



# Red Hat OpenShift Data Foundation 4.16

## Microsoft Azure を使用した OpenShift Data Foundation のデプロイ

Microsoft Azure を使用した OpenShift Data Foundation のデプロイ手順



# Red Hat OpenShift Data Foundation 4.16 Microsoft Azure を使用した OpenShift Data Foundation のデプロイ

---

Microsoft Azure を使用した OpenShift Data Foundation のデプロイ手順

## 法律上の通知

Copyright © 2024 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.

## 概要

Microsoft Azure 上の Red Hat OpenShift Container Platform を使用して Red Hat OpenShift Data Foundation をインストールおよび管理する方法について、このドキュメントをお読みください。

---

## 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 (英語のみ) .....	4
はじめに .....	5
第1章 OPENSIFT DATA FOUNDATION のデプロイの準備 .....	6
第2章 MICROSOFT AZURE への OPENSIFT DATA FOUNDATION のデプロイ .....	7
2.1. RED HAT OPENSIFT DATA FOUNDATION OPERATOR のインストール .....	7
2.2. トークン認証方法を使用した KMS を使用したクラスター全体の暗号化の有効化 .....	9
2.3. KUBERNETES 認証方式を使用した KMS でのクラスター全体の暗号化の有効化 .....	9
2.4. OPENSIFT DATA FOUNDATION クラスターの作成 .....	12
第3章 OPENSIFT DATA FOUNDATION デプロイメントの確認 .....	18
3.1. POD の状態の確認 .....	18
3.2. OPENSIFT DATA FOUNDATION クラスターが正常であることを確認 .....	20
3.3. MULTICLOUD OBJECT GATEWAY が正常であることを確認 .....	20
3.4. 特定のストレージクラスが存在することを確認 .....	20
第4章 スタンドアロンの MULTICLOUD OBJECT GATEWAY のデプロイ .....	22
4.1. RED HAT OPENSIFT DATA FOUNDATION OPERATOR のインストール .....	22
4.2. スタンドアロンの MULTICLOUD OBJECT GATEWAY の作成 .....	23
第5章 OPENSIFT DATA FOUNDATION トポロジーの表示 .....	27
第6章 OPENSIFT DATA FOUNDATION のアンインストール .....	28
6.1. 内部モードでの OPENSIFT DATA FOUNDATION のアンインストール .....	28



## 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、用語の置き換えは、今後の複数のリリースにわたって段階的に実施されます。詳細は、[Red Hat CTO である Chris Wright のメッセージ](#) をご覧ください。

## RED HAT ドキュメントへのフィードバック (英語のみ)

Red Hat ドキュメントに対する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。

フィードバックを送信するには、Bugzilla 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向けたご提案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載してください。
4. **Submit Bug** をクリックします。



## はじめに

Red Hat OpenShift Data Foundation では、既存の Red Hat OpenShift Container Platform (RHOCP) Azure クラスターでのデプロイメントをサポートします。



### 注記

Microsoft Azure では、内部の OpenShift Data Foundation クラスターのみがサポートされます。デプロイメント要件の詳細は、[Planning your deployment](#) を参照してください。

OpenShift Data Foundation をデプロイするには、[OpenShift Data Foundation のデプロイの準備](#) の章の要件を確認し、要件に応じて適切なデプロイメントプロセスを実行します。

- [Microsoft Azure への OpenShift Data Foundation のデプロイ](#)
- [スタンドアロンの Multicloud Object Gateway コンポーネントのデプロイ](#)

## 第1章 OPENSIFT DATA FOUNDATION のデプロイの準備

動的ストレージデバイスを使用して OpenShift Data Foundation を OpenShift Container Platform にデプロイすると、内部クラスターリソースを作成するオプションが提供されます。これにより、ベースサービスの内部プロビジョニングが可能になり、追加のストレージクラスをアプリケーションで使用できるようになります。

OpenShift Data Foundation のデプロイを開始する前に、以下を実行します。

1. chrony サーバーをセットアップします。[chrony タイムサービスの設定](#) を参照し、[ナレッジベースソリューション](#) を使用して、すべてのトラフィックを許可するルールを作成します。
2. オプション: 外部の鍵管理システム (KMS) HashiCorp Vault を使用してクラスター全体の暗号化を有効にする場合は、次の手順に従います。
  - 有効な Red Hat OpenShift Data Foundation Advanced サブスクリプションがあることを確認してください。OpenShift Data Foundation のサブスクリプションの仕組みを確認するには、[OpenShift Data Foundation subscriptions に関するナレッジベースの記事](#) を参照してください。
  - 暗号化にトークン認証方法が選択されている場合は、[Enabling cluster-wide encryption with the Token authentication using KMS](#) を参照してください。
  - 暗号化に Kubernetes 認証方式を選択した場合は、[KMS を使用した Kubernetes 認証によるクラスター全体の暗号化の有効化](#) を参照してください。
  - Vault サーバーで署名済みの証明書を使用していることを確認します。



### 注記

Thales CipherTrust Manager を KMS として使用している場合は、デプロイ時に有効にします。

3. ノードの最小要件  
OpenShift Data Foundation クラスターは、標準のデプロイメントリソース要件を満たしていない場合に、最小の設定でデプロイされます。プランニングガイドの [リソース要件](#) セクションを参照してください。
4. 障害復旧の要件 [テクノロジープレビュー]  
Red Hat OpenShift Data Foundation でサポートされる障害復旧機能では、障害復旧ソリューションを正常に実装するために以下の前提条件をすべて満たす必要があります。
  - 有効な Red Hat OpenShift Data Foundation Advanced サブスクリプション
  - 有効な Red Hat Advanced Cluster Management for Kubernetes サブスクリプション  
OpenShift Data Foundation のサブスクリプションの仕組みを確認するには、[OpenShift Data Foundation subscriptions に関するナレッジベースの記事](#) を参照してください。

