



# Red Hat OpenShift Data Foundation 4.10

## VMware vSphere에 OpenShift Data Foundation 배포

VMware vSphere 인프라를 사용하여 OpenShift Data Foundation 배포 방법



# Red Hat OpenShift Data Foundation 4.10 VMware vSphere에 OpenShift Data Foundation 배포

---

VMware vSphere 인프라를 사용하여 OpenShift Data Foundation 배포 방법

## 법적 공지

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.

## 초록

VMware vSphere 클러스터에 Red Hat OpenShift Container Platform을 사용하여 Red Hat OpenShift Data Foundation을 설치하는 방법에 대한 자세한 내용은 이 문서를 참조하십시오.

<b>차례</b>	
보다 포괄적 수용을 위한 오픈 소스 용어 교체 .....	3
RED HAT 문서에 관한 피드백 제공 .....	4
머리말 .....	5
<b>1장. OPENSIFT DATA FOUNDATION 배포 준비 .....</b>	<b>6</b>
1.1. 로컬 스토리지 장치를 사용하여 OPENSIFT DATA FOUNDATION을 설치하기 위한 요구사항	6
<b>2장. 동적 스토리지 장치를 사용하여 배포 .....</b>	<b>8</b>
2.1. RED HAT OPENSIFT DATA FOUNDATION OPERATOR 설치	8
2.2. 토큰 인증 방법을 사용하여 KMS로 클러스터 전체 암호화 활성화	9
2.3. KUBERNETES 인증 방법을 사용하여 KMS로 클러스터 전체 암호화 활성화	10
2.4. MULTUS 네트워크 생성 [기술 프리뷰]	13
2.5. OPENSIFT DATA FOUNDATION 클러스터 생성	15
<b>3장. 로컬 스토리지 장치를 사용하여 배포 .....</b>	<b>19</b>
3.1. 로컬 스토리지 OPERATOR 설치	19
3.2. RED HAT OPENSIFT DATA FOUNDATION OPERATOR 설치	19
3.3. MULTUS 네트워크 생성 [기술 프리뷰]	21
3.4. VMWARE VSPHERE에서 OPENSIFT DATA FOUNDATION 클러스터 생성	23
<b>4장. OPENSIFT DATA FOUNDATION 배포 확인 .....</b>	<b>28</b>
4.1. POD 상태 확인	28
4.2. OPENSIFT DATA FOUNDATION 클러스터 상태 확인	30
4.3. MULTICLOUD OBJECT GATEWAY의 상태 확인	30
4.4. OPENSIFT DATA FOUNDATION 특정 스토리지 클래스가 있는지 확인	30
4.5. MULTUS 네트워킹 확인	31
<b>5장. 독립 실행형 MULTICLOUD OBJECT GATEWAY 배포 .....</b>	<b>34</b>
5.1. 동적 스토리지 장치를 사용하여 독립 실행형 MULTICLOUD OBJECT GATEWAY 배포	34
5.2. 로컬 스토리지 장치를 사용하여 독립 실행형 MULTICLOUD OBJECT GATEWAY 배포	37
<b>6장. OPENSIFT DATA FOUNDATION 설치 제거 .....</b>	<b>43</b>
6.1. 내부 모드에서 OPENSIFT DATA FOUNDATION 설치 제거	43



## 보다 포괄적 수용을 위한 오픈 소스 용어 교체

Red Hat은 코드, 문서, 웹 속성에서 문제가 있는 용어를 교체하기 위해 최선을 다하고 있습니다. 먼저 마스터(master), 슬레이브(slave), 블랙리스트(blacklist), 화이트리스트(whitelist) 등 네 가지 용어를 교체하고 있습니다. 이러한 변경 작업은 작업 범위가 크므로 향후 여러 릴리스에 걸쳐 점차 구현할 예정입니다. 자세한 내용은 [CTO Chris Wright의 메시지](#)를 참조하십시오.

## RED HAT 문서에 관한 피드백 제공

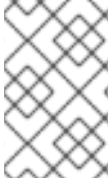
문서 개선을 위한 의견을 보내 주십시오. 개선할 내용에 대해 알려주십시오. 피드백을 보내주시려면 다음을 확인하십시오.

- 특정 문구에 대한 간단한 의견 작성 방법은 다음과 같습니다.
  1. 문서가 *Multi-page HTML* 형식으로 표시되는지 확인합니다. 또한 문서 오른쪽 상단에 **피드백** 버튼이 있는지 확인합니다.
  2. 마우스 커서를 사용하여 주석 처리하려는 텍스트 부분을 강조 표시합니다.
  3. 강조 표시된 텍스트 아래에 표시되는 **피드백 추가** 팝업을 클릭합니다.
  4. 표시된 지침을 따릅니다.
- 보다 상세하게 피드백을 제출하려면 다음과 같이 Bugzilla 티켓을 생성하십시오.
  1. [Bugzilla](#) 웹 사이트로 이동하십시오.
  2. **구성 요소** 섹션에서 **문서** 를 선택합니다.
  3. **설명** 필드에 문서 개선을 위한 제안 사항을 기입하십시오. 관련된 문서의 해당 부분 링크를 알려주십시오.
  4. **버그 제출**을 클릭합니다.



## 머리말

Red Hat OpenShift Data Foundation은 프록시 환경에 대한 기본 지원과 함께 연결되거나 연결이 끊긴 환경의 기존 Red Hat OpenShift Container Platform (RHOCP) vSphere 클러스터에 대한 배포를 지원합니다.



### 참고

VMware vSphere에서는 내부 및 외부 OpenShift Data Foundation 클러스터가 모두 지원됩니다. 배포 요구 사항에 대한 자세한 내용은 [배포 계획](#) 및 [OpenShift Data Foundation 배포 준비](#)를 참조하십시오.

OpenShift Data Foundation을 배포하려면 OpenShift Data Foundation [배포 준비](#) 장의 요구 사항부터 시작한 다음 사용자 환경에 따라 다음 배포 프로세스 중 하나를 따르십시오.

- 내부 모드
  - 동적 스토리지 장치를 사용하여 배포
  - 로컬 스토리지 장치를 사용하여 배포
  - 독립 실행형 Multicloud Object Gateway 구성 요소 배포
- 외부 모드

# 1장. OPENSIFT DATA FOUNDATION 배포 준비

동적 또는 로컬 스토리지 장치를 사용하여 OpenShift Container Platform에 OpenShift Data Foundation을 배포하면 내부 클러스터 리소스를 생성할 수 있는 옵션이 제공됩니다. 이렇게 하면 기본 서비스가 내부 프로비저닝되므로 애플리케이션에서 추가 스토리지 클래스를 사용할 수 있습니다.

동적 또는 로컬 스토리지를 사용하여 Red Hat OpenShift Data Foundation 배포를 시작하기 전에 리소스 요구 사항을 충족하는지 확인해야 합니다. [배포 계획](#)을 참조하십시오.

1. 선택 사항: 외부 키 관리 시스템(KMS)을 사용하여 클러스터 전체 암호화를 활성화하려면 다음 단계를 따르십시오.
  - 유효한 Red Hat OpenShift Data Foundation Advanced 서브스크립션이 있는지 확인합니다. OpenShift Data Foundation의 서브스크립션이 작동하는 방법을 알아보려면 [OpenShift Data Foundation 서브스크립션에 대한 기술 자료 문서](#)를 참조하십시오.
  - 암호화에 대해 토큰 인증 방법을 선택하면 [KMS를 사용하여 토큰 인증으로 클러스터 전체 암호화](#) 활성화를 참조하십시오.
  - 암호화에 대해 Kubernetes 인증 방법을 선택하면 [KMS를 사용하여 Kubernetes 인증으로 클러스터 전체 암호화](#) 활성화를 참조하십시오.
  - Vault 서버에서 서명된 인증서를 사용하고 있는지 확인합니다.
2. 최소 노드 시작 요구 사항  
표준 배포 리소스 요구 사항이 충족되지 않으면 OpenShift Data Foundation 클러스터는 최소 구성으로 배포됩니다. 계획 가이드의 [리소스 요구 사항](#) 섹션을 참조하십시오.
3. regional-DR 요구 사항 [Developer Preview]  
Red Hat OpenShift Data Foundation에서 지원하는 재해 복구 기능을 사용하려면 재해 복구 솔루션을 성공적으로 구현하기 위해 다음과 같은 사전 요구 사항이 모두 필요합니다.
  - 유효한 Red Hat OpenShift Data Foundation Advanced 서브스크립션
  - 유효한 Red Hat Advanced Cluster Management for Kubernetes 서브스크립션  
OpenShift Data Foundation의 서브스크립션이 작동하는 방법을 알아보려면 [OpenShift Data Foundation 서브스크립션에 대한 기술 자료 문서](#)를 참조하십시오.
 자세한 요구 사항은 [Regional-DR 요구 사항](#) 및 [RHACM 요구 사항](#)을 참조하십시오.
4. 로컬 스토리지 장치를 사용하여 배포하는 경우 [로컬 스토리지 장치를 사용하여 OpenShift Data Foundation 설치](#)를 위한 [요구사항](#)을 참조하십시오. 동적 스토리지 장치를 사용한 배포에는 적용되지 않습니다.

## 1.1. 로컬 스토리지 장치를 사용하여 OPENSIFT DATA FOUNDATION을 설치하기 위한 요구사항

### 노드 요구 사항

클러스터는 각각 로컬에 연결된 3개 이상의 OpenShift Container Platform 작업자 노드로 구성되어야 합니다.

- 선택한 세 개의 각 노드에는 OpenShift Data Foundation에서 사용할 수 있는 하나 이상의 원시 블록 장치를 사용할 수 있어야 합니다.

- 사용하는 장치는 비어 있어야 합니다. 디스크에 물리 볼륨(PV), 볼륨 그룹(VG) 또는 논리 볼륨(LV)이 디스크에 남아 있지 않아야 합니다.

자세한 내용은 계획 가이드의 [리소스 요구 사항](#) 섹션을 참조하십시오.

### regional-DR 요구 사항 [Developer Preview]

Red Hat OpenShift Data Foundation에서 지원하는 재해 복구 기능을 사용하려면 재해 복구 솔루션을 성공적으로 구현하기 위해 다음과 같은 사전 요구 사항이 모두 필요합니다.

- 유효한 Red Hat OpenShift Data Foundation Advanced 서브스크립션
- 유효한 Red Hat Advanced Cluster Management for Kubernetes 서브스크립션

OpenShift Data Foundation의 서브스크립션이 작동하는 방법을 알아보려면 [OpenShift Data Foundation 서브스크립션에 대한 기술 자료 문서](#)를 참조하십시오.

자세한 요구 사항은 [Regional-DR 요구 사항](#) 및 [RHACM 요구 사항](#)을 참조하십시오.

### Arbiter 확장 클러스터 요구 사항 [기술 프리뷰]

이 경우 단일 클러스터가 두 개의 영역으로 확장되고 세 번째 영역이 중재자의 위치로 확장됩니다. 이는 현재 OpenShift Container Platform 온-프레미스 및 동일한 데이터 센터에 배포하기 위한 기술 프리뷰 기능입니다. 이 솔루션은 여러 데이터 센터에 걸쳐 확장되는 배포에는 권장되지 않습니다. 대신 Metro-DR을 대기 시간이 짧은 여러 데이터 센터에 배포된 데이터 손실 DR 솔루션이 없는 첫 번째 옵션으로 간주합니다.

