NetApp Astra 및 Cisco Intersight, Red Hat OpenShift를 포함하는 FlexPod 하이브리드 클라우드

FlexPod

NetApp
March 25, 2024

This PDF was generated from https://docs.netapp.com/ko-kr/flexpod/hybrid-cloud/flexpod-rho-cvo-solution-overview.html on March 25, 2024. Always check docs.netapp.com for the latest.

목차

NetApp Astra 및 Cisco Intersight, Red Hat OpenShift를 포함하는 FlexPod 하이브리드 클라우드 1
TR-4936: NetApp Astra 및 Cisco Intersight for Red Hat OpenShift를 포함하는 FlexPod 하이브리드 클라우드 1
솔루션 구성 요소 2
설치 및 구성 11
솔루션 검증 33
결론 54

NetApp Astra 및 Cisco Intersight, Red Hat OpenShift를 포함하는 FlexPod 하이브리드 클라우드

TR-4936: NetApp Astra 및 Cisco Intersight for Red Hat OpenShift를 포함하는 FlexPod 하이브리드 클라우드

Abhinav Singh입니다

소개

컨테이너와 Kubernetes가 컨테이너화된 앱의 개발, 구축, 실행, 관리, 확장을 위한 실질적인 선택이 됨에 따라 기업에서 비즈니스 크리티컬 애플리케이션을 점차 실행하고 있습니다. 비즈니스 크리티컬 애플리케이션은 주로 상태에 따라 달라집니다. 상태 저장 응용 프로그램은 상태, 데이터 및 구성 정보를 연결하며 비즈니스 논리를 실행하기 위해 이전 데이터 트랜잭션에 의존합니다. Kubernetes에서 실행 중인 비즈니스 크리티컬 애플리케이션은 기존 애플리케이션과 같은 가용성 및 비즈니스 연속성 요구사항을 지속적으로 충족합니다. 서비스 중단이 매출, 생산성 및 회사 평판에 심각한 영향을 미칠 수 있습니다. 따라서 클러스터, 온프레미스 데이터 센터 및 하이브리드 클라우드 환경 내부와 클러스터 간에 Kubernetes 워크로드를 빠르고 쉽게 보호, 복구, 이동하는 것이 매우 중요합니다. 기업들은 비즈니스를 하이브리드 클라우드 모델로 전환하고 애플리케이션을 클라우드 네이티브 폼 팩터로 전환하는 이점을 누리고 있습니다.

이 기술 보고서는 FlexPod 통합 인프라 솔루션 기반 Red Hat OpenShift Container Platform과 NetApp Astra Control Center를 결합하여 AWS(Amazon Web Services)로 확장하여 하이브리드 클라우드 데이터 센터를 구축합니다. 친숙함을 바탕으로 구축 "FlexPod 및 Red Hat OpenShift", 이 문서에서는 설치, 구성, 애플리케이션 보호 워크플로우 및 사내/클라우드 간 애플리케이션 마이그레이션부터 시작되는 NetApp Astra Control Center에 대해 설명합니다. 또한, Red Hat OpenShift에서 실행되는 컨테이너형 애플리케이션에 NetApp Astra Control Center를 사용할 때 백업 및 복구, 비즈니스 연속성 등의 애플리케이션 인식 데이터 관리 기능의 이점에 대해 설명합니다.

다음 그림에서는 솔루션 개요를 보여 줍니다.

[오류: 그래픽 이미지가 없습니다]

대상

이 문서는 CTO(최고 기술 책임자), 애플리케이션 개발자, 클라우드 솔루션 설계자, SRE(사이트 안정성 엔지니어), DevOps 엔지니어, ITOps 및 컨테이너식 애플리케이션의 설계, 호스팅, 관리에 중점을 둔 프로페셔널 서비스 팀을 대상으로 합니다.

NetApp Astra Control – 주요 사용 사례

NetApp Astra Control은 클라우드 네이티브 마이크로서비스를 처리하는 고객을 위해 애플리케이션 보호를 단순화하는 것을 목표로 합니다.

- * 스냅샷을 사용한 시점(PiT) 애플리케이션 표현 * Astra Control을 사용하면 Kubernetes에서 실행되는 애플리케이션의 구성 세부 정보와 관련 영구 스토리지를 포함하는 컨테이너화된 애플리케이션의 엔드 투 엔드 스냅샷을 생성할 수 있습니다. 인시던트가 발생한 경우 버튼 클릭 시 정상 작동이 확인된 상태로 응용 프로그램을 복원할 수 있습니다.
- * 전체 복제 애플리케이션 백업. * Astra Control을 사용하면 사전 정의된 일정에 따라 전체 애플리케이션 백업을 수행할 수 있습니다. 이 백업은 애플리케이션을 동일한 K8s 클러스터 또는 자동화된 방식으로 필요에 따라 다른 K8s 클러스터로 복원하는 데 사용할 수 있습니다.

- * 클론을 사용한 애플리케이션 이식성 및 마이그레이션 * Astra Control을 사용하면 전체 애플리케이션과 함께 Kubernetes 클러스터 간에 또는 동일한 K8s 클러스터 내에서 데이터를 복제할 수 있습니다. 이 기능은 또한 클러스터의 위치에 관계없이 K8s 클러스터 간에 애플리케이션을 포팅하거나 마이그레이션하는 데 도움이 됩니다(클론 생성 후 소스 애플리케이션 인스턴스를 삭제하기만 함).
- * 응용 프로그램 일관성 사용자 지정. * Astra Control을 사용하면 실행 후크를 활용하여 응용 프로그램 중지 상태 정의를 제어할 수 있습니다. 스냅샷 및 백업 워크플로우에 '사전' 및 '사후' 실행 후크를 놓으면 스냅샷 또는 백업을 수행하기 전에 응용 프로그램이 자동으로 정지됩니다.
- * 응용 프로그램 수준 재해 복구(DR) 자동화 * Astra Control을 사용하면 컨테이너화된 응용 프로그램에 대한 BCDR(비즈니스 연속성 재해 복구) 계획을 구성할 수 있습니다. NetApp SnapMirror는 백엔드에서 사용되며 DR 워크플로우의 전체 구현이 자동화됩니다.

솔루션 토폴로지

이 섹션에서는 솔루션의 논리적 토폴로지를 설명합니다.

다음 그림은 OpenShift 컨테이너 플랫폼 클러스터를 실행하는 FlexPod 사내 환경과 NetApp Cloud Volumes ONTAP, Cisco Intersight, NetApp Cloud Manager SaaS 플랫폼을 지원하는 AWS 기반 셀프 관리 OpenShift Container Platform 클러스터로 구성된 솔루션 토폴로지를 나타낸 것입니다.

[오류: 그래픽 이미지가 없습니다]

첫 번째 OpenShift Container Platform 클러스터는 FlexPod에 베어 메탈 설치이고, 두 번째 OpenShift Container Platform 클러스터는 FlexPod에서 실행되는 VMware vSphere에 배포되며, 세 번째 OpenShift Container Platform 클러스터는 로 배포됩니다 "프라이빗 클러스터" 자체 관리 인프라로서 AWS에 있는 기존 VPC(가상 프라이빗 클라우드)에 통합할 수 있습니다.

이 솔루션에서 FlexPod는 사이트 간 VPN을 통해 AWS에 연결되지만, 고객은 직접 연결 구현을 사용하여 하이브리드 클라우드로 확장할 수도 있습니다. Cisco Intersight는 FlexPod 인프라 구성 요소를 관리하는 데 사용됩니다.

이 솔루션에서 Astra Control Center는 FlexPod 및 AWS에서 실행되는 OpenShift Container Platform 클러스터에서 호스팅되는 컨테이너식 애플리케이션을 관리합니다. Astra Control Center는 FlexPod에서 실행되는 OpenShift 베어 메탈 인스턴스에 설치됩니다. Astra Control은 마스터 노드의 kube-API와 통신하며 Kubernetes 클러스터의 변경 사항을 지속적으로 확인합니다. K8s 클러스터에 추가된 모든 새 애플리케이션이 자동으로 검색되어 관리 작업에 사용할 수 있게 됩니다.

Astra Control Center를 사용하여 컨테이너화된 애플리케이션을 스냅샷으로 캡처할 수 있습니다. 애플리케이션 스냅샷은 예약된 보호 정책 또는 필요 시 트리거될 수 있습니다. Astra가 지원하는 애플리케이션의 경우 스냅샷이 충돌에 따라 정합성이 보장됩니다. 애플리케이션 스냅샷은 영구 볼륨의 애플리케이션 데이터 스냅샷과 해당 애플리케이션과 관련된 다양한 Kubernetes 리소스의 애플리케이션 메타데이터를 구성합니다.

Astra Control을 사용하여 사전 정의된 백업 스케줄을 사용하거나 필요에 따라 애플리케이션 전체 복제본 백업을 생성할 수 있습니다. 오브젝트 스토리지는 애플리케이션 데이터의 백업을 저장하는 데 사용됩니다. NetApp ONTAP S3, NetApp StorageGRID 및 모든 일반 S3 구현을 오브젝트 저장소로 사용할 수 있습니다.

"다음: 솔루션 구성 요소."

솔루션 구성 요소

"이전: 솔루션 개요"

FlexPod

FlexPod는 가상화된 솔루션과 가상화되지 않은 솔루션 모두를 위한 통합된 기반을 형성하는 정의된 하드웨어 및 소프트웨어 세트입니다. FlexPod에는 NetApp ONTAP 스토리지, Cisco Nexus 네트워킹, Cisco MDS 스토리지 네트워킹, Cisco UCS(Unified Computing System)가 포함되어 있습니다. 이 설계는 네트워킹, 컴퓨팅 및 스토리지가 하나의 데이터 센터 랙에 장착되거나 고객의 데이터 센터 설계에 따라 배포될 수 있을 만큼 유연합니다. 포트 밀도를 통해 네트워킹 구성 요소가 여러 구성을 수용할 수 있습니다.

아스트라 컨트롤

Astra Control은 퍼블릭 클라우드와 사내 모두에서 호스팅되는 클라우드 네이티브 애플리케이션을 위한 애플리케이션 인식 데이터 보호 서비스를 제공합니다. Astra Control은 Kubernetes에서 실행되는 컨테이너식 애플리케이션을 위한 데이터 보호, 재해 복구 및 마이그레이션 기능을 제공합니다.

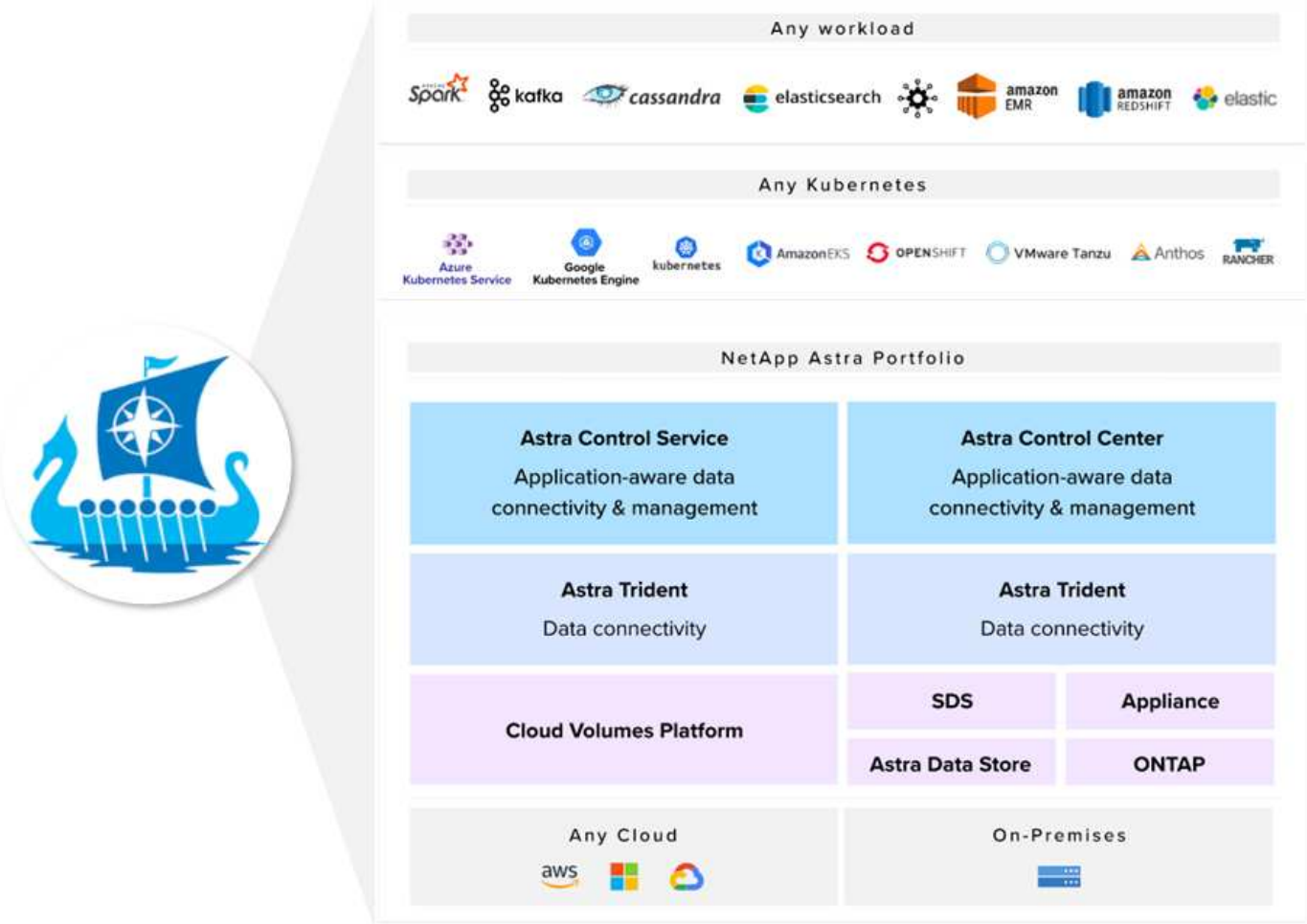
피처

Astra Control은 Kubernetes 애플리케이션 데이터 라이프사이클 관리에 중요한 기능을 제공합니다.

- 영구 스토리지를 자동으로 관리합니다
- 애플리케이션 정합성이 보장되는 온디맨드 스냅샷 및 백업을 생성합니다
- 자동화된 정책 기반 스냅샷 및 백업 작업
- 하이브리드 클라우드 설정에서 Kubernetes 클러스터 간에 애플리케이션 및 관련 데이터를 마이그레이션합니다
- 애플리케이션을 동일한 K8s 클러스터 또는 다른 K8s 클러스터로 클론 복제합니다
- 애플리케이션 보호 상태를 시각화합니다
- 그래픽 사용자 인터페이스 및 REST API의 전체 목록을 제공하여 기존 사내 툴에서 모든 보호 워크플로우를 구현합니다.

Astra Control은 Kubernetes 클러스터에서 생성된 관련 리소스에 대한 통찰력을 포함하는 컨테이너식 애플리케이션에 대한 단일 창 방식의 시각화를 제공합니다. 단일 포털을 통해 모든 클러스터, 모든 앱, 모든 클라우드 또는 모든 데이터 센터를 볼 수 있습니다. 온프레미스 또는 퍼블릭 클라우드의 모든 환경에서 Astra Control API를 사용하여 데이터 관리 워크플로우를 구축할 수 있습니다.

다음 이미지는 Astra Control 기능을 보여줍니다.



