup ls
```

8. 查找需要删除的 PV。

```
$ oc get pv -L kubernetes.io/hostname | grep localblock | grep Released
```

输出示例：

```
local-pv-d6bf175b      1490Gi   RWO      Delete   Released   openshift-
storage/ocs-deviceset-0-data-0-6c5pw  localblock  2d22h   compute-1
```

9. 删除 PV。

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

**<pv-name>**

是 PV 的名称。

10. 替换旧设备，并使用新设备创建新的 OpenShift Container Platform PV。

- 使用要替换的设备登录到 OpenShift Container Platform 节点。在本例中，OpenShift Container Platform 节点是 **worker-0**。

```
$ oc debug node/worker-0
```

输出示例：

```
Starting pod/worker-0-debug ...
```

```
To use host binaries, run `chroot /host`
Pod IP: 192.168.88.21
If you don't see a command prompt, try pressing enter.
# chroot /host
```

- b. 使用前面标识的设备名称 **vdc** 记录要替换的 **/dev/disk**。

```
# ls -alh /mnt/local-storage/localblock
```

输出示例：

```
total 0
drwxr-xr-x. 2 root root 17 Nov 18 15:23 .
drwxr-xr-x. 3 root root 24 Nov 18 15:23 ..
lrwxrwxrwx. 1 root root 8 Nov 18 15:23 vdc -> /dev/vdc
```

- c. 查找 **LocalVolume** CR 的名称，并删除或注释掉要替换的设备 **/dev/disk**。

```
$ oc get -n openshift-local-storage localvolume
```

输出示例：

```
NAME      AGE
localblock 25h
```

```
# oc edit -n openshift-local-storage localvolume localblock
```

输出示例：

```
[...]
storageClassDevices:
- devicePaths:
# - /dev/vdc
storageClassName: localblock
volumeMode: Block
[...]
```

确保在编辑 CR 后保存更改。

11. 使用要替换的设备登录到 OpenShift Container Platform 节点，并删除旧的符号链接。

```
$ oc debug node/worker-0
```

输出示例：

```
Starting pod/worker-0-debug ...
To use host binaries, run `chroot /host`
Pod IP: 192.168.88.21
If you don't see a command prompt, try pressing enter.
# chroot /host
```

- a. 确定要替换的设备名称的旧 **符号链接**。在本例中，设备名称为 **vdc**。

```
# ls -alh /mnt/local-storage/localblock
```

输出示例：

```
total 0
drwxr-xr-x. 2 root root 17 Nov 18 15:23 .
drwxr-xr-x. 3 root root 24 Nov 18 15:23 ..
lrwxrwxrwx. 1 root root 8 Nov 18 15:23 vdc -> /dev/vdc
```

- b. 删除 **符号链接**。

```
# rm /mnt/local-storage/localblock/vdc
```

- c. 验证是否删除了 **symlink**。

```
# ls -alh /mnt/local-storage/localblock
```

输出示例：

```
total 0
drwxr-xr-x. 2 root root 6 Nov 18 17:11 .
drwxr-xr-x. 3 root root 24 Nov 18 15:23 ..
```

12. 使用新设备替换旧设备。

13. 重新登录到正确的 OpenShift Container Platform 节点，并确定新驱动器的设备名称。设备名称必须更改，除非您要重置同一设备。

```
# lsblk
```

输出示例：

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINT
vda	252:0	0	40G	0	disk	
vda1	252:1	0	4M	0	part	
vda2	252:2	0	384M	0	part	/boot
`vda4	252:4	0	39.6G	0	part	
`-coreos-luks-root-nocrypt	253:0	0	39.6G	0	dm	/sysroot
vdb	252:16	0	512B	1	disk	
vdd	252:32	0	256G	0	disk	

在这个示例中，新设备名称为 **vdd**。

14. 在新的 **/dev/disk** 可用后，您可以向 LocalVolume CR 添加新磁盘条目。

- a. 编辑 LocalVolume CR 并添加新 **/dev/disk**。

在本例中，新设备为 **/dev/vdd**。

