



Red Hat OpenShift Container Storage 4.8

デバイスの置き換え

動作中または故障したデバイスを安全に交換するための手順

Red Hat OpenShift Container Storage 4.8 デバイスの置き換え

動作中または故障したデバイスを安全に交換するための手順

Enter your first name here. Enter your surname here.

Enter your organisation's name here. Enter your organisational division here.

Enter your email address here.

法律上の通知

Copyright © 2022 | You need to change the HOLDER entity in the en-US/Replacing_devices.ent file |.

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[®] is the registered trademark of Linus Torvalds in the United States and other countries.

Java[®] is a registered trademark of Oracle and/or its affiliates.

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

MySQL[®] is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js[®] 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[®] 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.

概要

本書では、Red Hat OpenShift Container Storage のストレージデバイスを安全に置き換える方法について説明します。

目次

多様性を受け入れるオープンソースの強化	3
RED HAT ドキュメントへのフィードバックの提供	4
はじめに	5
第1章 AMAZON WEB SERVICES への OPENSIFT CONTAINER STORAGE の動的プロビジョニング	6
1.1. AWS のユーザーによってプロビジョニングされるインフラストラクチャーで動作するストレージデバイスまたは失敗したストレージデバイスの置き換え	6
1.2. AWS のインストーラーでプロビジョニングされるインフラストラクチャーで動作するストレージデバイスまたは失敗したストレージデバイスの置き換え	6
第2章 VMWARE への OPENSIFT CONTAINER STORAGE の動的プロビジョニング	7
2.1. VMWAREインフラストラクチャーで動作するストレージデバイスまたは失敗したストレージデバイスの置き換え	7
第3章 RED HAT VIRTUALIZATION への OPENSIFT CONTAINER STORAGE の動的プロビジョニング	12
3.1. RED HAT VIRTUALIZATION のインストーラーでプロビジョニングされるインフラストラクチャーで動作するストレージデバイスまたは障害のあるストレージデバイスの置き換え	12
第4章 MICROSOFT AZURE への OPENSIFT CONTAINER STORAGE の動的プロビジョニング	17
4.1. AZURE のインストーラーでプロビジョニングされるインフラストラクチャーで動作するストレージデバイスまたは失敗したストレージデバイスの置き換え	17
第5章 ローカルストレージデバイスを使用した OPENSIFT CONTAINER STORAGE のデプロイ	18
5.1. ローカルストレージデバイスがサポートするクラスターで動作するストレージデバイスまたは障害のあるストレージデバイスの置き換え	18
5.2. IBM POWER SYSTEMS で動作するストレージデバイスまたは障害のあるストレージデバイスの置き換え	24
5.3. IBM Z または LINUXONE インフラストラクチャーで動作するストレージデバイスまたは失敗したストレージデバイスの置き換え	34

多様性を受け入れるオープンソースの強化

Red Hat では、コード、ドキュメント、Web プロパティにおける配慮に欠ける用語の置き換えに取り組んでいます。まずは、マスター (master)、スレーブ (slave)、ブラックリスト (blacklist)、ホワイトリスト (whitelist) の 4 つの用語の置き換えから始めます。この取り組みは膨大な作業を要するため、今後の複数のリリースで段階的に用語の置き換えを実施して参ります。詳細は、[弊社の CTO、Chris Wright のメッセージ](#) を参照してください。

RED HAT ドキュメントへのフィードバックの提供

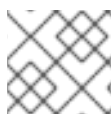
弊社のドキュメントについてのご意見をお聞かせください。ドキュメントの改善点があれば、ぜひお知らせください。フィードバックをお寄せいただくには、以下をご確認ください。

- 特定の部分についての簡単なコメントをお寄せいただく場合は、以下をご確認ください。
 1. ドキュメントの表示が **Multi-page HTML** 形式になっていて、ドキュメントの右上隅に **Feedback** ボタンがあることを確認してください。
 2. マウスカーソルで、コメントを追加する部分を強調表示します。
 3. そのテキストの下に表示される **Add Feedback** ポップアップをクリックします。
 4. 表示される手順に従ってください。
- より詳細なフィードバックを行う場合は、Bugzilla のチケットを作成します。
 1. [Bugzilla](#) の Web サイトに移動します。
 2. **Component** セクションで、**documentation** を選択します。
 3. **Description** フィールドに、ドキュメントの改善に関するご意見を記入してください。ドキュメントの該当部分へのリンクも記入してください。
 4. **Submit Bug** をクリックします。

はじめに

デプロイメントのタイプに応じて、以下のいずれかの手順を選択してストレージノードを置き換えることができます。

- AWS にデプロイされた動的に作成されたストレージクラスターについては、以下を参照してください。
 - 「AWS のユーザーによってプロビジョニングされるインフラストラクチャーで動作するストレージデバイスまたは失敗したストレージデバイスの置き換え」
 - 「AWS のインストーラーでプロビジョニングされるインフラストラクチャーで動作するストレージデバイスまたは失敗したストレージデバイスの置き換え」
- VMware にデプロイされた動的に作成されたストレージクラスターについては、「VMware インフラストラクチャーで動作するストレージデバイスまたは失敗したストレージデバイスの置き換え」を参照してください。
- Red Hat Virtualization にデプロイされた動的に作成されたストレージクラスターについては、「Red Hat Virtualization のインストーラーでプロビジョニングされるインフラストラクチャーで動作するストレージデバイスまたは障害のあるストレージデバイスの置き換え」を参照してください。
- Microsoft Azure にデプロイされた動的に作成されたストレージクラスターについては、「Azure のインストーラーでプロビジョニングされるインフラストラクチャーで動作するストレージデバイスまたは失敗したストレージデバイスの置き換え」を参照してください。
- ローカルストレージデバイスを使用してデプロイされたストレージクラスターについては、以下を参照してください。
 - 「ローカルストレージデバイスがサポートするクラスターで動作するストレージデバイスまたは障害のあるストレージデバイスの置き換え」
 - 「IBM Power Systems で動作するストレージデバイスまたは障害のあるストレージデバイスの置き換え」
 - 「IBM Z または LinuxONE インフラストラクチャーで動作するストレージデバイスまたは失敗したストレージデバイスの置き換え」