詳細な要件については、[OpenShift ワークロード用の OpenShift Data Foundation Disaster Recovery の設定](#) ガイド、および Red Hat Advanced Cluster Management for Kubernetes ドキュメントの [インストールガイド](#) の [要件と推奨事項](#) セクションを参照してください。

## 第2章 MICROSOFT AZURE への OPENSIFT DATA FOUNDATION のデプロイ

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 (IPI) (タイプ: **managed-csi**) によって提供される動的ストレージデバイスを使用して OpenShift Data Foundation を OpenShift Container Platform にデプロイすると、内部クラスターリソースを作成できます。これにより、ベースサービスの内部プロビジョニングが可能になり、追加のストレージクラスをアプリケーションで使用できるようになります。

また、OpenShift Data Foundation で Multicloud Object Gateway (MCG) コンポーネントのみをデプロイすることもできます。詳細は、[Deploy standalone Multicloud Object Gateway](#) を参照してください。



### 注記

Microsoft Azure では、内部の OpenShift Data Foundation クラスターのみがサポートされます。デプロイメント要件の詳細は、[Planning your deployment](#) を参照してください。

動的ストレージデバイスを使用してデプロイするための以下の手順に進む前に、[OpenShift Data Foundation のデプロイの準備](#) の章の要件を満たしていることを確認してください。

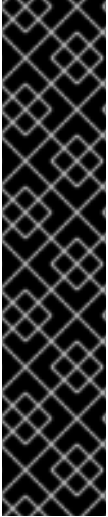
1. [Red Hat OpenShift Data Foundation Operator のインストール](#)
2. [OpenShift Data Foundation クラスターの作成](#)

### 2.1. RED HAT OPENSIFT DATA FOUNDATION OPERATOR のインストール

Red Hat OpenShift Data Foundation Operator は、Red Hat OpenShift Container Platform Operator Hub を使用してインストールできます。

#### 前提条件

- **cluster-admin** 権限および Operator インストール権限を持つアカウントを使用して OpenShift Container Platform クラスターにアクセスできる。
- Red Hat OpenShift Container Platform クラスターにワーカーノードまたはインフラストラクチャーノードが少なくとも 3 つある。
- その他のリソース要件は、[デプロイメントのプランニング](#) ガイドを参照してください。



## 重要

- OpenShift Data Foundation のクラスター全体でのデフォルトノードセレクターを上書きする必要がある場合は、以下のコマンドを使用して、**openshift-storage** namespace の空のノードセレクターを指定できます (この場合は **openshift-storage** を作成します)。

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

- ノードに Red Hat OpenShift Data Foundation リソースのみがスケジュールされるように **infra** のテイントを設定します。これにより、サブスクリプションコストを節約できます。詳細は、[ストレージリソースの管理と割り当て](#) ガイドの **Red Hat OpenShift Data Foundation に専用のワーカーノードを使用する方法** セクションを参照してください。

## 手順

1. OpenShift Web コンソールにログインします。
2. **Operators** → **OperatorHub** をクリックします。
3. スクロールするか、**OpenShift Data Foundation** を **Filter by keyword** ボックスに入力し、**OpenShift Data Foundation Operator** を検索します。
4. **Install** をクリックします。
5. **Install Operator** ページで、以下のオプションを設定します。
  - a. Update Channel を **stable-4.16** に設定します。
  - b. Installation Mode オプションに **A specific namespace on the cluster** を選択します。
  - c. Installed Namespace に **Operator recommended namespace openshift-storage** を選択します。namespace **openshift-storage** が存在しない場合は、Operator のインストール時に作成されます。
  - d. 承認ストラテジーを **Automatic** または **Manual** として選択します。  
**Automatic** (自動) 更新を選択すると、Operator Lifecycle Manager (OLM) は介入なしに、Operator の実行中のインスタンスを自動的にアップグレードします。  
  
**Manual** 更新を選択すると、OLM は更新要求を作成します。クラスター管理者は、Operator を新しいバージョンに更新できるように更新要求を手動で承認する必要があります。
  - e. **Console プラグイン** に **Enable** オプションが選択されていることを確認します。
  - f. **Install** をクリックします。

## 検証手順

- Operator が正常にインストールされると、**Web console update is available** メッセージを含むポップアップがユーザーインターフェイスに表示されます。このポップアップから **Refresh web console** をクリックして、反映するコンソールを変更します。
- Web コンソールに移動します。
  - Installed Operators に移動し、**OpenShift Data Foundation Operator** に、インストールが

正常に実行されたことを示す緑色のチェックマークが表示されていることを確認します。

- **Storage** に移動し、**Data Foundation** ダッシュボードが使用可能かどうかを確認します。

## 2.2. トークン認証方法を使用した KMS を使用したクラスター全体の暗号化の有効化

トークン認証のために、Vault でキーと値のバックエンドパスおよびポリシーを有効にできます。

### 前提条件

- Vault への管理者アクセス。
- 有効な Red Hat OpenShift Data Foundation Advanced サブスクリプション。詳細は、[OpenShift Data Foundation サブスクリプションに関するナレッジベースの記事](#) を参照してください。
- 後で変更できないため、命名規則に従って一意のパス名をバックエンド **path** として慎重に選択してください。

### 手順

1. Vault で Key/Value (KV) バックエンドパスを有効にします。  
Vault KV シークレットエンジン API の場合は、バージョン 1 です。

```
$ vault secrets enable -path=odf kv
```

Vault KV シークレットエンジン API の場合は、バージョン 2 を使用します。

```
$ vault secrets enable -path=odf kv-v2
```