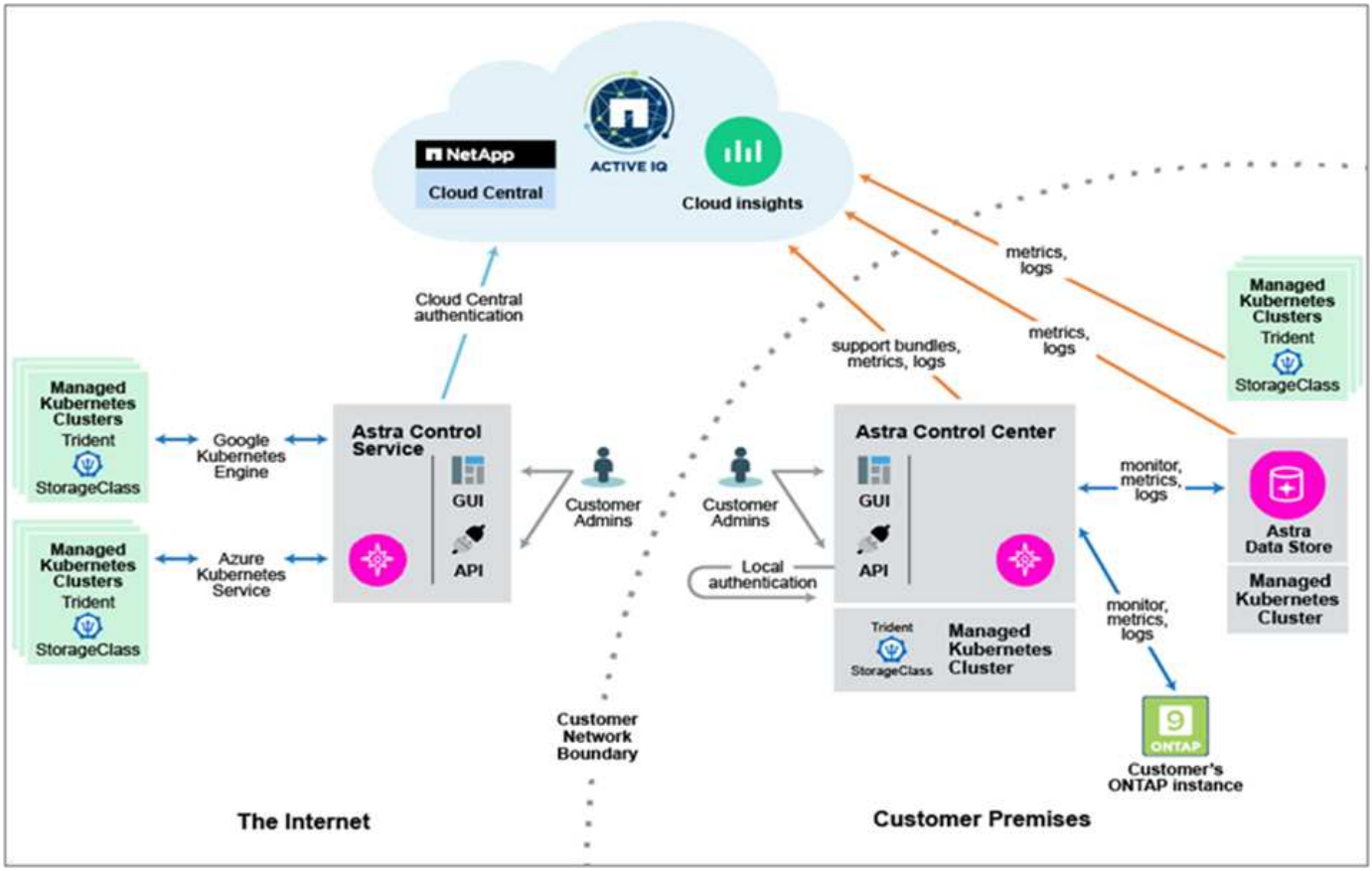
Astra Control 소비 모델

Astra Control은 두 가지 소비 모델로 제공됩니다.

- * Astra Control Service. * NetApp에서 호스팅하는 완전 관리형 서비스로, GKE(Google Kubernetes Engine), AKS(Azure Kubernetes Service)에서 Kubernetes 클러스터의 애플리케이션 인식 데이터 관리를 제공합니다.
- * Astra Control Center. * 사내 및 하이브리드 클라우드 환경에서 실행되는 Kubernetes 클러스터의 애플리케이션 인식 데이터 관리를 제공하는 자체 관리 소프트웨어입니다.

이 기술 보고서에서는 Kubernetes에서 실행되는 클라우드 네이티브 애플리케이션을 관리하기 위해 Astra Control Center를 활용합니다.

다음 이미지는 Astra Control 아키텍처를 보여 줍니다.



아스트라 트리덴트

Astra Trident는 컨테이너 및 Kubernetes 배포를 위한 완전히 지원되는 오픈 소스 스토리지 오케스트레이터입니다. 이 제품은 처음부터 와 같은 업계 표준 인터페이스를 사용하여 컨테이너화된 애플리케이션의 지속성 요구를 충족하도록 설계되었습니다 "컨테이너 스토리지 인터페이스(CSI)". Astra Trident를 사용하면 마이크로서비스 및 컨테이너식 애플리케이션에서 NetApp 스토리지 시스템 포트폴리오에서 제공하는 엔터프라이즈급 스토리지 서비스를 활용할 수 있습니다.

Astra Trident는 Kubernetes 클러스터에 Pod로 구축하고 Kubernetes 워크로드에 동적 스토리지 오케스트레이션 서비스를 제공합니다. 컨테이너화된 애플리케이션에서 NetApp ONTAP(NetApp AFF, NetApp FAS, NetApp ONTAP Select, 클라우드, Amazon FSx for NetApp ONTAP), NetApp Element 소프트웨어(NetApp SolidFire), Azure NetApp Files 서비스, Google Cloud 기반 Cloud Volume Service, AWS 기반 Cloud Volume Service FlexPod 환경에서 Astra Trident는 NetApp AFF, FAS 시스템, Cloud Volumes ONTAP와 같은 ONTAP 스토리지 플랫폼에서 호스팅되는 NetApp FlexVol 볼륨 및 LUN이 지원하는 컨테이너의 영구 볼륨을 동적으로 프로비저닝하고 관리하는 데 사용됩니다. 또한 Trident는 Astra Control에서 제공하는 애플리케이션 보호 체계를 구현하는 데 중요한 역할을 합니다. Astra Trident에 대한 자세한 내용은 를 참조하십시오 "Astra Trident 문서."

스토리지 백엔드

Astra Trident를 사용하려면 지원되는 스토리지 백엔드가 필요합니다. Trident 백엔드는 Trident와 스토리지 시스템 간의 관계를 정의합니다. Trident는 해당 스토리지 시스템과 통신하는 방법과 Trident가 스토리지 시스템에서 볼륨을 프로비저닝하는 방법을 알려줍니다. Trident는 스토리지 클래스에 정의된 요구 사항과 일치하는 백엔드에서 스토리지 풀을 자동으로 제공합니다.

- ONTAP AFF 및 FAS 스토리지 백엔드 스토리지 소프트웨어 및 하드웨어 플랫폼인 ONTAP는 핵심 스토리지 서비스, 다중 스토리지 액세스 프로토콜 지원 및 NetApp Snapshot 복사본 및 미러링과 같은 스토리지 관리 기능을 제공합니다.

- Cloud Volumes ONTAP 스토리지 백엔드
- "Astra 데이터 저장소" 스토리지 백엔드

NetApp Cloud Volumes ONTAP를 참조하십시오

NetApp Cloud Volumes ONTAP는 파일 및 블록 워크로드에 대한 고급 데이터 관리 기능을 제공하는 소프트웨어 정의 스토리지 오퍼링입니다. Cloud Volumes ONTAP을 사용하면 클라우드 스토리지 비용을 최적화하고 애플리케이션 성능을 높이는 동시에 데이터 보호, 보안 및 규정 준수를 향상할 수 있습니다.

주요 이점은 다음과 같습니다.

- 내장된 데이터 중복제거, 데이터 압축, 씬 프로비저닝 및 복제를 활용하여 스토리지 비용을 최소화합니다.
- 클라우드 환경에서 장애가 발생할 경우 엔터프라이즈급 안정성과 지속적인 운영을 보장합니다.
- Cloud Volumes ONTAP는 업계 최고 수준의 NetApp 복제 기술인 SnapMirror를 활용하여 사내 데이터를 클라우드로 복제하므로 여러 사용 사례에서 보조 복사본을 쉽게 사용할 수 있습니다.
- Cloud Volumes ONTAP은 또한 Cloud Backup Service과 통합되어 클라우드 데이터를 보호하고 장기적으로 보관하기 위한 백업 및 복원 기능을 제공합니다.
- 애플리케이션을 오프라인으로 전환하지 않고도 필요에 따라 고성능 및 고성능 스토리지 풀 간에 전환할 수 있습니다.
- NetApp SnapCenter를 사용하여 스냅샷 복사본의 일관성 보장
- Cloud Volumes ONTAP는 데이터 암호화를 지원하고 바이러스 및 랜섬웨어에 대한 보호를 제공합니다.
- 클라우드 데이터 센스와 통합되어 데이터 컨텍스트를 이해하고 중요한 데이터를 식별할 수 있습니다.

Cloud Central을 참조하십시오

Cloud Central은 NetApp 클라우드 데이터 서비스에 액세스하고 이를 관리할 수 있는 중앙의 위치입니다. 이러한 서비스를 사용하여 클라우드에서 중요한 애플리케이션을 실행하고, 자동화된 DR 사이트를 생성하고, 데이터를 백업하고, 여러 클라우드 간에 데이터를 효과적으로 마이그레이션 및 제어할 수 있습니다. 자세한 내용은 을 참조하십시오 "Cloud Central을 참조하십시오."

클라우드 관리자

Cloud Manager는 엔터프라이즈급 SaaS 기반 관리 플랫폼으로, IT 전문가 및 클라우드 설계자가 NetApp의 클라우드 솔루션을 사용하여 하이브리드 멀티 클라우드 인프라를 중앙에서 관리할 수 있도록 지원합니다. 이 솔루션은 사내 및 클라우드 스토리지를 중앙 집중식으로 확인 및 관리할 수 있는 시스템을 제공하여 하이브리드, 다중 클라우드 공급자 및 계정을 지원합니다. 자세한 내용은 을 참조하십시오 "클라우드 관리자".

커넥터

Connector는 Cloud Manager가 퍼블릭 클라우드 환경 내의 리소스 및 프로세스를 관리할 수 있도록 하는 인스턴스입니다. Cloud Manager가 제공하는 다양한 기능을 사용하려면 Connector가 필요합니다. Connector는 클라우드 또는 온프레미스 네트워크에 구축할 수 있습니다.

커넥터는 다음 위치에서 지원됩니다.

- 설치하고
- Microsoft Azure를 참조하십시오

- Google 클라우드
- 온프레미스

Connector에 대한 자세한 내용은 을 참조하십시오 "이 링크."

NetApp Cloud Insights를 참조하십시오

NetApp 클라우드 인프라 모니터링 툴인 Cloud Insights를 사용하면 Astra Control Center에서 관리하는 Kubernetes 클러스터의 성능과 활용률을 모니터링할 수 있습니다. Cloud Insights는 스토리지 사용량과 워크로드를 상호 연관시킵니다. Astra Control Center에서 Cloud Insights 연결을 활성화하면 Astra Control Center UI 페이지에 원격 측정 정보가 표시됩니다.

NetApp Active IQ Unified Manager를 참조하십시오

NetApp Active IQ Unified Manager를 사용하면 재설계된 직관적인 단일 인터페이스를 통해 ONTAP 스토리지 클러스터를 모니터링하여 커뮤니티의 지혜 및 AI 분석을 제공할 수 있습니다. 스토리지 환경과 VM(가상 시스템)에 대한 포괄적인 운영, 성능 및 사전 통찰력을 제공합니다. 스토리지 인프라에서 문제가 발생하면 Unified Manager에서 문제의 세부 정보를 알려 근본 원인을 파악하는 데 도움을 줄 수 있습니다. VM 대시보드에서는 VM의 성능 통계를 볼 수 있으므로 VMware vSphere 호스트에서 네트워크를 거쳐 마지막으로 스토리지까지 전체 입출력 경로를 조사할 수 있습니다. 일부 이벤트는 문제를 해결하기 위해 취할 수 있는 개선 조치도 제공합니다. 문제가 발생할 때 e-메일 및 SNMP 트랩을 통해 알림을 받도록 이벤트에 대한 사용자 지정 알림을 구성할 수 있습니다. Active IQ Unified Manager를 사용하면 용량 및 사용 추세를 예측하여 문제가 발생하기 전에 능동적으로 조치를 취함으로써 장기적인 문제를 해결할 수 있는 단기적인 의사 결정을 방지할 수 있으므로 사용자의 스토리지 요구 사항을 계획할 수 있습니다.

Cisco Intersight를 참조하십시오

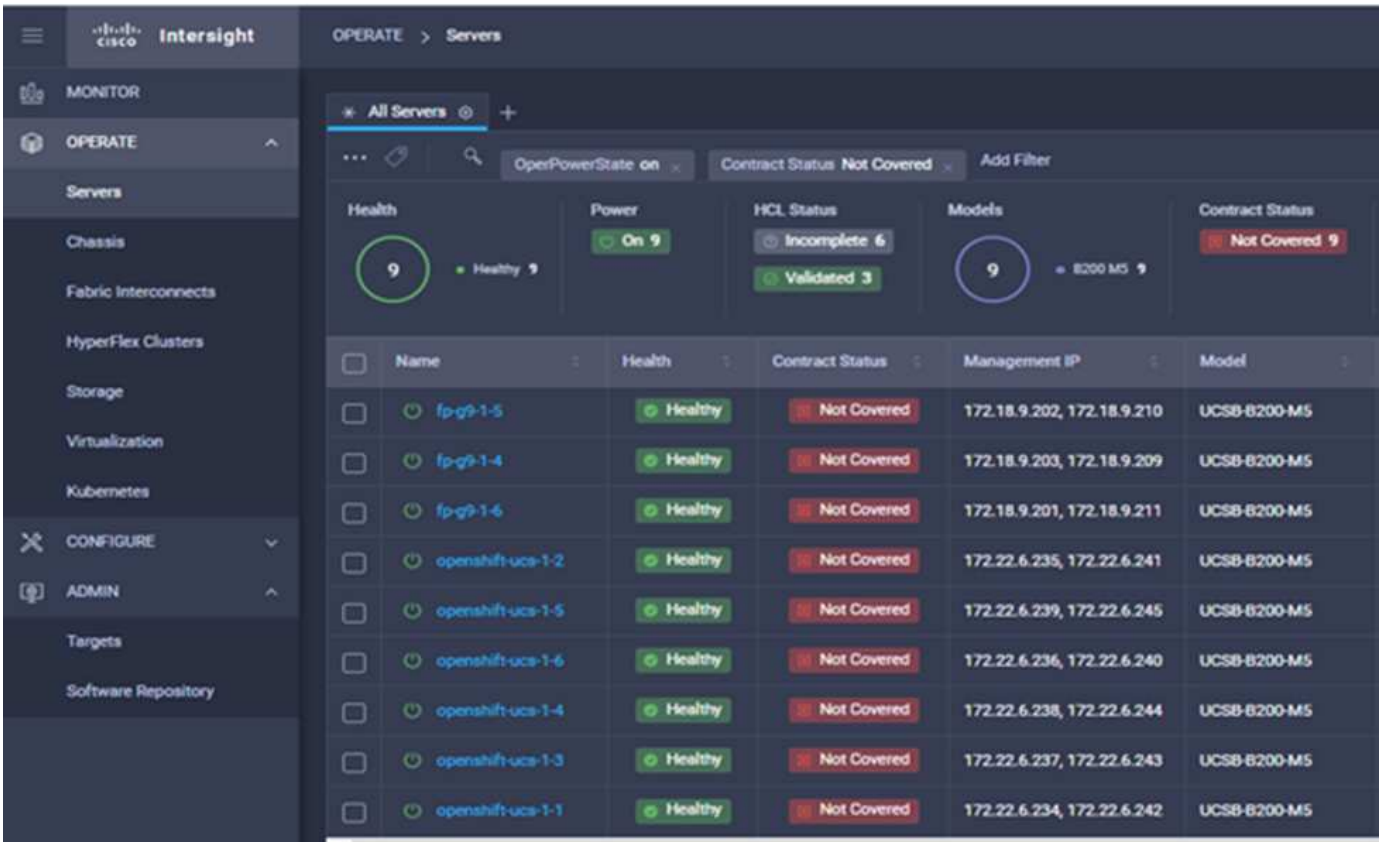
Cisco Intersight는 기존 및 클라우드 네이티브 애플리케이션과 인프라에 대한 지능형 자동화, 관찰 가능성 및 최적화를 제공하는 SaaS 플랫폼입니다. 이 플랫폼은 IT 팀의 변화를 이끌어내는 데 도움이 되며 하이브리드 클라우드용으로 설계된 운영 모델을 제공합니다.