자세한 요구 사항 및 지침은 [확장 클러스터를 위한 OpenShift Data Foundation](#) 구성을 참조하십시오.



#### 참고

유연한 확장 및 중재는 스케일링 논리가 충돌하므로 동시에 활성화할 수 없습니다. 유연한 확장을 사용하면 OpenShift Data Foundation 클러스터에 한 번에 하나의 노드를 추가할 수 있습니다. Arbiter 클러스터에서는 두 개의 데이터 영역 각각에 하나 이상의 노드를 추가해야 합니다.

### 최소 노드 시작 요구 사항

OpenShift Data Foundation 클러스터는 표준 배포 리소스 요구 사항이 충족되지 않은 경우 최소 구성으로 배포됩니다.

자세한 내용은 계획 가이드의 [리소스 요구 사항](#) 섹션을 참조하십시오.

## 2장. 동적 스토리지 장치를 사용하여 배포

VMware vSphere(디스크 형식: thin)에서 제공하는 동적 스토리지 장치를 사용하여 OpenShift Container Platform에 OpenShift Data Foundation을 배포하면 내부 클러스터 리소스를 생성할 수 있는 옵션이 제공 됩니다. 이렇게 하면 기본 서비스가 내부 프로비저닝되므로 애플리케이션에서 추가 스토리지 클래스를 사용할 수 있습니다.



### 참고

VMware vSphere에서는 내부 및 외부 OpenShift Data Foundation 클러스터가 모두 지원 됩니다 배포 요구 사항에 대한 자세한 내용은 [배포 계획](#)을 참조하십시오.

또한 동적 스토리지 장치를 사용하여 배포하기 위한 다음 단계를 진행하기 전에 [OpenShift Data Foundation 배포 준비](#)에 대한 요구 사항을 해결했는지 확인하십시오.

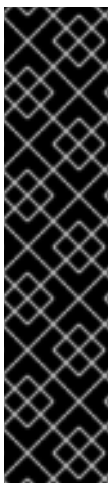
1. [Red Hat OpenShift Data Foundation Operator](#)를 설치합니다.
2. [OpenShift Data Foundation 클러스터](#)를 생성합니다.

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

Red Hat OpenShift Container Platform Operator Hub를 사용하여 Red Hat OpenShift Data Foundation Operator를 설치할 수 있습니다.

#### 사전 요구 사항

- **cluster-admin** 및 Operator 설치 권한이 있는 계정을 사용하여 OpenShift Container Platform 클러스터에 액세스할 수 있습니다.
- Red Hat OpenShift Container Platform 클러스터에 3개 이상의 작업자 노드가 있어야 합니다.
- 추가 리소스 요구 사항은 [배포 계획](#) 가이드를 참조하십시오.



### 중요

- OpenShift Data Foundation의 클러스터 수준 기본 노드 선택기를 재정의해야 하는 경우 명령줄 인터페이스에서 다음 명령을 사용하여 **openshift-storage** 네임스페이스에 대한 빈 노드 선택기를 지정할 수 있습니다(이 경우 openshift-storage 네임스페이스 생성).

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

- 노드에 Red Hat OpenShift Data Foundation 리소스만 예약되도록 **infra** 테인트를 구성합니다. 이를 통해 서브스크립션 비용을 절감할 수 있습니다. 자세한 내용은 [스토리지 리소스 관리 및 할당 가이드](#)의 [Red Hat OpenShift Data Foundation 전용 작업자 노드 사용 방법](#)을 참조하십시오.

#### 절차

1. OpenShift 웹 콘솔에 로그인합니다.
2. **Operators** → **OperatorHub**를 클릭합니다.

3. **OpenShift Data Foundation**을 키워드로 필터링 상자에 스크롤하여 **OpenShift Data Foundation Operator**를 찾습니다.
4. 설치를 클릭합니다.
5. **Operator** 설치 페이지에서 다음 옵션을 설정합니다.
  - a. Channel을 **stable-4.10**으로 업데이트합니다.
  - b. 설치 모드에서 클러스터의 특정 네임스페이스를 선택합니다.
  - c. 설치된 네임스페이스에서 **Operator** 권장 네임스페이스 **openshift-storage**를 선택합니다. 네임스페이스 **openshift-storage**가 없으면 Operator 설치 중에 생성됩니다.
  - d. 승인 전략을 **자동** 또는 **수동**으로 선택합니다.  
 자동 업데이트를 선택하면 OLM(Operator Lifecycle Manager)은 개입 없이 Operator의 실행 중인 인스턴스를 자동으로 업그레이드합니다.  
  
 수동 업데이트를 선택하면 OLM에서 업데이트 요청을 생성합니다. 클러스터 관리자는 Operator를 최신 버전으로 업데이트하기 위해 해당 업데이트 요청을 수동으로 승인해야 합니다.
  - e. **Console** 플러그인에 대해 **Enable** 옵션이 선택되어 있는지 확인합니다.
  - f. 설치를 클릭합니다.

#### 검증 단계

- **OpenShift Data Foundation Operator**에 설치 성공을 나타내는 녹색 눈금이 표시되는지 확인합니다.
- Operator가 성공적으로 설치되면 메시지가 포함된 팝업 **Web console update is available**이 사용자 인터페이스에 표시됩니다. 콘솔 변경 사항을 반영하려면 이 팝업 창에서 **웹 콘솔 새로 고침**을 클릭합니다.
  - 웹 콘솔에서 Operator로 이동하여 **OpenShift Data Foundation**을 사용할 수 있는지 확인합니다.

## 2.2. 토큰 인증 방법을 사용하여 KMS로 클러스터 전체 암호화 활성화

토큰 인증에 대해 Vault에서 키 값 백엔드 경로 및 정책을 활성화하려면 다음 절차를 따르십시오.

#### 사전 요구 사항

- Vault에 대한 관리자 액세스 권한이 있어야 합니다.
- 유효한 Red Hat OpenShift Data Foundation Advanced 서브스크립션. 자세한 내용은 [OpenShift Data Foundation 서브스크립션에 대한 지식베이스 문서](#)를 참조하십시오.
- 나중에 변경할 수 없으므로 이름 지정 규칙을 따르는 백엔드 경로로 고유한 경로 이름을 선택합니다.

#### 절차

1. Vault에서 KV(Key/Value) 백엔드 경로를 활성화합니다.  
Vault KV 시크릿 엔진 API의 경우 버전 1:

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

Vault KV 시크릿 엔진 API의 경우 버전 2:

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

- 다음 명령을 사용하여 시크릿에서 쓰기 또는 삭제 작업을 수행하도록 사용자를 제한하는 정책을 생성합니다.

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

- 위 정책과 일치하는 토큰을 생성합니다.

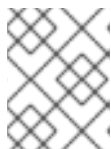
```
$ vault token create -policy=odf -format json
```

## 2.3. KUBERNETES 인증 방법을 사용하여 KMS로 클러스터 전체 암호화 활성화

KMS(Key Management System)를 사용하여 클러스터 전체 암호화에 대한 Kubernetes 인증 방법을 활성화할 수 있습니다.

사전 요구 사항

- Vault에 대한 관리자 액세스 권한이 있어야 합니다.
- 유효한 Red Hat OpenShift Data Foundation Advanced 서브스크립션. 자세한 내용은 [OpenShift Data Foundation 서브스크립션에 대한 지식베이스 문서](#)를 참조하십시오.
- OpenShift Data Foundation Operator는 Operator Hub에서 설치해야 합니다.
- 나중에 변경할 수 없으므로 이름 지정 규칙을 따르는 백엔드 경로로 고유한 경로 이름을 선택합니다.



참고

OpenShift Data Foundation 4.10에서 Kubernetes 인증 방법에서는 Vault 네임스페이스를 사용할 수 없습니다.

절차

1. 서비스 계정을 생성합니다.

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

여기서 `<serviceaccount_name>`은 서비스 계정의 이름을 지정합니다.

예를 들어 다음과 같습니다.

-

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

## 2. clusterrolebindings 및 clusterroles를 생성합니다.

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

예를 들어 다음과 같습니다.

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

## 3. OpenShift Container Platform 버전에 따라 다음 중 하나를 수행합니다.

- OpenShift Container Platform 4.10의 경우:

- 위에서 생성한 serviceaccount(SA)와 연결된 시크릿 이름을 식별합니다.

```
$ VAULT_SA_SECRET_NAME=$(oc -n openshift-storage get sa <SA_NAME> -o
jsonpath="{.secrets[*]['name']}") | grep -o "[^:space:]*-token-[^:space:]*"
```

예를 들어 다음과 같습니다.

```
$ VAULT_SA_SECRET_NAME=$(oc -n openshift-storage get sa odf-vault-auth -o
jsonpath="{.secrets[*]['name']}") | grep -o "[^:space:]*-token-[^:space:]*"
```

- OpenShift Container Platform 4.11의 경우:

- serviceaccount 토큰 및 CA 인증서에 대한 시크릿을 생성합니다.

```
$ 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
```

여기서 **<serviceaccount\_name>**은 이전 단계에서 생성한 서비스 계정입니다.

```
$ VAULT_SA_SECRET_NAME=odf-vault-auth-token
```

## 4. 시크릿에서 토큰 및 CA 인증서를 가져옵니다.

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

## 5. OCP 클러스터 끝점을 검색합니다.

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

## 6. 서비스 계정 발행자를 가져옵니다.

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

## 7. 위의 단계에서 수집된 정보를 사용하여 다음과 같이 Vault에서 Kubernetes 인증 방법을 설정합니다.

```
$ 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"
```



## 중요

issuer가 비어 있을 때 Vault에서 Kubernetes 인증 방법을 구성하려면 다음을 수행합니다.

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

## 8. Vault에서 KV(Key/Value) 백엔드 경로를 활성화합니다. Vault KV 시크릿 엔진 API의 경우 버전 1입니다.

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

Vault KV 시크릿 엔진 API의 경우 버전 2입니다.