2. シークレットに対して書き込み操作または削除操作を実行するようにユーザーを制限するポリシーを作成します。

```
echo '
path "odf/*" {
  capabilities = ["create", "read", "update", "delete", "list"]
}
path "sys/mounts" {
  capabilities = ["read"]
}' | vault policy write odf -
```

3. 上記のポリシーに一致するトークンを作成します。

```
$ vault token create -policy=odf -format json
```

## 2.3. KUBERNETES 認証方式を使用した KMS でのクラスター全体の暗号化の有効化

キー管理システム (KMS) を使用して、クラスター全体の暗号化に対して Kubernetes 認証方式を有効にできます。

## 前提条件

- Vault への管理者アクセス。
- 有効な Red Hat OpenShift Data Foundation Advanced サブスクリプション。詳細は、[OpenShift Data Foundation サブスクリプションに関するナレッジベースの記事](#) を参照してください。
- OpenShift Data Foundation Operator が Operator Hub からインストールされている。
- バックエンド **path** として一意のパス名を選択する。これは命名規則に厳密に準拠する必要があります。このパス名は後で変更できません。

## 手順

1. サービスアカウントを作成します。

```
$ oc -n openshift-storage create serviceaccount <serviceaccount_name>
```

ここで、**<serviceaccount\_name>** はサービスアカウントの名前を指定します。

以下に例を示します。

```
$ oc -n openshift-storage create serviceaccount odf-vault-auth
```

2. **clusterrolebindings** と **clusterroles** を作成します。

```
$ oc -n openshift-storage create clusterrolebinding vault-tokenreview-binding --
clusterrole=system:auth-delegator --serviceaccount=openshift-
storage:_<serviceaccount_name>_
```

以下に例を示します。

```
$ oc -n openshift-storage create clusterrolebinding vault-tokenreview-binding --
clusterrole=system:auth-delegator --serviceaccount=openshift-storage:odf-vault-auth
```

3. **serviceaccount** トークンおよび CA 証明書のシークレットを作成します。

```
$ cat <<EOF | oc create -f -
apiVersion: v1
kind: Secret
metadata:
  name: odf-vault-auth-token
  namespace: openshift-storage
  annotations:
    kubernetes.io/service-account.name: <serviceaccount_name>
type: kubernetes.io/service-account-token
data: {}
EOF
```

ここで、**<serviceaccount\_name>** は、前の手順で作成したサービスアカウントです。

4. シークレットからトークンと CA 証明書を取得します。

```
$ SA_JWT_TOKEN=$(oc -n openshift-storage get secret odf-vault-auth-token -o jsonpath=
```

```
{.data["token"]} | base64 --decode; echo)
$ SA_CA_CERT=$(oc -n openshift-storage get secret odf-vault-auth-token -o jsonpath="{.data['ca.crt']}" | base64 --decode; echo)
```

5. OCP クラスターエンドポイントを取得します。

```
$ OCP_HOST=$(oc config view --minify --flatten -o jsonpath="{.clusters[0].cluster.server}")
```

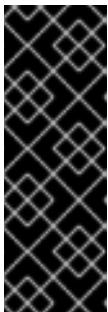
6. サービスアカウントの発行者を取得します。

```
$ oc proxy &
$ proxy_pid=$!
$ issuer="$( curl --silent http://127.0.0.1:8001/.well-known/openid-configuration | jq -r
.issuer)"
$ kill $proxy_pid
```

7. 前の手順で収集した情報を使用して、Vault で Kubernetes 認証方法を設定します。

```
$ vault auth enable kubernetes
```

```
$ vault write auth/kubernetes/config \
  token_reviewer_jwt="$SA_JWT_TOKEN" \
  kubernetes_host="$OCP_HOST" \
  kubernetes_ca_cert="$SA_CA_CERT" \
  issuer="$issuer"
```



### 重要

発行者が空の場合は Vault で Kubernetes 認証方法を設定します。

```
$ vault write auth/kubernetes/config \
  token_reviewer_jwt="$SA_JWT_TOKEN" \
  kubernetes_host="$OCP_HOST" \
  kubernetes_ca_cert="$SA_CA_CERT"
```

8. Vault で Key/Value (KV) バックエンドパスを有効にします。  
Vault KV シークレットエンジン API の場合は、バージョン 1 を使用します。

```
$ vault secrets enable -path=odf kv
```

Vault KV シークレットエンジン API の場合は、バージョン 2 を使用します。

```
$ vault secrets enable -path=odf kv-v2
```

9. シークレットに対して **write** または **delete** 操作を実行するようにユーザーを制限するポリシーを作成します。

```
echo '
path "odf/*" {
  capabilities = ["create", "read", "update", "delete", "list"]
}
```

```
path "sys/mounts" {
  capabilities = ["read"]
}'| vault policy write odf -
```

10. ロールを作成します。

```
$ vault write auth/kubernetes/role/odf-rook-ceph-op \
  bound_service_account_names=rook-ceph-system,rook-ceph-osd,noobaa \
  bound_service_account_namespaces=openshift-storage \
  policies=odf \
  ttl=1440h
```

ロール **odf-rook-ceph-op** は、後でストレージシステムの作成中に KMS 接続の詳細を設定するときに使用されます。

```
$ vault write auth/kubernetes/role/odf-rook-ceph-osd \
  bound_service_account_names=rook-ceph-osd \
  bound_service_account_namespaces=openshift-storage \
  policies=odf \
  ttl=1440h
```

## 2.4. OPENSIFT DATA FOUNDATION クラスターの作成

OpenShift Data Foundation Operator のインストール後に OpenShift Data Foundation クラスターを作成します。