Cisco Intersight는 다음과 같은 이점을 제공합니다.

- 신속한 제공.* 민첩한 기반 소프트웨어 개발 모델로 인해 잦은 업데이트와 지속적인 혁신을 통해 클라우드 또는 고객의 데이터 센터에서 서비스로 제공됩니다. 따라서 고객은 LOB에 대한 제공 속도를 높이는 데 주력할 수 있습니다.
- * 운영 간소화. * 공통의 인벤토리, 인증 및 API가 포함된 안전한 단일 툴을 사용하여 운영을 간소화함으로써 전체 스택 및 모든 위치에서 작업을 수행할 수 있으므로 팀 간의 사일로를 제거할 수 있습니다. 사내 물리적 서버 및 하이퍼바이저 관리부터 VM, K8s, 서버리스, 자동화, 사내 및 퍼블릭 클라우드 전반에서 최적화 및 비용 제어
- * 지속적인 최적화. * Cisco Intersight가 모든 계층 및 Cisco TAC에 제공하는 인텔리전스를 사용하여 환경을 지속적으로 최적화합니다. 이 인텔리전스는 권장 및 자동화 작업으로 변환되므로 워크로드 이동, 물리적 서버 상태 모니터링, K8s 클러스터 자동 크기 조정, 작업 중인 퍼블릭 클라우드 비용 절감 권장 사항 등 모든 변화에 실시간으로 대응할 수 있습니다.

Cisco Intersight를 사용하면 UCSM 관리 모드(UMM)와 Intersight 관리 모드(IMM)의 두 가지 관리 작업 모드를 사용할 수 있습니다. 패브릭 상호 연결의 초기 설정 중에 패브릭 연결 Cisco UCS 시스템에 대한 네이티브 UMM 또는 IMM을 선택할 수 있습니다. 이 솔루션에서는 네이티브 UMM이 사용됩니다.

다음 이미지는 Cisco Intersight 대시보드를 보여 줍니다.

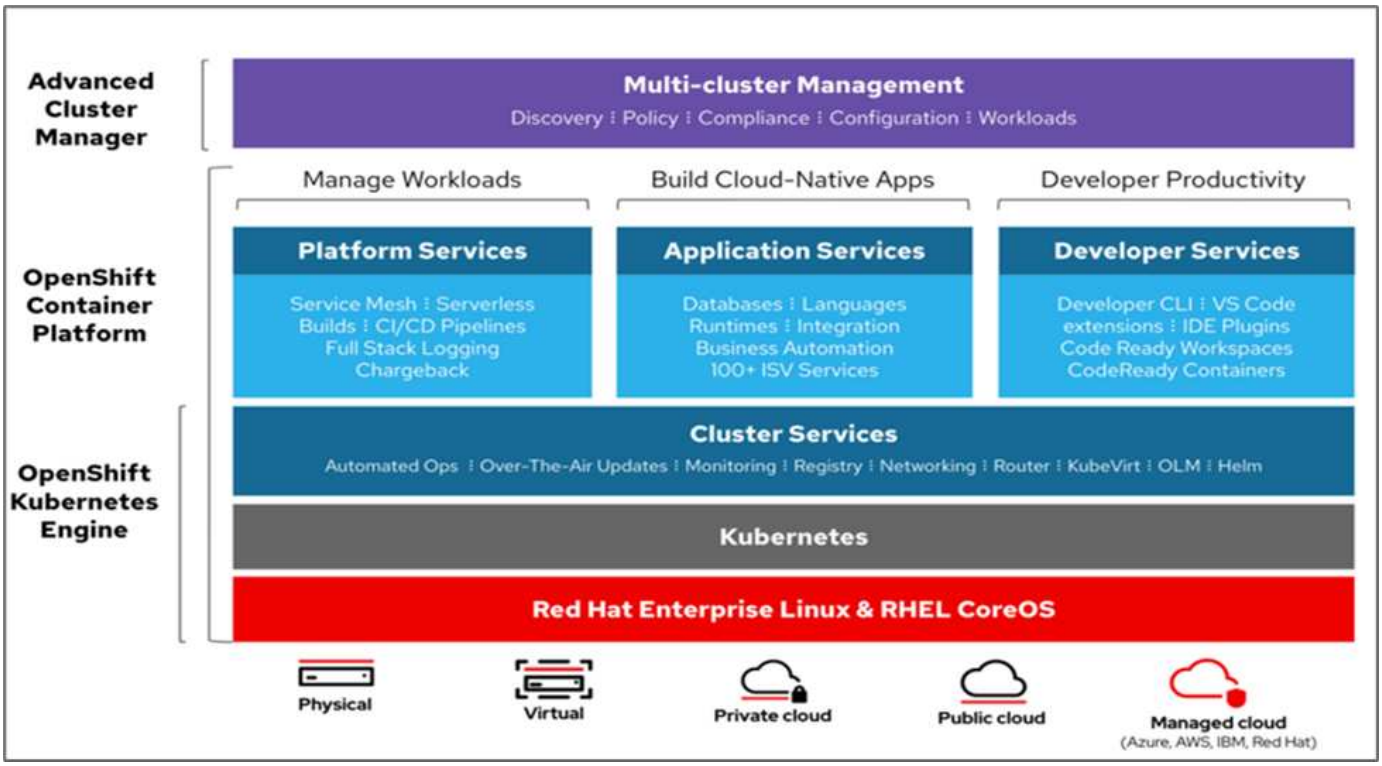


Red Hat OpenShift Container Platform

Red Hat OpenShift Container Platform은 컨테이너 애플리케이션 플랫폼으로, CRI-O와 Kubernetes를 통합하고 API 및 웹 인터페이스를 제공하여 이러한 서비스를 관리합니다. CRI-O는 Kubernetes 컨테이너 런타임 인터페이스(CRI)를 구현하여 OCI(Open Container Initiative) 호환 런타임 사용을 지원합니다. Kubernetes의 런타임 동안 Docker를 사용하는 것이 아니라 경량입니다.

OpenShift Container Platform을 사용하면 컨테이너를 만들고 관리할 수 있습니다. 컨테이너는 운영체제 및 기본 인프라와 관계없이 자체 환경에서 실행되는 독립 실행형 프로세스입니다. OpenShift Container Platform은 컨테이너 기반 애플리케이션을 개발, 배포 및 관리하는 데 도움이 됩니다. 또한 필요에 따라 애플리케이션을 생성, 수정 및 배포할 수 있는 셀프 서비스 플랫폼을 제공하므로 개발 및 릴리스 라이프사이클 단축이 가능합니다. OpenShift Container Platform은 마이크로서비스 기반의 작은 분리된 유닛으로, 서로 연동되는 아키텍처를 제공합니다. 이 제품은 Kubernetes 클러스터 위에서 실행되며, 안정적인 클러스터 키-가치 저장소인 etcd에 저장된 오브젝트 관련 데이터가 함께 사용됩니다.

다음 이미지는 Red Hat OpenShift Container 플랫폼의 개요입니다.



Kubernetes 인프라

OpenShift Container Platform에서 Kubernetes는 CRI-O 런타임 호스트 세트 전반에서 컨테이너식 애플리케이션을 관리하고 구축, 유지 관리, 애플리케이션 확장을 위한 메커니즘을 제공합니다. CRI-O 서비스 패키지는 컨테이너화된 애플리케이션을 인스턴스화하고 실행합니다.

Kubernetes 클러스터는 하나 이상의 마스터와 작업자 노드 집합으로 구성됩니다. 이 솔루션 설계에는 하드웨어 및 소프트웨어 스택의 고가용성(HA) 기능이 포함되어 있습니다. Kubernetes 클러스터는 3개의 마스터 노드와 2개 이상의 작업자 노드로 HA 모드에서 실행되도록 설계되어 있어 클러스터의 단일 장애 지점이 없도록 합니다.

Red Hat Core OS

OpenShift Container Platform은 CoreOS 및 Red Hat Atomic Host 운영 체제의 일부 최상의 기능을 결합한 컨테이너 중심의 운영 체제인 Red Hat Enterprise Linux CoreOS(RHCOS)를 사용합니다. RHCOS는 OpenShift Container Platform에서 컨테이너화된 애플리케이션을 실행하도록 특별히 설계되었으며, 새로운 툴과 함께 작동하여 빠른 설치, 운영자 기반 관리 및 간편한 업그레이드를 제공합니다.

RHCOS에는 다음과 같은 기능이 포함되어 있습니다.

- 점화 - OpenShift Container Platform이 처음 시스템을 가져와서 구성할 때 첫 번째 부팅 시스템 구성으로 사용하는 점화 시스템입니다.
- 운영 체제와 긴밀하게 통합된 Kubernetes 네이티브 컨테이너 런타임 구현인 CRI-O를 통해 효율적이고 최적화된 Kubernetes 경험을 제공할 수 있습니다. CRI-O는 컨테이너를 실행, 중지 및 재시작할 수 있는 기능을 제공합니다. OpenShift Container Platform 3에서 사용된 Docker 컨테이너 엔진을 완전히 대체합니다.
- Kubernetes의 기본 노드 에이전트인 Kubelet는 컨테이너 시작 및 모니터링을 담당합니다.

VMware vSphere 7.0

VMware vSphere는 CPU, 스토리지 및 네트워킹을 포함한 대규모 인프라 컬렉션을 완벽하고 다재다능한 동적 운영

환경으로 전체적으로 관리하는 가상화 플랫폼입니다. 개별 시스템을 관리하는 기존 운영 체제와 달리 VMware vSphere는 전체 데이터 센터의 인프라를 통합하여 필요한 애플리케이션에 빠르고 동적으로 할당할 수 있는 리소스를 갖춘 강력한 단일 시스템을 구축합니다.

자세한 내용은 을 참조하십시오 "VMware vSphere를 참조하십시오".

VMware vSphere vCenter를 참조하십시오

VMware vCenter Server는 단일 콘솔에서 모든 호스트와 VM을 통합 관리하고 클러스터, 호스트 및 VM의 성능 모니터링을 통합합니다. VMware vCenter Server를 통해 관리자는 컴퓨팅 클러스터, 호스트, VM, 스토리지, 게스트 OS, 가상 인프라스트럭처의 기타 주요 구성 요소 VMware vCenter는 VMware vSphere 환경에서 사용할 수 있는 다양한 기능을 관리합니다.

하드웨어 및 소프트웨어 개정

이 솔루션은 에 정의된 대로 지원되는 소프트웨어, 펌웨어 및 하드웨어 버전을 실행하는 FlexPod 환경으로 확장할 수 있습니다 "NetApp 상호 운용성 매트릭스 툴" 및 "Cisco UCS 하드웨어 호환성 목록" OpenShift 클러스터는 VMware vSphere뿐만 아니라 베어 메탈 방식으로 FlexPod에 설치됩니다.

여러 OpenShift(k8s) 클러스터를 관리하려면 Astra Control Center의 단일 인스턴스만 필요하고, 각 OpenShift 클러스터마다 Trident CSI가 설치됩니다. Astra Control Center는 이러한 OpenShift 클러스터에 설치할 수 있습니다. 이 솔루션에서 Astra Control Center는 OpenShift 베어 메탈 클러스터에 설치됩니다.

다음 표에는 OpenShift의 FlexPod 하드웨어 및 소프트웨어 개정 버전이 나와 있습니다.

구성 요소	제품	버전
컴퓨팅	Cisco UCS Fabric 인터커넥트 6454	4.1(3c)
	Cisco UCS B200 M5 서버	4.1(3c)
네트워크	Cisco Nexus 9336C-FX2 NX-OS	9.3(8)
스토리지	NetApp AFF A700	9.11.1
	NetApp Astra Control Center를 참조하십시오	22.04.0
	NetApp Astra Trident CSI 플러그인	22.04.0
	NetApp Active IQ Unified Manager를 참조하십시오	9.11
소프트웨어	VMware ESXi nenic 이더넷 드라이버	1.0.35.0
	vSphere ESXi	7.0(U2)
	VMware vCenter 어플라이언스	7.0 U2b
	Cisco Intersight Assist 가상 어플라이언스	1.0.9-342
	OpenShift 컨테이너 플랫폼	4.9
	OpenShift Container Platform 마스터 노드	RHCOS 4.9
	OpenShift 컨테이너 플랫폼 작업자 노드	RHCOS 4.9

다음 표에는 AWS의 OpenShift용 소프트웨어 버전이 나와 있습니다.

구성 요소	제품	버전
컴퓨팅	마스터 인스턴스 유형: M5.xLarge	해당 없음
	작업자 인스턴스 유형: M5.Large	해당 없음
네트워크	가상 프라이빗 클라우드 전송 게이트웨이	해당 없음
스토리지	NetApp Cloud Volumes ONTAP를 참조하십시오	9.11.1
	NetApp Astra Trident CSI 플러그인	22.04.0
소프트웨어	OpenShift 컨테이너 플랫폼	4.9
	OpenShift Container Platform 마스터 노드	RHCOS 4.9
	OpenShift 컨테이너 플랫폼 작업자 노드	RHCOS 4.9

"다음: OpenShift Container Platform 4 베어메탈 설치용 FlexPod"

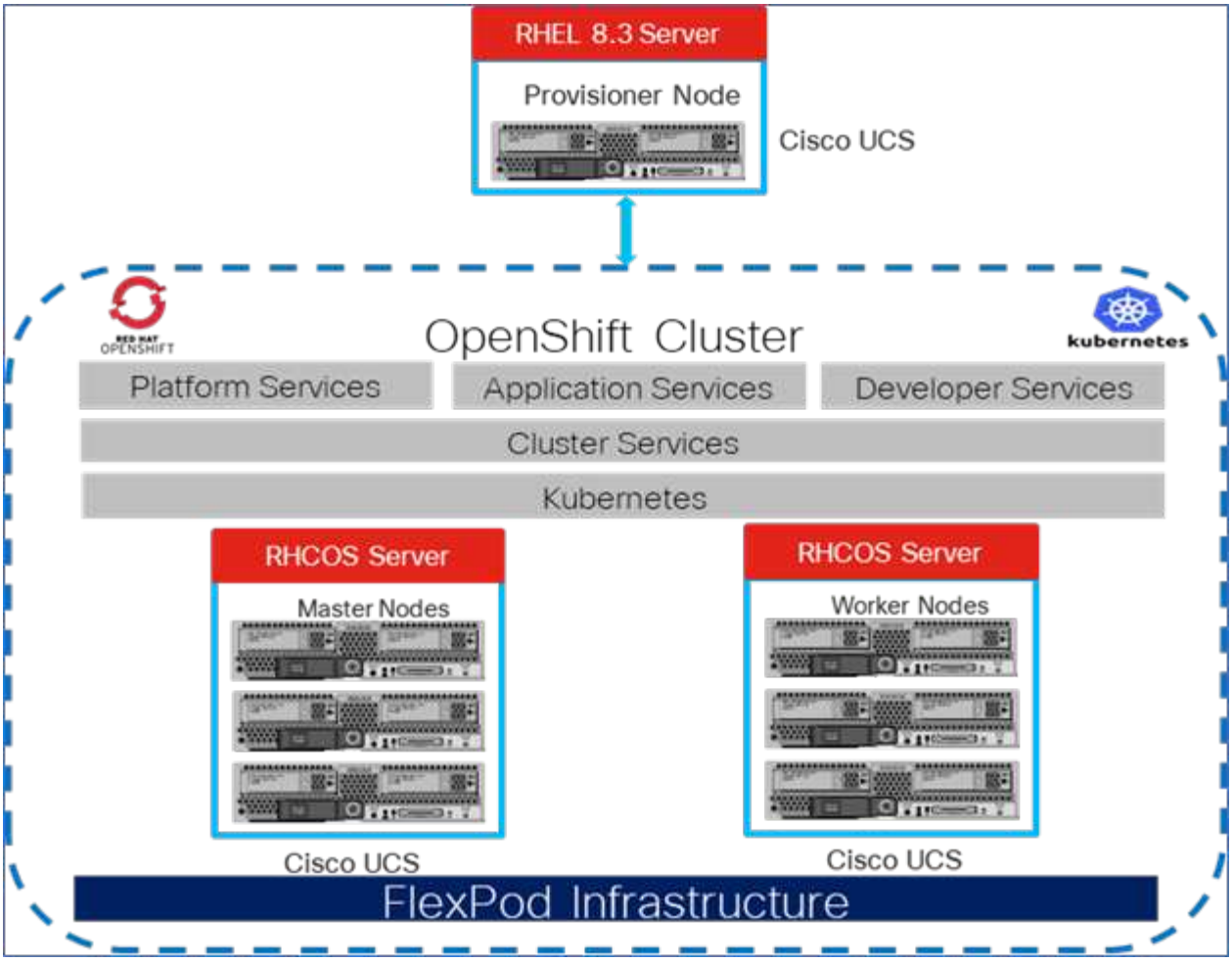
설치 및 구성

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.