```
# oc edit -n openshift-local-storage localvolume localblock
```

输出示例：

```
[...]
```

```

storageClassDevices:
- devicePaths:
# - /dev/vdc
- /dev/vdd
storageClassName: localblock
volumeMode: Block
[...]

```

确保在编辑 CR 后保存更改。

15. 验证是否有新 PV 处于 **Available** 状态且大小正确。

```
$ oc get pv | grep 256Gi
```

输出示例：

```

local-pv-1e31f771 256Gi RWO Delete Bound openshift-storage/ocs-deviceset-
localblock-2-data-0-6xhkf localblock 24h
local-pv-ec7f2b80 256Gi RWO Delete Bound openshift-storage/ocs-deviceset-
localblock-1-data-0-hr2fx localblock 24h
local-pv-8137c873 256Gi RWO Delete Available
localblock 32m

```

16. 为新设备创建一个新 OSD。  
部署新的 OSD。您需要重启 **rook-ceph-operator** 来强制协调 Operator。

- a. 识别 **rook-ceph-operator** 的名称。

```
$ oc get -n openshift-storage pod -l app=rook-ceph-operator
```

输出示例：

```

NAME                                READY STATUS RESTARTS AGE
rook-ceph-operator-85f6494db4-sg62v 1/1   Running 0     1d20h

```

- b. 删除 **rook-ceph-operator**。

```
$ oc delete -n openshift-storage pod rook-ceph-operator-85f6494db4-sg62v
```

输出示例：

```
pod "rook-ceph-operator-85f6494db4-sg62v" deleted
```

在本例中，rook-ceph-operator pod 名称为 **rook-ceph-operator-85f6494db4-sg62v**。

- c. 验证 **rook-ceph-operator** pod 是否已重启。

```
$ oc get -n openshift-storage pod -l app=rook-ceph-operator
```

输出示例：

```

NAME                                READY STATUS RESTARTS AGE
rook-ceph-operator-85f6494db4-wx9xx 1/1   Running 0     50s

```

在操作器重启后，创建新 OSD 可能需要几分钟时间。

#### 17. 删除 `ocs-osd-removal` 任务。

```
$ oc delete -n openshift-storage job ocs-osd-removal-job
```

输出示例：

```
job.batch "ocs-osd-removal-job" deleted
```



#### 注意

使用带有数据加密的外部密钥管理系统(KMS)时，可以从 Vault 服务器中删除旧的 OSD 加密密钥，因为它现在是孤立的密钥。

#### 验证步骤

##### 1. 验证是否有新的 OSD 正在运行。

```
$ oc get -n openshift-storage pods -l app=rook-ceph-osd
```

输出示例：

```
rook-ceph-osd-0-76d8fb97f9-mn8qz 1/1 Running 0 23m
rook-ceph-osd-1-7c99657cfb-jdzvz 1/1 Running 1 25h
rook-ceph-osd-2-5f9f6dfb5b-2mnw9 1/1 Running 0 25h
```

##### 2. 验证是否创建了新 PVC。

```
$ oc get -n openshift-storage pvc | grep localblock
```

输出示例：

```
ocs-deviceset-localblock-0-data-0-q4q6b Bound local-pv-8137c873 256Gi RWO
localblock 10m
ocs-deviceset-localblock-1-data-0-hr2fx Bound local-pv-ec7f2b80 256Gi RWO
localblock 1d20h
ocs-deviceset-localblock-2-data-0-6xhkf Bound local-pv-1e31f771 256Gi RWO
localblock 1d20h
```

##### 3. 可选：如果在集群中启用了集群范围的加密，请验证新 OSD 设备是否已加密。

###### a. 识别运行新 OSD pod 的节点。

```
$ oc get -n openshift-storage -o=custom-columns=NODE:.spec.nodeName pod/<OSD-
pod-name>
```

**<OSD-pod-name>**

是 OSD pod 的名称。

例如：

```
$ oc get -n openshift-storage -o=custom-columns=NODE:.spec.nodeName pod/rook-
ceph-osd-0-544db49d7f-qrgqm
```

输出示例：

