



설치 및 구성 FlexPod

NetApp
October 30, 2025

목차

설치 및 구성	1
OpenShift Container Platform 4용 FlexPod 베어 메탈 설치	1
VMware 설치 기반 OpenShift Container Platform 4용 FlexPod	1
AWS 기반 Red Hat OpenShift	2
NetApp Cloud Volumes ONTAP를 참조하십시오	3
OpenShift Container Platform에 Astra Control Center 설치	3
확인하십시오	4
OpenShift OperatorHub를 사용하여 Astra Control Center를 설치합니다	5
Astra Control Center를 설정합니다	18

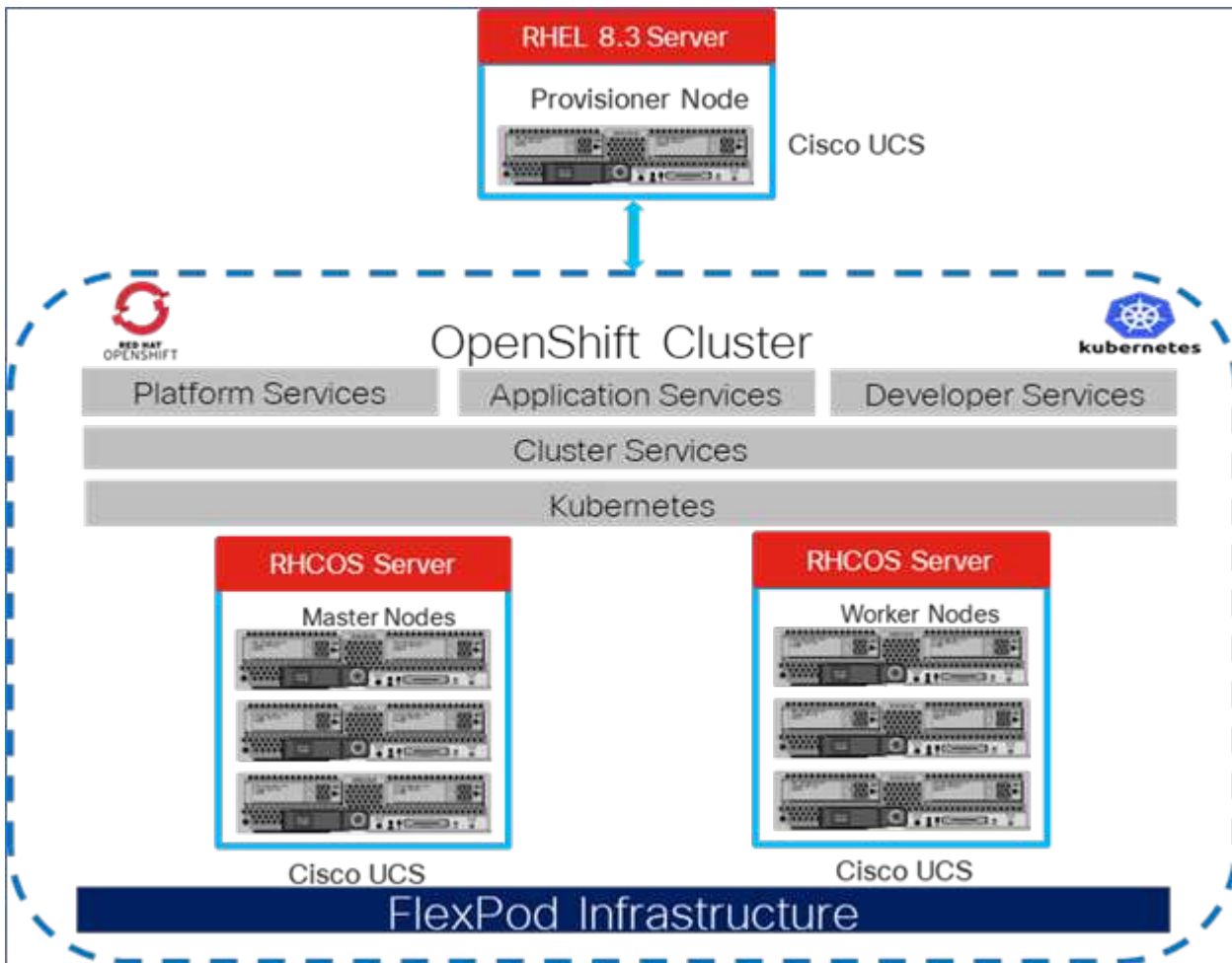
설치 및 구성

OpenShift Container Platform 4용 FlexPod 베어 메탈 설치

"이전: 솔루션 구성 요소."

OpenShift Container Platform 4 베어메탈 설계, 구축 세부 정보, NetApp Astra Trident 설치 및 구성을 위한 FlexPod를 알아보려면 를 참조하십시오 ["FlexPod 및 OpenShift CVD\(Cisco Validated Design and Deployment Guide\)"](#). 이 CVD에서는 Ansible을 사용한 FlexPod 및 OpenShift 컨테이너 플랫폼 구축에 대해 설명합니다. 또한 CVD는 Astra Control Center를 구축 및 구성하기 위한 몇 가지 사전 요구 사항에 따라 작업자 노드, Astra Trident 설치, 스토리지 백엔드 및 스토리지 클래스 구성을 준비하는 방법에 대한 자세한 정보를 제공합니다.

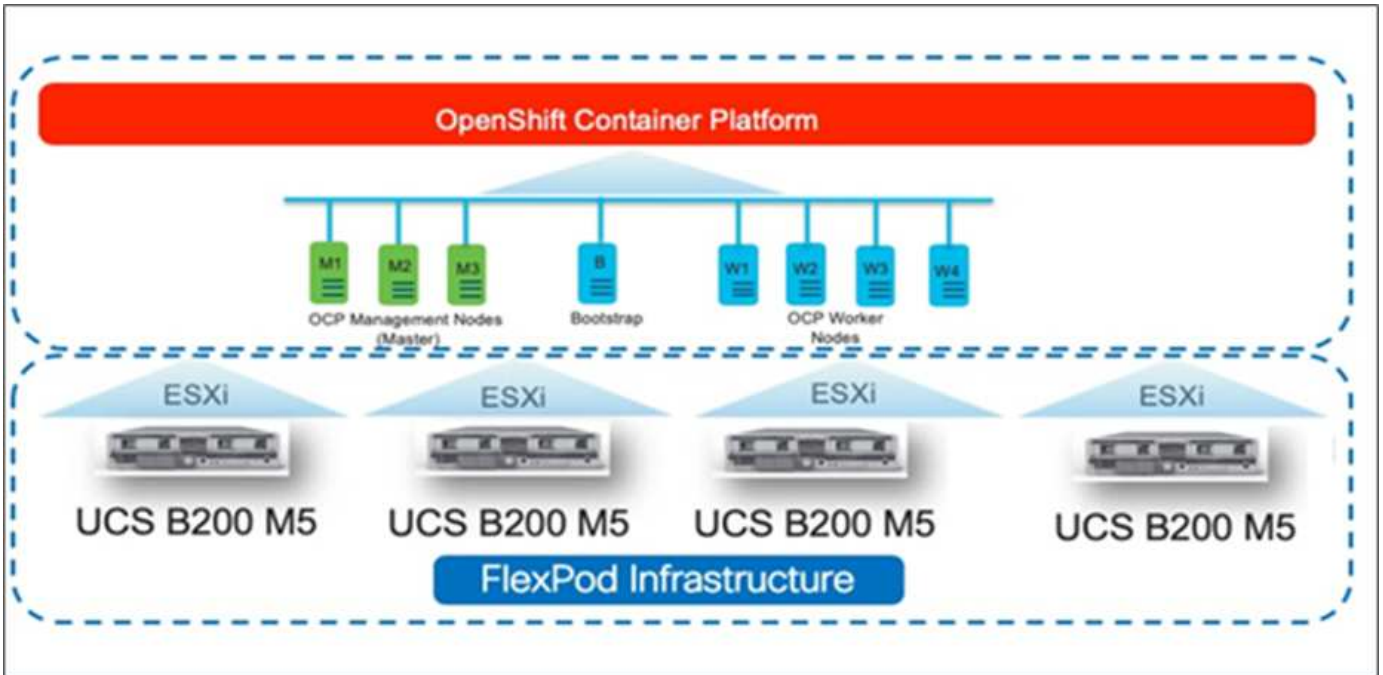
다음 그림은 FlexPod의 OpenShift 컨테이너 플랫폼 4 베어 메탈 을 보여 줍니다.



VMware 설치 기반 OpenShift Container Platform 4용 FlexPod

VMware vSphere를 실행하는 FlexPod에서 Red Hat OpenShift Container Platform 4 배포에 대한 자세한 내용은 을 참조하십시오 ["OpenShift 컨테이너 플랫폼용 FlexPod 데이터 센터 4"](#).

다음 그림에서는 vSphere 기반의 OpenShift Container Platform 4용 FlexPod를 보여 줍니다.



"다음으로, AWS의 Red Hat OpenShift입니다."