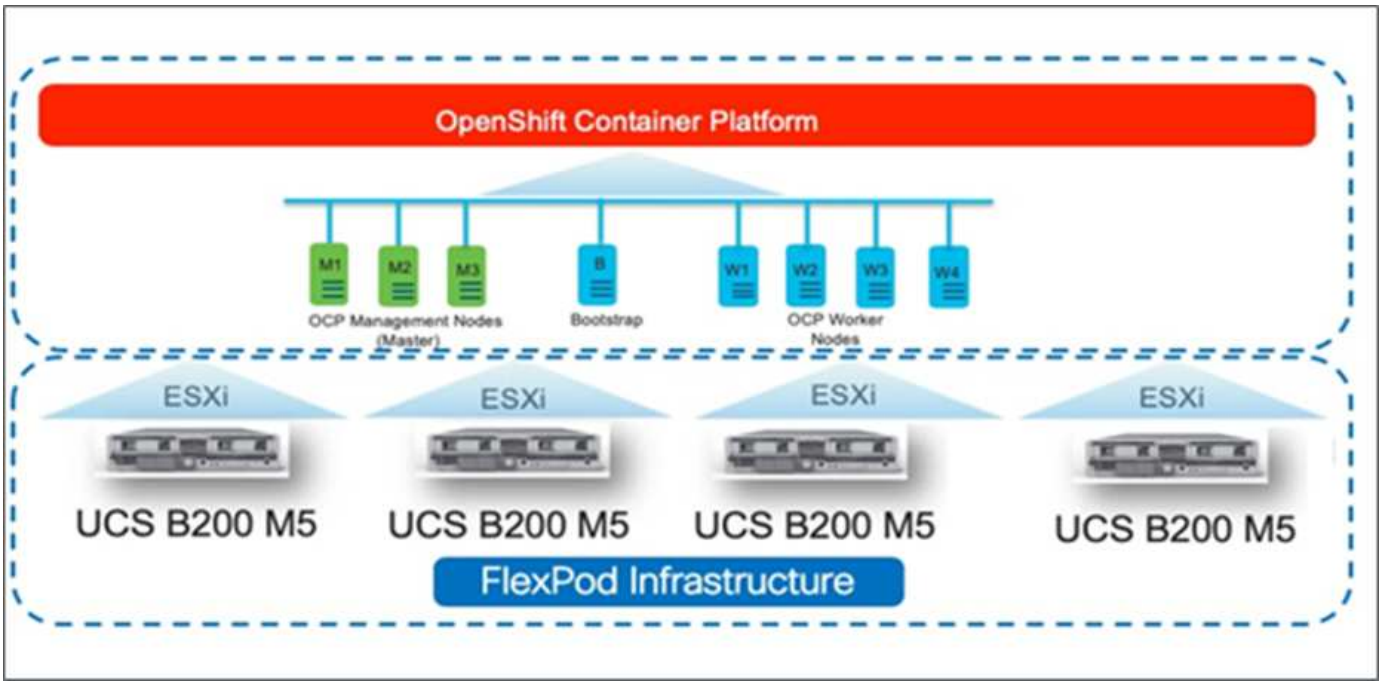
다음 그림은 FlexPod의 OpenShift 컨테이너 플랫폼 4 베어 메탈 을 보여 줍니다.



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 노드는 고가용성을 위해 세 개의 가용성 영역에 걸쳐 있습니다.

Instances (6) Info

Search

ocp | Clear filters

Name	Instance ID	Instance state	Instance type	Availability Zone	Private IP a...	Key name
ocpaws-v58kn-master-0	i-0d2d81ca91a54276d	Running	m5.xlarge	us-east-1b	172.30.165.160	–
ocpaws-v58kn-master-1	i-0b161945421d2a23c	Running	m5.xlarge	us-east-1c	172.30.166.162	–
ocpaws-v58kn-master-2	i-0146a665e1060ea59	Running	m5.xlarge	us-east-1a	172.30.164.209	–
ocpaws-v58kn-worker-us-east-1a-zj8dj	i-05e6efa18d136c842	Running	m5.large	us-east-1a	172.30.164.128	–
ocpaws-v58kn-worker-us-east-1b-7nmbc	i-0879a088b50d2d966	Running	m5.large	us-east-1b	172.30.165.93	–
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 엑스포트 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

- * 라이센스. * Astra Control Center를 사용하여 애플리케이션을 관리하려면 Astra Control Center 라이센스가 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	뮤atingWebhook	PersistentVolumeClaim
포드	팟캐스트 예산	팟캐스트 템플릿
네트워크 정책	ReplicaSet입니다	역할
RoleBinding 을 클릭합니다	루트	비밀
Webhook을 확인합니다		

OpenShift OperatorHub를 사용하여 **Astra Control Center**를 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.

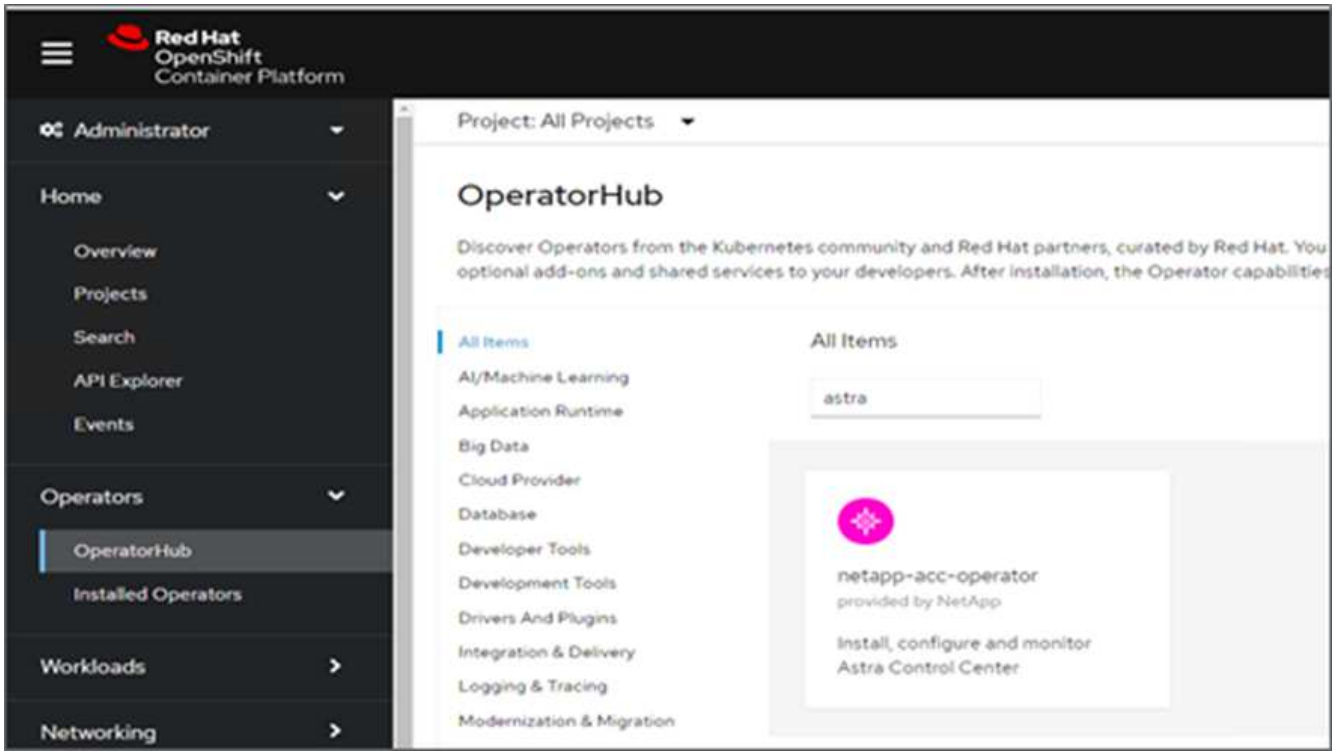
   Docker의 경우:

```
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:

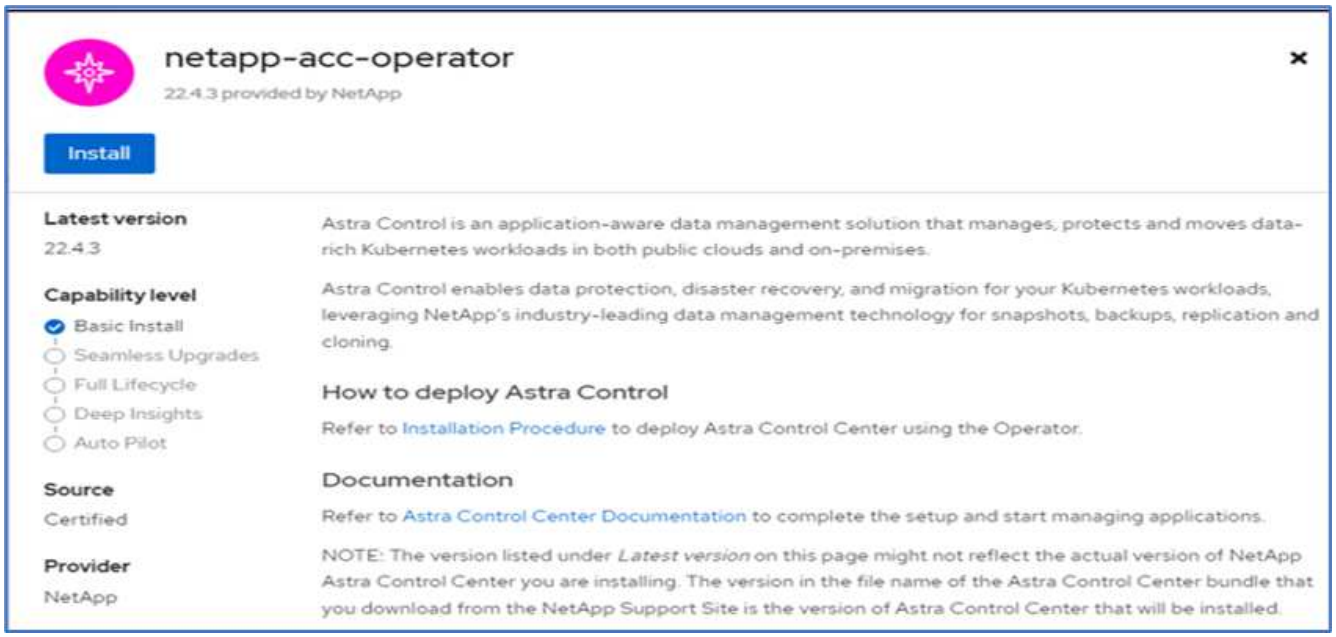
```
export REGISTRY=[Registry_path]
for astraImageFile in $(ls images/*.tar) ; do
  # Load to local cache. And store the name of the loaded image trimming
the 'Loaded images: '
  astraImage=$(podman load --input ${astraImageFile} | sed 's/Loaded
image(s): //')
  astraImage=$(echo ${astraImage} | sed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

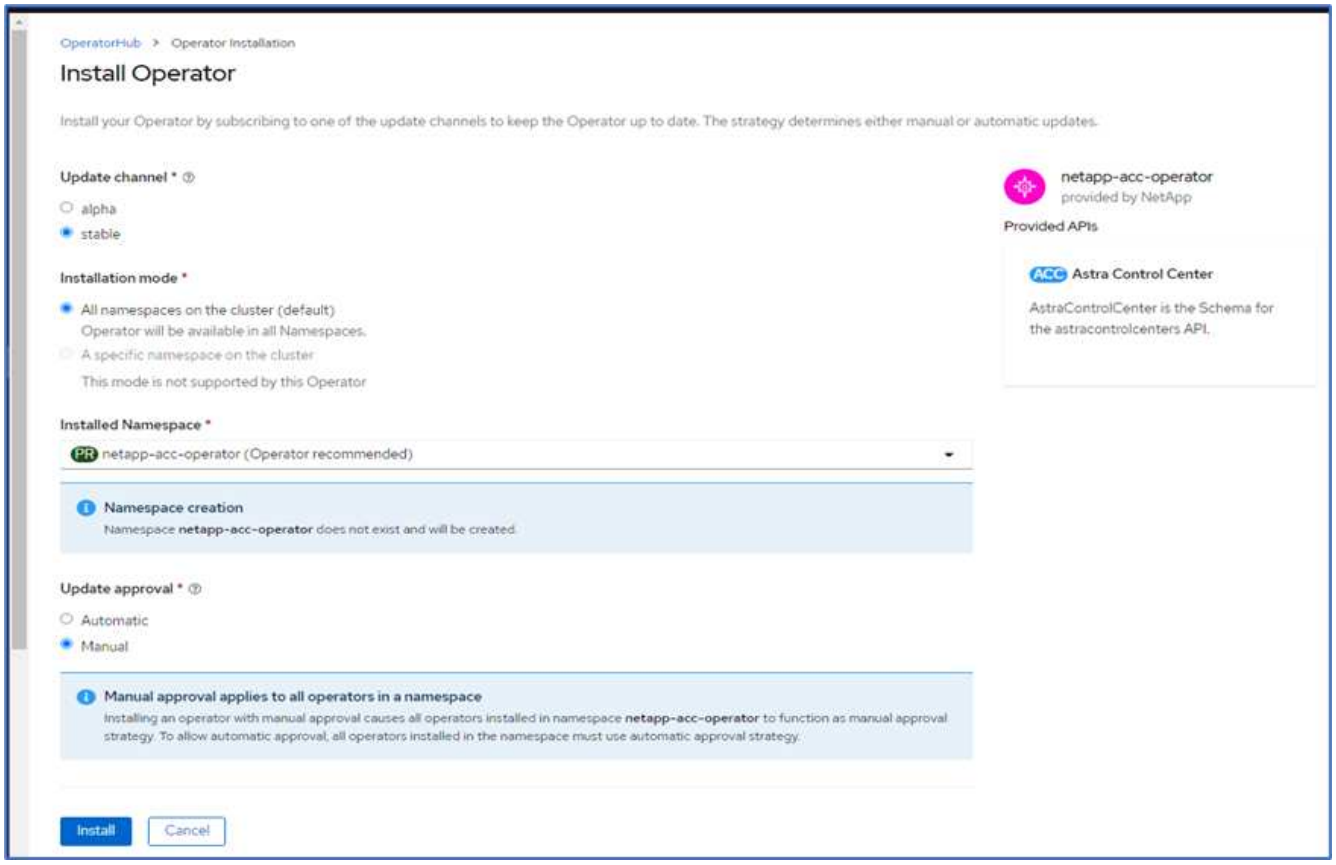


> 'NetApp-acc-operator'는 인증된 Red Hat OpenShift Operator로, OperatorHub 카탈로그 아래에 나열되어 있습니다.