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

## 9. 시크릿에서 쓰기 또는 삭제 작업을 수행하도록 사용자를 제한하는 정책을 생성합니다.

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

## 10. 역할을 생성합니다.



```
$ 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
```

스토리지 시스템을 생성하는 동안 KMS 연결 세부 정보를 구성하는 동안 **odf-rook-ceph-op** 역할이 사용됩니다.

```
$ 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. MULTUS 네트워크 생성 [기술 프리뷰]

OpenShift Container Platform은 Multus CNI 플러그인을 사용하여 CNI 플러그인 연결을 허용합니다. 클러스터 설치 중에 기본 Pod 네트워크를 구성할 수 있습니다. 기본 네트워크는 클러스터의 모든 일반 네트워크 트래픽을 처리합니다. 사용 가능한 CNI 플러그인을 기반으로 추가 네트워크를 정의하고 이러한 네트워크 중 하나 이상을 Pod에 연결할 수 있습니다. Pod에 추가 네트워크 인터페이스를 연결하려면 인터페이스 연결 방법을 정의하는 구성을 생성해야 합니다. NetworkAttachmentDefinition CR(사용자 정의 리소스)을 사용하여 각 인터페이스를 지정합니다. 각 NetworkAttachmentDefinition 내부의 CNI 구성은 해당 인터페이스의 생성 방법을 정의합니다.

OpenShift Data Foundation은 macvlan라는 CNI 플러그인을 사용합니다. macvlan 기반 추가 네트워크를 생성하면 호스트의 Pod가 물리적 네트워크 인터페이스를 사용하여 해당 호스트의 다른 호스트 및 Pod와 통신할 수 있습니다. macvlan 기반 추가 네트워크에 연결된 각 pod에는 고유한 MAC 주소가 제공됩니다.

### 중요

Multus 지원은 베어 메탈 및 VMWare 배포에서 지원 및 테스트된 기술 프리뷰 기능입니다. 기술 프리뷰 기능은 Red Hat 프로덕션 서비스 수준 계약(SLA)에서 지원되지 않으며 기능적으로 완전하지 않을 수 있습니다. 따라서 프로덕션 환경에서 사용하는 것은 권장하지 않습니다. 이러한 기능을 사용하면 향후 제품 기능을 조기에 이용할 수 있어 개발 과정에서 고객이 기능을 테스트하고 피드백을 제공할 수 있습니다.

자세한 내용은 [기술 프리뷰 기능 지원 범위를 참조하십시오](#).

### 2.4.1. 네트워크 연결 정의 생성

Multus를 사용하려면 올바른 네트워킹 구성이 이미 작동 중인 클러스터가 필요한 경우 [권장 네트워크 구성 및 Multus 구성에 대한 요구 사항](#)을 참조하십시오. 새로 생성된 NetworkAttachmentDefinition(NAD)은 스토리지 클러스터 설치 중에 선택할 수 있습니다. 이는 스토리지 클러스터 전에 생성해야 하는 이유입니다.

계획 가이드에 자세히 설명된 대로 생성한 Multus 네트워크는 OpenShift Data Foundation 트래픽에 대해 보유하고 있는 사용 가능한 네트워크 인터페이스 수에 따라 달라집니다. 모든 스토리지 트래픽을 두 인터페이스 중 하나(기본 OpenShift SDN에 사용되는 인터페이스 하나)로 분리하거나 스토리지 트래픽을 클라이언트 스토리지 트래픽(퍼블릭) 및 스토리지 복제 트래픽(프라이빗 또는 클러스터)으로 추가로 분리할 수 있습니다.

다음은 동일한 인터페이스에서 모든 스토리지 트래픽, 공용 및 클러스터에 대한 NetworkAttachmentDefinition 예입니다. 모든 예약 가능한 노드에 하나의 추가 인터페이스가 필요합니다(별도의 네트워크 인터페이스에서 OpenShift 기본 SDN).

```

apiVersion: "k8s.cni.cncf.io/v1"
kind: NetworkAttachmentDefinition
metadata:
  name: ocs-public-cluster
  namespace: openshift-storage
spec:
  config: '{
    "cniVersion": "0.3.1",
    "type": "macvlan",
    "master": "ens2",
    "mode": "bridge",
    "ipam": {
      "type": "whereabouts",
      "range": "192.168.1.0/24"
    }
  }'
```



### 참고

모든 네트워크 인터페이스 이름은 Multus 네트워크에 연결된 모든 노드에서 동일해야 합니다(즉, **ocs-public-cluster**의 경우 **ens 2**).

다음은 별도의 Multus 네트워크, 공용 네트워크, 클라이언트 스토리지 트래픽, 복제 트래픽 클러스터 상의 스토리지 트래픽에 대한 **NetworkAttachmentDefinition**의 예입니다. OSD pod를 호스팅하는 OpenShift 노드에 2개의 추가 인터페이스가 필요하고 다른 모든 스케줄링 가능한 노드에 1개의 추가 인터페이스가 필요합니다(별도의 네트워크 인터페이스에 OpenShift 기본 SDN).

```

apiVersion: "k8s.cni.cncf.io/v1"
kind: NetworkAttachmentDefinition
metadata:
  name: ocs-public
  namespace: openshift-storage
spec:
  config: '{
    "cniVersion": "0.3.1",
    "type": "macvlan",
    "master": "ens2",
    "mode": "bridge",
    "ipam": {
      "type": "whereabouts",
      "range": "192.168.1.0/24"
    }
  }'
```

### 예 NetworkAttachmentDefinition:

```

apiVersion: "k8s.cni.cncf.io/v1"
kind: NetworkAttachmentDefinition
metadata:
  name: ocs-cluster
  namespace: openshift-storage
spec:
  config: '{
    "cniVersion": "0.3.1",
```