### 前提条件

- OpenShift Data Foundation Operator が Operator Hub からインストールされている。詳細は、[Operator Hub を使用した OpenShift Data Foundation Operator のインストール](#) を参照してください。
- Azure Vault [テクノロジープレビュー] をキー管理サービスプロバイダーとして使用する場合は、次の手順に従ってクライアント認証を設定し、Azure からクライアント認証情報を取得してください。
  1. Azure Vault を作成します。詳細は、Microsoft 製品ドキュメントの [Quickstart: Create a key vault using the Azure portal](#) を参照してください。
  2. 証明書ベースの認証を使用してサービスプリンシパルを作成します。詳細は、Microsoft 製品ドキュメントの [Create an Azure service principal with Azure CLI](#) を参照してください。
  3. Azure Key Vault のロールベースのアクセス制御 (RBAC) を設定します。詳細は、[Enable Azure RBAC permissions on Key Vault](#) を参照してください。

### 手順

1. OpenShift Web コンソールで、**Operators → Installed Operators** をクリックし、インストールされた Operator を表示します。  
選択された **Project** が **openshift-storage** であることを確認します。
2. **OpenShift Data Foundation Operator** をクリックした後、**Create StorageSystem** をクリックします。



3. **Backing storage** ページで、以下を選択します。
  - a. **Deployment type** オプションで **Full Deployment** を選択します。
  - b. **Use an existing StorageClass** オプションを選択します。
  - c. **Storage Class** を選択します。  
デフォルトでは、これは **managed-csi** に設定されています。
  - d. オプション: 外部 PostgreSQL を使用するには、**Use external PostgreSQL** チェックボックスを選択します [テクノロジープレビュー]。  
これにより、PostgreSQL Pod が単一障害点となるマルチクラウドオブジェクトゲートウェイの高可用性ソリューションが提供されます。
    - i. 以下の接続の詳細を指定します。
      - ユーザー名
      - パスワード
      - サーバー名とポート
      - データベース名
    - ii. **Enable TLS/SSL** チェックボックスを選択して、Postgres サーバーの暗号化を有効にします。
  - e. **Next** をクリックします。
4. **Capacity and nodes** ページで、必要な情報を提供します。
  - a. ドロップダウンリストから **Requested Capacity** の値を選択します。デフォルトで、これは **2 TiB** に設定されます。



#### 注記

初期ストレージ容量を選択すると、クラスターの拡張は、選択された使用可能な容量を使用してのみ実行されます (raw ストレージの 3 倍)。

- b. **Select Nodes** セクションで、少なくとも 3 つの利用可能なノードを選択します。
- c. **Configure performance** セクションで、以下のパフォーマンスプロファイルのいずれかを選択します。
  - Lean  
これは、最小リソースが推奨値よりも少ない、リソースに制約のある環境で使用します。このプロファイルでは、割り当てられる CPU とメモリーの数が少なくなり、リソースの消費が最小限に抑えられます。
  - balanced (デフォルト)  
推奨リソースが利用可能な場合にこれを使用します。このプロファイルは、さまざまなワークロードのリソース消費とパフォーマンスのバランスを提供します。
  - パフォーマンス  
最高のパフォーマンスを得るために十分なリソースがある環境でこれを使用してください。このプロファイルは、負荷の高いワークロードを最適に実行できるように十分なメモリーと CPU を割り当てることで、高いパフォーマンスを実現するように調整されて

います。



### 注記

**StorageSystems** タブのオプションメニューから **Configure performance** オプションを使用して、デプロイメント後にパフォーマンスプロファイルを設定するオプションがあります。



### 重要

リソースプロファイルを選択する前に、クラスター内のリソースの現在の可用性を必ず確認してください。リソースが不十分なクラスターでより高いリソースプロファイルを選択すると、インストールが失敗する可能性があります。

リソース要件の詳細は、[パフォーマンスプロファイルのリソース要件](#) を参照してください。

- d. オプション: 選択したノードを OpenShift Data Foundation 専用にする場合は、**Taint nodes** チェックボックスを選択します。

複数のアベイラビリティゾーンを持つクラウドプラットフォームの場合は、ノードが異なる場所/アベイラビリティゾーンに分散されていることを確認します。

選択したノードが集約された 30 CPU および 72 GiB の RAM の OpenShift Data Foundation クラスターの要件と一致しない場合は、最小クラスターがデプロイされます。ノードの最小要件については、[プランニングガイドのリソース要件](#) セクションを参照してください。

- e. **Next** をクリックします。

5. オプション: **Security and network** ページで、要件に応じて以下を設定します。

- a. 暗号化を有効にするには、**Enable data encryption for block and file storage** を選択します。

- i. 暗号化レベルのいずれかまたは両方を選択します。

- **クラスター全体の暗号化**

クラスター全体を暗号化します (ブロックおよびファイル)。

- **StorageClass の暗号化**

暗号化対応のストレージクラスを使用して、暗号化された永続ボリューム (ブロックのみ) を作成します。

- ii. オプション: **Connect to an external key management service** チェックボックスを選択します。これはクラスター全体の暗号化の場合はオプションになります。

- A. **Key Management Service Provider** ドロップダウンリストから、次のいずれかのプロバイダーを選択し、必要な詳細情報を入力します。