注記

OpenShift Container Storage は異なる OSD サイズをサポートしません。

第1章 AMAZON WEB SERVICES への OPENSIFT CONTAINER STORAGE の動的プロビジョニング

AWS ユーザーまたはインストーラーでプロビジョニングされたインフラストラクチャーで動作中または障害が発生したストレージデバイスを交換するには、それぞれのセクションのリンクをたどってください。

1.1. AWS のユーザーによってプロビジョニングされるインフラストラクチャーで動作するストレージデバイスまたは失敗したストレージデバイスの置き換え

AWS のユーザーによってプロビジョニングされるインフラストラクチャーの動的に作成されたストレージクラスターのデバイスを置き換える場合は、ストレージノードを置き換える必要があります。ノードを置き換える方法の詳細については、以下を参照してください。

- [ユーザーによってプロビジョニングされるインフラストラクチャーで動作する AWS ノードの置き換え](#)
- [ユーザーによってプロビジョニングされるインフラストラクチャーでの失敗した AWS ノードの置き換え](#)

1.2. AWS のインストーラーでプロビジョニングされるインフラストラクチャーで動作するストレージデバイスまたは失敗したストレージデバイスの置き換え

AWS のインストーラーでプロビジョニングされるインフラストラクチャーの動的に作成されたストレージクラスターのデバイスを置き換える場合は、ストレージノードを置き換える必要があります。ノードを置き換える方法の詳細については、以下を参照してください。

- [インストーラーでプロビジョニングされるインフラストラクチャーで動作する AWS ノードの置き換え](#)
- [インストーラーでプロビジョニングされるインフラストラクチャーでの失敗した AWS ノードの置き換え](#)

第2章 VMWARE への OPENSIFT CONTAINER STORAGE の動的プロビジョニング

VMware インフラストラクチャーで動作中または故障したストレージデバイスを交換するには、次のセクションの手順を実行します。

2.1. VMWARE インフラストラクチャーで動作するストレージデバイスまたは失敗したストレージデバイスの置き換え

VMware インフラストラクチャーに動的にデプロイされた OpenShift Container Storage 内の1つ以上の仮想マシンディスク (VMDK) を置き換える場合は、この手順のステップを実行します。この手順は、新規ボリュームで新規の Persistent Volume Claim (永続ボリューム要求、PVC) を作成し、古いオブジェクトストレージデバイス (OSD) を削除するのに役立ちます。

前提条件

- データに耐久性があることを確認します。
 - OpenShift Web コンソールで、**Storage → Overview** にナビゲートします。
 - **Status** カードの **Block and File** で、**Data Resiliency** に緑色のチェックマークが付いていることを確認します。

手順

1. 置き換える必要がある OSD と、その OSD がスケジュールされている OpenShift Container Platform ノードを特定します。

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

出力例:

```
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>
```

この例では、**rook-ceph-osd-0-6d77d6c7c6-m8xj6** を置き換える必要があり、**compute-2** は OSD がスケジュールされる OpenShift Container platform ノードです。



注記

置き換える OSD が正常であれば、Pod のステータスは **Running** になります。

2. 置き換えられる OSD の OSD デプロイメントをスケールダウンします。
OSD を置き換えるたびに、**osd_id_to_remove** パラメーターを OSD ID で更新してこの手順を繰り返します。

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

ここで、**osd_id_to_remove** は **rook-ceph-osd** プレフィックスの直後にくる Pod 名の整数です。この例では、デプロイメント名は **rook-ceph-osd-0** です。

出力例:

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

3. **rook-ceph-osd** Pod が停止していることを確認します。

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

出力例:

```
No resources found.
```



注記

rook-ceph-osd Pod が **terminating** 状態にある場合は、**force** オプションを使用して Pod を削除します。

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

出力例:

```
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. クラスタから古い OSD を削除して、新しい OSD を追加します。

- a. 古い **ocs-osd-removal** ジョブを削除します。

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

出力例:

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

- b. **openshift-storage** プロジェクトを変更します。

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

- c. クラスタから以前の OSD を削除します。

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

コマンドにコンマ区切りの OSD ID を追加して、複数の OSD を削除できます。(例: FAILED_OSD_IDS=0,1,2)

**警告**

この手順により、OSD はクラスターから完全に削除されます。**osd_id_to_remove** の正しい値が指定されていることを確認します。

5. **ocs-osd-removal** Pod のステータスをチェックして、OSD が正常に削除されたことを確認します。**Completed** のステータスで、OSD の削除ジョブが正常に完了したことを確認します。

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

**注記**

ocs-osd-removal が失敗し、Pod が予想される **Completed** の状態にない場合、追加のデバッグのために Pod ログを確認します。以下は例になります。

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

6. 暗号化がインストール時に有効にされている場合は、それぞれの OpenShift Container Storage ノードから削除された OSD デバイスから **dm-crypt** で管理される **device-mapper** マッピングを削除します。

- a. **ocs-osd-removal-job** Pod のログから置き換えられた OSD の PVC 名を取得します。

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

以下は例になります。

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

- b. 前のステップで識別されたノードごとに、以下を実行します。

- i. **デバッグ** Pod を作成し、ストレージノードのホストに対して **chroot** を作成します。

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

- ii. 直前の手順で特定された PVC 名に基づいて関連するデバイス名を検索します。