```
"type": "macvlan",
"master": "ens3",
"mode": "bridge",
"ipam": {
  "type": "whereabouts",
  "range": "192.168.2.0/24"
}
}'
```



### 참고

모든 네트워크 인터페이스 이름은 Multus 네트워크에 연결된 모든 노드에서 동일해야 합니다(즉, `ocs-public`의 경우 `ens2`, `ocs-cluster`의 경우 `ens3`).

## 2.5. OPENSIFT DATA FOUNDATION 클러스터 생성

OpenShift Data Foundation Operator를 설치한 후 OpenShift Data Foundation 클러스터를 생성합니다.

### 사전 요구 사항

- OpenShift Data Foundation Operator는 Operator Hub에서 설치해야 합니다. 자세한 내용은 [OpenShift Data Foundation Operator 설치](#)를 참조하십시오.
- VMware의 VM의 경우 `disk.EnableUUID` 옵션이 `TRUE`로 설정되어 있는지 확인합니다. VM을 구성하려면 vCenter 계정 권한이 있어야 합니다. 자세한 내용은 [Required vCenter 계정 권한](#)을 참조하십시오. `disk.EnableUUID` 옵션을 설정하려면 `Customize hardware` 탭에서 VM Options의 고급 옵션을 사용합니다. 자세한 내용은 [vSphere에 설치](#)를 참조하십시오.
- 선택 사항: 유연성을 위해 두께 프로비저닝된 스토리지를 사용하려면 `Oedthick` 또는 `eager zeroedthick` 디스크 형식의 스토리지 클래스를 생성해야 합니다. 자세한 내용은 [VMware vSphere 개체 정의](#)를 참조하십시오.
- `multus` 지원의 기술 프리뷰 기능을 사용하려면 배포 전에 나중에 클러스터에 연결된 네트워크 연결 정의(NAD)를 생성해야 합니다. 자세한 내용은 [다중 네트워크 플러그인\(Multus\) 지원 및 네트워크 연결 정의 생성](#)을 참조하십시오.

### 절차

1. OpenShift 웹 콘솔에서 Operator → 설치된 Operator를 클릭하여 설치된 모든 Operator를 확인합니다.  
선택한 프로젝트가 `openshift-storage` 인지 확인합니다.
2. OpenShift Data Foundation Operator를 클릭한 다음 스토리지 시스템 만들기를 클릭합니다.
3. 백업 스토리지 페이지에서 다음을 선택합니다.
  - a. 배포 유형 옵션에 대해 `Full Deployment`를 선택합니다.
  - b. 기존 스토리지 클래스 사용 옵션을 선택합니다.
  - c. 스토리지 클래스를 선택합니다.  
기본적으로 이 값은 `thin`으로 설정됩니다. 씩 프로비저닝된 스토리지에 대해 `zeroedthick` 또는 `eagerzeroedthick` 디스크 형식의 스토리지 클래스를 생성한 경우 해당 스토리지 클래스가 기본 `thin` 스토리지 클래스와 함께 나열됩니다.

- d. 다음을 클릭합니다.
4. 용량 및 노드 페이지에서 필요한 정보를 제공합니다.
- a. 드롭다운 목록에서 요청 용량 값을 선택합니다. 기본적으로 **2TiB**로 설정됩니다.



**참고**

초기 스토리지 용량을 선택하면 선택한 사용 가능한 용량(원시 스토리지의 3배)만을 사용하여 클러스터 확장이 수행됩니다.

- b. 노드 선택 섹션에서 사용 가능한 노드를 3개 이상 선택합니다.
- c. 선택 사항: Taint 노드 확인란을 선택하여 OpenShift Data Foundation에 대해 선택한 노드를 전용으로 지정합니다.  
고가용성을 위해 작업자 노드를 3개의 물리적 노드, 랙 또는 장애 도메인에 분산합니다.

동일한 물리 새시에서 두 개의 작업자 노드를 스케줄링하지 않으려면 vCenter 유사성 방지를 사용하여 OpenShift Data Foundation 랙 라벨을 데이터 센터의 물리적 노드 및 랙에 맞추십시오.

선택한 노드가 집계된 30 개의 CPU 및 72GiB RAM의 OpenShift Data Foundation 클러스터 요구 사항과 일치하지 않으면 최소 클러스터가 배포됩니다. 최소 노드 요구 사항은 계획 가이드의 [리소스 요구 사항](#) 섹션을 참조하십시오.

테인트 노드 확인란을 선택하여 OpenShift Data Foundation 전용 노드를 선택합니다.

- d. 다음을 클릭합니다.
5. 선택 사항: 보안 및 네트워크 페이지에서 요구 사항에 따라 다음을 구성합니다.
- a. 암호화를 활성화하려면 블록 및 파일 스토리지에 데이터 암호화 사용을 선택합니다.
  - b. 암호화 수준 중 하나 또는 둘 다를 선택합니다.
    - 클러스터 전체 암호화  
전체 클러스터(블록 및 파일)를 암호화합니다.
    - 스토리지 클래스 암호화  
암호화 활성화된 스토리지 클래스를 사용하여 암호화된 영구 볼륨(블록만 해당)을 생성합니다.
  - c. 외부 키 관리 서비스에 연결 확인란을 선택합니다. 이는 클러스터 전체 암호화의 경우 선택 사항입니다.
    - i. 키 관리 서비스 공급자는 기본적으로 **Vault**로 설정됩니다.
    - ii. 인증 방법을 선택합니다.

**토큰 인증 방법 사용**

- Vault 서버의 고유한 연결 이름, 호스트 주소 ('https://<hostname 또는 ip>'), 포트 번호 및 토큰을 입력합니다.
- 고급 설정을 확장하여 Vault 구성에 따라 추가 설정 및 인증서 세부 정보를 입력합니다.

- OpenShift Data Foundation 전용 및 고유한 백엔드 경로에 키 값 시크릿 경로를 입력합니다.
- 선택 사항: TLS 서버 이름 및 Vault Enterprise Namespace를 입력합니다.
- 각 PEM 인코딩 인증서 파일을 업로드하여 CA 인증서, 클라이언트 인증서 및 클라이언트 개인 키를 제공합니다.
- 저장을 클릭합니다.

#### Kubernetes 인증 방법 사용

- 자격 증명 모음 연결 이름, Vault 서버의 호스트주소 ('https://<hostname 또는 ip>'), 포트 번호 및 역할 이름을 입력합니다.
  - 고급 설정을 확장하여 Vault 구성에 따라 추가 설정 및 인증서 세부 정보를 입력합니다.
    - OpenShift Data Foundation 전용 및 고유한 백엔드 경로에 키 값 시크릿 경로를 입력합니다.
    - 선택 사항: 해당하는 경우 TLS 서버 이름 및 인증 경로를 입력합니다.
    - 각 PEM 인코딩 인증서 파일을 업로드하여 CA 인증서, 클라이언트 인증서 및 클라이언트 개인 키를 제공합니다.
    - 저장을 클릭합니다.
- d. 여러 네트워크 인터페이스를 사용하려는 경우 사용자 지정 (Multus) 네트워크를 선택하고 단일 네트워크를 사용하는 경우 Default (SDN)를 선택합니다.
- i. 드롭다운에서 공용 네트워크 인터페이스를 선택합니다.
  - ii. 드롭다운에서 클러스터 네트워크 인터페이스를 선택합니다.



#### 참고

하나의 추가 네트워크 인터페이스만 사용하는 경우 공용 네트워크 인터페이스에 대해 단일 **NetworkAttachmentDefinition** (즉, **ocs-public-cluster**)를 선택하고 클러스터 네트워크 인터페이스를 비워 둡니다.

e. 다음을 클릭합니다.

6. 검토 및 생성 페이지에서 구성 세부 정보를 검토합니다.  
구성 설정을 수정하려면 뒤로를 클릭합니다.
7. 스토리지 시스템 생성을 클릭합니다.

#### 검증 단계

- 설치된 스토리지 클러스터의 최종 상태를 확인하려면 다음을 수행합니다.
  - a. OpenShift 웹 콘솔에서 설치된 Operator → OpenShift Data Foundation → 스토리지 시스템 → **ocs-storagecluster-storage-system** → Resources 로 이동합니다.
  - b. **StorageCluster**의 Status가 **Ready** 이고 옆에 녹색 눈금이 표시되어 있는지 확인합니다.

1. OpenShift Data Foundation의 모든 구성 요소가 성공적으로 설치되었는지 확인하려면 [OpenShift Data Foundation 배포 확인](#)을 참조하십시오.
2. 다중 네트워킹(Multus)을 확인하려면 [Multus 네트워킹 확인](#)을 참조하십시오.

#### 추가 리소스

Overprovision Control 경고를 활성화하려면 모니터링 가이드의 [경고](#)를 참조하십시오.

## 3장. 로컬 스토리지 장치를 사용하여 배포

로컬 스토리지 장치를 사용하여 OpenShift Container Platform에 OpenShift Data Foundation을 배포하면 내부 클러스터 리소스를 생성할 수 있는 옵션이 제공됩니다. 이렇게 하면 기본 서비스가 내부 프로비저닝 되므로 애플리케이션에서 추가 스토리지 클래스를 사용할 수 있습니다.

이 섹션을 사용하여 OpenShift Container Platform이 이미 설치된 VMware에 OpenShift Data Foundation을 배포합니다.

또한 다음 단계를 진행하기 전에 [OpenShift Data Foundation 배포 준비](#)에 있는 요구 사항을 해결했는지 확인하십시오.

1. [로컬 스토리지 Operator 설치](#)
2. [Red Hat OpenShift Data Foundation Operator를 설치합니다.](#)
3. [OpenShift Data Foundation 클러스터를 생성합니다.](#)

### 3.1. 로컬 스토리지 OPERATOR 설치

로컬 스토리지 장치에서 Red Hat OpenShift Data Foundation 클러스터를 생성하기 전에 Operator Hub에서 Local Storage Operator를 설치합니다.

절차

1. OpenShift 웹 콘솔에 로그인합니다.
2. Operators → OperatorHub를 클릭합니다.
3. 키워드로 필터링에 로컬 스토리지를 입력합니다. Operator 목록에서 Local Storage Operator를 찾아서 클릭합니다.
4. Operator 설치 페이지에서 다음 옵션을 설정합니다.
  - a. 채널을 **4.10** 또는 **stable**로 업데이트합니다.
  - b. 설치 모드에서 클러스터의 특정 네임스페이스를 선택합니다.
  - c. 설치된 네임스페이스를 Operator 권장 네임스페이스 **openshift-local-storage**를 선택합니다.
  - d. 승인을 자동으로 업데이트합니다.
5. 설치를 클릭합니다.

검증 단계

- Local Storage Operator에 성공적인 설치를 나타내는 녹색 눈금이 표시되는지 확인합니다.

### 3.2. RED HAT OPENSIFT DATA FOUNDATION OPERATOR 설치

Red Hat OpenShift Container Platform Operator Hub를 사용하여 Red Hat OpenShift Data Foundation Operator를 설치할 수 있습니다.

사전 요구 사항

- **cluster-admin** 및 **Operator** 설치 권한이 있는 계정을 사용하여 **OpenShift Container Platform** 클러스터에 액세스할 수 있습니다.
- Red Hat OpenShift Container Platform 클러스터에 3개 이상의 작업자 노드가 있어야 합니다.
- 추가 리소스 요구 사항은 **배포 계획** 가이드를 참조하십시오.



**중요**

- OpenShift Data Foundation의 클러스터 수준 기본 노드 선택기를 재정의해야 하는 경우 명령줄 인터페이스에서 다음 명령을 사용하여 **openshift-storage** 네임스페이스에 대한 빈 노드 선택기를 지정할 수 있습니다(이 경우 **openshift-storage** 네임스페이스 생성).

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

- 노드에 Red Hat OpenShift Data Foundation 리소스만 예약되도록 **infra** 테인트를 구성합니다. 이를 통해 서브스크립션 비용을 절감할 수 있습니다. 자세한 내용은 **스토리지 리소스 관리 및 할당 가이드**의 **Red Hat OpenShift Data Foundation 전용 작업자 노드 사용 방법**을 참조하십시오.

**절차**

1. OpenShift 웹 콘솔에 로그인합니다.
2. Operators → OperatorHub를 클릭합니다.
3. **OpenShift Data Foundation**을 키워드로 필터링 상자에 스크롤하여 OpenShift Data Foundation Operator를 찾습니다.
4. 설치를 클릭합니다.
5. Operator 설치 페이지에서 다음 옵션을 설정합니다.
  - a. Channel을 **stable-4.10**으로 업데이트합니다.
  - b. 설치 모드에서 클러스터의 특정 네임스페이스를 선택합니다.
  - c. 설치된 네임스페이스에서 Operator 권장 네임스페이스 **openshift-storage**를 선택합니다. 네임스페이스 **openshift-storage**가 없으면 Operator 설치 중에 생성됩니다.
  - d. 승인 전략을 자동 또는 수동으로 선택합니다.  
 자동 업데이트를 선택하면 OLM(Operator Lifecycle Manager)은 개입 없이 Operator의 실행 중인 인스턴스를 자동으로 업그레이드합니다.  
  
 수동 업데이트를 선택하면 OLM에서 업데이트 요청을 생성합니다. 클러스터 관리자는 Operator를 최신 버전으로 업데이트하기 위해 해당 업데이트 요청을 수동으로 승인해야 합니다.
  - e. Console 플러그인에 대해 **Enable** 옵션이 선택되어 있는지 확인합니다.
  - f. 설치를 클릭합니다.

**검증 단계**

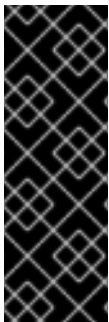


- OpenShift Data Foundation Operator에 설치 성공을 나타내는 녹색 눈금이 표시되는지 확인합니다.
- Operator가 성공적으로 설치되면 메시지가 포함된 팝업 **Web console update is available**이 사용자 인터페이스에 표시됩니다. 콘솔 변경 사항을 반영하려면 이 팝업 창에서 웹 콘솔 새로 고침을 클릭합니다.
  - 웹 콘솔에서 Operator로 이동하여 OpenShift Data Foundation 을 사용할 수 있는지 확인합니다.

### 3.3. MULTUS 네트워크 생성 [기술 프리뷰]

OpenShift Container Platform은 Multus CNI 플러그인을 사용하여 CNI 플러그인 연결을 허용합니다. 클러스터 설치 중에 기본 Pod 네트워크를 구성할 수 있습니다. 기본 네트워크는 클러스터의 모든 일반 네트워크 트래픽을 처리합니다. 사용 가능한 CNI 플러그인을 기반으로 추가 네트워크를 정의하고 이러한 네트워크 중 하나 이상을 Pod에 연결할 수 있습니다. Pod에 추가 네트워크 인터페이스를 연결하려면 인터페이스 연결 방법을 정의하는 구성을 생성해야 합니다. NetworkAttachmentDefinition CR(사용자 정의 리소스)을 사용하여 각 인터페이스를 지정합니다. 각 NetworkAttachmentDefinition 내부의 CNI 구성은 해당 인터페이스의 생성 방법을 정의합니다.

OpenShift Data Foundation은 macvlan라는 CNI 플러그인을 사용합니다. macvlan 기반 추가 네트워크를 생성하면 호스트의 Pod가 물리적 네트워크 인터페이스를 사용하여 해당 호스트의 다른 호스트 및 Pod와 통신할 수 있습니다. macvlan 기반 추가 네트워크에 연결된 각 pod에는 고유 한 MAC 주소가 제공됩니다.



#### 중요

Multus 지원은 베어 메탈 및 VMWare 배포에서 지원 및 테스트된 기술 프리뷰 기능입니다. 기술 프리뷰 기능은 Red Hat 프로덕션 서비스 수준 계약(SLA)에서 지원되지 않으며 기능적으로 완전하지 않을 수 있습니다. 따라서 프로덕션 환경에서 사용하는 것은 권장하지 않습니다. 이러한 기능을 사용하면 향후 제품 기능을 조기에 이용할 수 있어 개발 과정에서 고객이 기능을 테스트하고 피드백을 제공할 수 있습니다.

자세한 내용은 [기술 프리뷰 기능 지원 범위를 참조하십시오](#).

#### 3.3.1. 네트워크 연결 정의 생성

Multus를 사용하려면 올바른 네트워킹 구성이 이미 작동 중인 클러스터가 필요한 경우 [권장 네트워크 구성 및 Multus 구성에 대한 요구 사항](#)을 참조하십시오. 새로 생성된 NetworkAttachmentDefinition(NAD)은 스토리지 클러스터 설치 중에 선택할 수 있습니다. 이는 스토리지 클러스터 전에 생성해야 하는 이유입니다.

계획 가이드에 자세히 설명된 대로 생성한 Multus 네트워크는 OpenShift Data Foundation 트래픽에 대해 보유하고 있는 사용 가능한 네트워크 인터페이스 수에 따라 달라집니다. 모든 스토리지 트래픽을 두 인터페이스 중 하나(기본 OpenShift SDN에 사용되는 인터페이스 하나)로 분리하거나 스토리지 트래픽을 클라이언트 스토리지 트래픽(퍼블릭) 및 스토리지 복제 트래픽(프라이빗 또는 클러스터)으로 추가로 분리할 수 있습니다.

다음은 동일한 인터페이스에서 모든 스토리지 트래픽, 공용 및 클러스터에 대한 **NetworkAttachmentDefinition** 예입니다. 모든 예약 가능한 노드에 하나의 추가 인터페이스가 필요합니다 (별도의 네트워크 인터페이스에서 OpenShift 기본 SDN).