- **Vault**

- i. **Authentication Method** を選択します。

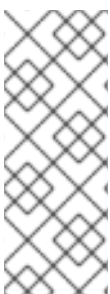
- **トークン認証方式の使用**

- Vault ('https://<hostname or ip>') サーバーの一意的 **Connection Name**、ホストの **Address**、**Port** 番号および **Token** を入力します。
- **Advanced Settings** をデプロイメントして、**Vault** 設定に基づいて追加の設定および証明書の詳細を入力します。
  - OpenShift Data Foundation 専用かつ特有のキーと値のシークレットパスを **Backend Path** に入力します。
- オプション: **TLS Server Name** および **Vault Enterprise Namespace** を入力します。
- PEM でエンコードされた、該当の証明書ファイルをアップロードし、**CA Certificate**、**Client Certificate**、および **Client Private Key** を指定します。
- **Save** をクリックします。
- **Kubernetes 認証方式の使用**
  - Vault ('https://<hostname or ip>') サーバーの一意的 **Connection Name**、ホストの **Address**、**Port** 番号、および **Role** 名を入力します。
  - **Advanced Settings** をデプロイメントして、**Vault** 設定に基づいて追加の設定および証明書の詳細を入力します。
    - OpenShift Data Foundation 専用かつ特有のキーと値のシークレットパスを **Backend Path** に入力します。
    - 該当する場合は、**TLS Server Name** および **Authentication Path** を入力します。
    - PEM でエンコードされた、該当の証明書ファイルをアップロードし、**CA Certificate**、**Client Certificate**、および **Client Private Key** を指定します。
  - **Save** をクリックします。
- **Thales CipherTrust Manager (KMIP を使用)**
  - I. プロジェクト内のキー管理サービスの一意的 **Connection Name** を入力します。
  - II. **Address** および **Port** セクションで、Thales CipherTrust Manager の IP と、KMIP インターフェイスが有効になっているポートを入力します。以下に例を示します。
    - **Address:** 123.34.3.2
    - **Port:** 5696
  - III. **Client Certificate**、**CA certificate**、および **Client Private Key** をアップロードします。
  - IV. StorageClass 暗号化が有効になっている場合は、上記で生成された暗号化および復号化に使用する一意の識別子を入力します。

- V. **TLS Server** フィールドはオプションであり、KMIP エンドポイントの DNS エントリーがない場合に使用します。たとえば、**kmip\_all\_<port>.ciphertrustmanager.local** などです。
- **Azure Key Vault** [テクノロジープレビュー]
 

Azure プラットフォームでクライアント認証を設定し、クライアント認証情報を取得する方法については、この手順の「前提条件」セクションを参照してください。

  - I. プロジェクト内のキー管理サービスの一意の **Connection name** を入力します。
  - II. **Azure Vault URL** を入力します。
  - III. **Client ID** を入力します。
  - IV. **Tenant ID** を入力します。
  - V. **Certificate** ファイルを **.PEM** 形式でアップロードします。証明書ファイルには、クライアント証明書と秘密鍵が含まれている必要があります。
- b. 転送中の暗号化を有効にするには、**In-transit encryption** を選択します。
  - i. **Network** を選択します。
  - ii. **Next** をクリックします。
- 6. **Data Protection** ページで、OpenShift Data Foundation の Regional DR ソリューションを設定している場合は、**Prepare cluster for disaster recovery(Regional-DR only)** チェックボックスを選択し、それ以外の場合は **Next** をクリックします。
- 7. **Review and create** ページで、設定の詳細を確認します。設定を変更するには、**Back** をクリックします。
- 8. **Create StorageSystem** をクリックします。



## 注記

デプロイメントに5つ以上のノード、ラック、またはルームがあり、デプロイメント内に5つ以上の障害ドメインが存在する場合、ラックまたはゾーンの数に基づいて Ceph モニター数を設定できます。OpenShift Web コンソールの通知パネルまたはアラートセンターにアラートが表示され、Ceph モニター数を増やすオプションが示されます。アラートで **Configure** オプションを使用して、Ceph モニター数を設定できます。詳細は、[Ceph モニター数が少ないというアラートの解決](#) を参照してください。

## 検証手順

- インストールされたストレージクラスターの最終ステータスを確認するには、以下を実行します。
  - a. OpenShift Web コンソールで、**Installed Operators** → **OpenShift Data Foundation** → **Storage System** → **ocs-storagecluster-storagesystem** → **Resources** の順に移動します。
  - b. **StorageCluster** の **Status** が **Ready** になっており、その横に緑色のチェックマークが表示されていることを確認します。

- OpenShift Data Foundation のすべてのコンポーネントが正常にインストールされていることを確認するには、[Verifying your OpenShift Data Foundation deployment](#) を参照してください。

## 関連情報

Overprovision Control アラートを有効にするには、モニタリングガイドの [アラート](#) を参照してください。

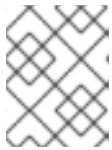
## 第3章 OPENSIFT DATA FOUNDATION デプロイメントの確認

このセクションを使用して、OpenShift Data Foundation が正しくデプロイされていることを確認します。

### 3.1. POD の状態の確認

#### 手順

1. OpenShift Web コンソールから **Workloads** → **Pods** をクリックします。
2. **Project** ドロップダウンリストから **openshift-storage** を選択します。



#### 注記

**Show default projects** オプションが無効になっている場合は、切り替えボタンを使用して、すべてのデフォルトプロジェクトをリスト表示します。

各コンポーネントの予想される Pod 数と、それがノード数によってどのように変化するかについては、次の表を参照してください。

1. 実行中および完了した Pod のフィルターを設定して、次の Pod が **Running** および **Completed** 状態であることを確認します。

コンポーネント	対応する Pod
OpenShift Data Foundation Operator	<ul style="list-style-type: none"> <li>● <b>ocs-operator-*</b> (任意のストレージノードに 1Pod)</li> <li>● <b>ocs-metrics-exporter-*</b> (任意のストレージノードに 1Pod)</li> <li>● <b>odf-operator-controller-manager-*</b> (任意のストレージノードに 1Pod)</li> <li>● <b>odf-console-*</b> (任意のストレージノードに 1Pod)</li> <li>● <b>csi-addons-controller-manager-*</b> (任意のストレージノードに 1Pod)</li> <li>● <b>ux-backend-server-*</b> (任意のストレージノードに 1Pod)</li> <li>● <b>*ocs-client-operator-*</b> (任意のストレージノードに 1Pod)</li> <li>● <b>ocs-client-operator-console-*</b> (任意のストレージノードに 1Pod)</li> <li>● <b>ocs-provider-server-*</b> (任意のストレージノードに 1Pod)</li> </ul>