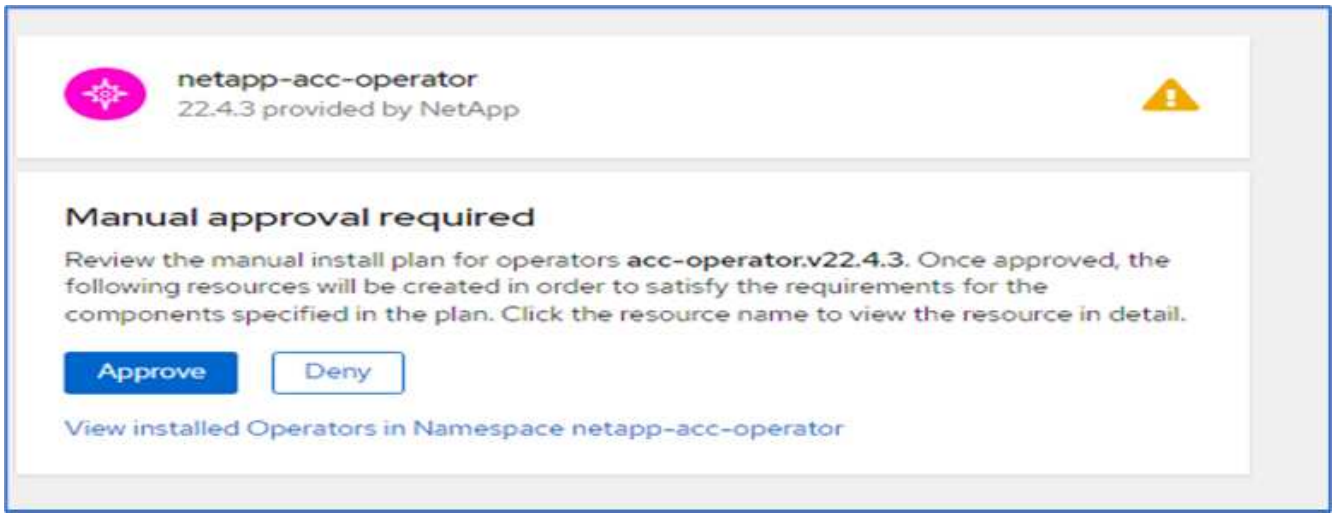
9. 'NetApp-acc-operator'를 선택하고 '설치'를 클릭합니다.



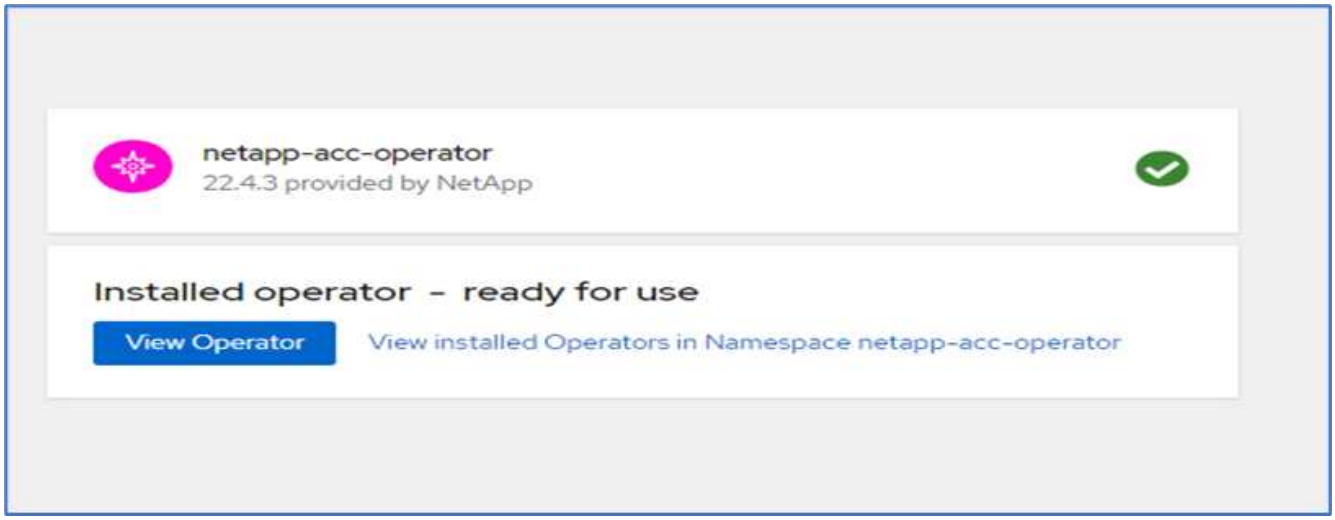
10. 적절한 옵션을 선택하고 설치 를 클릭합니다.



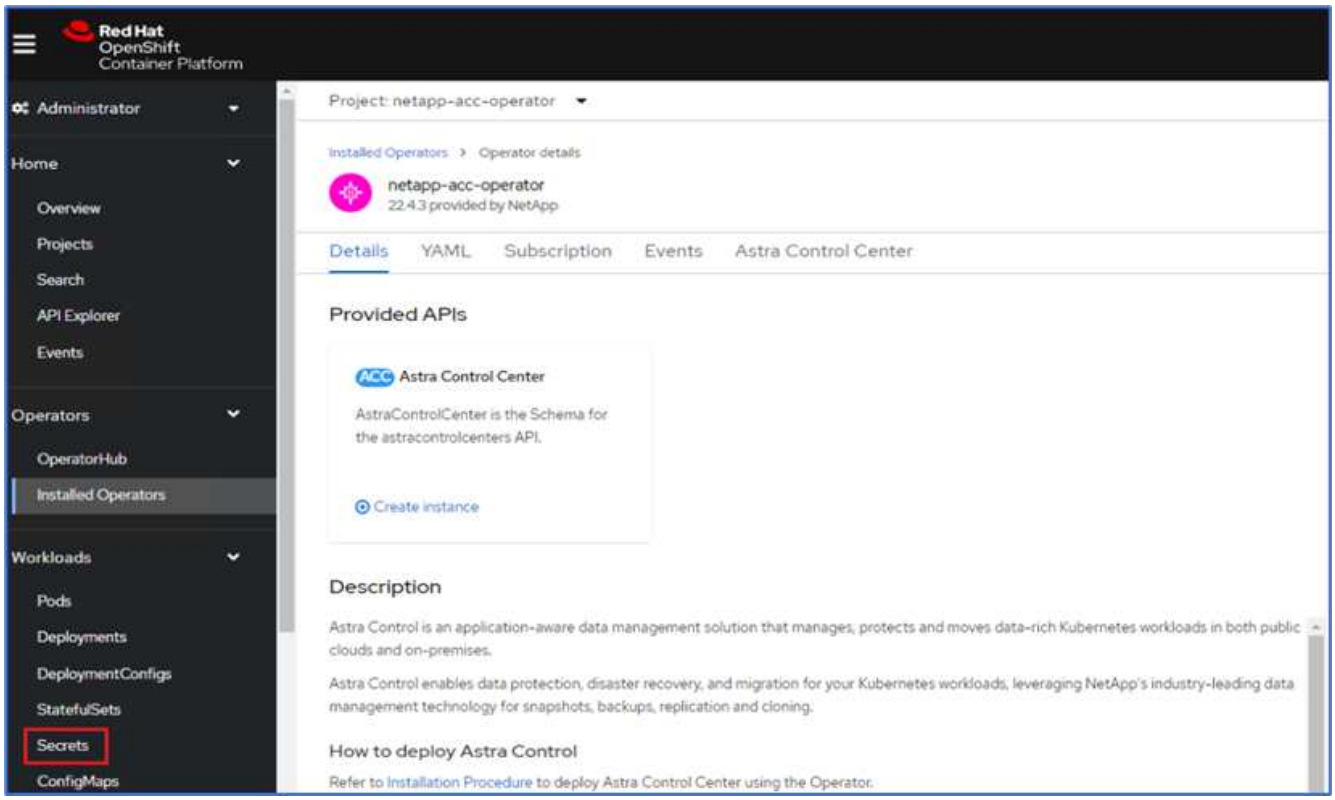
11. 설치를 승인하고 운영자가 설치될 때까지 기다립니다.



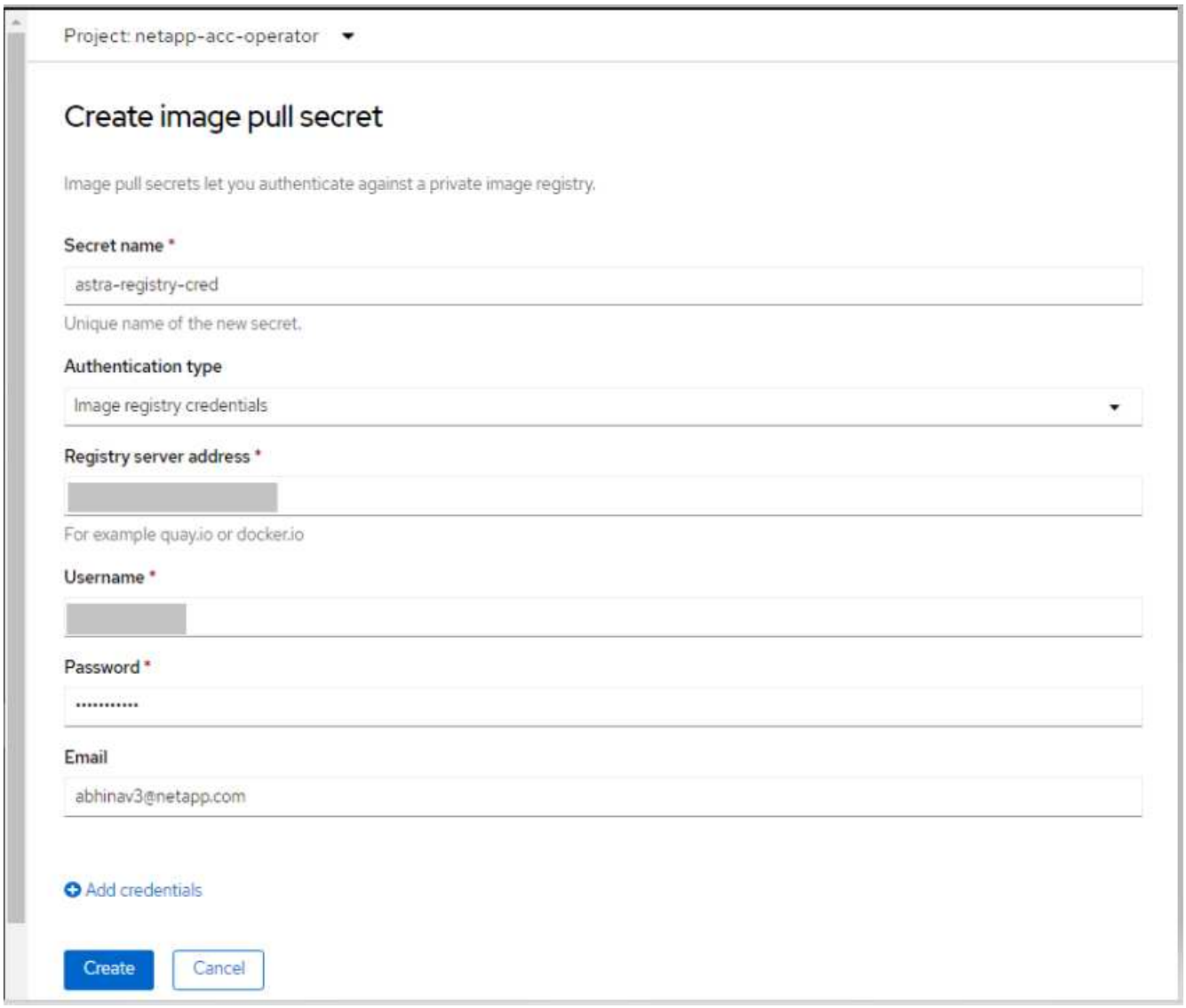
12. 이 단계에서 오퍼레이터는 성공적으로 설치되고 사용할 준비가 완료됩니다. View Operator(작업자 보기) 를 클릭하여 Astra Control Center 설치를 시작합니다.



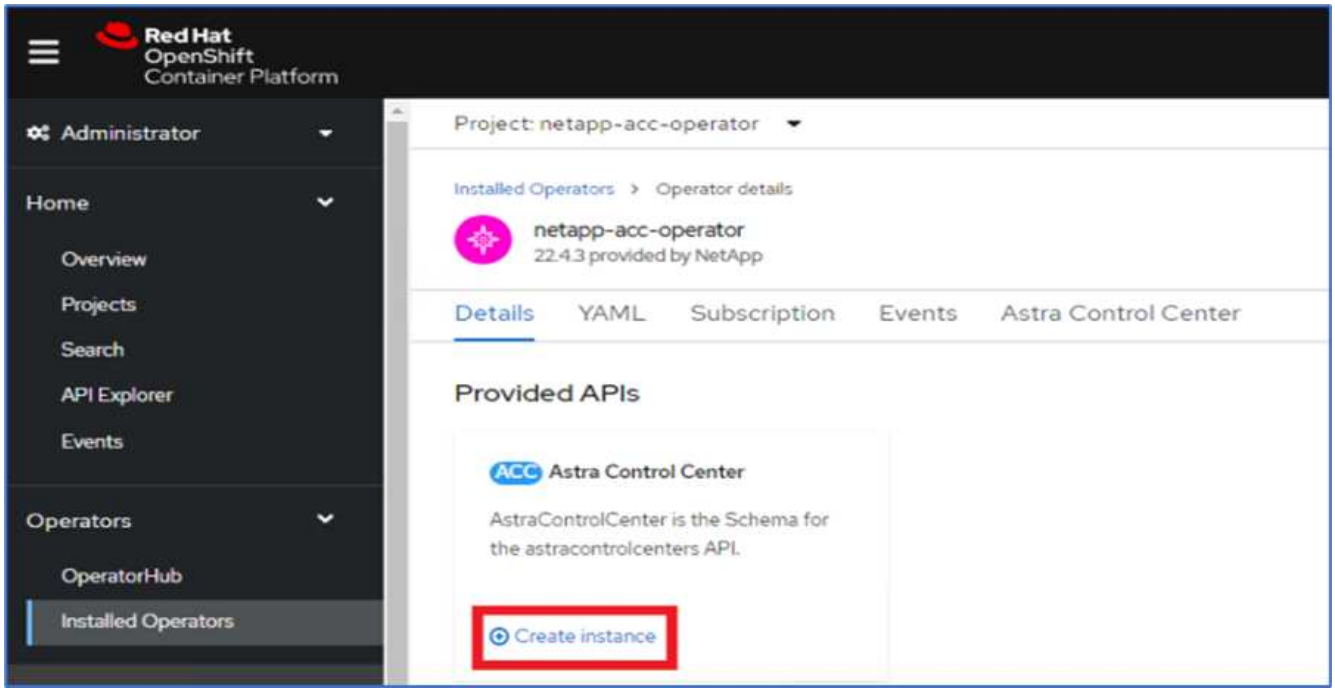
13. Astra Control Center를 설치하기 전에 앞서 푸시한 Docker 레지스트리에서 Astra 이미지를 다운로드하기 위한 풀 비밀을 생성합니다.