```
apiVersion: "k8s.cni.cncf.io/v1"
kind: NetworkAttachmentDefinition
metadata:
  name: ocs-public-cluster
  namespace: openshift-storage
```

```
spec:
  config: '{
    "cniVersion": "0.3.1",
    "type": "macvlan",
    "master": "ens2",
    "mode": "bridge",
    "ipam": {
      "type": "whereabouts",
      "range": "192.168.1.0/24"
    }
  }'
```



## 참고

모든 네트워크 인터페이스 이름은 Multus 네트워크에 연결된 모든 노드에서 동일해야 합니다(즉, **ocs-public-cluster**의 경우 **ens 2**).

다음은 별도의 Multus 네트워크, 공용 네트워크, 클라이언트 스토리지 트래픽, 복제 트래픽 클러스터 상의 스토리지 트래픽에 대한 **NetworkAttachmentDefinition**의 예입니다. OSD pod를 호스팅하는 OpenShift 노드에 2개의 추가 인터페이스가 필요하고 다른 모든 스케줄링 가능한 노드에 1개의 추가 인터페이스가 필요합니다(별도의 네트워크 인터페이스에 OpenShift 기본 SDN).

```
apiVersion: "k8s.cni.cncf.io/v1"
kind: NetworkAttachmentDefinition
metadata:
  name: ocs-public
  namespace: openshift-storage
spec:
  config: '{
    "cniVersion": "0.3.1",
    "type": "macvlan",
    "master": "ens2",
    "mode": "bridge",
    "ipam": {
      "type": "whereabouts",
      "range": "192.168.1.0/24"
    }
  }'
```

## 예 NetworkAttachmentDefinition:

```
apiVersion: "k8s.cni.cncf.io/v1"
kind: NetworkAttachmentDefinition
metadata:
  name: ocs-cluster
  namespace: openshift-storage
spec:
  config: '{
    "cniVersion": "0.3.1",
    "type": "macvlan",
    "master": "ens3",
    "mode": "bridge",
    "ipam": {
      "type": "whereabouts",
```

```
"range": "192.168.2.0/24"
```

```
}  
}'
```



#### 참고

모든 네트워크 인터페이스 이름은 Multus 네트워크에 연결된 모든 노드에서 동일해야 합니다(즉, **ocs-public**의 경우 **ens2**, **ocs-cluster**의 경우 **ens3**).

### 3.4. VMWARE VSPHERE에서 OPENSIFT DATA FOUNDATION 클러스터 생성

VMware vSphere는 다음 세 가지 유형의 로컬 스토리지를 지원합니다.

- VMDK(가상 머신 디스크)
- RDM(원시 장치 매핑)
- VMDirectPath I/O

#### 사전 요구 사항

- [로컬 스토리지 장치 섹션을 사용하여 OpenShift Data Foundation 설치 요구 사항](#)의 모든 요구 사항이 충족되었는지 확인합니다.
- VMware에서 로컬 스토리지 장치를 사용하려면 최소 3개의 작업자 노드가 있고 각 노드에 동일한 스토리지 유형의 크기가 연결되어 있어야 합니다.
- VMware vSphere의 VM의 경우 **disk.EnableUUID** 옵션이 **TRUE** 로 설정되어 있는지 확인합니다. VM을 구성하려면 vCenter 계정 권한이 있어야 합니다. 자세한 내용은 [Required vCenter 계정 권한](#)을 참조하십시오. **disk.EnableUUID** 옵션을 설정하려면 **Customize hardware** 탭에서 **VM Options**의 고급 옵션을 사용합니다. 자세한 내용은 [vSphere에 설치](#)를 참조하십시오.
- **multus** 지원의 기술 프리뷰 기능을 사용하려면 배포 전에 나중에 클러스터에 연결될 네트워크 연결 정의(NAD)를 생성해야 합니다. 자세한 내용은 [다중 네트워크 플러그인\(Multus\) 지원 및 네트워크 연결 정의 생성](#)을 참조하십시오.

#### 절차

1. OpenShift 웹 콘솔에서 **Operator** → **설치된 Operator**를 클릭하여 설치된 모든 **Operator**를 확인합니다.  
선택한 프로젝트가 **openshift-storage** 인지 확인합니다.
2. **OpenShift Data Foundation Operator**를 클릭한 다음 **스토리지 시스템 만들기**를 클릭합니다.
3. 백업 스토리지 페이지에서 다음을 수행합니다.
  - a. 배포 유형 옵션에 대해 **Full Deployment**를 선택합니다.
  - b. 로컬 스토리지 장치 옵션을 사용하여 새 **StorageClass 만들기** 옵션을 선택합니다.
  - c. 다음을 클릭합니다.



**참고**

아직 설치되지 않은 경우 **Local Storage Operator**를 설치하라는 메시지가 표시됩니다. 설치를 클릭하고 **Local Storage Operator 설치**에 설명된 절차를 따릅니다.

4. 로컬 볼륨 세트 생성 페이지에서 다음 정보를 제공합니다.

- a. 로컬 볼륨 세트의 이름과 스토리지 클래스를 입력합니다.  
기본적으로 스토리지 클래스 이름에 로컬 볼륨 세트 이름이 표시됩니다. 이름을 변경할 수 있습니다.
- b. 다음 명령 중 하나를 선택합니다.
  - 모든 노드의 디스크는 모든 노드에서 선택한 필터와 일치하는 사용 가능한 디스크를 사용합니다.
  - 선택한 노드의 디스크가 선택한 노드에서만 선택한 필터와 일치하는 사용 가능한 디스크를 사용합니다.



**중요**

- 3개 이상의 노드로 생성한 스토리지 클러스터가 3개 이상의 가용성 영역의 최소 요구 사항보다 적은 경우에만 유연한 확장 기능을 사용할 수 있습니다.  
유연한 확장에 대한 자세한 내용은 스토리지 확장 가이드의 [YAML을 사용하여 용량 추가](#)를 참조하십시오.
- 선택한 노드가 집계된 30 개의 CPU 및 72GiB RAM의 OpenShift Data Foundation 클러스터 요구 사항과 일치하지 않으면 최소 클러스터가 배포됩니다.  
최소 노드 요구 사항은 계획 가이드의 [리소스 요구 사항](#) 섹션을 참조하십시오.

- c. 사용 가능한 디스크 유형 목록에서 **SSD/NVMe**을 선택합니다.
- d. 고급 섹션을 확장하고 다음 옵션을 설정합니다.

볼륨 모드	블록은 기본적으로 선택됩니다.
장치 유형	드롭다운 목록에서 하나 이상의 장치 유형을 선택합니다.
디스크 크기	장치에 대해 최소 크기 100GB와 포함되어야 하는 장치의 사용 가능한 최대 크기를 설정합니다.
최대 디스크 제한	이는 노드에서 생성할 수 있는 최대 PV 수를 나타냅니다. 이 필드가 비어 있으면 일치하는 노드에서 사용 가능한 모든 디스크에 PV가 생성됩니다.

- e. 다음을 클릭합니다.  
로컬 볼륨 세트 생성이 표시되는지 확인하는 팝업이 표시됩니다.
- f. 계속하려면 **예**를 클릭합니다.

5. 용량 및 노드 페이지에서 다음을 구성합니다.

- a. 사용 가능한 원시 용량은 스토리지 클래스와 연결된 모든 디스크에 따라 용량 값으로 채워집니다. 이 작업을 수행하는 데 시간이 다소 걸립니다. 선택한 노드 목록에는 스토리지 클래스를 기반으로 하는 노드가 표시됩니다.
  - b. 선택 사항: Taint 노드 확인란을 선택하여 OpenShift Data Foundation에 대해 선택한 노드를 전용으로 지정합니다.
  - c. 다음을 클릭합니다.
6. 선택 사항: 보안 및 네트워크 페이지에서 요구 사항에 따라 다음을 구성합니다.
- a. 암호화를 활성화하려면 블록 및 파일 스토리지에 데이터 암호화 사용을 선택합니다.
  - b. 다음 암호화 수준 중 하나를 선택합니다.
    - 전체 클러스터를 암호화하는 클러스터 전체 (블록 및 파일) 암호화
    - 암호화가 활성화된 스토리지 클래스를 사용하여 암호화된 영구 볼륨(블록만 해당)을 생성하는 스토리지 클래스 암호화입니다.
  - c. 외부 키 관리 서비스에 연결을 선택합니다. 이는 클러스터 전체 암호화의 경우 선택 사항입니다.
    - i. 키 관리 서비스 공급자는 기본적으로 **Vault**로 설정됩니다.
    - ii. 인증 방법을 선택합니다.
 

토큰 인증 방법 사용

      - Vault 서버의 고유한 연결 이름, 호스트 주소 ('https://<hostname 또는 ip>'), 포트 번호 및 토큰을 입력합니다.
      - 고급 설정을 확장하여 Vault 구성에 따라 추가 설정 및 인증서 세부 정보를 입력합니다.
        - OpenShift Data Foundation 전용 및 고유한 백엔드 경로에 키 값 시크릿 경로를 입력합니다.
        - 선택 사항: TLS 서버 이름 및 Vault Enterprise Namespace를 입력합니다.
        - 각 PEM 인코딩 인증서 파일을 업로드하여 CA 인증서, 클라이언트 인증서 및 클라이언트 개인 키를 제공합니다.
        - 저장을 클릭합니다.

Kubernetes 인증 방법 사용

      - 자격 증명 모음 연결 이름, Vault 서버의 호스트 주소 ('https://<hostname 또는 ip>'), 포트 번호 및 역할 이름을 입력합니다.
      - 고급 설정을 확장하여 Vault 구성에 따라 추가 설정 및 인증서 세부 정보를 입력합니다.
        - OpenShift Data Foundation 전용 및 고유한 백엔드 경로에 키 값 시크릿 경로를 입력합니다.
        - 선택 사항: 해당하는 경우 TLS 서버 이름 및 인증 경로를 입력합니다.

- 각 PEM 인코딩 인증서 파일을 업로드하여 CA 인증서, 클라이언트 인증서 및 클라이언트 개인 키를 제공합니다.
- 저장을 클릭합니다.

d. 다음 명령 중 하나를 선택합니다.

- 단일 네트워크를 사용하는 경우 **Default (SDN)** 를 선택합니다.
- 여러 네트워크 인터페이스를 사용하는 경우 **Custom (Multus)** 을 선택합니다.
  - i. 드롭다운에서 **공용 네트워크 인터페이스**를 선택합니다.
  - ii. 드롭다운에서 **클러스터 네트워크 인터페이스**를 선택합니다.



**참고**

하나의 추가 네트워크 인터페이스만 사용하는 경우 공용 네트워크 인터페이스에 대해 단일 **NetworkAttachmentDefinition** (즉, **ocs-public-cluster**)를 선택하고 클러스터 네트워크 인터페이스를 비워 둡니다.

e. 다음을 클릭합니다.

7. 검토 및 생성 페이지에서 구성 세부 정보를 검토합니다.

- 구성 설정을 수정하려면 뒤로 이동하여 이전 구성 페이지로 돌아갑니다.

8. 스토리지 시스템 생성을 클릭합니다.

**검증 단계**

- 설치된 스토리지 클러스터의 최종 상태를 확인하려면 다음을 수행합니다.
  - a. OpenShift 웹 콘솔에서 설치된 Operator → OpenShift Data Foundation → 스토리지 시스템 → **ocs-storagecluster-storagesystem** → Resources 로 이동합니다.
  - b. **StorageCluster**의 **Status**가 **Ready** 이고 옆에 녹색 눈금이 표시되어 있는지 확인합니다.
- 스토리지 클러스터에서 유연한 확장이 활성화되어 있는지 확인하려면 다음 단계를 수행합니다 (rbiter 모드의 경우 유연한 확장은 비활성화됨).
  - 1. OpenShift 웹 콘솔에서 설치된 Operator → OpenShift Data Foundation → 스토리지 시스템 → **ocs-storagecluster-storagesystem** → Resources 로 이동합니다.
  - 2. YAML 탭의 **spec** 섹션에서 **flexibleScaling** 키를 검색하고 **status** 섹션에서 **failureDomain**을 검색합니다. **flexible scaling**이 true이고 **failureDomain** 이 host로 설정되면 유연한 확장 기능이 활성화됩니다.