```
sh-4.4# dmsetup ls| grep <pvc name>
ocs-deviceset-xxx-xxx-xxx-xxx-block-dmccrypt (253:0)
```

- iii. マップ済みデバイスを削除します。

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



注記

権限が十分でないため、コマンドがスタックした場合には、以下のコマンドを実行します。

- **CTRL+Z** を押して上記のコマンドを終了します。
- スタックしているプロセスの PID を見つけます。

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

- **kill** コマンドを使用してプロセスを終了します。

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

- デバイス名が削除されていることを確認します。

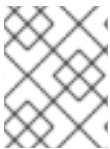
```
$ dmsetup ls
```

7. **ocs-osd-removal** ジョブを削除します。

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

出力例:

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



注記

データ暗号化で外部の鍵管理システム (KMS) を使用する場合は、古い OSD 暗号化キーは孤立したキーであるために Vault サーバーから削除できます。

検証手順

1. 新しい OSD が実行していることを確認します。

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

出力例:

```
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. **Bound** 状態の新しい PVC が作成されていることを確認します。

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

出力例:

```
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. (オプション) クラスタでクラスタ全体の暗号化が有効な場合には、新規 OSD デバイスが暗号化されていることを確認します。
 - a. 新規 OSD Pod が実行しているノードを特定します。

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

以下は例になります。

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

- b. 前のステップで識別されたノードごとに、以下を実行します。
 - i. **debug** Pod を作成し、選択したホストの **chroot** 環境を開きます。

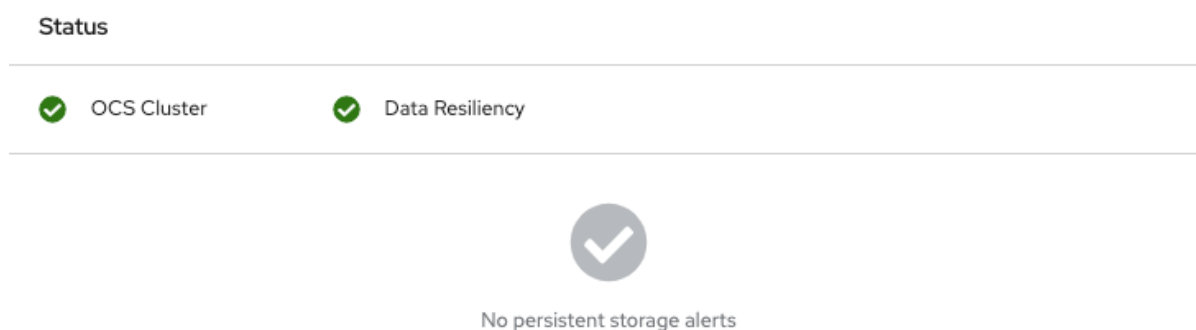
```
$ oc debug node/<node name>
$ chroot /host
```

- ii. **lsblk** を実行し、**ocs-deviceset** 名の横にある **crypt** キーワードを確認します。

```
$ lsblk
```

4. OpenShift Web コンソールにログインし、ストレージダッシュボードを表示します。

図2.1 デバイスの置き換え後の OpenShift Container Platform ストレージダッシュボードの OSD ステータス



第3章 RED HAT VIRTUALIZATION への OPENSIFT CONTAINER STORAGE の動的プロビジョニング

Red Hat Virtualization インストーラーでプロビジョニングされたインフラストラクチャーで動作中または故障したストレージデバイスを交換するには、次のセクションの手順を実行します。

3.1. RED HAT VIRTUALIZATION のインストーラーでプロビジョニングされるインフラストラクチャーで動作するストレージデバイスまたは障害のあるストレージデバイスの置き換え

OpenShift Container Storage にデプロイされた Red Hat 仮想化インフラストラクチャーの1つ以上の仮想マシンディスク (VMDK) を交換する場合は、この手順のステップを実行します。この手順は、新規ボリュームで新規の Persistent Volume Claim (永続ボリューム要求、PVC) を作成し、古いオブジェクトストレージデバイス (OSD) を削除するのに役立ちます。

前提条件

- データに耐久性があることを確認します。
 - OpenShift Web コンソールで、**Storage → Overview** にナビゲートします。
 - **Status** カードの **Block and File** で、**Data Resiliency** に緑色のチェックマークが付いていることを確認します。

手順

1. 置き換える必要がある OSD と、その OSD がスケジュールされている OpenShift Container Platform ノードを特定します。

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

出力例:

```
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>
```

この例では、**rook-ceph-osd-0-6d77d6c7c6-m8xj6** を置き換える必要があり、**compute-2** は OSD がスケジュールされる OpenShift Container platform ノードです。



注記

置き換える OSD が正常であれば、Pod のステータスは **Running** になります。

2. 置き換えられる OSD の OSD デプロイメントをスケールダウンします。OSD を置き換えるたびに、**osd_id_to_remove** パラメーターを OSD ID で更新してこの手順を繰り返します。


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

ここで、**osd_id_to_remove** は **rook-ceph-osd** プレフィックスの直後にくる Pod 名の整数です。この例では、デプロイメント名は **rook-ceph-osd-0** です。

出力例:

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

3. **rook-ceph-osd** Pod が終了していることを確認します。

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

出力例:

```
No resources found.
```



注記

rook-ceph-osd Pod が **terminating** 状態にある場合は、**force** オプションを使用して Pod を削除します。

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

出力例:

```
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. クラスタから古い OSD を削除して、新しい OSD を追加します。

- a. 古い **ocs-osd-removal** ジョブを削除します。

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

出力例:

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

- b. **openshift-storage** プロジェクトを変更します。

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

- c. クラスタから以前の OSD を削除します。

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

コマンドにコンマ区切りの OSD ID を追加して、複数の OSD を削除できます。(例: FAILED_OSD_IDS=0,1,2)

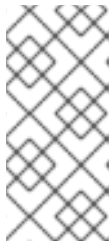


警告

この手順により、OSD はクラスターから完全に削除されます。**osd_id_to_remove** の正しい値が指定されていることを確認します。

5. **ocs-osd-removal** Pod のステータスをチェックして、OSD が正常に削除されたことを確認します。**Completed** のステータスで、OSD の削除ジョブが正常に完了したことを確認します。

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



注記

ocs-osd-removal が失敗し、Pod が予想される **Completed** の状態にない場合、追加のデバッグのために Pod ログを確認します。以下は例になります。

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

6. 暗号化がインストール時に有効にされている場合は、それぞれの OpenShift Container Storage ノードから削除された OSD デバイスから **dm-crypt** で管理される **device-mapper** マッピングを削除します。

- a. **ocs-osd-removal-job** Pod のログから置き換えられた OSD の PVC 名を取得します。

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

以下は例になります。

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

- b. 前のステップで識別されたノードごとに、以下を実行します。

- i. **デバッグ** Pod を作成し、ストレージノードのホストに対して **chroot** を作成します。

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

- ii. 直前の手順で特定された PVC 名に基づいて関連するデバイス名を検索します。

```
sh-4.4# dmsetup ls| grep <pvc name>
ocs-deviceset-xxx-xxx-xxx-xxx-block-dmccrypt (253:0)
```

- iii. マップ済みデバイスを削除します。

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

注記

権限が十分でないため、コマンドがスタックした場合には、以下のコマンドを実行します。

- **CTRL+Z** を押して上記のコマンドを終了します。
- スタックしているプロセスの PID を見つけます。

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

- **kill** コマンドを使用してプロセスを終了します。

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

- デバイス名が削除されていることを確認します。

```
$ dmsetup ls
```

7. **ocs-osd-removal** ジョブを削除します。

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

出力例:

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

注記

データ暗号化で外部の鍵管理システム (KMS) を使用する場合は、古い OSD 暗号化キーは孤立したキーであるために Vault サーバーから削除できます。

検証手順

1. 新しい OSD が実行していることを確認します。

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

出力例:

```
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. **Bound** 状態の新しい PVC が作成されていることを確認します。

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

3. (オプション) クラスタでクラスタ全体の暗号化が有効な場合には、新規 OSD デバイスが暗号化されていることを確認します。

- a. 新規 OSD Pod が実行しているノードを特定します。

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

以下は例になります。

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

- b. 前のステップで識別されたノードごとに、以下を実行します。

- i. **debug** Pod を作成し、選択したホストの **chroot** 環境を開きます。

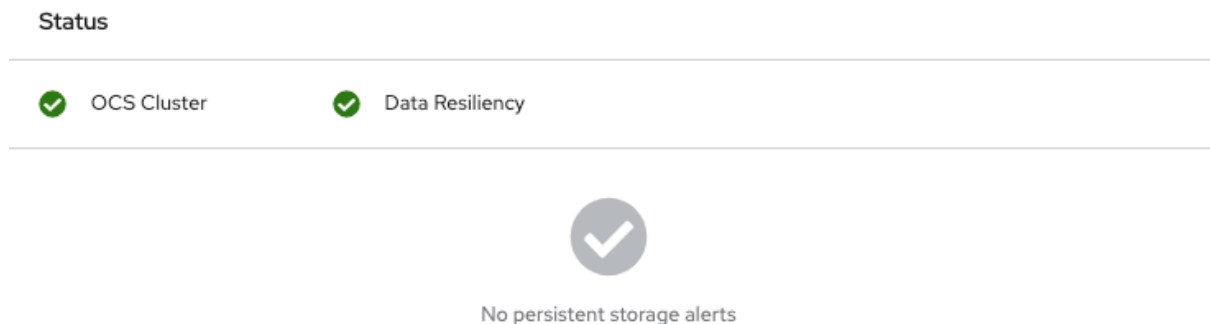
```
$ oc debug node/<node name>  
$ chroot /host
```

- ii. **lsblk** を実行し、**ocs-deviceset** 名の横にある **crypt** キーワードを確認します。