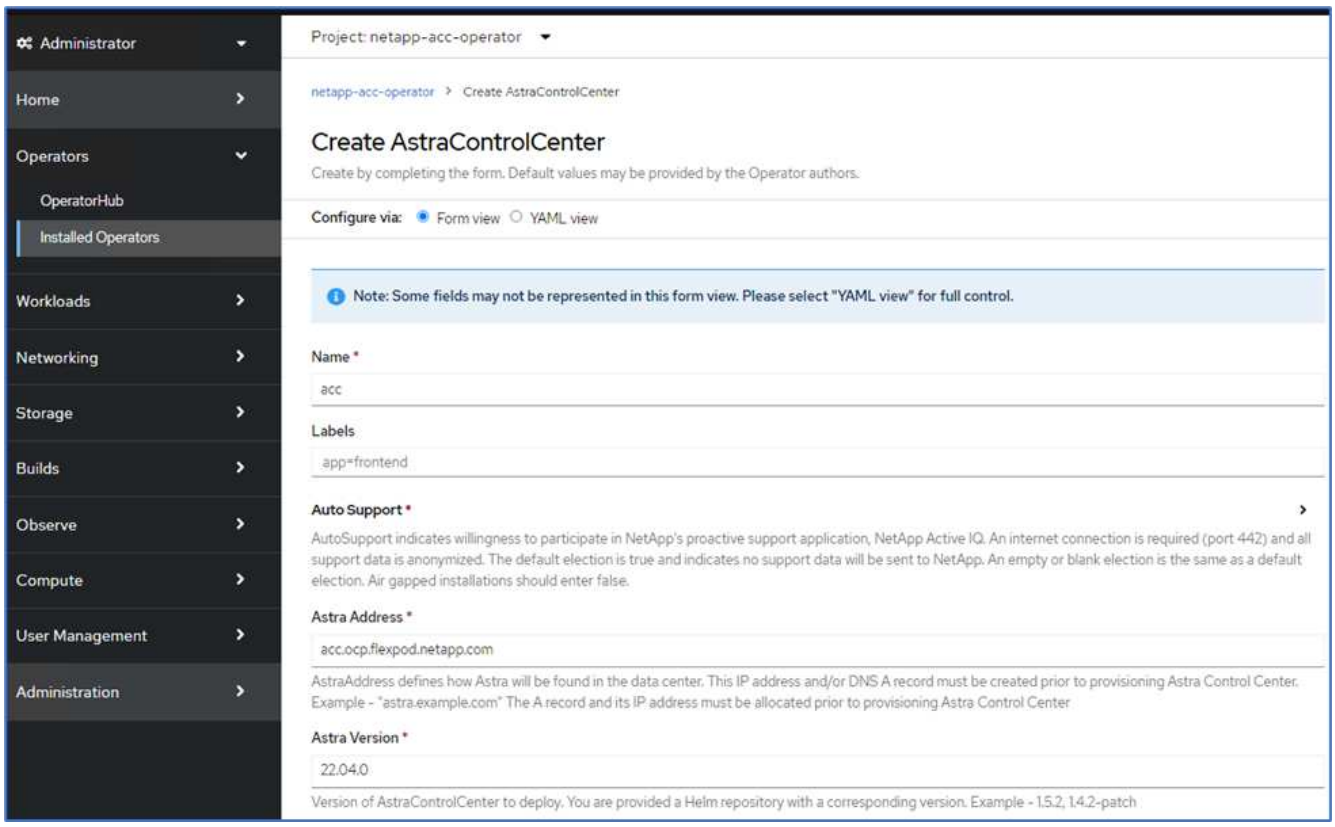
14. Docker 프라이빗 레포에서 Astra Control Center 이미지를 풀려면 'NetApp-acc-operator' 네임스페이스에서 암호를 생성하십시오. 이 비밀 이름은 나중에 Astra Control Center YAML manifest에서 제공합니다.



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



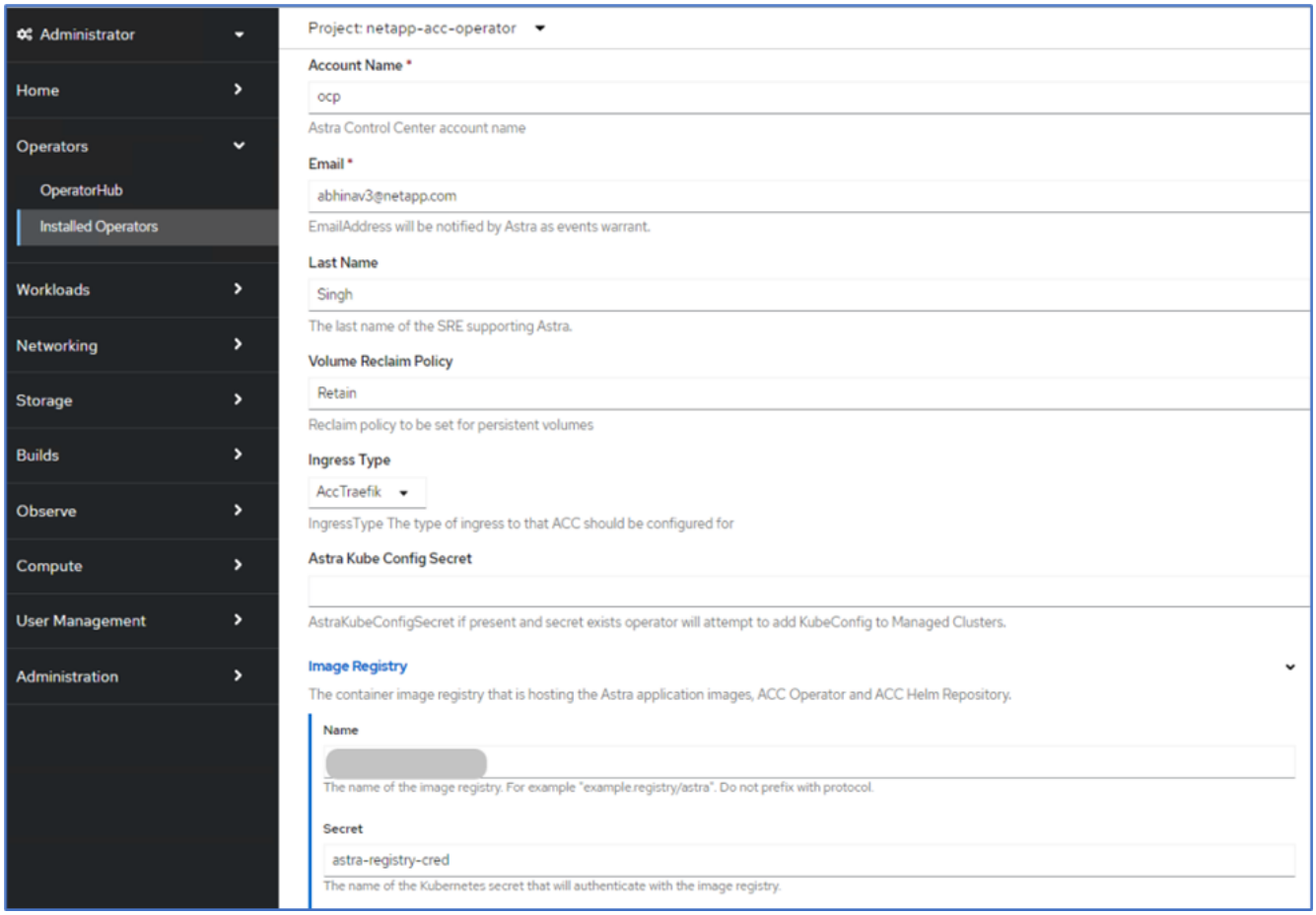
16. Create AstraControlCenter 양식을 완료합니다. 이름, Astra 주소 및 Astra 버전을 입력합니다.



> 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.

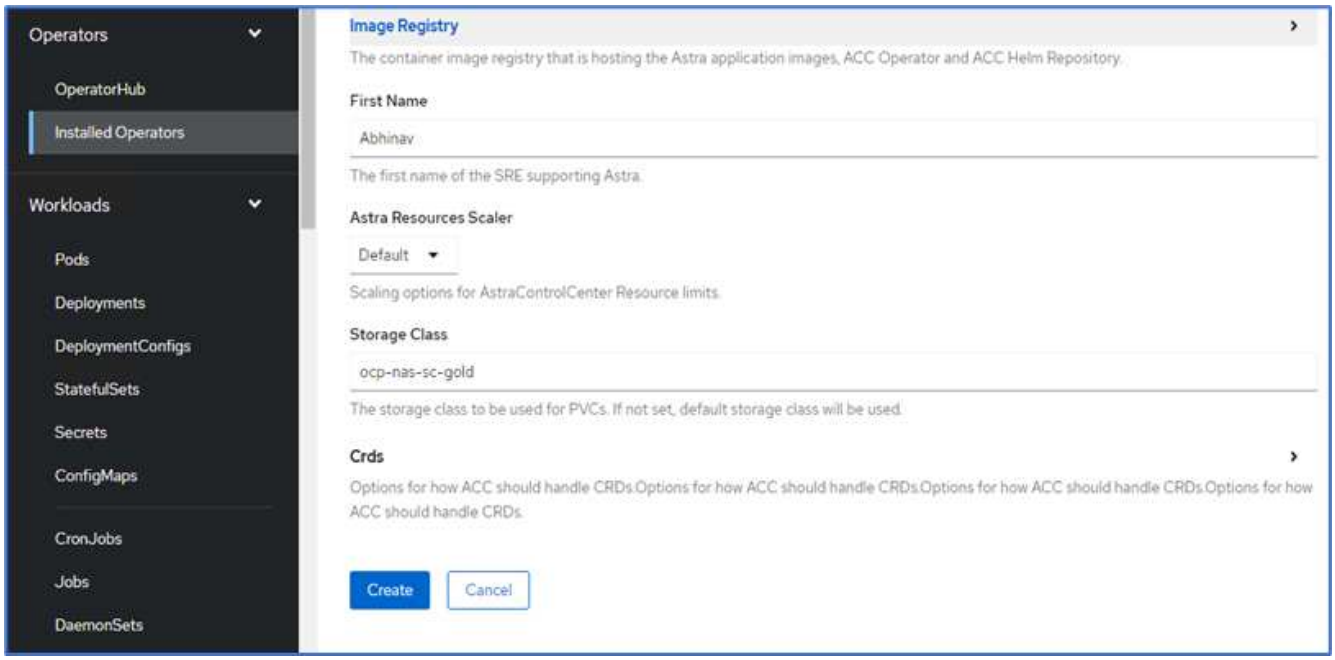
17. 계정 이름, 이메일 주소, 관리자 성을 입력하고 기본 볼륨 복원 정책을 유지합니다. 부하 분산 장치를 사용하는 경우 수신 유형을 "AccTraefik"로 설정합니다. 그렇지 않으면 Ingress.Controller로 Generic을 선택합니다. 이미지

레지스트리에서 컨테이너 이미지 레지스트리 경로 및 암호를 입력합니다.

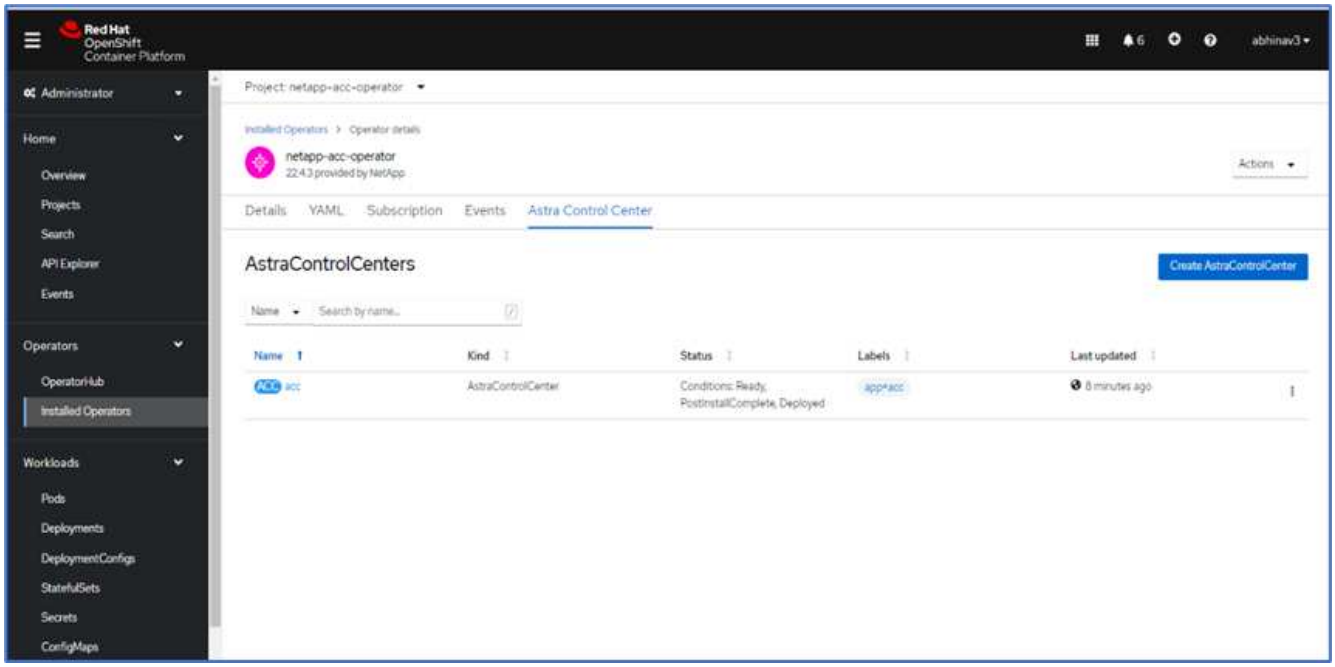


이 솔루션에서는 Metallb 로드 밸런서가 사용됩니다. 따라서 수신 유형은 AccTraefik입니다. 그러면 Astra Control Center traefik 게이트웨이가 워크로드 밸런서의 Kubernetes 서비스로 표시됩니다.

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



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                                                  1/1     Running   0
10m
nats-1                                                  1/1     Running   0
9m43s
nats-2                                                  1/1     Running   0
9m23s
nautilus-7bb469f857-4hlc6                               1/1     Running   0
6m3s
nautilus-7bb469f857-vz94m                               1/1     Running   0
4m42s
openapi-8586db4bcd-gwwvf                                1/1     Running   0
5m41s
packages-6bdb949cfb-nrq8l                               1/1     Running   0
6m35s
polaris-consul-consul-server-0                          1/1     Running   0
9m22s
polaris-consul-consul-server-1                          1/1     Running   0
9m22s
polaris-consul-consul-server-2                          1/1     Running   0
9m22s
polaris-mongodb-0                                       2/2     Running   0
9m22s
polaris-mongodb-1                                       2/2     Running   0
8m58s
polaris-mongodb-2                                       2/2     Running   0
8m34s
polaris-ui-5df7687dbd-trcnf                             1/1     Running   0
3m18s
polaris-vault-0                                         1/1     Running   0
9m18s
polaris-vault-1                                         1/1     Running   0
9m18s
polaris-vault-2                                         1/1     Running   0
9m18s
public-metrics-7b96476f64-j88bw                         1/1     Running   0
5m48s
storage-backend-metrics-5fd6d7cd9c-vcb4j                1/1     Running   0
5m59s
storage-provider-bb85ff965-m7qrq                        1/1     Running   0
5m25s
telegraf-ds-4zqgz                                       1/1     Running   0
3m36s
telegraf-ds-cp9x4                                       1/1     Running   0
3m36s
telegraf-ds-h4n59                                       1/1     Running   0
3m36s
```

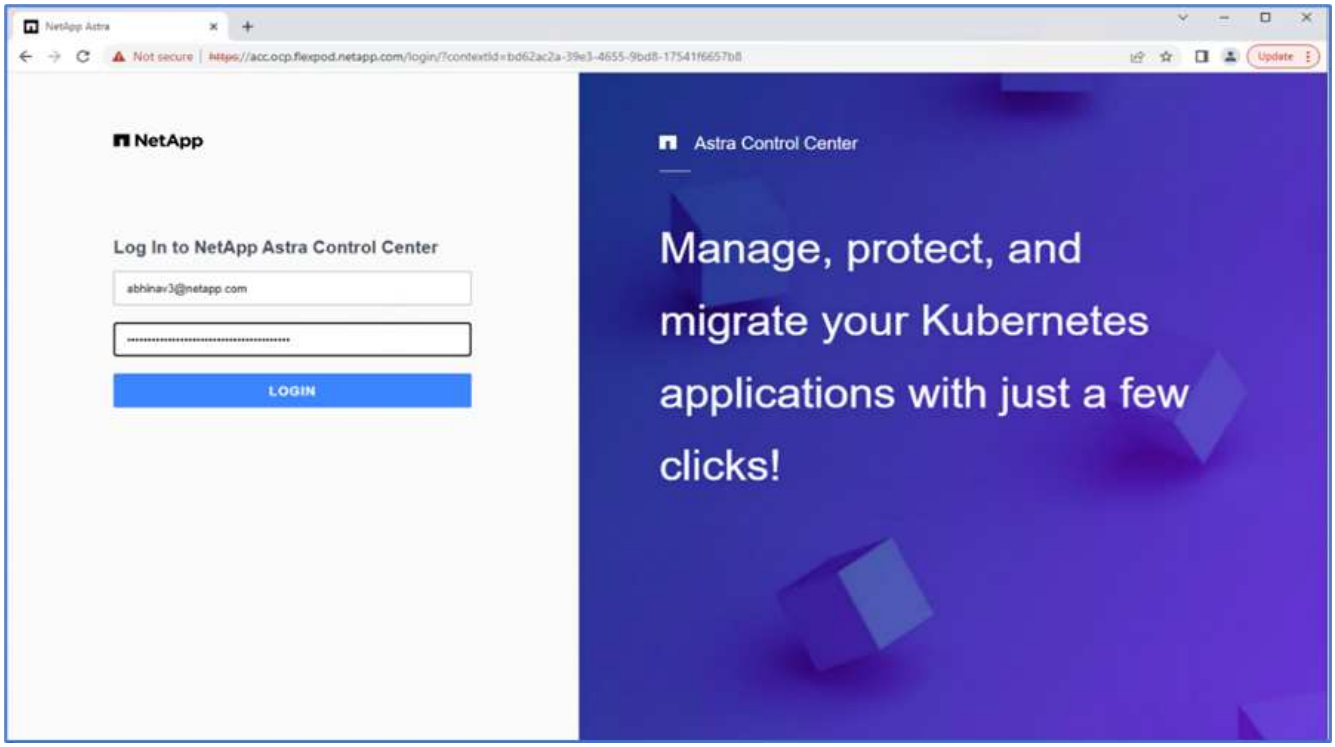
```
telegraf-ds-jnp2q                                    1/1     Running   0
3m36s
telegraf-ds-pdz5j                                    1/1     Running   0
3m36s
telegraf-ds-znqtp                                    1/1     Running   0
3m36s
telegraf-rs-rt64j                                    1/1   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

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]).