AWS 기반 Red Hat OpenShift

"이전: OpenShift Container Platform 4 베어메탈 설치용 FlexPod"

DR 사이트로 AWS에 별도의 자가 관리 OpenShift Container Platform 4 클러스터가 구축됩니다. 마스터 노드와 작업자 노드는고가용성을 위해 세 개의 가용성 영역에 걸쳐 있습니다.

Instances (6) Info								
<input type="text" value="Search"/> <input type="button" value="Clear filters"/>								
<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Availability Zone	Private IP a...	Key name	
<input type="checkbox"/>	ocpaws-v58kn-master-0	i-0d2d81ca91a54276d	Running	m5.xlarge	us-east-1b	172.30.165.160	-	
<input type="checkbox"/>	ocpaws-v58kn-master-1	i-0b161945421d2a23c	Running	m5.xlarge	us-east-1c	172.30.166.162	-	
<input type="checkbox"/>	ocpaws-v58kn-master-2	i-0146a665e1060ea59	Running	m5.xlarge	us-east-1a	172.30.164.209	-	
<input type="checkbox"/>	ocpaws-v58kn-worker-us-east-1a-zj8dj	i-05e6efa18d136c842	Running	m5.large	us-east-1a	172.30.164.128	-	
<input type="checkbox"/>	ocpaws-v58kn-worker-us-east-1b-7nmbc	i-0879a088b50d2d966	Running	m5.large	us-east-1b	172.30.165.93	-	
<input type="checkbox"/>	ocpaws-v58kn-worker-us-east-1c-96j6n	i-0c24ff3c2d701f82c	Running	m5.large	us-east-1c	172.30.166.51	-	

```
[ec2-user@ip-172-30-164-92 ~]$ oc get nodes
```

NAME	STATUS	ROLES	AGE	VERSION
ip-172-30-164-128.ec2.internal	Ready	worker	29m	v1.22.8+f34b40c
ip-172-30-164-209.ec2.internal	Ready	master	36m	v1.22.8+f34b40c
ip-172-30-165-160.ec2.internal	Ready	master	33m	v1.22.8+f34b40c
ip-172-30-165-93.ec2.internal	Ready	worker	30m	v1.22.8+f34b40c
ip-172-30-166-162.ec2.internal	Ready	master	36m	v1.22.8+f34b40c
ip-172-30-166-51.ec2.internal	Ready	worker	28m	v1.22.8+f34b40c

OpenShift는 로 배포됩니다 **"프라이빗 클러스터"** AWS 기반 기존 VPC에. 사설 OpenShift Container Platform 클러스터는 외부 엔드포인트를 노출하지 않으며 내부 네트워크에서만 액세스할 수 있고 인터넷에서는 볼 수 없습니다. NetApp Cloud Manager를 사용하여 단일 노드 NetApp Cloud Volumes ONTAP를 구축하고 Astra Trident에 스토리지 백엔드를 제공합니다.

AWS에 OpenShift를 설치하는 방법에 대한 자세한 내용은 를 참조하십시오 **"OpenShift 문서"**.

"다음은 NetApp Cloud Volumes ONTAP의 차례입니다."

NetApp Cloud Volumes ONTAP를 참조하십시오

"이전: AWS의 Red Hat OpenShift"

NetApp Cloud Volumes ONTAP 인스턴스는 AWS에 구축되며 Astra Trident에 백엔드 스토리지 역할을 합니다. Cloud Volumes ONTAP 작업 환경을 추가하기 전에 커넥터를 구축해야 합니다. 커넥터가 없는 상태에서 첫 번째 Cloud Volumes ONTAP 작업 환경을 생성하려고 하면 Cloud Manager에서 메시지가 표시됩니다. AWS에 Connector를 구축하려면 를 참조하십시오 **"커넥터를 작성합니다"**.

AWS에 Cloud Volumes ONTAP를 구축하려면 를 참조하십시오 **"AWS 환경을 위한 빠른 시작"**.

Cloud Volumes ONTAP를 구축한 후 Astra Trident를 설치하고 OpenShift Container Platform 클러스터에서 스토리지 백엔드 및 스냅샷 클래스를 구성할 수 있습니다.

"다음: OpenShift Container Platform에 Astra Control Center를 설치합니다."

OpenShift Container Platform에 Astra Control Center 설치

"이전: NetApp Cloud Volumes ONTAP."

FlexPod에서 실행되는 OpenShift 클러스터나 Cloud Volumes ONTAP 스토리지 백엔드가 있는 AWS에 Astra Control Center를 설치할 수 있습니다. 이 솔루션에서 Astra Control Center는 OpenShift 베어 메탈 클러스터에 구축됩니다.

Astra Control Center는 설명된 표준 프로세스를 사용하여 설치할 수 있습니다 **"여기"** 또는 Red Hat OpenShift OperatorHub를 참조하십시오. Astra Control Operator는 Red Hat 공인 운영자입니다. 이 솔루션에서는 Red Hat OperatorHub를 사용하여 Astra Control Center를 설치합니다.

확인하십시오

- Astra Control Center는 여러 Kubernetes 배포를 지원합니다. Red Hat OpenShift의 경우 지원되는 버전에 Red Hat OpenShift Container Platform 4.8 또는 4.9가 포함됩니다.
- Astra Control Center에는 환경 및 최종 사용자의 애플리케이션 리소스 요구 사항 외에 다음과 같은 리소스가 필요합니다.

구성 요소	요구 사항
스토리지 백엔드 용량입니다	최소 500GB가 제공됩니다
작업자 노드	최소 3개의 작업자 노드, 각각 4개의 CPU 코어, 12GB RAM
FQDN(정규화된 도메인 이름) 주소입니다	Astra Control Center의 FQDN 주소입니다
아스트라 트리덴트	Astra Trident 21.04 이상 설치 및 구성
수신 컨트롤러 또는 로드 밸런서	FQDN으로 확인할 IP 주소를 제공하도록 수신 컨트롤러를 URL 또는 로드 밸런서와 함께 Astra Control Center를 노출하도록 구성합니다

- Astra Control Center 빌드 이미지를 푸시할 수 있는 기존 개인 이미지 레지스트리가 있어야 합니다. 이미지를 업로드하는 이미지 레지스트리의 URL을 제공해야 합니다.



일부 이미지는 특정 워크플로우를 실행하는 동안 가져오며, 필요에 따라 컨테이너가 생성되고 제거됩니다.

- Astra Control Center에서는 스토리지 클래스를 생성하고 기본 스토리지 클래스로 설정해야 합니다. Astra Control Center는 Astra Trident에서 제공하는 다음과 같은 ONTAP 드라이버를 지원합니다.
 - ONTAP - NAS
 - ONTAP-NAS-Flexgroup입니다
 - ONTAP-SAN
 - ONTAP-SAN - 경제성



배포된 OpenShift 클러스터에 ONTAP 백엔드로 설치 및 구성된 Astra Trident가 있으며 기본 스토리지 클래스도 정의되어 있다고 가정합니다.

- OpenShift 환경에서 애플리케이션을 복제하려면 OpenShift가 볼륨을 마운트하고 파일 소유권을 변경할 수 있도록 Astra Control Center를 허용해야 합니다. 이러한 작업을 허용하도록 ONTAP 익스포트 정책을 수정하려면 다음 명령을 실행합니다.

```
export-policy rule modify -vserver <storage virtual machine name>
-policyname <policy name> -ruleindex 1 -superuser sys
export-policy rule modify -vserver <storage virtual machine name>
-policyname <policy name> -ruleindex 1 -anon 65534
```



두 번째 OpenShift 운영 환경을 관리되는 컴퓨팅 리소스로 추가하려면 Astra Trident 볼륨 스냅샷 기능이 활성화되어 있는지 확인합니다. Astra Trident를 사용하여 볼륨 스냅샷을 활성화하고 테스트하려면 공식 을 참조하십시오 "[Astra Trident 지침](#)".

- A "[VolumeSnapClass](#)" 애플리케이션이 관리되는 모든 Kubernetes 클러스터에서 구성되어야 합니다. 여기에는 Astra Control Center가 설치된 K8s 클러스터도 포함될 수 있습니다. Astra Control Center는 실행 중인 K8s 클러스터에서 애플리케이션을 관리할 수 있습니다.

설명합니다

- * 라이선스. * Astra Control Center를 사용하여 애플리케이션을 관리하려면 Astra Control Center 라이선스가 필요합니다.
- * Namespaces. * 네임스페이스는 Astra Control Center에서 응용 프로그램으로 관리할 수 있는 가장 큰 엔터티입니다. 기존 네임스페이스에서 응용 프로그램 레이블 및 사용자 지정 레이블을 기준으로 구성 요소를 필터링하고 리소스의 하위 집합을 응용 프로그램으로 관리하도록 선택할 수 있습니다.
- * StorageClass. * StorageClass가 명시적으로 설정된 응용 프로그램을 설치하고 응용 프로그램을 복제해야 하는 경우 클론 작업의 타겟 클러스터에 원래 지정된 StorageClass가 있어야 합니다. 명시적으로 StorageClass를 동일한 StorageClass가 없는 클러스터로 설정한 응용 프로그램을 클론 복제하지 못합니다.
- Kubernetes 리소스 * Astra Control에서 캡처하지 않은 Kubernetes 리소스를 사용하는 애플리케이션에는 전체 애플리케이션 데이터 관리 기능이 없을 수 있습니다. Astra Control은 다음과 같은 Kubernetes 리소스를 캡처할 수 있습니다.