```
spec:
  flexibleScaling: true
  [...]
status:
  failureDomain: host
```

- OpenShift Data Foundation의 모든 구성 요소가 성공적으로 설치되었는지 확인하려면 [OpenShift Data Foundation 배포 확인](#)을 참조하십시오.

- 다중 네트워킹(Multus)을 확인하려면 [Multus 네트워킹 확인](#)을 참조하십시오.

#### 추가 리소스

- 초기 클러스터의 용량을 확장하려면 [스토리지 확장 가이드](#)를 참조하십시오.

## 4장. OPENSIFT DATA FOUNDATION 배포 확인

이 섹션을 사용하여 OpenShift Data Foundation이 올바르게 배포되었는지 확인합니다.

### 4.1. POD 상태 확인

절차

1. OpenShift 웹 콘솔에서 워크로드 → Pod를 클릭합니다.
2. 프로젝트 드롭다운 목록에서 **openshift-storage** 를 선택합니다.



#### 참고

기본 프로젝트 표시 옵션이 비활성화된 경우 토글 버튼을 사용하여 모든 기본 프로젝트를 나열합니다.

각 구성 요소에 대해 예상되는 Pod 수와 노드 수에 따라 달라지는 방법에 대한 자세한 내용은 표 4.1. "OpenShift Data Foundation 클러스터에 해당하는 Pod" 을 참조하십시오.

3. 실행 및 완료 탭을 클릭하여 다음 Pod가 **Running** 및 **Completed** 상태인지 확인합니다.

표 4.1. OpenShift Data Foundation 클러스터에 해당하는 Pod

구성 요소	해당 Pod
OpenShift Data Foundation Operator	<ul style="list-style-type: none"> <li>● <b>OCS-operator-*</b> (모든 작업자 노드에 1 Pod)</li> <li>● <b>OCS-metrics-exporter-*</b> (모든 작업자 노드에 1 Pod)</li> <li>● <b>ODF-operator-controller-manager-*</b> (모든 작업자 노드에 1 Pod)</li> <li>● <b>ODF-console-*</b> (모든 작업자 노드에 1 Pod)</li> <li>● <b>CSI-addons-controller-manager-*</b> (1개의 작업자 노드에서 Pod)</li> </ul>
Rook-ceph Operator	<p><b>rook-ceph-operator-*</b></p> <p>(모든 작업자 노드에 1 Pod)</p>



구성 요소	해당 Pod
Multicloud Object Gateway	<ul style="list-style-type: none"> <li>● <b>NooBaa-operator-*</b> (모든 작업자 노드에 1 Pod)</li> <li>● <b>NooBaa-core-*</b> (모든 스토리지 노드에 1 Pod)</li> <li>● <b>NooBaa-db-pg-*</b> (모든 스토리지 노드에 1 Pod)</li> <li>● <b>NooBaa-endpoint-*</b> (모든 스토리지 노드에 1 Pod)</li> </ul>
MON	<p><b>rook-ceph-mon-*</b></p> <p>(스토리지 노드에 분산된 3 Pod)</p>
MGR	<p><b>rook-ceph-mgr-*</b></p> <p>(모든 스토리지 노드에 1 Pod)</p>
MDS	<p><b>rook-ceph-mds-ocs-storagecluster-cephfilesystem-*</b></p> <p>(스토리지 노드에 분산된 2 Pod)</p>
RGW	<p><b>rook-ceph-rgw-ocs-storagecluster-cephobjectstore-*</b> (모든 스토리지 노드에 1 Pod)</p>
CSI	<ul style="list-style-type: none"> <li>● <b>cephfs</b> <ul style="list-style-type: none"> <li>○ <b>CSI-cephfsplugin-*</b> (모든 작업자 노드에 1 Pod)</li> <li>○ <b>CSI-cephfsplugin-provisioner-*</b> (작업자 노드에 분산된 2 Pod)</li> </ul> </li> <li>● <b>rbd</b> <ul style="list-style-type: none"> <li>○ <b>CSI-rbdplugin-*</b> (각 작업자 노드에서 1 Pod)</li> <li>○ <b>CSI-rbdplugin-provisioner-*</b> (2 작업자 노드에 분산된 2 Pod)</li> </ul> </li> </ul>
rook-ceph-crashcollector	<p><b>rook-ceph-crashcollector-*</b></p> <p>(각 스토리지 노드에서 1 pod)</p>

구성 요소	해당 Pod
OSD	<ul style="list-style-type: none"> <li>● <b>rook-ceph-osd-*</b> (각 장치의 1Pod)</li> <li>● <b>rook-ceph-osd-prepare-ocs-deviceset-*</b> (각 장치의 1Pod)</li> </ul>

## 4.2. OPENSIFT DATA FOUNDATION 클러스터 상태 확인

### 절차

1. OpenShift 웹 콘솔에서 스토리지 → 데이터 기반 을 클릭합니다.
2. 개요 탭의 상태 카드에서 스토리지 시스템을 클릭한 다음 해당 팝업에서 스토리지 시스템 링크를 클릭합니다.
3. 블록 및 파일 탭의 상태 카드에서 스토리지 클러스터에 녹색 체크 표시가 있는지 확인합니다.
4. 세부 정보 카드에서 클러스터 정보가 표시되는지 확인합니다.

블록 및 파일 대시보드를 사용하는 OpenShift Data Foundation 클러스터의 상태에 대한 자세한 내용은 [OpenShift Data Foundation 모니터링](#) 을 참조하십시오.

## 4.3. MULTICLOUD OBJECT GATEWAY의 상태 확인

### 절차

1. OpenShift 웹 콘솔에서 스토리지 → 데이터 기반 을 클릭합니다.
2. 개요 탭의 상태 카드에서 스토리지 시스템을 클릭한 다음 해당 팝업에서 스토리지 시스템 링크를 클릭합니다.
  - a. Object 탭의 상태 카드에서 *Object Service* 및 *Data Resiliency* 모두 녹색 눈금이 있는지 확인합니다.
  - b. 세부 정보 카드에 MCG 정보가 표시되는지 확인합니다.

오브젝트 서비스 대시보드를 사용하는 OpenShift Data Foundation 클러스터의 상태에 대한 자세한 내용은 [OpenShift Data Foundation 모니터링](#) 을 참조하십시오.

## 4.4. OPENSIFT DATA FOUNDATION 특정 스토리지 클래스가 있는지 확인

### 절차

1. OpenShift 웹 콘솔의 왼쪽 창에서 스토리지 → 스토리지 클래스를 클릭합니다.
2. OpenShift Data Foundation 클러스터 생성과 함께 다음 스토리지 클래스가 생성되었는지 확인합니다.
  - **ocs-storagecluster-ceph-rbd**
  - **ocs-storagecluster-cephfs**

- `openshift-storage.noobaa.io`
- `ocs-storagecluster-ceph-rgw`

## 4.5. MULTUS 네트워킹 확인

Multus가 클러스터에서 작동하는지 확인하려면 Multus 네트워킹을 확인합니다.

### 절차

네트워크 구성 선택 사항에 따라 OpenShift Data Foundation Operator는 다음 중 하나를 수행합니다.

- 공용 네트워크 인터페이스에 대해 단일 NetworkAttachmentDefinition(예: `ocs-public-cluster`)만 선택한 경우 애플리케이션 pod와 OpenShift Data Foundation 클러스터 간의 트래픽이 이 네트워크에서 수행됩니다. 또한 클러스터는 OSD 간 복제 및 트래픽 재조정에 이 네트워크를 사용하도록 자체 구성됩니다.
- 스토리지 클러스터 설치 중에 공용 네트워크 인터페이스와 클러스터 네트워크 인터페이스에 대해 각각 NetworkAttachmentDefinitions(예: `ocs-public` 및 `ocs-cluster`)를 선택한 경우, OSD 간 트래픽 복제 및 균형 재조정을 위해 클라이언트 스토리지 트래픽이 공용 네트워크와 클러스터 네트워크에 있게 됩니다.

네트워크 구성이 올바른지 확인하려면 다음을 완료합니다.

OpenShift 콘솔에서 설치된 Operator → OpenShift Data Foundation → 스토리지 시스템 → `ocs-storagecluster-storagesystem` → 리소스 → `ocs-storagecluster` 로 이동합니다.

YAML 탭에서 `spec` 섹션에서 `rnetwork`를 검색하고 네트워크 인터페이스 선택 사항에 대한 구성이 올바른지 확인합니다. 이 예에서는 클라이언트 스토리지 트래픽을 스토리지 복제 트래픽과 분리하기 위한 것입니다.

샘플 출력:

```
[..]
spec:
  [..]
  network:
    ipFamily: IPv4
    provider: multus
    selectors:
      cluster: openshift-storage/ocs-cluster
      public: openshift-storage/ocs-public
  [..]
```

명령줄 인터페이스를 사용하여 네트워크 구성이 올바른지 확인하려면 다음 명령을 실행합니다.

```
$ oc get storagecluster ocs-storagecluster \
-n openshift-storage \
-o=jsonpath='{.spec.network}'
```

샘플 출력:

```
{"ipFamily":"IPv4","provider":"multus","selectors":{"cluster":"openshift-storage/ocs-cluster","public":"openshift-storage/ocs-public"}}
```

## OSD pod에서 올바른 네트워크를 사용하고 있는지 확인

**openshift-storage** 네임스페이스에서 OSD pod 중 하나를 사용하여 Pod가 올바른 네트워크에 연결되어 있는지 확인합니다. 이 예에서는 클라이언트 스토리지 트래픽을 스토리지 복제 트래픽과 분리하기 위한 것입니다.



### 참고

둘 다 생성되는 경우 OSD pod만 Multus 공용 및 클러스터 네트워크에 연결됩니다. 기타 모든 OCS Pod는 Multus 공용 네트워크에 연결됩니다.