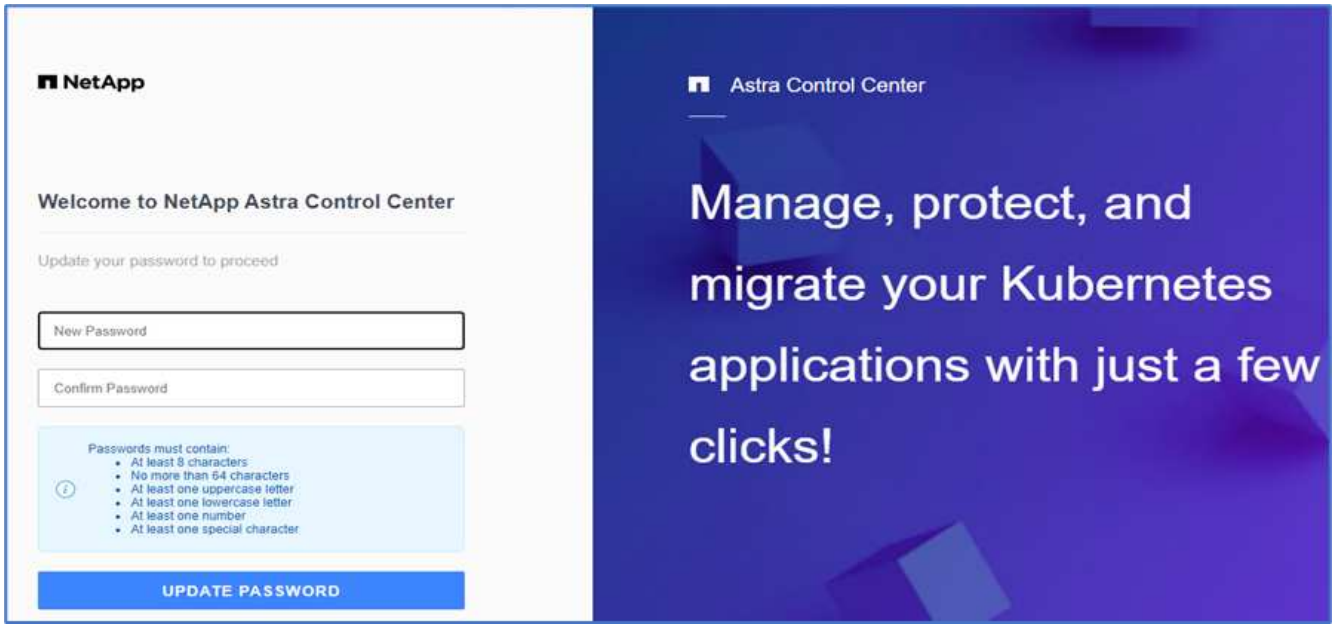
```
root@abhinav-ansible# oc get acc -o yaml -n netapp-acc-operator
```

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.

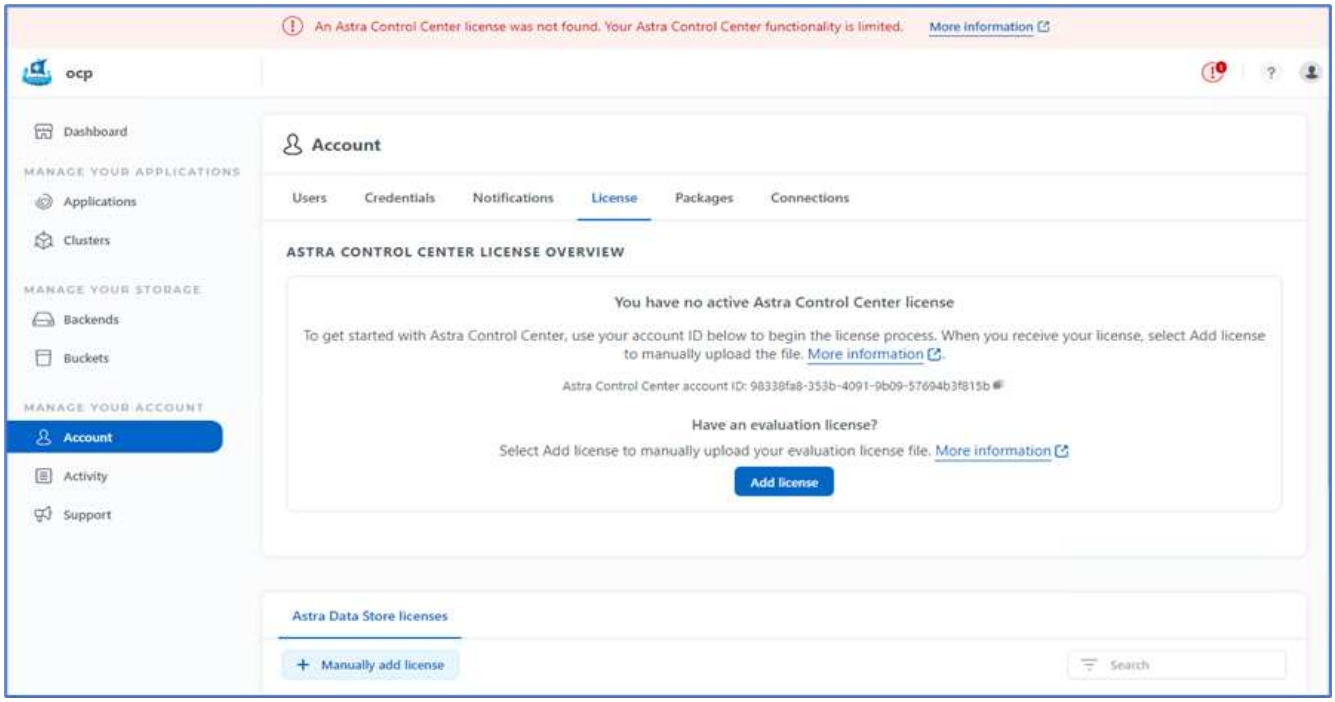


Astra Control Center 설치에 대한 자세한 내용은 를 참조하십시오 "Astra Control Center 설치 개요" 페이지.

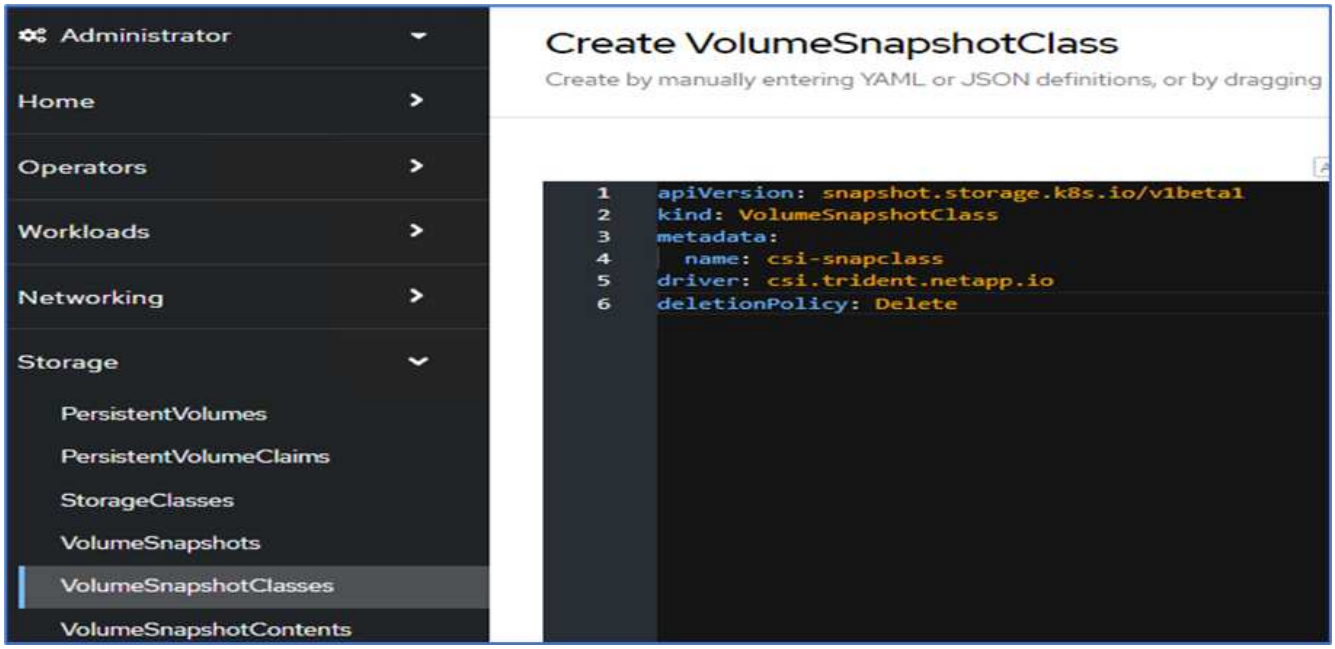
Astra Control Center를 설정합니다

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

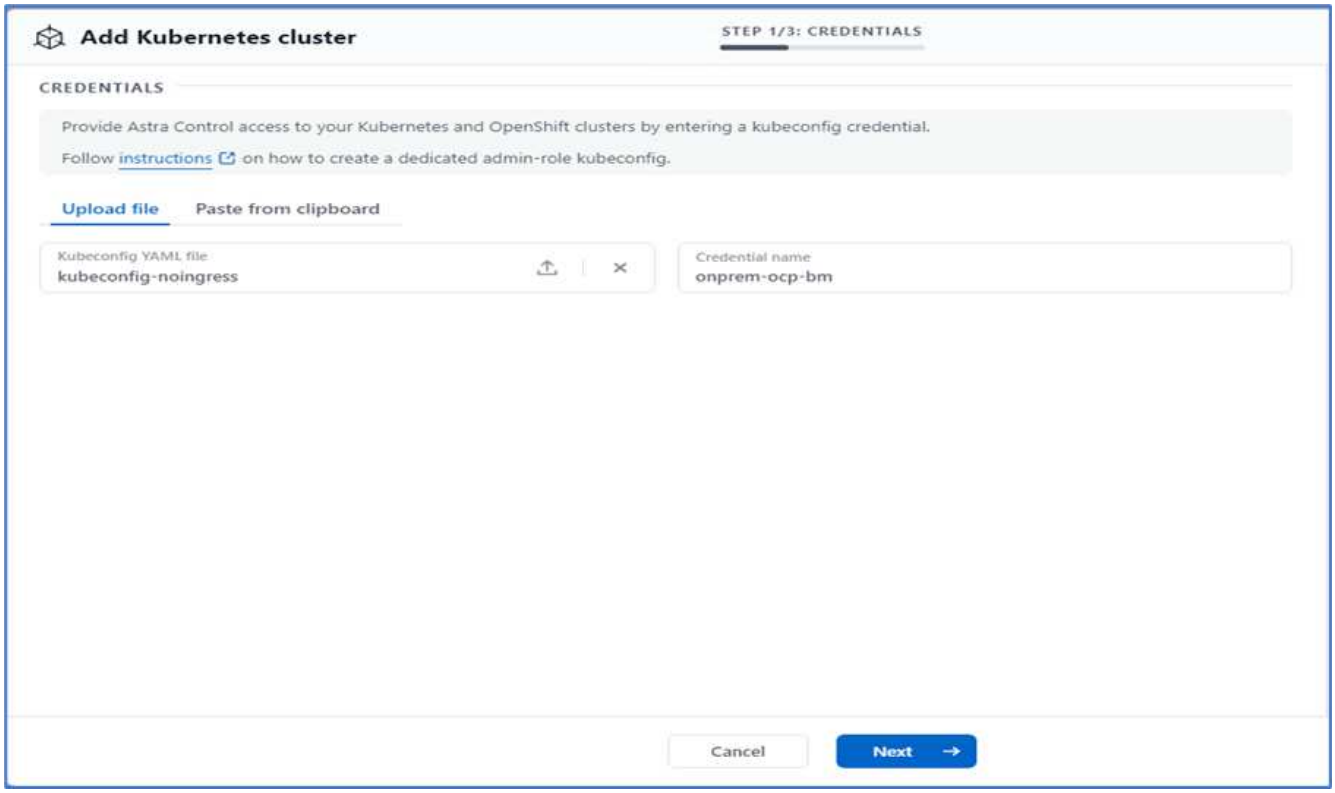
1. 홈 페이지의 계정 에서 라이센스 탭으로 이동하고 라이센스 추가 를 선택하여 Astra 라이센스를 업로드합니다.



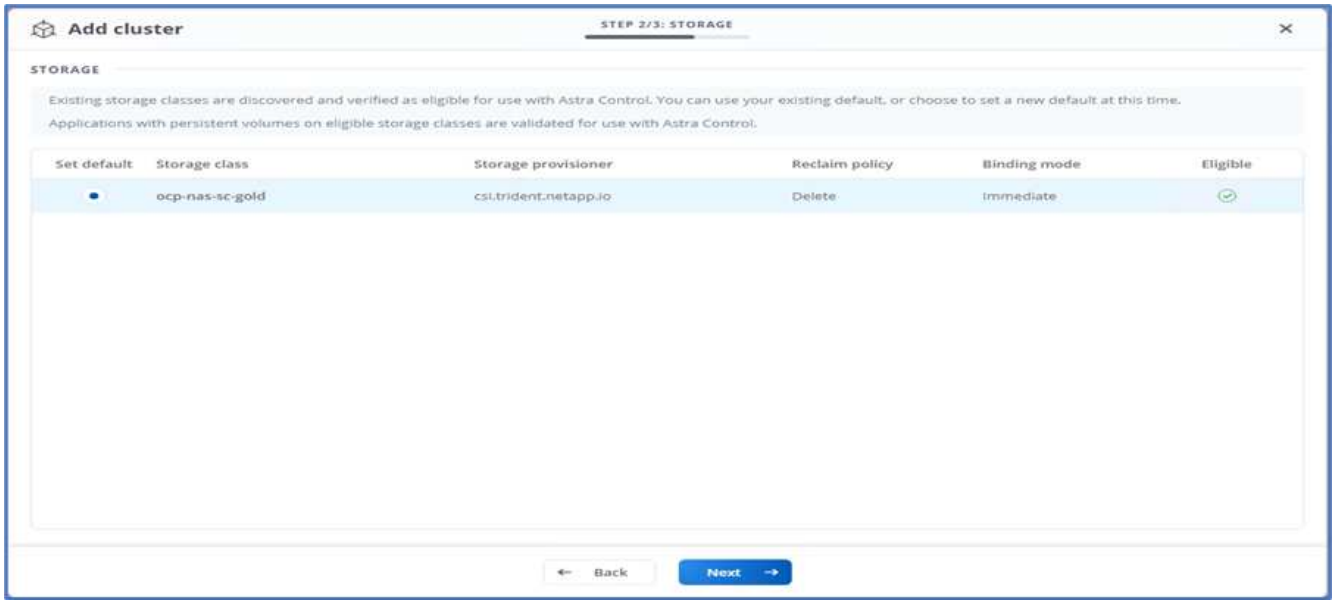
2. OpenShift 클러스터를 추가하기 전에 OpenShift 웹 콘솔에서 Astra Trident Volume 스냅샷 클래스를 생성합니다. 볼륨 스냅샷 클래스는 'csi.trident.netapp.io` 드라이버'로 구성됩니다.