```
NODE
compute-1
```

- b. 对于之前标识的每个节点，执行以下操作：
  - i. 创建调试 pod，并为所选主机打开 chroot 环境。

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

**<node name>**

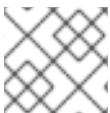
是节点的名称。

```
$ chroot /host
```

- ii. 检查 **ocs-device-set** 名称旁边的 **crypt** 关键字。

```
$ lsblk
```

4. 登录 OpenShift Web 控制台，并在存储下的 OpenShift Data Foundation 控制面板中检查状态卡。



### 注意

根据正在恢复的数据量，完整数据恢复可能需要更长的时间。

## 5.3. 在 IBM Z 或 LINUXONE 基础架构中替换操作或失败的存储设备

您可以使用新的小型计算机系统接口(SCSI)磁盘替换 IBM Z 或 LinuxONE 基础架构上的操作或失败存储设备。

IBM Z 或 LinuxONE 支持 SCSI FCP 磁盘逻辑单元 (SCSI 磁盘) 作为来自外部磁盘存储的持久性存储设备。您可以使用其 FCP 设备号、两个目标全球端口名称 (WWPN1 和 WWPN2) 以及逻辑单元号(LUN)识别 SCSI 磁盘。如需更多信息，请参阅

[https://www.ibm.com/support/knowledgecenter/SSB27U\\_6.4.0/com.ibm.zvm.v640.hcpa5/scsiover.html](https://www.ibm.com/support/knowledgecenter/SSB27U_6.4.0/com.ibm.zvm.v640.hcpa5/scsiover.html)

### 先决条件

- 确保数据具有弹性。
  - 在 OpenShift Web 控制台中，点 **Storage** → **Data Foundation**。
  - 点 **Storage Systems** 选项卡，然后点 **ocs-storagecluster-storagesystem**。
  - 在 **Block and File** 仪表板的 **Status** 卡中，在 **Overview** 选项卡中，验证 **Data Resiliency** 是否具有绿色勾号。

### 流程



11613

1. 列出所有磁盘。

```
$ lszdev
```

输出示例：

```
TYPE      ID
zfcplib  0.0.8204
zfcplib  0.0.8204:0x102107630b1b5060:0x4001402900000000 yes no  sda sg0
zfcplib  0.0.8204:0x500407630c0b50a4:0x3002b03000000000 yes yes sdb sg1
qeth     0.0.bdd0:0.0.bdd1:0.0.bdd2
generic-ccw 0.0.0009
yes no
```

SCSI 磁盘在 ID 以 **zfcplib** 表示，格式为 **<device-id>:<wwpn>:<lun-id>**。第一个磁盘用于操作系统。如果一个存储设备失败，您可以将其替换为新磁盘。

2. 删除磁盘。

在磁盘中运行以下命令，使用要替换的磁盘的 SCSI 磁盘标识符替换 **scsi-id**：

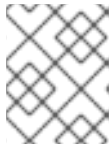
```
$ chzdev -d scsi-id
```

例如，以下命令删除设备 ID 为 **0.0.8204**、WWPN **0x500507630a0b50a4** 和 LUN **0x4002403000000000** 的一个磁盘：

```
$ chzdev -d 0.0.8204:0x500407630c0b50a4:0x3002b03000000000
```

3. 附加一个新的 SCSI 磁盘。

```
$ chzdev -e 0.0.8204:0x500507630b1b50a4:0x4001302a00000000
```



### 注意

新磁盘的设备 ID 必须与要替换的磁盘相同。新磁盘通过其 WWPN 和 LUN ID 进行标识。

4. 列出所有 FCP 设备以验证新磁盘是否已配置。

```
$ lszdev zfcplib
```

输出示例：

```
TYPE      ID
zfcplib  0.0.8204:0x102107630b1b5060:0x4001402900000000 yes no  sda sg0
zfcplib  0.0.8204:0x500507630b1b50a4:0x4001302a00000000 yes yes sdb sg1
```