Kubernetes 리소스		
클러스터 역할	ClusterRoleBinding 을 참조하십시오	ConfigMap을 클릭합니다
사용자 지정 리소스 정의	CustomResource 를 선택합니다	경작업
DemonSet	HorizontalPodAutoscaler	침투
DeploymentConfig(배포 구성	mutatingWebhook	PersistentVolumeClaim
포드	팟캐스트 예산	팟캐스트 템플릿
네트워크 정책	ReplicaSet입니다	역할
RoleBinding 을 클릭합니다	루트	비밀
Webhook을 확인합니다		

OpenShift OperatorHub를 사용하여 Astra Control Center를 설치합니다

다음 절차에서는 Red Hat OperatorHub를 사용하여 Astra Control Center를 설치합니다. 이 솔루션에서 Astra Control Center는 FlexPod에서 실행되는 베어 메탈 OpenShift 클러스터에 설치됩니다.

1. 에서 Astra Control Center 번들('Astra-control-center-[version].tar.gz')을 다운로드합니다 "[NetApp Support 사이트](#)".
2. Astra Control Center 인증서 및 키용 .zip 파일을 에서 다운로드합니다 "[NetApp Support 사이트](#)".
3. 번들의 서명을 확인합니다.

```
openssl dgst -sha256 -verify astra-control-center[version].pub  
-signature <astra-control-center[version].sig astra-control-  
center[version].tar.gz
```

4. Astra 이미지를 추출합니다.

```
tar -vxzf astra-control-center-[version].tar.gz
```

5. Astra 디렉토리로 변경합니다.

```
cd astra-control-center-[version]
```

6. 이미지를 로컬 레지스트리에 추가합니다.

```
For Docker:  
docker login [your_registry_path]OR  
For Podman:  
podman login [your_registry_path]
```

7. 적절한 스크립트를 사용하여 이미지를 로드하고, 이미지에 태그를 지정한 다음 로컬 레지스트리에 해당 이미지를 푸시합니다.

Docker의 경우:

```
export REGISTRY=[Docker_registry_path]  
for astraImageFile in $(ls images/*.tar) ; do  
    # Load to local cache. And store the name of the loaded image trimming  
    the 'Loaded images: '  
    astraImage=$(docker load --input ${astraImageFile} | sed 's/Loaded  
image: //' )  
    astraImage=$(echo ${astraImage} | sed 's!localhost/!!')  
    # Tag with local image repo.  
    docker tag ${astraImage} ${REGISTRY}/${astraImage}  
    # Push to the local repo.  
    docker push ${REGISTRY}/${astraImage}  
done
```

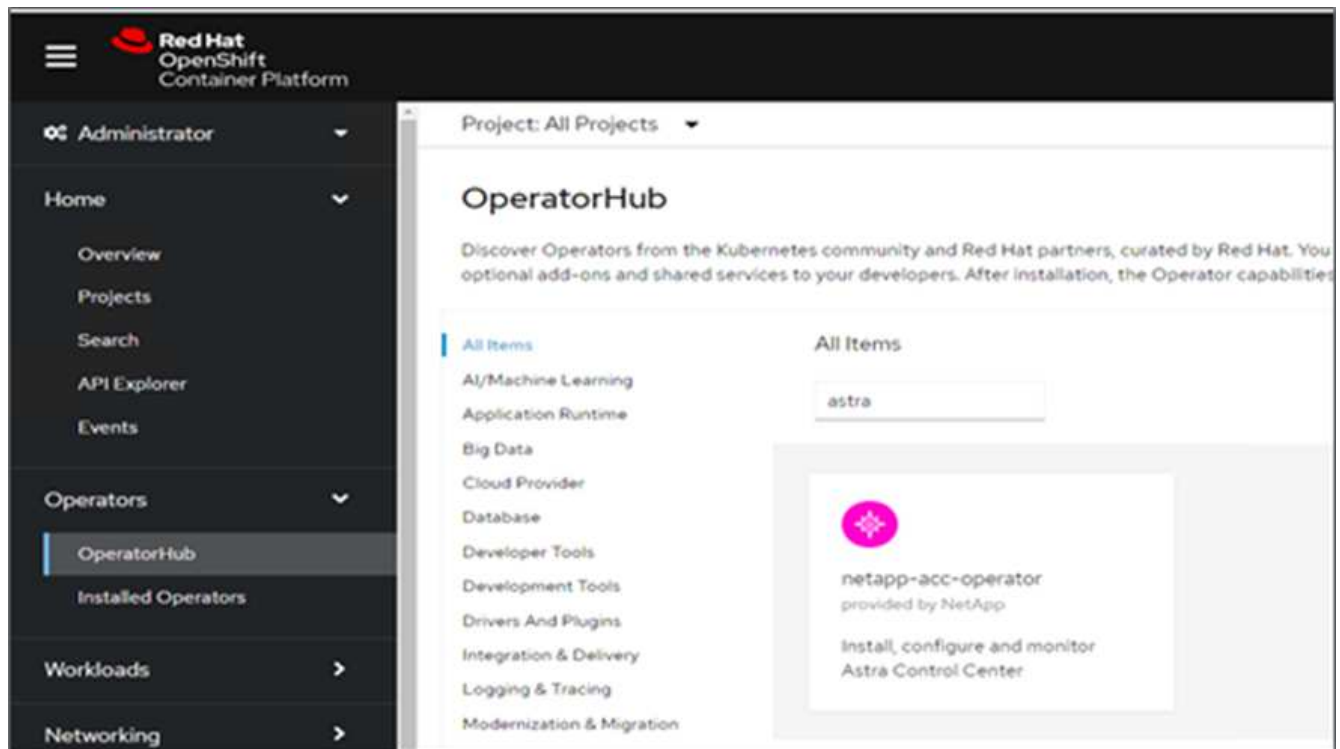
Podman의 경우:


```

export REGISTRY=[Registry_path]
for astraImageFile in $(ls images/*.tar) ; do
    # Load to local cache. And store the name of the loaded image trimming
    the 'Loaded images: '
    astraImage=$(podman load --input ${astraImageFile} | sed 's/Loaded
image(s): //'')
    astraImage=$(echo ${astraImage} | sed 's!localhost/!!')
    # Tag with local image repo.
    podman tag ${astraImage} ${REGISTRY}/${astraImage}
    # Push to the local repo.
    podman push ${REGISTRY}/${astraImage}
done


```

8. 베어 메탈 OpenShift 클러스터 웹 콘솔에 로그인합니다. 측면 메뉴에서 Operators > OperatorHub 를 선택합니다. NetApp-acc-operator를 나열하려면 "Astra"를 입력합니다.



'NetApp-acc-operator'는 인증된 Red Hat OpenShift Operator로, OperatorHub 카탈로그 아래에 나열되어 있습니다.

9. 'NetApp-acc-operator'를 선택하고 '설치'를 클릭합니다.



netapp-acc-operator
 22.4.3 provided by NetApp

Install

Latest version
 22.4.3

Capability level
☒ Basic Install
☐ Seamless Upgrades
☐ Full Lifecycle
☐ Deep Insights
☐ Auto Pilot

Source
 Certified

Provider
 NetApp

Astra Control is an application-aware data management solution that manages, protects and moves data-rich Kubernetes workloads in both public clouds and on-premises.

Astra Control enables data protection, disaster recovery, and migration for your Kubernetes workloads, leveraging NetApp's industry-leading data management technology for snapshots, backups, replication and cloning.

How to deploy Astra Control

Refer to [Installation Procedure](#) to deploy Astra Control Center using the Operator.

Documentation

Refer to [Astra Control Center Documentation](#) to complete the setup and start managing applications.

NOTE: The version listed under *Latest version* on this page might not reflect the actual version of NetApp Astra Control Center you are installing. The version in the file name of the Astra Control Center bundle that you download from the NetApp Support Site is the version of Astra Control Center that will be installed.

10. 적절한 옵션을 선택하고 설치 를 클릭합니다.

OperatorHub > Operator Installation

Install Operator

Install your Operator by subscribing to one of the update channels to keep the Operator up to date. The strategy determines either manual or automatic updates.