```
$ oc get -n openshift-storage $(oc get pods -n openshift-storage -o name -l app=rook-ceph-osd | grep 'osd-0') -o=jsonpath='{.metadata.annotations.k8s.v1.cni.cncf.io/network-status}'
```

샘플 출력:

```
[[
  {
    "name": "openshift-sdn",
    "interface": "eth0",
    "ips": [
      "10.129.2.30"
    ],
    "default": true,
    "dns": {}
  }, {
    "name": "openshift-storage/ocs-cluster",
    "interface": "net1",
    "ips": [
      "192.168.2.1"
    ],
    "mac": "e2:04:c6:81:52:f1",
    "dns": {}
  }, {
    "name": "openshift-storage/ocs-public",
    "interface": "net2",
    "ips": [
      "192.168.1.1"
    ],
    "mac": "ee:a0:b6:a4:07:94",
    "dns": {}
  }
]]
```

명령줄 인터페이스를 사용하여 OSD Pod가 올바른 네트워크를 사용하고 있는지 확인하려면 다음 명령을 실행합니다(jq 유틸리티 필요).

```
$ oc get -n openshift-storage $(oc get pods -n openshift-storage -o name -l app=rook-ceph-osd | grep 'osd-0') -o=jsonpath='{.metadata.annotations.k8s.v1.cni.cncf.io/network-status}' | jq -r '.[].name'
```

샘플 출력:

```
openshift-sdn
openshift-storage/ocs-cluster
openshift-storage/ocs-public
```

-

## 5장. 독립 실행형 MULTICLOUD OBJECT GATEWAY 배포

OpenShift Data Foundation을 사용하여 Multicloud Object Gateway 구성 요소만 배포하면 배포 유연성을 제공하고 리소스 소비를 줄이는 데 도움이 됩니다. 동적 스토리지 장치를 사용하거나 로컬 스토리지 장치를 사용하여 Multicloud Object Gateway 구성 요소를 배포할 수 있습니다.

### 5.1. 동적 스토리지 장치를 사용하여 독립 실행형 MULTICLOUD OBJECT GATEWAY 배포

다음 단계를 포함하는 독립 실행형 Multicloud Object Gateway 구성 요소만 배포하려면 이 섹션을 사용합니다.

- Red Hat OpenShift Data Foundation Operator 설치
- 독립 실행형 Multicloud Object Gateway 생성

#### 5.1.1. Red Hat OpenShift Data Foundation Operator 설치

Red Hat OpenShift Container Platform Operator Hub를 사용하여 Red Hat OpenShift Data Foundation Operator를 설치할 수 있습니다.

사전 요구 사항

- **cluster-admin** 및 Operator 설치 권한이 있는 계정을 사용하여 OpenShift Container Platform 클러스터에 액세스할 수 있습니다.
- Red Hat OpenShift Container Platform 클러스터에 3개 이상의 작업자 노드가 있어야 합니다.
- 추가 리소스 요구 사항은 [배포 계획 가이드](#)를 참조하십시오.

중요

- OpenShift Data Foundation의 클러스터 수준 기본 노드 선택기를 재정의해야 하는 경우 명령줄 인터페이스에서 다음 명령을 사용하여 **openshift-storage** 네임스페이스에 대한 빈 노드 선택기를 지정할 수 있습니다(이 경우 **openshift-storage** 네임스페이스 생성).

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

- 노드에 Red Hat OpenShift Data Foundation 리소스만 예약되도록 **infra** 테인트를 구성합니다. 이를 통해 서브스크립션 비용을 절감할 수 있습니다. 자세한 내용은 [스토리지 리소스 관리 및 할당 가이드](#)의 [Red Hat OpenShift Data Foundation 전용 작업자 노드 사용 방법](#)을 참조하십시오.

절차

1. OpenShift 웹 콘솔에 로그인합니다.
2. Operators → OperatorHub를 클릭합니다.
3. **OpenShift Data Foundation**을 키워드로 필터링 상자에 스크롤하여 OpenShift Data Foundation Operator를 찾습니다.
4. 설치를 클릭합니다.

5. Operator 설치 페이지에서 다음 옵션을 설정합니다.
  - a. Channel을 `stable-4.10`으로 업데이트합니다.
  - b. 설치 모드에서 클러스터의 특정 네임스페이스를 선택합니다.
  - c. 설치된 네임스페이스에서 Operator 권장 네임스페이스 `openshift-storage`를 선택합니다. 네임스페이스 `openshift-storage`가 없으면 Operator 설치 중에 생성됩니다.
  - d. 승인 전략을 자동 또는 수동으로 선택합니다.  
자동 업데이트를 선택하면 OLM(Operator Lifecycle Manager)은 개입 없이 Operator의 실행 중인 인스턴스를 자동으로 업그레이드합니다.  
  
수동 업데이트를 선택하면 OLM에서 업데이트 요청을 생성합니다. 클러스터 관리자는 Operator를 최신 버전으로 업데이트하기 위해 해당 업데이트 요청을 수동으로 승인해야 합니다.
  - e. Console 플러그인에 대해 Enable 옵션이 선택되어 있는지 확인합니다.
  - f. 설치를 클릭합니다.

#### 검증 단계

- OpenShift Data Foundation Operator에 설치 성공을 나타내는 녹색 눈금이 표시되는지 확인합니다.
- Operator가 성공적으로 설치되면 메시지가 포함된 팝업 **Web console update is available**이 사용자 인터페이스에 표시됩니다. 콘솔 변경 사항을 반영하려면 이 팝업 창에서 웹 콘솔 새로 고침을 클릭합니다.
  - 웹 콘솔에서 Operator로 이동하여 OpenShift Data Foundation 을 사용할 수 있는지 확인합니다.

### 5.1.2. 독립 실행형 Multicloud Object Gateway 생성

OpenShift Data Foundation을 배포하는 동안 독립 실행형 Multicloud Object Gateway 구성 요소만 생성할 수 있습니다.

#### 사전 요구 사항

- OpenShift Data Foundation Operator가 설치되어 있는지 확인합니다.

#### 절차

1. OpenShift 웹 콘솔에서 Operators → 설치된 Operator를 클릭하여 설치된 모든 Operator를 확인합니다.  
선택한 프로젝트가 `openshift-storage` 인지 확인합니다.
2. OpenShift Data Foundation Operator를 클릭한 다음 스토리지 시스템 생성을 클릭합니다.
3. 백업 스토리지 페이지에서 다음을 선택합니다.
  - a. 배포 유형 용으로 Multicloud Object Gateway를 선택합니다.
  - b. 기존 스토리지 클래스 사용 옵션을 선택합니다.

- c. 다음을 클릭합니다.
- 4. 선택 사항: 보안 페이지에서 외부 키 관리 서비스에 연결을 선택합니다.
  - a. 키 관리 서비스 공급자는 기본적으로 **Vault**로 설정됩니다.
  - b. **Vault** 서비스 이름, **Vault** 서버의 호스트 주소 ('https://<hostname or ip>'), 포트 번호, 토큰을 입력합니다.
  - c. 고급 설정을 확장하여 **Vault** 구성에 따라 추가 설정 및 인증서 세부 정보를 입력합니다.
    - i. **OpenShift Data Foundation** 전용 및 고유한 백엔드 경로에 키 값 시크릿 경로를 입력합니다.
    - ii. 선택 사항: TLS 서버 이름 및 **Vault Enterprise Namespace**를 입력합니다.
    - iii. 각 **PEM** 인코딩 인증서 파일을 업로드하여 **CA** 인증서, 클라이언트 인증서 및 클라이언트 개인 키를 제공합니다.
    - iv. 저장을 클릭합니다.
  - d. 다음을 클릭합니다.
- 5. 검토 및 생성 페이지에서 구성 세부 정보를 검토합니다.  
구성 설정을 수정하려면 뒤로를 클릭합니다.
- 6. 스토리지 시스템 생성을 클릭합니다.

## 검증 단계

### OpenShift Data Foundation 클러스터 상태 확인

1. **OpenShift** 웹 콘솔에서 스토리지 → 데이터 기반 을 클릭합니다.
2. 개요 탭의 상태 카드에서 스토리지 시스템을 클릭한 다음 해당 팝업에서 스토리지 시스템 링크를 클릭합니다.
  - a. **Object** 탭의 상태 카드에서 **Object Service** 및 **Data Resiliency** 모두 녹색 눈금이 있는지 확인합니다.
  - b. 세부 정보 카드에 **MCG** 정보가 표시되는지 확인합니다.

### Pod 상태 확인

1. **OpenShift** 웹 콘솔에서 워크로드 → Pod를 클릭합니다.
2. 프로젝트 드롭다운 목록에서 **openshift-storage**를 선택하고 다음 Pod가 **Running** 상태인지 확인합니다.



#### 참고

기본 프로젝트 표시 옵션이 비활성화된 경우 토글 버튼을 사용하여 모든 기본 프로젝트를 나열합니다.



구성 요소	해당 Pod
OpenShift Data Foundation Operator	<ul style="list-style-type: none"> <li>● <b>OCS-operator-*</b> (모든 작업자 노드에 1 Pod)</li> <li>● <b>OCS-metrics-exporter-*</b> (모든 작업자 노드에 1 Pod)</li> <li>● <b>ODF-operator-controller-manager-*</b> (모든 작업자 노드에 1 Pod)</li> <li>● <b>ODF-console-*</b> (모든 작업자 노드에 1 Pod)</li> <li>● <b>CSI-addons-controller-manager-*</b> (1개의 작업자 노드에서 Pod)</li> </ul>
Rook-ceph Operator	<b>rook-ceph-operator-*</b> (모든 작업자 노드에 1 Pod)
Multicloud Object Gateway	<ul style="list-style-type: none"> <li>● <b>noobaa-operator-*</b> (모든 작업자 노드에 1 Pod)</li> <li>● <b>noobaa-core-*</b> (모든 작업자 노드에 1 Pod)</li> <li>● <b>noobaa-db-pg-*</b> (모든 작업자 노드에서 1 pod)</li> <li>● <b>noobaa-endpoint-*</b> (모든 작업자 노드에서 1 Pod)</li> </ul>

## 5.2. 로컬 스토리지 장치를 사용하여 독립 실행형 MULTICLOUD OBJECT GATEWAY 배포

다음 단계를 포함하는 독립 실행형 Multicloud Object Gateway 구성 요소만 배포하려면 이 섹션을 사용합니다.

- Local Storage Operator 설치
- Red Hat OpenShift Data Foundation Operator 설치
- 독립 실행형 Multicloud Object Gateway 생성

### 5.2.1. 로컬 스토리지 Operator 설치

로컬 스토리지 장치에서 Red Hat OpenShift Data Foundation 클러스터를 생성하기 전에 Operator Hub에서 Local Storage Operator를 설치합니다.

절차

1. OpenShift 웹 콘솔에 로그인합니다.
2. Operators → OperatorHub를 클릭합니다.
3. 키워드로 필터링에 로컬 스토리지를 입력합니다. Operator 목록에서 Local Storage Operator를 찾아서 클릭합니다.

4. Operator 설치 페이지에서 다음 옵션을 설정합니다.
  - a. 채널을 **4.10** 또는 **stable**로 업데이트합니다.
  - b. 설치 모드에서 클러스터의 특정 네임스페이스를 선택합니다.
  - c. 설치된 네임스페이스를 Operator 권장 네임스페이스 **openshift-local-storage**를 선택합니다.
  - d. 승인을 자동으로 업데이트합니다.
5. 설치를 클릭합니다.

#### 검증 단계

- Local Storage Operator에 성공적인 설치를 나타내는 녹색 눈금이 표시되는지 확인합니다.

### 5.2.2. Red Hat OpenShift Data Foundation Operator 설치

Red Hat OpenShift Container Platform Operator Hub를 사용하여 Red Hat OpenShift Data Foundation Operator를 설치할 수 있습니다.

#### 사전 요구 사항

- **cluster-admin** 및 Operator 설치 권한이 있는 계정을 사용하여 OpenShift Container Platform 클러스터에 액세스할 수 있습니다.
- Red Hat OpenShift Container Platform 클러스터에 3개 이상의 작업자 노드가 있어야 합니다.
- 추가 리소스 요구 사항은 [배포 계획 가이드](#)를 참조하십시오.



#### 중요

- OpenShift Data Foundation의 클러스터 수준 기본 노드 선택기를 재정의해야 하는 경우 명령줄 인터페이스에서 다음 명령을 사용하여 **openshift-storage** 네임스페이스에 대한 빈 노드 선택기를 지정할 수 있습니다(이 경우 **openshift-storage** 네임스페이스 생성).

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

- 노드에 Red Hat OpenShift Data Foundation 리소스만 예약되도록 **infra** 테인트를 구성합니다. 이를 통해 서브스크립션 비용을 절감할 수 있습니다. 자세한 내용은 [스토리지 리소스 관리 및 할당 가이드](#)의 [Red Hat OpenShift Data Foundation 전용 작업자 노드 사용 방법](#)을 참조하십시오.

#### 절차

1. OpenShift 웹 콘솔에 로그인합니다.
2. Operators → OperatorHub를 클릭합니다.
3. **OpenShift Data Foundation**을 키워드로 필터링 상자에 스크롤하여 OpenShift Data Foundation Operator를 찾습니다.
4. 설치를 클릭합니다.

5. Operator 설치 페이지에서 다음 옵션을 설정합니다.
  - a. Channel을 stable-4.10으로 업데이트합니다.
  - b. 설치 모드에서 클러스터의 특정 네임스페이스를 선택합니다.
  - c. 설치된 네임스페이스에서 Operator 권장 네임스페이스 openshift-storage를 선택합니다. 네임스페이스 openshift-storage가 없으면 Operator 설치 중에 생성됩니다.
  - d. 승인 전략을 자동 또는 수동으로 선택합니다.  
자동 업데이트를 선택하면 OLM(Operator Lifecycle Manager)은 개입 없이 Operator의 실행 중인 인스턴스를 자동으로 업그레이드합니다.  
  
수동 업데이트를 선택하면 OLM에서 업데이트 요청을 생성합니다. 클러스터 관리자는 Operator를 최신 버전으로 업데이트하기 위해 해당 업데이트 요청을 수동으로 승인해야 합니다.
  - e. Console 플러그인에 대해 Enable 옵션이 선택되어 있는지 확인합니다.
  - f. 설치를 클릭합니다.

#### 검증 단계

- OpenShift Data Foundation Operator에 설치 성공을 나타내는 녹색 눈금이 표시되는지 확인합니다.
- Operator가 성공적으로 설치되면 메시지가 포함된 팝업 **Web console update is available**이 사용자 인터페이스에 표시됩니다. 콘솔 변경 사항을 반영하려면 이 팝업 창에서 웹 콘솔 새로 고침을 클릭합니다.
  - 웹 콘솔에서 Operator로 이동하여 OpenShift Data Foundation 을 사용할 수 있는지 확인합니다.

### 5.2.3. 독립 실행형 Multicloud Object Gateway 생성

OpenShift Data Foundation을 배포하는 동안 독립 실행형 Multicloud Object Gateway 구성 요소만 생성할 수 있습니다.

#### 사전 요구 사항

- OpenShift Data Foundation Operator가 설치되어 있는지 확인합니다.

#### 절차

1. OpenShift 웹 콘솔에서 Operators → 설치된 Operator를 클릭하여 설치된 모든 Operator를 확인합니다.  
선택한 프로젝트가 openshift-storage 인지 확인합니다.
2. OpenShift Data Foundation Operator를 클릭한 다음 스토리지 시스템 생성을 클릭합니다.
3. 백업 스토리지 페이지에서 다음을 선택합니다.
  - a. 배포 유형 용으로 Multicloud Object Gateway를 선택합니다.
  - b. 로컬 스토리지 장치 옵션을 사용하여 새 StorageClass 만들기 옵션을 선택합니다.

c. 다음을 클릭합니다.



**참고**

아직 설치되지 않은 경우 **Local Storage Operator**를 설치하라는 메시지가 표시됩니다. 설치를 클릭하고 **Local Storage Operator 설치**에 설명된 절차를 따릅니다.

4. 로컬 볼륨 세트 생성 페이지에서 다음 정보를 제공합니다.

- a. 로컬 볼륨 세트의 이름과 스토리지 클래스를 입력합니다.  
기본적으로 스토리지 클래스 이름에 로컬 볼륨 세트 이름이 표시됩니다. 이름을 변경할 수 있습니다.
- b. 다음 중 하나를 선택합니다.
  - 모든 노드의 디스크  
모든 노드에서 선택한 필터와 일치하는 사용 가능한 디스크를 사용합니다.
  - 선택한 노드의 디스크  
선택한 노드에서만 선택한 필터와 일치하는 사용 가능한 디스크를 사용합니다.
- c. 사용 가능한 디스크 유형 목록에서 **SSD/NVMe**을 선택합니다.
- d. 고급 섹션을 확장하고 다음 옵션을 설정합니다.

볼륨 모드	파일 시스템은 기본적으로 선택됩니다. 항상 볼륨 모드의 파일 시스템이 선택되어 있는지 확인하십시오.
장치 유형	드롭다운 목록에서 하나 이상의 장치 유형을 선택합니다.
디스크 크기	장치에 대해 최소 크기 100GB와 포함되어야 하는 장치의 사용 가능한 최대 크기를 설정합니다.
최대 디스크 제한	이는 노드에서 생성할 수 있는 최대 PV 수를 나타냅니다. 이 필드가 비어 있으면 일치하는 노드에서 사용 가능한 모든 디스크에 PV가 생성됩니다.

- e. 다음을 클릭합니다.  
로컬 볼륨 세트 생성이 표시되는지 확인하는 팝업이 표시됩니다.
- f. 계속하려면 예를 클릭합니다.

5. 용량 및 노드 페이지에서 다음을 구성합니다.

- a. 사용 가능한 원시 용량은 스토리지 클래스와 연결된 모든 디스크에 따라 용량 값으로 채워집니다. 이 작업을 수행하는 데 시간이 다소 걸립니다. 선택한 노드 목록에는 스토리지 클래스를 기반으로 하는 노드가 표시됩니다.
- b. 다음을 클릭합니다.

6. 선택 사항: 보안 페이지에서 외부 키 관리 서비스에 연결을 선택합니다.

- a. 키 관리 서비스 공급자는 기본적으로 **Vault**로 설정됩니다.

- b. Vault 서비스 이름, Vault 서버의 호스트 주소 ('https://<hostname or ip>'), 포트 번호, 토큰을 입력합니다.
  - c. 고급 설정을 확장하여 Vault 구성에 따라 추가 설정 및 인증서 세부 정보를 입력합니다.
    - i. OpenShift Data Foundation 전용 및 고유한 백엔드 경로에 키 값 시크릿 경로를 입력합니다.
    - ii. 선택 사항: TLS 서버 이름 및 Vault Enterprise Namespace를 입력합니다.
    - iii. 각 PEM 인코딩 인증서 파일을 업로드하여 CA 인증서, 클라이언트 인증서 및 클라이언트 개인 키를 제공합니다.
    - iv. 저장을 클릭합니다.
  - d. 다음을 클릭합니다.
7. 검토 및 생성 페이지에서 구성 세부 정보를 검토합니다.  
구성 설정을 수정하려면 뒤로를 클릭합니다.
  8. 스토리지 시스템 생성을 클릭합니다.

## 검증 단계

### OpenShift Data Foundation 클러스터 상태 확인

1. OpenShift 웹 콘솔에서 스토리지 → 데이터 기반 을 클릭합니다.
2. 개요 탭의 상태 카드에서 스토리지 시스템을 클릭한 다음 해당 팝업에서 스토리지 시스템 링크를 클릭합니다.
  - a. Object 탭의 상태 카드에서 Object Service 및 Data Resiliency 모두 녹색 눈금이 있는지 확인합니다.
  - b. 세부 정보 카드에 MCG 정보가 표시되는지 확인합니다.

### Pod 상태 확인

1. OpenShift 웹 콘솔에서 워크로드 → Pod를 클릭합니다.
2. 프로젝트 드롭다운 목록에서 openshift-storage를 선택하고 다음 Pod가 Running 상태인지 확인합니다.



#### 참고

기본 프로젝트 표시 옵션이 비활성화된 경우 토글 버튼을 사용하여 모든 기본 프로젝트를 나열합니다.

구성 요소	해당 Pod
-------	--------

구성 요소	해당 Pod
OpenShift Data Foundation Operator	<ul style="list-style-type: none"> <li>● <b>OCS-operator-*</b> (모든 작업자 노드에 1 Pod)</li> <li>● <b>OCS-metrics-exporter-*</b> (모든 작업자 노드에 1 Pod)</li> <li>● <b>ODF-operator-controller-manager-*</b> (모든 작업자 노드에 1 Pod)</li> <li>● <b>ODF-console-*</b> (모든 작업자 노드에 1 Pod)</li> <li>● <b>CSI-addons-controller-manager-*</b> (1개의 작업자 노드에서 Pod)</li> </ul>
Rook-ceph Operator	<p><b>rook-ceph-operator-*</b></p> <p>(모든 작업자 노드에 1 Pod)</p>
Multicloud Object Gateway	<ul style="list-style-type: none"> <li>● <b>noobaa-operator-*</b> (모든 작업자 노드에 1 Pod)</li> <li>● <b>noobaa-core-*</b> (모든 작업자 노드에 1 Pod)</li> <li>● <b>noobaa-db-pg-*</b> (모든 작업자 노드에서 1 pod)</li> <li>● <b>noobaa-endpoint-*</b> (모든 작업자 노드에서 1 Pod)</li> </ul>

## 6장. OPENSIFT DATA FOUNDATION 설치 제거

### 6.1. 내부 모드에서 OPENSIFT DATA FOUNDATION 설치 제거

내부 모드에서 OpenShift Data Foundation을 설치 제거하려면 [OpenShift Data Foundation 설치 제거에 대한 지식 기반 문서](#)를 참조하십시오.