Rook-ceph Operator	<b>rook-ceph-operator-*</b> (任意のストレージノードに 1Pod)
Multicloud Object Gateway	<ul style="list-style-type: none"> <li>● <b>noobaa-operator-*</b> (任意のストレージノードに 1Pod)</li> <li>● <b>noobaa-core-*</b> (任意のストレージノードに 1Pod)</li> <li>● <b>noobaa-db-pg-*</b> (任意のストレージノードに 1Pod)</li> <li>● <b>noobaa-endpoint-*</b> (任意のストレージノードに 1Pod)</li> </ul>
MON	<b>rook-ceph-mon-*</b> (ストレージノードに分散する 3 Pod)
MGR	<b>rook-ceph-mgr-*</b> (任意のストレージノードに 1Pod)
MDS	<b>rook-ceph-mds-ocs-storagecluster-cephfilesystem-*</b> (ストレージノードに分散する 2 Pod)
CSI	<ul style="list-style-type: none"> <li>● <b>cephfs</b> <ul style="list-style-type: none"> <li>○ <b>csi-cephfsplugin-*</b> (各ストレージノードに 1Pod)</li> <li>○ <b>csi-cephfsplugin-provisioner-*</b> (ストレージノードに分散する 2 Pod)</li> </ul> </li> <li>● <b>rbd</b> <ul style="list-style-type: none"> <li>○ <b>csi-rbdplugin-*</b> (各ストレージノードに 1Pod)</li> <li>○ <b>csi-rbdplugin-provisioner-*</b> (ストレージノードに分散する 2 Pod)</li> </ul> </li> </ul>
rook-ceph-crashcollector	<b>rook-ceph-crashcollector-*</b> (各ストレージノードに 1Pod)
OSD	<ul style="list-style-type: none"> <li>● <b>rook-ceph-osd-*</b> (各デバイス用に 1Pod)</li> <li>● <b>rook-ceph-osd-prepare-ocs-device-*</b> (各デバイス用に 1Pod)</li> </ul>

## 3.2. OPENSIFT DATA FOUNDATION クラスタが正常であることを確認

### 手順

1. OpenShift Web コンソールで、**Storage → Data Foundation** をクリックします。
2. **Overview** タブの **Status** カードで **Storage System** をクリックし、表示されたポップアップからストレージシステムリンクをクリックします。
3. **Block and File** タブの **Status** カードで、**Storage Cluster** に緑色のチェックマークが表示されていることを確認します。
4. **Details** カードで、クラスタ情報が表示されていることを確認します。

ブロックおよびファイルダッシュボードを使用した OpenShift Data Foundation クラスタの正常性については、[Monitoring OpenShift Data Foundation](#) を参照してください。

## 3.3. MULTICLOUD OBJECT GATEWAY が正常であることを確認

### 手順

1. OpenShift Web コンソールで、**Storage → Data Foundation** をクリックします。
2. **Overview** タブの **Status** カードで **Storage System** をクリックし、表示されたポップアップからストレージシステムリンクをクリックします。
  - a. **Object** タブの **Status card** で、**Object Service** と **Data Resiliency** の両方に緑色のチェックマークが表示されていることを確認します。
  - b. **Details** カードで、MCG 情報が表示されることを確認します。

ブロックおよびファイルダッシュボードを使用した OpenShift Data Foundation クラスタの正常性については、[OpenShift Data Foundation の監視](#) を参照してください。



### 重要

Multicloud Object Gateway には、データベースのコピー (NooBaa DB) が1つだけあります。つまり、NooBaa DB PVC が破損し、回復できない場合は、Multicloud Object Gateway に存在するアプリケーションデータが完全に失われる可能性があります。このため、Red Hat では NooBaa DB PVC のバックアップを定期的にとることを推奨しています。NooBaa DB に障害が発生して回復できない場合は、最新のバックアップバージョンに戻すことができます。NooBaa DB をバックアップする手順は、[こちらのナレッジベースの記事](#) の手順に従ってください。

## 3.4. 特定のストレージクラスが存在することを確認

### 手順

1. OpenShift Web コンソールの左側のペインから **Storage → Storage Classes** をクリックします。
2. 以下のストレージクラスが OpenShift Data Foundation クラスタの作成時に作成されることを確認します。

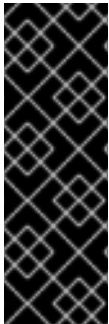


- **ocs-storagecluster-ceph-rbd**
- **ocs-storagecluster-cephfs**
- **openshift-storage.noobaa.io**

## 第4章 スタンドアロンの MULTICLOUD OBJECT GATEWAY のデプロイ

OpenShift Data Foundation で Multicloud Object Gateway コンポーネントのみをデプロイすると、デプロイメントで柔軟性が高まり、リソース消費を減らすことができます。このセクションでは、以下のステップで、スタンドアロンの Multicloud Object Gateway コンポーネントのみをデプロイします。

- Red Hat OpenShift Data Foundation Operator のインストール
- スタンドアロンの Multicloud Object Gateway の作成



### 重要

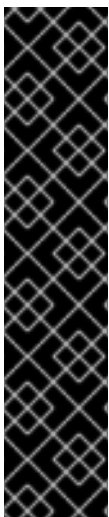
Multicloud Object Gateway には、データベースのコピー (NooBaa DB) が1つだけあります。つまり、NooBaa DB PVC が破損し、回復できない場合は、Multicloud Object Gateway に存在するアプリケーションデータが完全に失われる可能性があります。このため、Red Hat では NooBaa DB PVC のバックアップを定期的にとることを推奨しています。NooBaa DB に障害が発生して回復できない場合は、最新のバックアップバージョンに戻すことができます。NooBaa DB をバックアップする手順は、[こちらのナレッジベースの記事](#) の手順に従ってください。