Update channel * ⓘ

☐ alpha
 ☒ stable

Installation mode *


☒ All namespaces on the cluster (default)
 Operator will be available in all Namespaces.
 ☐ A specific namespace on the cluster
 This mode is not supported by this Operator

Installed Namespace *


 netapp-acc-operator (Operator recommended)


Update approval * ⓘ


☐ Automatic
 ☒ Manual


netapp-acc-operator
 provided by NetApp

Provided APIs

 **Astra Control Center**
 AstraControlCenter is the Schema for the astracontrolcenters API.

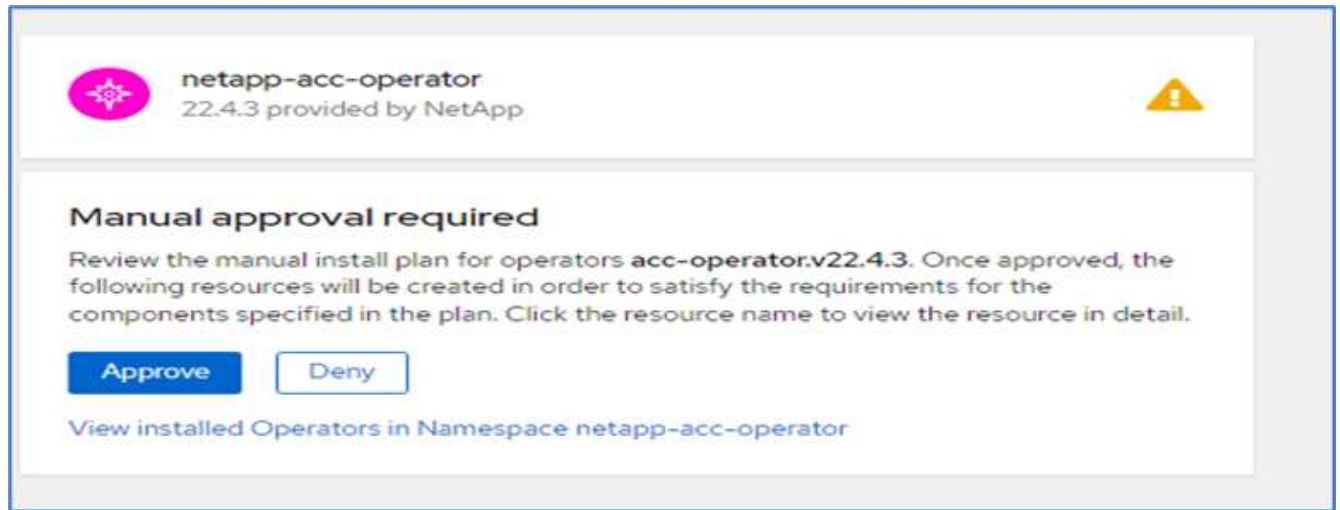
 **Namespace creation**
 Namespace **netapp-acc-operator** does not exist and will be created.

 **Manual approval applies to all operators in a namespace**
 Installing an operator with manual approval causes all operators installed in namespace **netapp-acc-operator** to function as manual approval strategy. To allow automatic approval, all operators installed in the namespace must use automatic approval strategy.

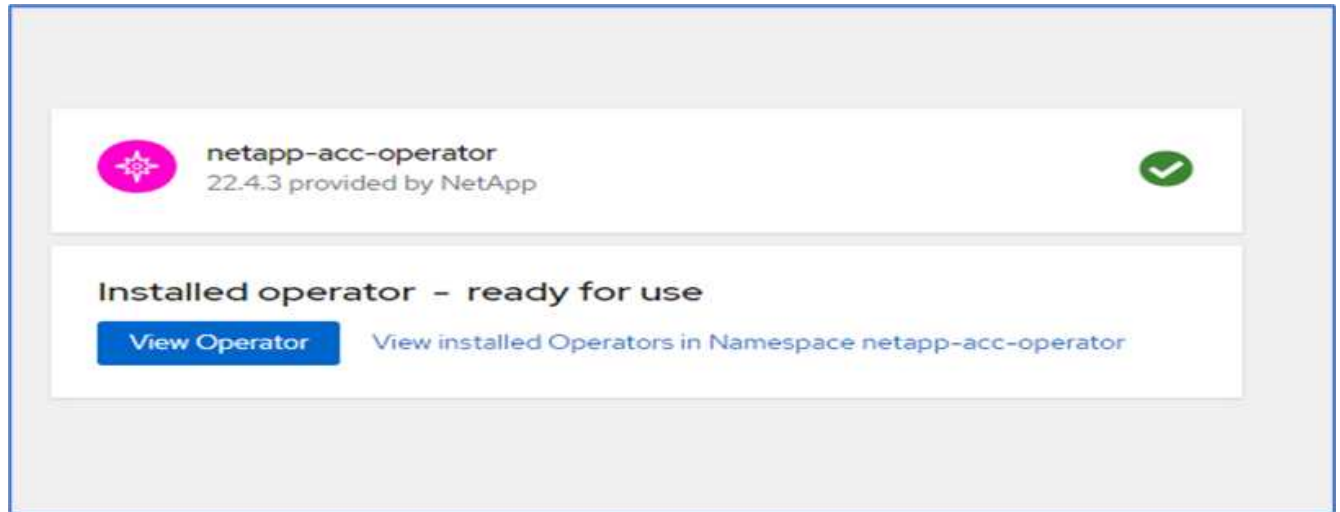
Install

Cancel

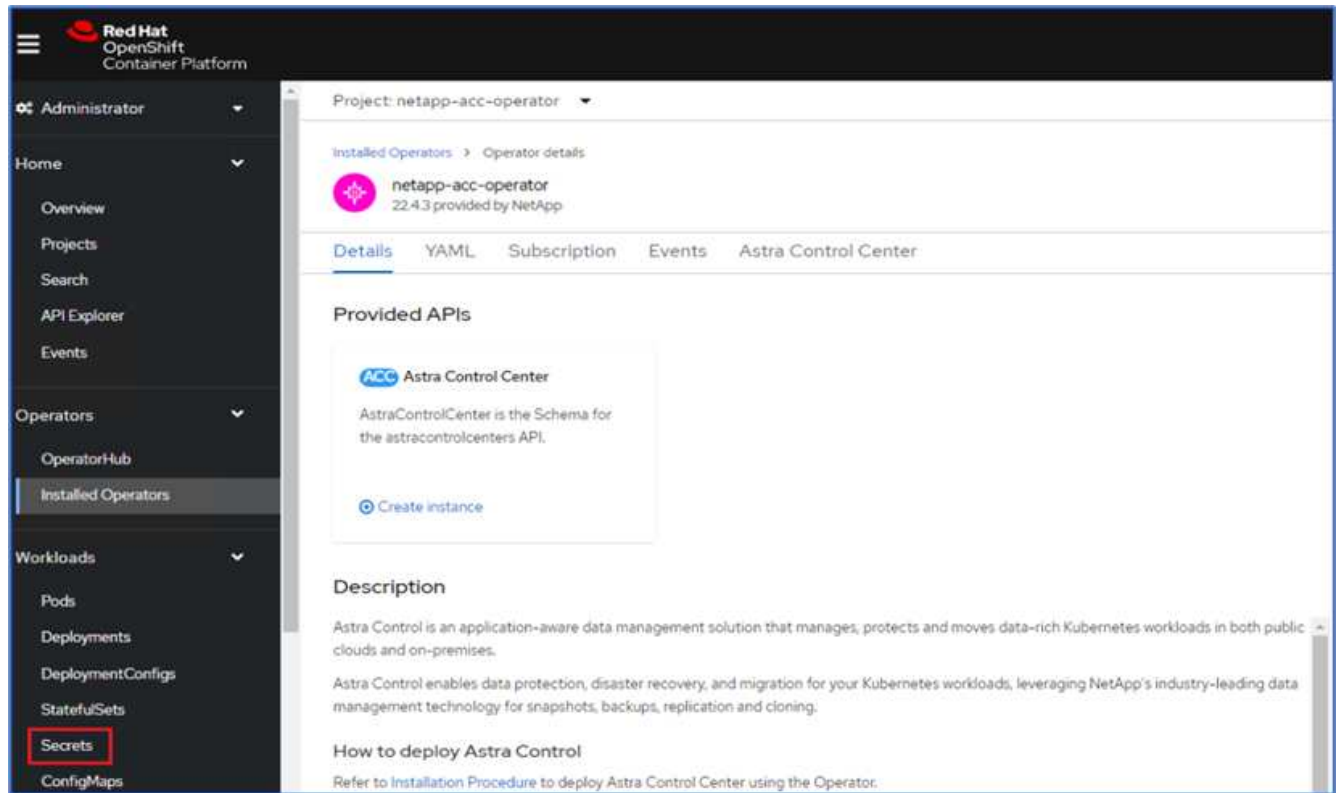
11. 설치를 승인하고 운영자가 설치될 때까지 기다립니다.



- 이 단계에서 오퍼레이터는 성공적으로 설치되고 사용할 준비가 완료됩니다. View Operator(작업자 보기) 를 클릭하여 Astra Control Center 설치를 시작합니다.



- Astra Control Center를 설치하기 전에 앞서 푸시한 Docker 레지스트리에서 Astra 이미지를 다운로드하기 위한 비밀번호를 생성합니다.



14. Docker 프라이빗 레포에서 Astra Control Center 이미지를 풀려면 'NetApp-acc-operator' 네임스페이스에서 암호를 생성하십시오. 이 비밀 이름은 나중에 Astra Control Center YAML manifest에서 제공합니다.

Project: netapp-acc-operator ▼

Create image pull secret

Image pull secrets let you authenticate against a private image registry.

Secret name *

Unique name of the new secret.