```
$ lsblk
```

4. OpenShift Web コンソールにログインし、ストレージダッシュボードを表示します。

図3.1 デバイスの置き換え後の OpenShift Container Platform ストレージダッシュボードの OSD ステータス



第4章 MICROSOFT AZURE への OPENSIFT CONTAINER STORAGE の動的プロビジョニング

Microsoft Azure インストーラーでプロビジョニングされたインフラストラクチャーで動作中または故障したストレージデバイスを交換するには、次のセクションの手順を実行します。

4.1. AZURE のインストーラーでプロビジョニングされるインフラストラクチャーで動作するストレージデバイスまたは失敗したストレージデバイスの置き換え

Azure のインストーラーでプロビジョニングされるインフラストラクチャーの動的に作成されたストレージクラスターのデバイスを置き換える必要がある場合は、ストレージノードを置き換える必要があります。ノードを置き換える方法は、以下を参照してください。

- [Azure のインストーラーでプロビジョニングされるインフラストラクチャーで動作するノードの置き換え](#)
- [Azure のインストーラーでプロビジョニングされるインフラストラクチャーでの失敗したノードの置き換え](#)

第5章 ローカルストレージデバイスを使用した OPENSIFT CONTAINER STORAGE のデプロイ

5.1. ローカルストレージデバイスがサポートするクラスターで動作するストレージデバイスまたは障害のあるストレージデバイスの置き換え

以下のインフラストラクチャーでローカルストレージデバイスを使用してデプロイされた OpenShift Container Storage のオブジェクトストレージデバイス (OSD) を置き換えることができます。

- ベアメタル
- VMware
- Red Hat Virtualization

基盤のストレージデバイスを1つまたは複数置き換える必要がある場合は、この手順を使用します。

前提条件

- Red Hat は、交換用デバイスを、交換するデバイスと同様のインフラストラクチャーおよびリソースで構成することを推奨します。
- 以前のバージョンから OpenShift Container Storage バージョン 4.8 にアップグレードし、**LocalVolumeDiscovery** および **LocalVolumeSet** オブジェクトを作成していない場合は、[Post-update configuration changes for clusters backed by local storage](#) に説明されている手順を行います。
- データに耐久性があることを確認します。
 - OpenShift Web コンソールで、**Storage → Overview** にナビゲートします。
 - **Status** カードの **Block and File** で、**Data Resiliency** に緑色のチェックマークが付いていることを確認します。

手順

1. 関連するワーカーノードから基礎となるストレージデバイスを削除します。
2. 関連する OSD Pod が CrashLoopBackOff 状態になったことを確認します。
置き換える必要がある OSD と、その OSD がスケジュールされている OpenShift Container Platform ノードを特定します。

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

出力例:

```
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>
```

この例では、**rook-ceph-osd-0-6d77d6c7c6-m8xj6** を置き換える必要があり、**compute-2** は OSD がスケジュールされる OpenShift Container platform ノードです。

3. 置き換えられる OSD の OSD デプロイメントをスケールダウンします。

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

ここで、**osd_id_to_remove** は **rook-ceph-osd** プレフィックスの直後にくる Pod 名の整数です。この例では、デプロイメント名は **rook-ceph-osd-0** です。

出力例:

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

4. **rook-ceph-osd** Pod が停止していることを確認します。

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

出力例:

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

注記

rook-ceph-osd Pod が数分以上 **terminating** 状態である場合は、**force** オプションを使用して Pod を削除します。

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

出力例:

```
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. 新規 OSD を追加できるようにクラスターから古い OSD を削除します。

- a. 古い **ocs-osd-removal** ジョブを削除します。

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

出力例:

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

- b. **openshift-storage** プロジェクトを変更します。

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

- c. クラスタから以前の OSD を削除します。

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

コマンドにコンマ区切りの OSD ID を追加して、複数の OSD を削除できます。(例: FAILED_OSD_IDS=0,1,2)



警告

この手順により、OSD はクラスタから完全に削除されます。**osd_id_to_remove** の正しい値が指定されていることを確認します。

6. **ocs-osd-removal** Pod のステータスをチェックして、OSD が正常に削除されたことを確認します。**Completed** のステータスで、OSD の削除ジョブが正常に完了したことを確認します。

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



注記

ocs-osd-removal が失敗し、Pod が予想される **Completed** の状態にない場合、追加のデバッグのために Pod ログを確認します。以下は例になります。

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

7. 暗号化がインストール時に有効にされている場合は、それぞれの OpenShift Container Storage ノードから削除された OSD デバイスから **dm-crypt** で管理される **device-mapper** マッピングを削除します。

- a. **ocs-osd-removal-job** Pod のログから、置き換えられた OSD の PVC 名を取得します。

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

以下は例になります。

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

- b. 手順 #1 で特定されたノードごとに、以下を実行します。

- i. デバッグ Pod を作成し、ストレージノードのホストに対して **chroot** を作成します。

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

- ii. 直前の手順で特定された PVC 名に基づいて関連するデバイス名を検索します。


```
sh-4.4# dmsetup ls| grep <pvc name>
ocs-deviceset-xxx-xxx-xxx-xxx-block-dmccrypt (253:0)
```

- iii. マップ済みデバイスを削除します。

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



注記

権限が十分でないため、コマンドがスタックした場合には、以下のコマンドを実行します。

- **CTRL+Z** を押して上記のコマンドを終了します。

- スタックしたプロセスの PID を検索します。

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

- **kill** コマンドを使用してプロセスを終了します。

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

- デバイス名が削除されていることを確認します。

```
$ dmsetup ls
```

8. コマンドで削除する必要のある永続ボリューム (PV) を検索します。

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

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

9. 永続ボリュームを削除します。

```
$ oc delete pv local-pv-d6bf175b
```

10. 物理的に新規デバイスをノードに追加します。

11. 以下のコマンドを使用して、**deviceInclusionSpec** に一致するデバイスの永続ボリュームのプロビジョニングを追跡します。永続ボリュームをプロビジョニングするのに数分かかる場合があります。

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

出力例:

```
[...]
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

```

永続ボリュームがプロビジョニングされると、新しい OSD Pod がプロビジョニングボリューム用に自動作成されます。

12. **ocs-osd-removal** ジョブを削除します。

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

出力例:

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



注記

データ暗号化で外部の鍵管理システム (KMS) を使用する場合は、古い OSD 暗号化キーは孤立したキーであるために Vault サーバーから削除できます。

検証手順

1. 新しい OSD が実行されていることを確認します。

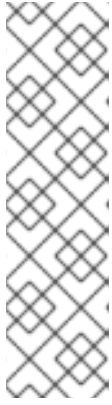
```
$ oc get -n openshift-storage pods -l app=rook-ceph-osd
```

出力例:

```

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

```



注記

数分後に新規 OSD が **Running** と表示されない場合は、**rook-ceph-operator** Pod を再起動して強制的に調整を行います。

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

出力例:

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

2. 新規 PVC が作成されていることを確認します。

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

出力例:

```
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. (オプション) クラスタでクラスタ全体の暗号化が有効な場合には、新規 OSD デバイスが暗号化されていることを確認します。
 - a. 新規 OSD Pod が実行しているノードを特定します。

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

以下は例になります。

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

- b. 直前の手順で特定されたノードごとに、以下を実行します。
 - i. デバッグ Pod を作成し、選択したホストの chroot 環境を開きます。

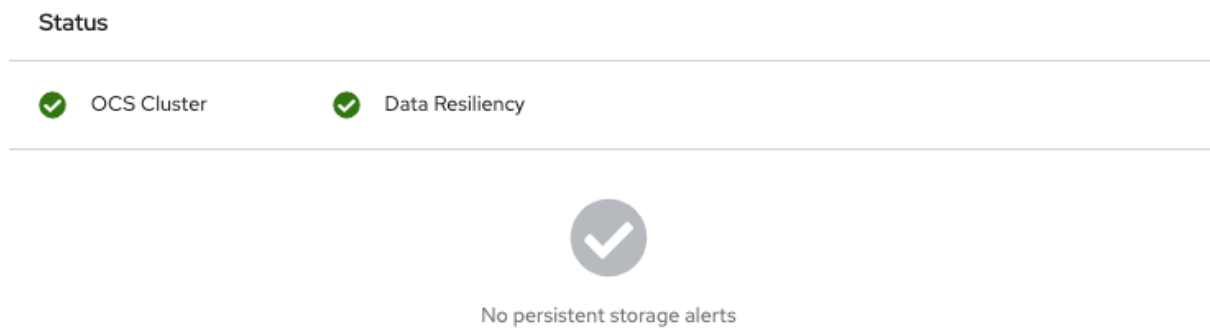
```
$ oc debug node/<node name>
$ chroot /host
```

- ii. 「lsblk」を実行し、**ocs-deviceset** 名の横にある「crypt」キーワードを確認します。

```
$ lsblk
```

4. OpenShift Web コンソールにログインし、ストレージダッシュボードで OSD のステータスを確認します。

図5.1 デバイスの置き換え後の OpenShift Container Platform ストレージダッシュボードの OSD ステータス



注記

データの完全復旧には、復元されるデータ量により、時間がかかる場合があります。

5.2. IBM POWER SYSTEMS で動作するストレージデバイスまたは障害のあるストレージデバイスの置き換え

IBM Power Systems でローカルストレージデバイスを使用してデプロイされた OpenShift Container Storage のオブジェクトストレージデバイス (OSD) を置き換えることができます。基礎となるストレージデバイスを置き換える必要がある場合は、この手順を使用します。

前提条件

- Red Hat は、交換用デバイスを、交換するデバイスと同様のインフラストラクチャーおよびリソースで構成することを推奨します。
- 以前のバージョンから OpenShift Container Storage 4.8 にアップグレードし、**LocalVolumeDiscovery** オブジェクトを作成していない場合は、[Post-update configuration changes for clusters backed by local storage](#) に説明されている手順を行います。
- データに耐久性があることを確認します。
 - OpenShift Web コンソールで、**Storage → Overview** にナビゲートします。
 - **Status** カードの **Block and File** で、**Data Resiliency** に緑色のチェックマークが付いていることを確認します。

手順

1. 置き換える必要がある OSD と、その OSD がスケジュールされている OpenShift Container Platform ノードを特定します。

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

出力例:

```
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>
```

この例では、**rook-ceph-osd-0-86bf8cdc8-4nb5t** を置き換える必要があり、**worker-0** は OSD がスケジュールされる RHOCP ノードです。



注記

置き換える OSD が正常である場合、Pod のステータスは **Running** になります。

- 置き換えられる OSD の OSD デプロイメントをスケールダウンします。

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

ここで、**osd_id_to_remove** は **rook-ceph-osd** プレフィックスの直後にくる Pod 名の整数です。この例では、デプロイメント名は **rook-ceph-osd-0** です。

出力例:

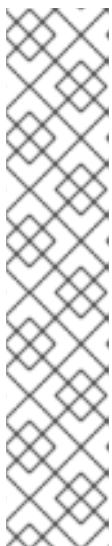
```
deployment.apps/rook-ceph-osd-0 scaled
```

- rook-ceph-osd** Pod が停止していることを確認します。

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

出力例:

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



注記

rook-ceph-osd Pod が **terminating** 状態にある場合は、**force** オプションを使用して Pod を削除します。

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

出力例:

```
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
```

- 新規 OSD を追加できるようにクラスターから古い OSD を削除します。
 - 置き換える OSD に関連付けられた **DeviceSet** を特定します。

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

出力例:

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

この例では、PVC 名は **ocs-deviceset-localblock-0-data-0-64xjl** です。

- b. PVC に関連付けられた PV を特定します。

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

ここで、**x**、**y**、および **pvc-suffix** は、ステップ 4(a) で特定された **DeviceSet** の値です。

出力例:

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

この例では、関連付けられた PV は **local-pv-8137c873** です。

- c. 置き換えるデバイスの名前を特定します。

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

ここで、**pvc-suffix** は、前のステップで特定された PV 名の値です。

出力例:

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

この例では、デバイス名は **vdc** です。

- d. 置き換える OSD に関連付けられた **prepare-pod** を特定します。

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

ここで、**x**、**y**、および **pvc-suffix** は、直前の手順で特定された **DeviceSet** の値です。

出力例:

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

この例では、**prepare-pod** の名前は **rook-ceph-osd-prepare-ocs-deviceset-localblock-0-data-0-64knzkc** です。

- e. 古い **ocs-osd-removal** ジョブを削除します。

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

出力例:

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

- f. **openshift-storage** プロジェクトを変更します。

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

- g. クラスターから以前の OSD を削除します。

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

コマンドにコンマ区切りの OSD ID を追加して、複数の OSD を削除できます。(例: FAILED_OSD_IDS=0,1,2)



警告

この手順により、OSD はクラスターから完全に削除されます。**osd_id_to_remove** の正しい値が指定されていることを確認します。

5. **ocs-osd-removal** Pod のステータスをチェックして、OSD が正常に削除されたことを確認します。**Completed** のステータスで、OSD の削除ジョブが正常に完了したことを確認します。

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



注記

ocs-osd-removal が失敗し、Pod が予想される **Completed** の状態にない場合、追加のデバッグのために Pod ログを確認します。以下は例になります。

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

6. 暗号化がインストール時に有効にされている場合は、それぞれの OpenShift Container Storage ノードから削除された OSD デバイスから **dm-crypt** で管理される **device-mapper** マッピングを削除します。

- a. **ocs-osd-removal-job** Pod のログから、置き換えられた OSD の PVC 名を取得します。

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

以下は例になります。

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

b. 手順 #1 で特定されたノードごとに、以下を実行します。

i. デバッグ Pod を作成し、ストレージノードのホストに対して **chroot** を作成します。

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

ii. 直前の手順で特定された PVC 名に基づいて関連するデバイス名を検索します。

```
sh-4.4# dmsetup ls| grep <pvc name>
ocs-deviceset-xxx-xxx-xxx-xxx-block-dmccrypt (253:0)
```

iii. マップ済みデバイスを削除します。

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

注記

権限が十分でないため、コマンドがスタックした場合には、以下のコマンドを実行します。

- **CTRL+Z** を押して上記のコマンドを終了します。
- スタックしたプロセスの PID を検索します。

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

- **kill** コマンドを使用してプロセスを終了します。

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

- デバイス名が削除されていることを確認します。

```
$ dmsetup ls
```

7. 古いデバイスを置き換え、新規デバイスを使用して新規の OpenShift Container Platform PV を作成します。

a. 置き換えるデバイスで OpenShift Container Platform ノードにログインします。この例では、OpenShift Container Platform ノードは **worker-0** です。

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

出力例:

```
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. 先に特定したデバイス名 **vdc** を使用して置き換える **/dev/disk** の内容を記録します。

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

出力例:

```
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. **LocalVolume** CR の名前を見つけ、置き換えるデバイス **/dev/disk** を削除またはコメントアウトします。

```
$ oc get -n openshift-local-storage localvolume
NAME      AGE
localblock 25h
```

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

出力例:

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

CR の編集後に変更を保存するようにしてください。

8. 置き換えるデバイスで OpenShift Container Platform ノードにログインし、古い **symlink** を削除します。

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

出力例:

```
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. 置き換えるデバイス名の古い **symlink** を特定します。この例では、デバイス名は **vdc** です。

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

出力例:

```
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. **symlink** を削除します。

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

- c. **symlink** が削除されていることを確認します。

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

出力例:

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



重要

OpenShift Container Storage 4.5 以降の新規デプロイメントでは、LVM が使用されていないため、**ceph-volume** raw モードが動作します。そのため、追加の検証は不要であり、次のステップに進むことができます。

9. 以下のコマンドを使用して削除する必要のある永続ボリューム (PV) を検索します。

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

```
local-pv-8137c873      256Gi      RWO      Delete      Released      openshift-
storage/ocs-deviceset-localblock-0-data-0-64xjl  localblock  2d22h      worker-0
```

10. 永続ボリュームを削除します。

```
$ oc delete pv local-pv-8137c873
```

11. デバイスを新しいデバイスに置き換えます。

12. 正しい OpenShift Container Platform ノードにログインし、新規ドライブのデバイス名を特定します。同じデバイスを使用しない限り、デバイス名は変更する必要があります。

```
# lsblk
```

出力例:

```
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
```

この例では、新しいデバイス名は **vdd** です。

13. 新規の **/dev/disk** が利用可能になると、新規ディスクエントリーを LocalVolume CR に追加できます。
 - a. LocalVolume CR を編集し、新規の **/dev/disk** を追加します。この例では、新しいデバイスは **/dev/vdd** です。

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

出力例:

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

CR の編集後に変更を保存するようにしてください。

14. 新規 PV が **Available** 状態にあり、正しいサイズであることを確認します。

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

出力例:

```
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
```

15. 新規デバイス用に新規 OSD を作成します。
 - a. 置き換えられる OSD のデプロイメントを削除します。

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

出力例:

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

- b. **rook-ceph-operator** を再起動して Operator の調整を強制的に実行して新規 OSD をデプロイします。
 - i. **rook-ceph-operator** の名前を特定します。

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

出力例:

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

- ii. **rook-ceph-operator** を削除します。

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

出力例:

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

この例では、rook-ceph-operator Pod 名は **rook-ceph-operator-85f6494db4-sg62v** です。

- iii. **rook-ceph-operator** Pod が再起動していることを確認します。

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

出力例:

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

新規 OSD の作成には、Operator が再起動するまでに数分かかる場合があります。

16. **ocs-osd-removal** ジョブを削除します。

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

出力例:

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



注記

データ暗号化で外部の鍵管理システム (KMS) を使用する場合は、古い OSD 暗号化キーは孤立したキーであるために Vault サーバーから削除できます。

検証手順

1. 新しい OSD が実行されていることを確認します。

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

出力例:

```
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. 新規 PVC が作成されていることを確認します。

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

出力例:

```
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. (オプション) クラスタでクラスタ全体の暗号化が有効な場合には、新規 OSD デバイスが暗号化されていることを確認します。
- a. 新規 OSD Pod が実行しているノードを特定します。

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

以下は例になります。

```
oc get -o=custom-columns=NODE:.spec.nodeName pod/rook-ceph-osd-0-76d8fb97f9-
mn8qz
```

- b. 直前の手順で特定されたノードごとに、以下を実行します。

- i. デバッグ Pod を作成し、選択したホストの chroot 環境を開きます。

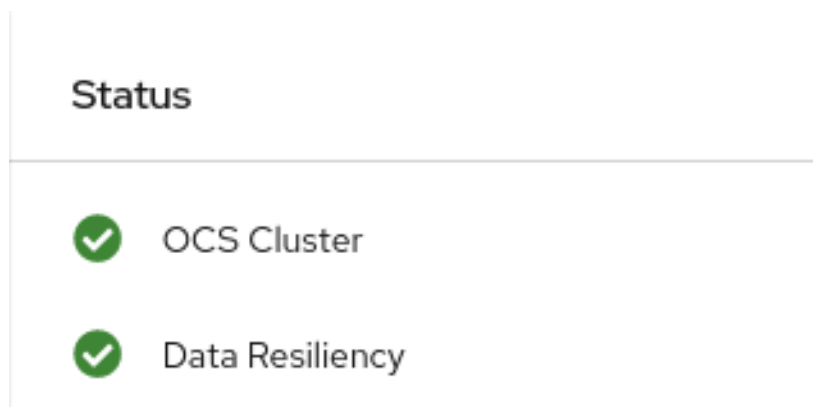
```
$ oc debug node/<node name>
$ chroot /host
```

- ii. 「lsblk」を実行し、**ocs-deviceset** 名の横にある「crypt」キーワードを確認します。

```
$ lsblk
```

4. OpenShift Web コンソールにログインし、ストレージダッシュボードを表示します。

図5.2 デバイスの置き換え後の OpenShift Container Platform ストレージダッシュボードの OSD ステータス





注記

データの完全復旧には、復元されるデータ量により、時間がかかる場合があります。

5.3. IBM Z または LINUXONE インフラストラクチャーで動作するストレージデバイスまたは失敗したストレージデバイスの置き換え

IBM Z または LinuxONE インフラストラクチャーの動作するストレージデバイスまたは障害のあるストレージデバイスを、新規 SCSI ディスクに置き換えることができます。

IBM Z または LinuxONE は、外部ディスクストレージからの永続ストレージとして SCSI FCP ディスク論理ユニット (SCSI ディスク) に対応します。SCSI ディスクは、FCP デバイス番号、2つのターゲットのワールドワイドポート名 (WWPN1 および WWPN2) と、論理ユニット番号 (LUN) を使用して識別できます。詳細は、次を参照してください

い。 https://www.ibm.com/support/knowledgecenter/SSB27U_6.4.0/com.ibm.zvm.v640.hcpa5/scsiover.h

前提条件

- データに耐久性があることを確認します。
 - OpenShift Web コンソールで、**Storage → Overview** にナビゲートします。
 - **Status** カードの **Block and File** で、**Data Resiliency** に緑色のチェックマークが付いていることを確認します。

手順

1. 以下のコマンドを使用してすべてのディスクを一覧表示します。

```
$ lszdev
```

出力例:

```
TYPE      ID
zfcplib  0.0.8204                yes yes
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                yes no  encbdd0
generic-ccw 0.0.0009                yes no
```

SCSI ディスクは、ID セクションの **<device-id>:<wwpn>:<lun-id>** 構造で **zfcplib** として表されます。最初のディスクはオペレーティングシステムに使用されます。1つのストレージデバイスに障害が発生した場合は、これを新しいディスクに置き換えることができます。

2. ディスクを削除します。
ディスクで以下のコマンドを実行し、**scsi-id** を、置き換えるディスクの SCSI ディスク識別子に置き換えます。

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

たとえば、以下のコマンドはデバイス ID **0.0.8204**、WWPN **0x500507630a0b50a4**、および LUN **0x4002403000000000** のディスクを1つ削除します。

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

- 以下のコマンドを使用して新規 SCSI ディスクを追加します。

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



注記

新規ディスクのデバイス ID は、置き換えるディスクと同じである必要があります。新規ディスクは、WWPN および LUN ID で識別されます。

- すべての FCP デバイスを一覧表示して、新規ディスクが設定されていることを確認します。

```
$ lszdev zfcplun
TYPE      ID                                     ON PERS NAMES
zfcplun   0.0.8204:0x102107630b1b5060:0x4001402900000000  yes no   sda sg0
zfcplun   0.0.8204:0x500507630b1b50a4:0x4001302a00000000  yes yes  sdb sg1
```