### 4.1. RED HAT OPENSIFT DATA FOUNDATION OPERATOR のインストール

Red Hat OpenShift Data Foundation Operator は、Red Hat OpenShift Container Platform Operator Hub を使用してインストールできます。

#### 前提条件

- **cluster-admin** 権限および Operator インストール権限を持つアカウントを使用して OpenShift Container Platform クラスタにアクセスできる。
- Red Hat OpenShift Container Platform クラスタにワーカーノードまたはインフラストラクチャーノードが少なくとも3つある。
- その他のリソース要件は、[デプロイメントのプランニング](#) ガイドを参照してください。



### 重要

- OpenShift Data Foundation のクラスタ全体でのデフォルトノードセレクターを上書きする必要がある場合は、以下のコマンドを使用して、**openshift-storage** namespace の空のノードセレクターを指定できます (この場合は **openshift-storage** を作成します)。

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

- ノードに Red Hat OpenShift Data Foundation リソースのみがスケジュールされるように **infra** のテイントを設定します。これにより、サブスクリプションコストを節約できます。詳細は、[ストレージリソースの管理と割り当て](#) ガイドの **Red Hat OpenShift Data Foundation に専用のワーカーノードを使用する方法** セクションを参照してください。

#### 手順

1. OpenShift Web コンソールにログインします。
2. **Operators** → **OperatorHub** をクリックします。
3. スクロールするか、**OpenShift Data Foundation** を **Filter by keyword** ボックスに入力し、**OpenShift Data Foundation Operator** を検索します。
4. **Install** をクリックします。
5. **Install Operator** ページで、以下のオプションを設定します。
  - a. Update Channel を **stable-4.16** に設定します。
  - b. Installation Mode オプションに **A specific namespace on the cluster** を選択します。
  - c. Installed Namespace に **Operator recommended namespace openshift-storage** を選択します。namespace **openshift-storage** が存在しない場合は、Operator のインストール時に作成されます。
  - d. 承認ストラテジーを **Automatic** または **Manual** として選択します。

**Automatic** (自動) 更新を選択すると、Operator Lifecycle Manager (OLM) は介入なしに、Operator の実行中のインスタンスを自動的にアップグレードします。

**Manual** 更新を選択すると、OLM は更新要求を作成します。クラスター管理者は、Operator を新しいバージョンに更新できるように更新要求を手動で承認する必要があります。
  - e. **Console プラグイン** に **Enable** オプションが選択されていることを確認します。
  - f. **Install** をクリックします。

#### 検証手順

- Operator が正常にインストールされると、**Web console update is available** メッセージを含むポップアップがユーザーインターフェイスに表示されます。このポップアップから **Refresh web console** をクリックして、反映するコンソールを変更します。
- Web コンソールに移動します。
  - Installed Operators に移動し、**OpenShift Data Foundation Operator** に、インストールが正常に実行されたことを示す緑色のチェックマークが表示されていることを確認します。
  - **Storage** に移動し、**Data Foundation** ダッシュボードが使用可能かどうかを確認します。

## 4.2. スタンドアロンの MULTICLOUD OBJECT GATEWAY の作成

OpenShift Data Foundation のデプロイ中には、スタンドアロンの Multicloud Object Gateway コンポーネントのみを作成できます。

#### 前提条件

- OpenShift Data Foundation Operator がインストールされている。

#### 手順

1. OpenShift Web コンソールで、**Operators** → **Installed Operators** をクリックし、インストールされた Operator を表示します。  
選択された **Project** が **openshift-storage** であることを確認します。
2. **OpenShift Data Foundation Operator** をクリックした後、**Create StorageSystem** をクリックします。
3. **Backing storage** ページで、以下を選択します。
  - a. **Deployment type** の **Multicloud Object Gateway** を選択します。
  - b. **Use an existing StorageClass** オプションを選択します。
  - c. **Next** をクリックします。
4. オプション: **Connect to an external key management service** チェックボックスを選択します。これはクラスター全体の暗号化の場合はオプションになります。
  - a. **Key Management Service Provider** ドロップダウンリストから、**Vault** または **Thales CipherTrust Manager (using KMIP)** を選択します。**Vault** を選択した場合は、次の手順に進みます。**Thales CipherTrust Manager (using KMIP)** を選択した場合は、手順 iii に進みます。
  - b. **Authentication Method** を選択します。

#### トークン認証方式の使用

- Vault ('https://<hostname or ip>') サーバーの一意の **Connection Name**、ホストの **Address**、**Port** 番号および **Token** を入力します。
- **Advanced Settings** をデプロイメントして、**Vault** 設定に基づいて追加の設定および証明書の詳細を入力します。
  - OpenShift Data Foundation 専用かつ特有のキーと値のシークレットパスを **Backend Path** に入力します。
  - オプション: **TLS Server Name** および **Vault Enterprise Namespace** を入力します。
  - PEM でエンコードされた、該当の証明書ファイルをアップロードし、**CA Certificate**、**Client Certificate**、および **Client Private Key** を指定します。
  - **Save** をクリックして、手順 iv に進みます。

#### Kubernetes 認証方式の使用

- Vault ('https://<hostname or ip>') サーバーの一意の **Connection Name**、ホストの **Address**、**Port** 番号、および **Role** 名を入力します。
- **Advanced Settings** をデプロイメントして、**Vault** 設定に基づいて追加の設定および証明書の詳細を入力します。
  - OpenShift Data Foundation 専用かつ特有のキーと値のシークレットパスを **Backend Path** に入力します。
  - 該当する場合は、**TLS Server Name** および **Authentication Path** を入力します。