Authentication type

Registry server address *

For example quay.io or docker.io

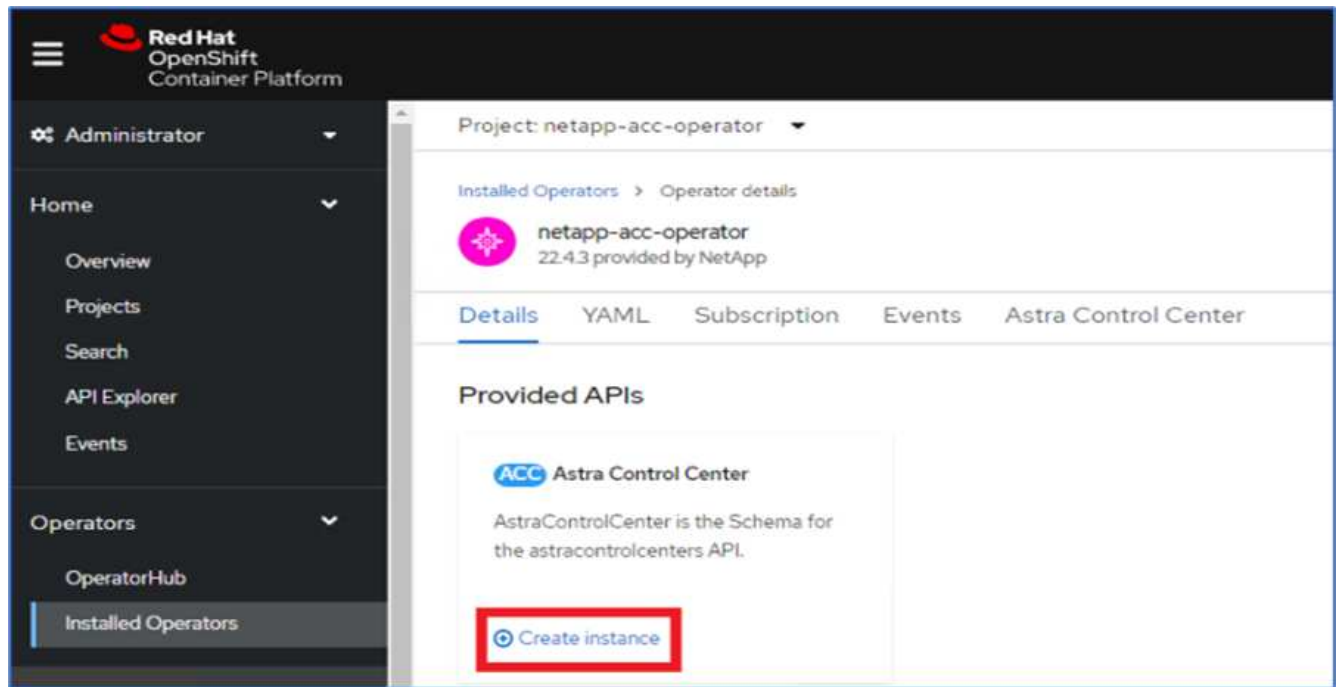
Username *

Password *

Email

[+ Add credentials](#)

15. 측면 메뉴에서 Operators > Installed Operators 를 선택하고 제공된 API 섹션에서 Create instance 를 클릭합니다.



16. Create AstraControlCenter 양식을 완료합니다. 이름, Astra 주소 및 Astra 버전을 입력합니다.

The screenshot shows the 'Create AstraControlCenter' form. The form is titled 'Create AstraControlCenter' and includes a note: 'Create by completing the form. Default values may be provided by the Operator authors.' The 'Configure via' section has 'Form view' selected. A note states: 'Note: Some fields may not be represented in this form view. Please select "YAML view" for full control.' The form fields are as follows:

- Name ***: acc
- Labels**: app=frontend
- Auto Support ***: (checked)
- Astra Address ***: acc.ocp.flexpod.netapp.com
- Astra Version ***: 22.04.0

Below the Astra Version field, there is a note: 'Version of AstraControlCenter to deploy. You are provided a Helm repository with a corresponding version. Example - 1.5.2, 1.4.2-patch'.



Astra Address 아래에 Astra Control Center의 FQDN 주소를 입력합니다. 이 주소는 Astra Control Center 웹 콘솔에 액세스하는 데 사용됩니다. FQDN도 연결 가능한 IP 네트워크로 확인되어야 하며 DNS에서 구성해야 합니다.

17. 계정 이름, 이메일 주소, 관리자 성을 입력하고 기본 볼륨 복원 정책을 유지합니다. 부하 분산 장치를 사용하는 경우 수신 유형을 "AccTraefik"로 설정합니다. 그렇지 않으면 Ingress.Controller로 Generic을 선택합니다. 이미지

레지스트리에서 컨테이너 이미지 레지스트리 경로 및 암호를 입력합니다.

Project: netapp-acc-operator

Account Name *
ocp
Astra Control Center account name

Email *
abhinav3@netapp.com
EmailAddress will be notified by Astra as events warrant.

Last Name
Singh
The last name of the SRE supporting Astra.

Volume Reclaim Policy
Retain
Reclaim policy to be set for persistent volumes

Ingress Type
AccTraefik
IngressType The type of ingress to that ACC should be configured for

Astra Kube Config Secret

AstraKubeConfigSecret if present and secret exists operator will attempt to add KubeConfig to Managed Clusters.

Image Registry
The container image registry that is hosting the Astra application images, ACC Operator and ACC Helm Repository.

Name

The name of the image registry. For example "example.registry/astra". Do not prefix with protocol.

Secret
astra-registry-cred
The name of the Kubernetes secret that will authenticate with the image registry.



이 솔루션에서는 Metallb 로드 밸런서가 사용됩니다. 따라서 수신 유형은 AccTraefik입니다. 그러면 Astra Control Center traefik 게이트웨이가 워크로드 밸런서의 Kubernetes 서비스로 표시됩니다.

18. admin 이름을 입력하고, 리소스 확장을 구성하고, 스토리지 클래스를 제공합니다. 생성 을 클릭합니다.

Image Registry
The container image registry that is hosting the Astra application images, ACC Operator and ACC Helm Repository.

First Name
Abhinav
The first name of the SRE supporting Astra.

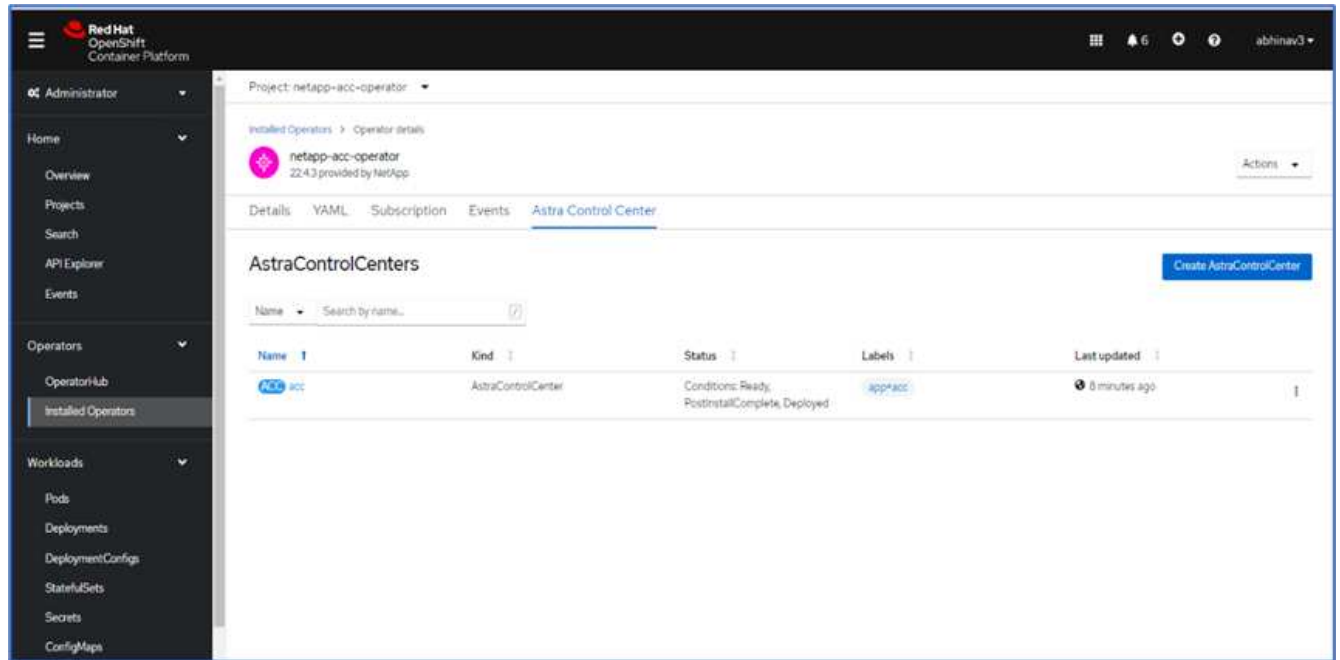
Astra Resources Scaler
Default
Scaling options for AstraControlCenter Resource limits.

Storage Class
ocp-nas-sc-gold
The storage class to be used for PVCs. If not set, default storage class will be used.

Crds
Options for how ACC should handle CRDs.Options for how ACC should handle CRDs.Options for how ACC should handle CRDs.Options for how ACC should handle CRDs.