- PEM でエンコードされた、該当の証明書ファイルをアップロードし、**CA Certificate**、**Client Certificate**、および **Client Private Key** を指定します。
  - **Save** をクリックして、手順 iv に進みます。
- c. **Thales CipherTrust Manager (using KMIP)** を KMS プロバイダーとして使用するには、次の手順に従います。
- i. プロジェクト内のキー管理サービスの一意の **Connection Name** を入力します。
  - ii. **Address** および **Port** セクションで、Thales CipherTrust Manager の IP と、KMIP インターフェイスが有効になっているポートを入力します。以下に例を示します。
    - **Address:** 123.34.3.2
    - **Port:** 5696
  - iii. **Client Certificate**、**CA certificate**、および **Client Private Key** をアップロードします。
  - iv. StorageClass 暗号化が有効になっている場合は、上記で生成された暗号化および復号化に使用する一意の識別子を入力します。
  - v. **TLS Server** フィールドはオプションであり、KMIP エンドポイントの DNS エントリがない場合に使用します。たとえば、**kmip\_all\_<port>.ciphertrustmanager.local** などです。
- d. **Network** を選択します。
- e. **Next** をクリックします。
5. **Review and create** ページで、設定の詳細を確認します。  
設定を変更するには、**Back** をクリックします。
6. **Create StorageSystem** をクリックします。

## 検証手順

### OpenShift Data Foundation クラスタが正常であることの確認

1. OpenShift Web コンソールで、**Storage → Data Foundation** をクリックします。
2. **Overview** タブの **Status** カードで **Storage System** をクリックし、表示されたポップアップからストレージシステムリンクをクリックします。
  - a. **Object** タブの **Status card** で、**Object Service** と **Data Resiliency** の両方に緑色のチェックマークが表示されていることを確認します。
  - b. **Details** カードで、MCG 情報が表示されることを確認します。

### Pod の状態の確認

1. OpenShift Web コンソールから **Workloads → Pods** をクリックします。
2. **Project** ドロップダウンリストから **openshift-storage** を選択し、以下の Pod が **Running** 状態にあることを確認します。



## 注記

Show default projects オプションが無効になっている場合は、切り替えボタンを使用して、すべてのデフォルトプロジェクトをリスト表示します。

コンポーネント	対応する Pod
OpenShift Data Foundation Operator	<ul style="list-style-type: none"> <li>● <b>ocs-operator-*</b> (任意のストレージノードに 1Pod)</li> <li>● <b>ocs-metrics-exporter-*</b> (任意のストレージノードに 1Pod)</li> <li>● <b>odf-operator-controller-manager-*</b> (任意のストレージノードに 1Pod)</li> <li>● <b>odf-console-*</b> (任意のストレージノードに 1Pod)</li> <li>● <b>csi-addons-controller-manager-*</b> (任意のストレージノードに 1Pod)</li> </ul>
Rook-ceph Operator	<p><b>rook-ceph-operator-*</b></p> <p>(任意のストレージノードに 1Pod)</p>
Multicloud Object Gateway	<ul style="list-style-type: none"> <li>● <b>noobaa-operator-*</b> (任意のストレージノードに 1Pod)</li> <li>● <b>noobaa-core-*</b> (任意のストレージノードに 1Pod)</li> <li>● <b>noobaa-db-pg-*</b> (任意のストレージノードに 1Pod)</li> <li>● <b>noobaa-endpoint-*</b> (任意のストレージノードに 1Pod)</li> </ul>

## 第5章 OPENSIFT DATA FOUNDATION トポロジーの表示

トポロジーは、OpenShift Data Foundation ストレージクラスターをマップしたた視覚情報をさまざまな抽象化レベルで示し、このような階層の操作も可能にします。このビューでは、ストレージクラスターがさまざまな要素でどのように構成されているかがわかります。

### 手順

1. OpenShift Web コンソールで、**Storage** → **Data Foundation** → **Topology** に移動します。  
このビューには、ストレージクラスターとその内部のゾーンが表示されます。ノードがゾーン内(点線で示されている)にある円形のエンティティーで表示されていることがわかります。各アイテムまたはリソースのラベルには、ステータスやヘルス、アラートの状態などの基本情報が含まれています。
2. ノードを選択すると、右側のパネルにノードの詳細が表示されます。検索/プレビューデコレーターアイコンをクリックして、ノード内のリソースまたはデプロイメントにアクセスすることもできます。
3. デプロイメントの詳細を表示します。
  - a. ノード上のプレビューデコレーターをクリックします。ノードの上にモーダルウィンドウが表示され、そのノードに関連付けられているすべてのデプロイメントとそのステータスが表示されます。
  - b. モデルの左上隅にある **Back to main view** ボタンをクリックしてモデルを閉じ、前のビューに戻ります。
  - c. 特定のデプロイメントを選択すると、そのデプロイメントに関する詳細が表示されます。関連するデータがすべてサイドパネルに表示されます。
4. **Resources** タブをクリックして Pod 情報を表示します。このタブを使用すると、問題の理解を深めることができるだけでなく、複数の詳細レベルが提供されるので適切にトラブルシューティングができるようになります。
5. Pod のリンクをクリックして、OpenShift Container Platform の Pod 情報ページを表示します。リンクは新しいウィンドウで開きます。

## 第6章 OPENSIFT DATA FOUNDATION のアンインストール

### 6.1. 内部モードでの OPENSIFT DATA FOUNDATION のアンインストール

内部モードで OpenShift Data Foundation をアンインストールするには、[OpenShift Data Foundation のアンインストールに関するナレッジベース記事](#) を参照してください。