Create **Cancel**

Astra Control Center 인스턴스의 상태는 배포 에서 준비 로 변경해야 합니다.



19. 모든 시스템 구성 요소가 성공적으로 설치되어 있고 모든 Pod가 실행 중인지 확인합니다.

```
root@abhinav-ansible# oc get pods -n netapp-acc-operator
NAME                                     READY   STATUS
RESTARTS   AGE
acc-helm-repo-77745b49b5-7zg2v         1/1     Running   0
10m
acc-operator-controller-manager-5c656c44c6-tqnmn  2/2     Running   0
13m
activity-589c6d59f4-x2sfs              1/1     Running   0
6m4s
api-token-authentication-4q5lj          1/1     Running   0
5m26s
api-token-authentication-pzptd          1/1     Running   0
5m27s
api-token-authentication-tbtg6          1/1     Running   0
5m27s
asup-669df8d49-qps54                   1/1     Running   0
5m26s
authentication-5867c5f56f-dnpp2         1/1     Running   0
3m54s
bucketservice-85495bc475-5zcc5          1/1     Running   0
5m55s
cert-manager-67f486bbc6-txhh6           1/1     Running   0
9m5s
cert-manager-cainjector-75959db744-4l5p5  1/1     Running   0
9m6s
```


cert-manager-webhook-765556b869-g6wdf	1/1	Running	0
9m6s			
cloud-extension-5d595f85f-txrfl	1/1	Running	0
5m27s			
cloud-insights-service-674649567b-5s4wd	1/1	Running	0
5m49s			
composite-compute-6b58d48c69-46vhc	1/1	Running	0
6m11s			
composite-volume-6d447fd959-chnrt	1/1	Running	0
5m27s			
credentials-66668f8ddd-8qc5b	1/1	Running	0
7m20s			
entitlement-fd6fc5c58-wxnmh	1/1	Running	0
6m20s			
features-756bbb7c7c-rgcrm	1/1	Running	0
5m26s			
fluent-bit-ds-278pg	1/1	Running	0
3m35s			
fluent-bit-ds-5pqc6	1/1	Running	0
3m35s			
fluent-bit-ds-8l7cq	1/1	Running	0
3m35s			
fluent-bit-ds-9qbft	1/1	Running	0
3m35s			
fluent-bit-ds-nj475	1/1	Running	0
3m35s			
fluent-bit-ds-x9pd8	1/1	Running	0
3m35s			
graphql-server-698d6f4bf-kftwc	1/1	Running	0
3m20s			
identity-5d4f4c87c9-wjz6c	1/1	Running	0
6m27s			
influxdb2-0	1/1	Running	0
9m33s			
krakend-657d44bf54-8cb56	1/1	Running	0
3m21s			
license-594bbdc-rghdg	1/1	Running	0
6m28s			
login-ui-6c65fbbbd4-jg8wz	1/1	Running	0
3m17s			
loki-0	1/1	Running	0
9m30s			
metrics-facade-75575f69d7-hnlk6	1/1	Running	0
6m10s			
monitoring-operator-65dff79cfb-z78vk	2/2	Running	0
3m47s			

nats-0 10m	1/1	Running	0
nats-1 9m43s	1/1	Running	0
nats-2 9m23s	1/1	Running	0
nautilus-7bb469f857-4hlc6 6m3s	1/1	Running	0
nautilus-7bb469f857-vz94m 4m42s	1/1	Running	0
openapi-8586db4bcd-gwwvf 5m41s	1/1	Running	0
packages-6bdb949cfb-nrq8l 6m35s	1/1	Running	0
polaris-consul-consul-server-0 9m22s	1/1	Running	0
polaris-consul-consul-server-1 9m22s	1/1	Running	0
polaris-consul-consul-server-2 9m22s	1/1	Running	0
polaris-mongodb-0 9m22s	2/2	Running	0
polaris-mongodb-1 8m58s	2/2	Running	0
polaris-mongodb-2 8m34s	2/2	Running	0
polaris-ui-5df7687dbd-trcnf 3m18s	1/1	Running	0
polaris-vault-0 9m18s	1/1	Running	0
polaris-vault-1 9m18s	1/1	Running	0
polaris-vault-2 9m18s	1/1	Running	0
public-metrics-7b96476f64-j88bw 5m48s	1/1	Running	0
storage-backend-metrics-5fd6d7cd9c-vc4j 5m59s	1/1	Running	0
storage-provider-bb85ff965-m7qrq 5m25s	1/1	Running	0
telegraf-ds-4zqgz 3m36s	1/1	Running	0
telegraf-ds-cp9x4 3m36s	1/1	Running	0
telegraf-ds-h4n59 3m36s	1/1	Running	0

telegraf-ds-jnp2q 3m36s	1/1	Running	0
telegraf-ds-pdz5j 3m36s	1/1	Running	0
telegraf-ds-znqtp 3m36s	1/1	Running	0
telegraf-rs-rt64j 3m36s	1/1	Running	0
telemetry-service-7dd9c74bfc-sfkzt 6m19s	1/1	Running	0
tenancy-d878b7fb6-wf8x9 6m37s	1/1	Running	0
traefik-6548496576-5v2g6 98s	1/1	Running	0
traefik-6548496576-g82pq 3m8s	1/1	Running	0
traefik-6548496576-psn49 38s	1/1	Running	0
traefik-6548496576-qrkfd 2m53s	1/1	Running	0
traefik-6548496576-srs6r 98s	1/1	Running	0
trident-svc-679856c67-78kbt 5m27s	1/1	Running	0
vault-controller-747d664964-xmn6c 7m37s	1/1	Running	0

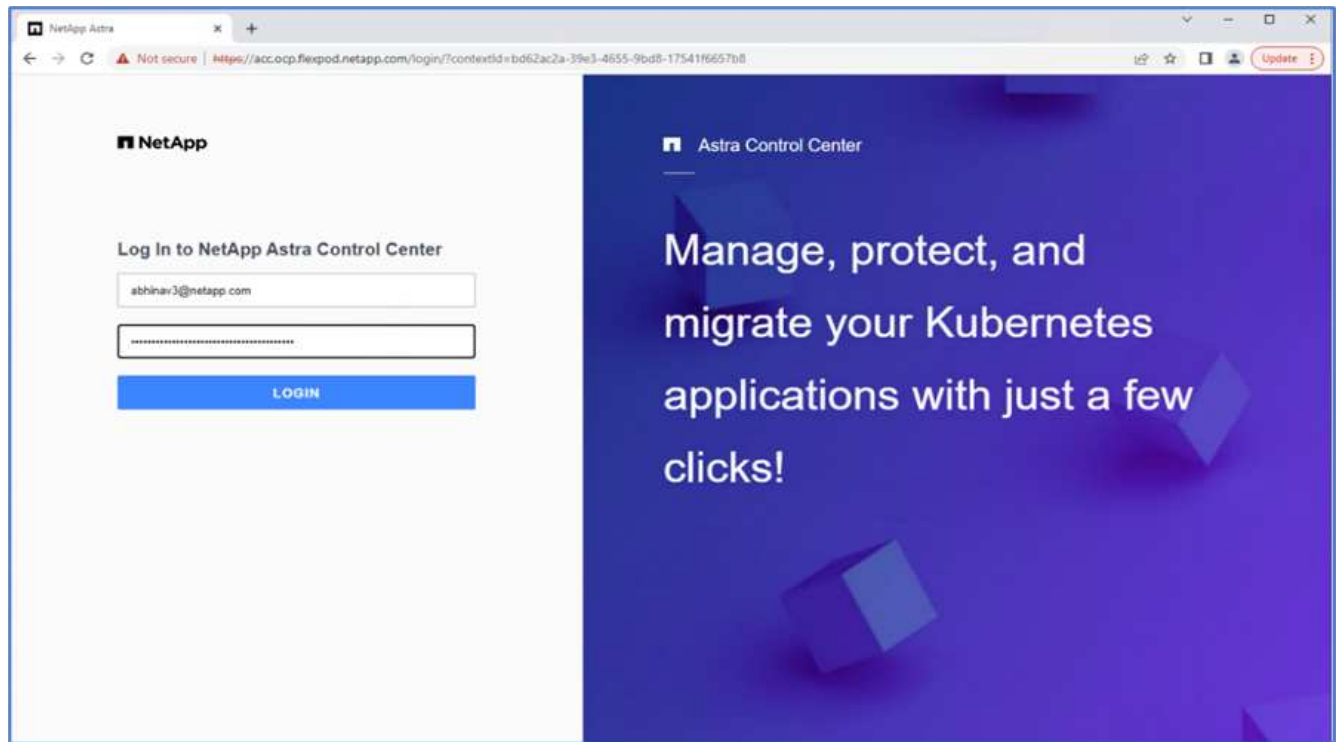


각 POD는 실행 중 상태여야 합니다. 시스템 포드를 구축하는 데 몇 분 정도 걸릴 수 있습니다.

20. 모든 Pod가 실행 중인 경우 다음 명령을 실행하여 1회 암호를 검색합니다. YAML 출력 버전에서 'tatus.deploymentState' 필드에 배포된 값을 확인한 후 'Status.uuid' 값을 복사합니다. 암호는 'ACC-'로, 이어서 UUID 값이 된다. (ACC - [UUID]).

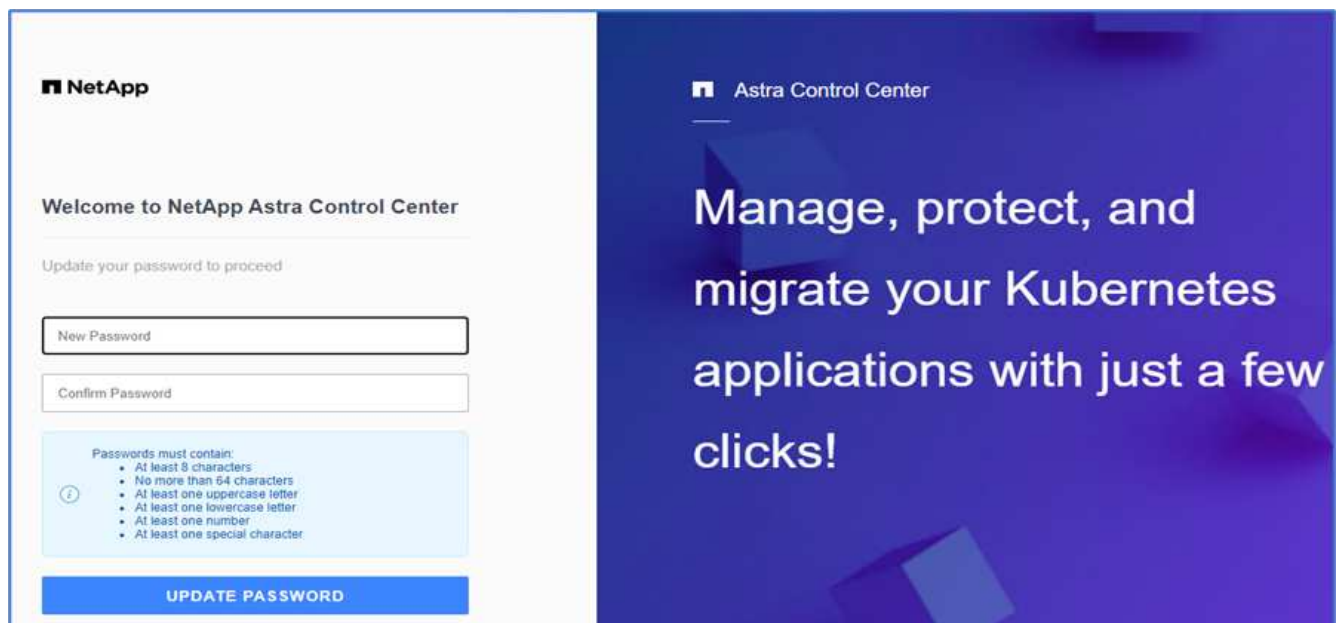
```
root@abhinav-ansible# oc get acc -o yaml -n netapp-acc-operator
```

21. 브라우저에서 제공한 FQDN을 사용하여 URL로 이동합니다.
22. 설치 중에 제공된 이메일 주소인 기본 사용자 이름과 1회 암호 ACC-[UUID]를 사용하여 로그인합니다.



잘못된 암호를 세 번 입력하면 관리자 계정이 15분 동안 잠깁니다.

23. 암호를 변경하고 계속 진행합니다.

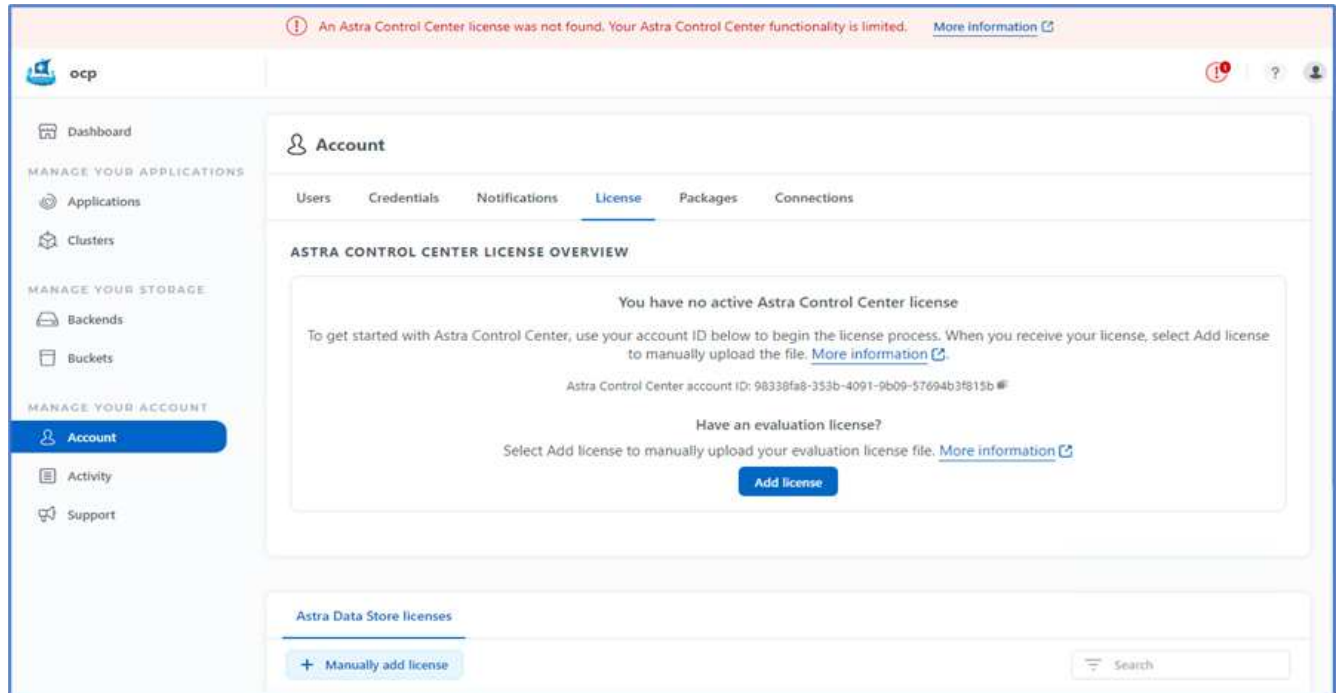


Astra Control Center 설치에 대한 자세한 내용은 를 참조하십시오 "[Astra Control Center 설치 개요](#)" 페이지.

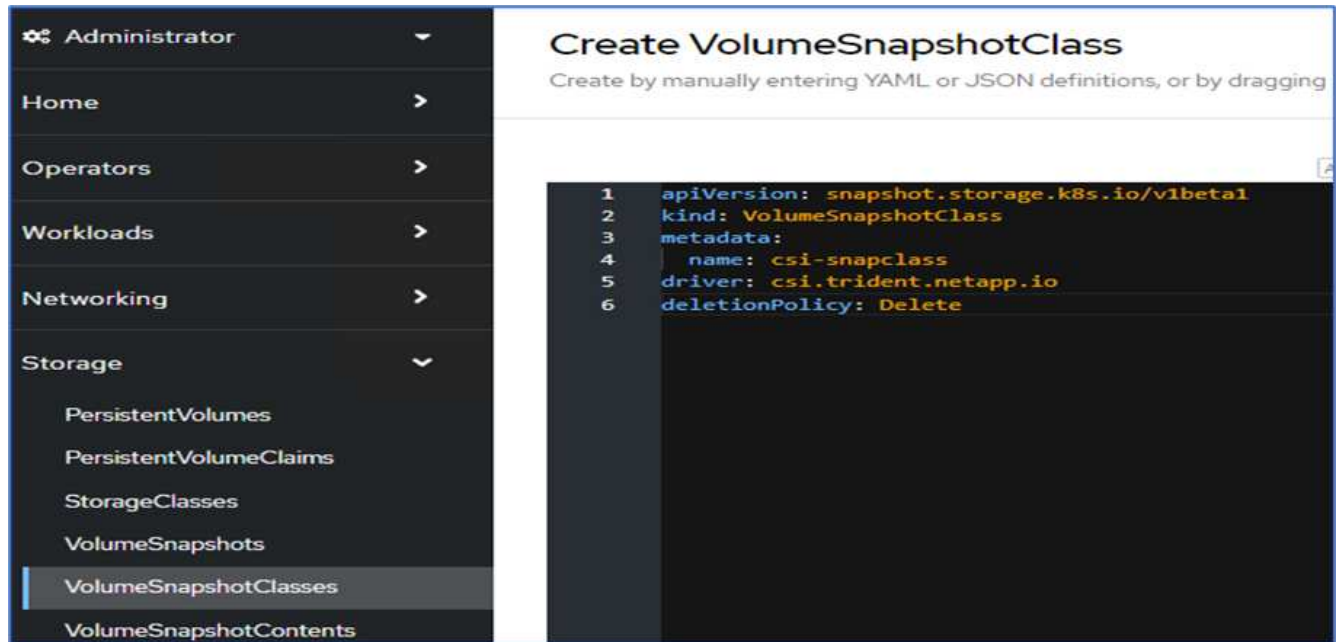
Astra Control Center를 설정합니다

Astra Control Center를 설치한 후 UI에 로그인하고 라이선스를 업로드하고, 클러스터를 추가하고, 스토리지를 관리하고, 버킷을 추가합니다.

1. 홈 페이지의 계정 에서 라이센스 탭으로 이동하고 라이센스 추가 를 선택하여 Astra 라이센스를 업로드합니다.



2. OpenShift 클러스터를 추가하기 전에 OpenShift 웹 콘솔에서 Astra Trident Volume 스냅샷 클래스를 생성합니다. 볼륨 스냅샷 클래스는 'csi.trident.netapp.io' 드라이버'로 구성됩니다.



3. Kubernetes 클러스터를 추가하려면 홈 페이지의 클러스터 로 이동하고 Kubernetes 클러스터 추가 를 클릭합니다. 그런 다음 클러스터에 대한 kubecononfig 파일을 업로드하고 자격 증명 이름을 제공합니다. 다음 을 클릭합니다.

Add Kubernetes cluster STEP 1/3: CREDENTIALS

CREDENTIALS

Provide Astra Control access to your Kubernetes and OpenShift clusters by entering a kubeconfig credential.
Follow [instructions](#) on how to create a dedicated admin-role kubeconfig.

Upload file Paste from clipboard

Kubeconfig YAML file
kubeconfig-noingress

Credential name
onprem-ocp-bm

Cancel Next →

4. 기존 스토리지 클래스가 자동으로 검색됩니다. 기본 스토리지 클래스를 선택하고 다음 을 클릭한 다음 클러스터 추가 를 클릭합니다.

Add cluster STEP 2/3: STORAGE

STORAGE

Existing storage classes are discovered and verified as eligible for use with Astra Control. You can use your existing default, or choose to set a new default at this time.
Applications with persistent volumes on eligible storage classes are validated for use with Astra Control.

Set default	Storage class	Storage provisioner	Reclaim policy	Binding mode	Eligible
<input checked="" type="radio"/>	ocp-nas-sc-gold	csi.trident.netapp.io	Delete	immediate	

← Back Next →

5. 몇 분 내에 클러스터가 추가됩니다. OpenShift Container Platform 클러스터를 추가하려면 1-4단계를 반복합니다.



OpenShift 운영 환경을 관리되는 컴퓨팅 리소스로 추가하려면 Astra Trident를 확인하십시오
"VolumeSnapshotClass 객체" 정의됩니다.

6. 스토리지를 관리하려면 백엔드로 이동하여 관리하려는 백엔드에 대한 작업에서 세 개의 점을 클릭합니다. 관리 를 클릭합니다.

Name	State	Capacity	Throughput	Type	Cluster	Cloud	Actions
c190-cluster	Discovered	Not available yet	Not available yet	ONTAP 9.11.1	Not applicable	Not applicable	⋮
healthylife	Discovered	Not available yet	Not available yet	ONTAP 9.11.1	Not applicable	Not applicable	⋮
singlevoaws	Discovered	Not available yet	Not available yet	ONTAP 9.11.1	Not applicable	Not applicable	⋮

7. ONTAP 자격 증명을 입력하고 Next를 클릭합니다. 정보를 검토하고 관리 를 클릭합니다. 백엔드는 다음 예제와 같아야 합니다.

Name	State	Capacity	Throughput	Type	Cluster	Cloud	Actions
c190-cluster	Available	0.4/10.64 TiB: 3.8%	Not available yet	ONTAP 9.11.1	Not applicable	Not applicable	⋮
healthylife	Available	5.16/106.42 TiB: 4.8%	Not available yet	ONTAP 9.11.1	Not applicable	Not applicable	⋮
singlevoaws	Available	0.07/0.62 TiB: 11.9%	Not available yet	ONTAP 9.11.1	Not applicable	Not applicable	⋮

8. Astra Control에 버킷을 추가하려면 Bucket 을 선택하고 Add 를 클릭합니다.

Name	Description	State	Type
------	-------------	-------	------

9. 버킷 유형을 선택하고 버킷 이름, S3 서버 이름 또는 IP 주소와 S3 자격 증명을 제공합니다. 업데이트 를 클릭합니다.



이 솔루션에서는 AWS S3 및 ONTAP S3 버킷을 모두 사용합니다. StorageGRID를 사용할 수도 있습니다.

Bucket 상태는 정상 상태여야 합니다.

Name	Description	State	Type	Actions
acc-aws-bucket		Healthy	Generic S3	
astra-bucket	On Prem S3 Bucket	Healthy	NetApp ONTAP S3	

Astra Control Center에서 애플리케이션 인식 데이터 관리를 위한 Kubernetes 클러스터 등록 시, Astra Control은 역할 바인딩 및 NetApp 모니터링 네임스페이스를 자동으로 생성하여 애플리케이션 Pod 및 작업자 노드에서 메트릭 및 로그를 수집합니다. 지원되는 ONTAP 기반 스토리지 클래스 중 하나를 기본값으로 설정합니다.

먼저 해 "[Astra Control 관리에 클러스터를 추가합니다](#)" 클러스터(Astra Control 외)에 앱을 설치한 다음 Astra Control의 앱 페이지로 이동하여 앱과 리소스를 관리할 수 있습니다. Astra로 앱을 관리하는 방법에 대한 자세한 내용은 [참조하십시오 "설명합니다"](#).

"다음: 솔루션 검증 개요"

저작권 정보

Copyright © 2025 NetApp, Inc. All Rights Reserved. 미국에서 인쇄된 본 문서의 어떠한 부분도 저작권 소유자의 사전 서면 승인 없이는 어떠한 형식이나 수단(복사, 녹음, 녹화 또는 전자 검색 시스템에 저장하는 것을 비롯한 그래픽, 전자적 또는 기계적 방법)으로도 복제될 수 없습니다.

NetApp이 저작권을 가진 자료에 있는 소프트웨어에는 아래의 라이선스와 고지사항이 적용됩니다.

본 소프트웨어는 NetApp에 의해 '있는 그대로' 제공되며 상품성 및 특정 목적에의 적합성에 대한 명시적 또는 묵시적 보증을 포함하여(이에 제한되지 않음) 어떠한 보증도 하지 않습니다. NetApp은 대체품 또는 대체 서비스의 조달, 사용 불능, 데이터 손실, 이익 손실, 영업 중단을 포함하여(이에 국한되지 않음), 이 소프트웨어의 사용으로 인해 발생하는 모든 직접 및 간접 손해, 우발적 손해, 특별 손해, 징벌적 손해, 결과적 손해의 발생에 대하여 그 발생 이유, 책임론, 계약 여부, 엄격한 책임, 불법 행위(과실 또는 그렇지 않은 경우)와 관계없이 어떠한 책임도 지지 않으며, 이와 같은 손실의 발생 가능성이 통지되었다 하더라도 마찬가지입니다.

NetApp은 본 문서에 설명된 제품을 언제든지 예고 없이 변경할 권리를 보유합니다. NetApp은 NetApp의 명시적인 서면 동의를 받은 경우를 제외하고 본 문서에 설명된 제품을 사용하여 발생하는 어떠한 문제에도 책임을 지지 않습니다. 본 제품의 사용 또는 구매의 경우 NetApp에서는 어떠한 특허권, 상표권 또는 기타 지적 재산권이 적용되는 라이선스도 제공하지 않습니다.

본 설명서에 설명된 제품은 하나 이상의 미국 특허, 해외 특허 또는 출원 중인 특허로 보호됩니다.

제한적 권리 표시: 정부에 의한 사용, 복제 또는 공개에는 DFARS 252.227-7013(2014년 2월) 및 FAR 52.227-19(2007년 12월)의 기술 데이터-비상업적 품목에 대한 권리(Rights in Technical Data -Noncommercial Items) 조항의 하위 조항 (b)(3)에 설명된 제한사항이 적용됩니다.

여기에 포함된 데이터는 상업용 제품 및/또는 상업용 서비스(FAR 2.101에 정의)에 해당하며 NetApp, Inc.의 독점 자산입니다. 본 계약에 따라 제공되는 모든 NetApp 기술 데이터 및 컴퓨터 소프트웨어는 본질적으로 상업용이며 개인 비용만으로 개발되었습니다. 미국 정부는 데이터가 제공된 미국 계약과 관련하여 해당 계약을 지원하는 데에만 데이터에 대한 전 세계적으로 비독점적이고 양도할 수 없으며 재사용이 불가능하며 취소 불가능한 라이선스를 제한적으로 가집니다. 여기에 제공된 경우를 제외하고 NetApp, Inc.의 사전 서면 승인 없이는 이 데이터를 사용, 공개, 재생산, 수정, 수행 또는 표시할 수 없습니다. 미국 국방부에 대한 정부 라이선스는 DFARS 조항 252.227-7015(b)(2014년 2월)에 명시된 권한으로 제한됩니다.

상표 정보

NETAPP, NETAPP 로고 및 <http://www.netapp.com/TM>에 나열된 마크는 NetApp, Inc.의 상표입니다. 기타 회사 및 제품 이름은 해당 소유자의 상표일 수 있습니다.