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.



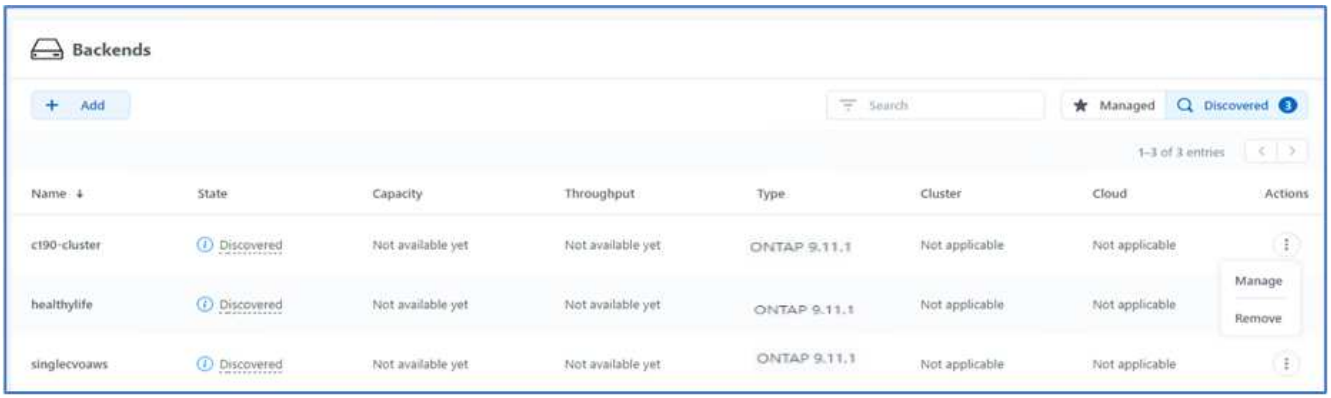
4. 기존 스토리지 클래스가 자동으로 검색됩니다. 기본 스토리지 클래스를 선택하고 다음 을 클릭한 다음 클러스터 추가 를 클릭합니다.



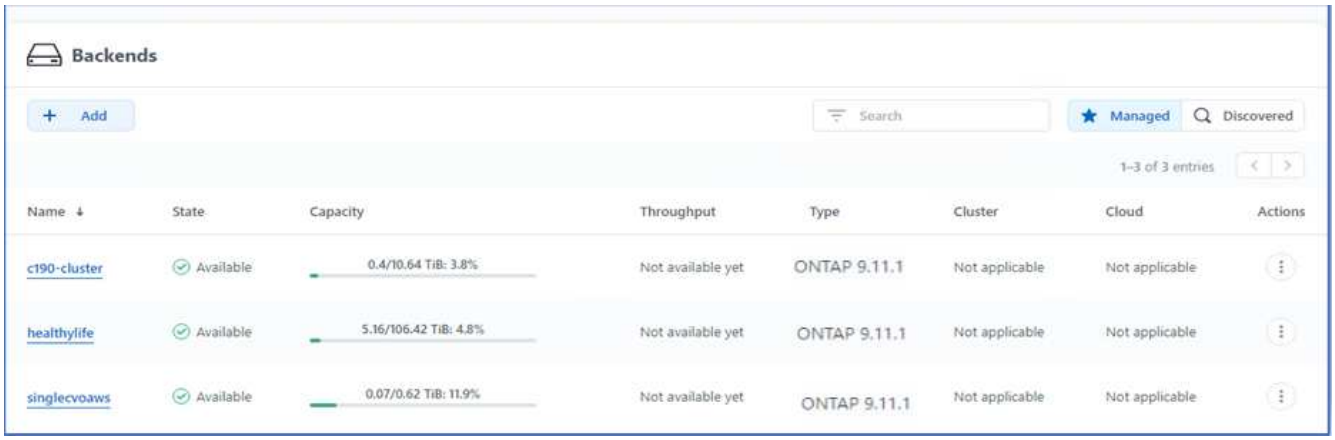
5. 몇 분 내에 클러스터가 추가됩니다. OpenShift Container Platform 클러스터를 추가하려면 1-4단계를 반복합니다.

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

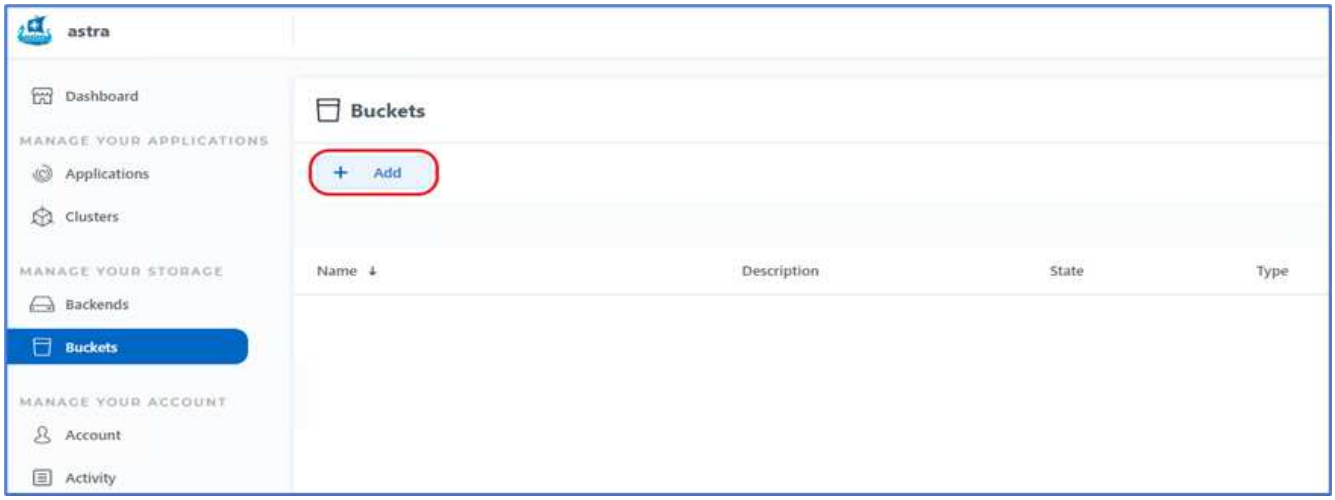
6. 스토리지를 관리하려면 백엔드로 이동하여 관리하려는 백엔드에 대한 작업에서 세 개의 점을 클릭합니다. 관리 를 클릭합니다.



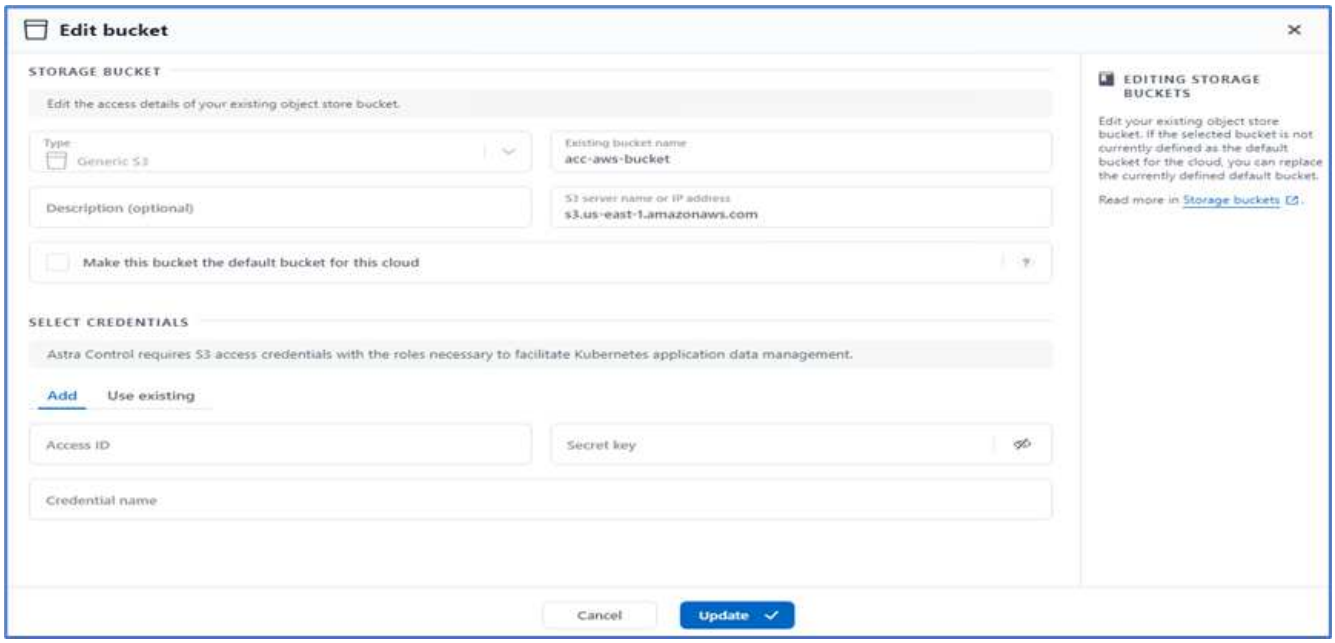
7. ONTAP 자격 증명을 입력하고 Next를 클릭합니다. 정보를 검토하고 관리 를 클릭합니다. 백엔드는 다음 예제와 같아야 합니다.



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

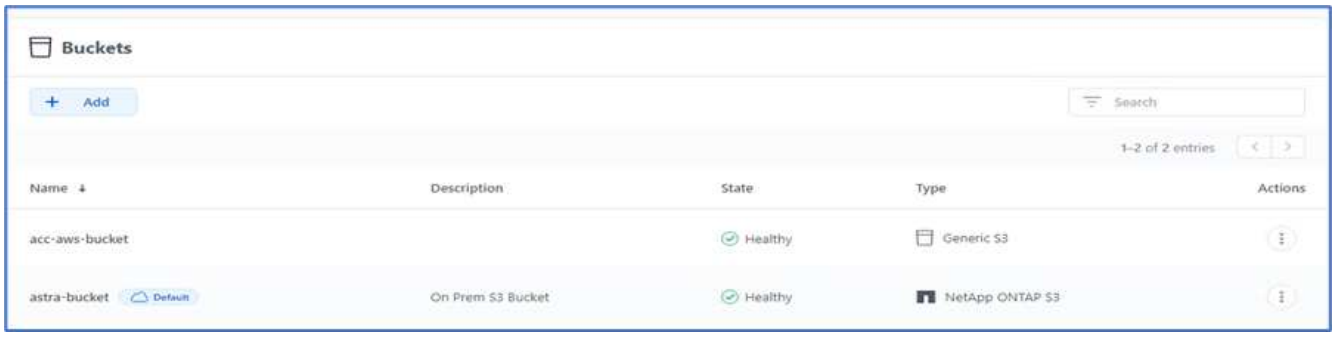


9. 버킷 유형을 선택하고 버킷 이름, S3 서버 이름 또는 IP 주소와 S3 자격 증명을 제공합니다. 업데이트 를 클릭합니다.



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

Bucket 상태는 정상 상태여야 합니다.



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 을 참조하십시오 "설명합니다".

"다음: 솔루션 검증 개요"

솔루션 검증

개요

"이전: OpenShift Container Platform에 Astra Control Center 설치."

이 섹션에서는 몇 가지 사용 사례와 함께 솔루션을 다시 살펴봅니다.

- 원격 백업에서 클라우드 실행 중인 다른 OpenShift 클러스터로 Stateful 애플리케이션을 복원합니다.
- OpenShift 클러스터의 동일한 네임스페이스에 stateful 애플리케이션을 복원합니다.
- 한 FlexPod 시스템(OpenShift Container Platform Bare Metal)에서 다른 FlexPod 시스템(VMware의 OpenShift Container Platform)으로 복제하여 애플리케이션 이동성 보장

그뿐만 아니라 이 솔루션에서 검증된 사용 사례는 몇 가지 뿐입니다. 이 검증은 Astra Control Center의 전체 기능을 나타내는 것은 아닙니다.

"다음: 원격 백업을 사용한 애플리케이션 복구"

원격 백업을 사용한 애플리케이션 복구

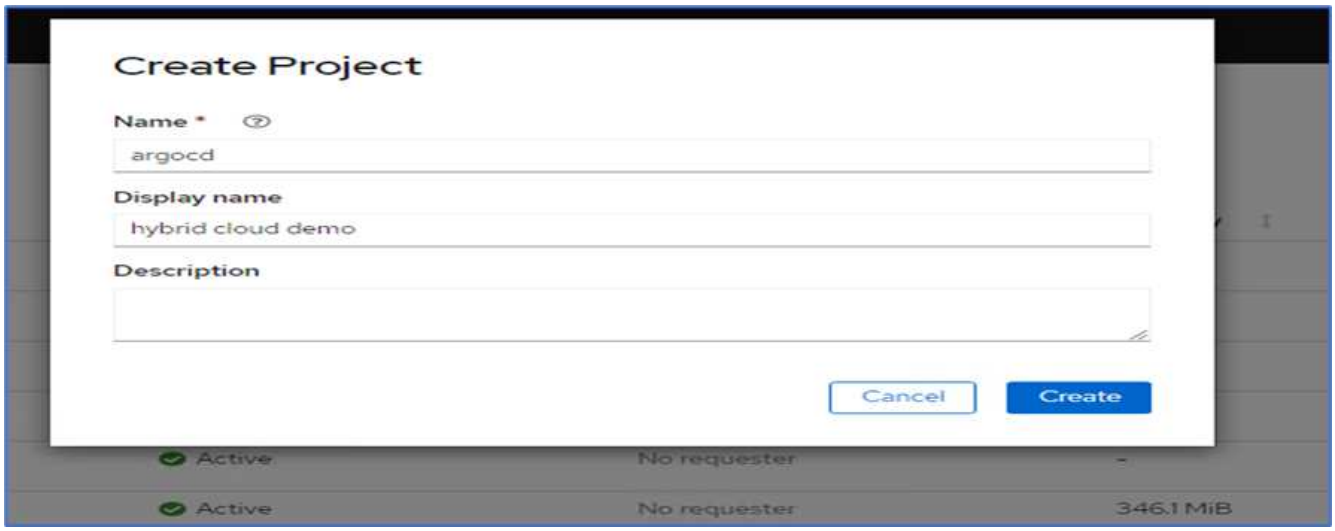
"이전: 솔루션 검증 개요"

Astra를 사용하면 애플리케이션 정합성이 보장된 전체 백업을 수행할 수 있습니다. 이 백업을 사용하여 사내 데이터 센터 또는 퍼블릭 클라우드에서 실행 중인 다른 Kubernetes 클러스터로 데이터를 가진 애플리케이션을 복원할 수 있습니다.

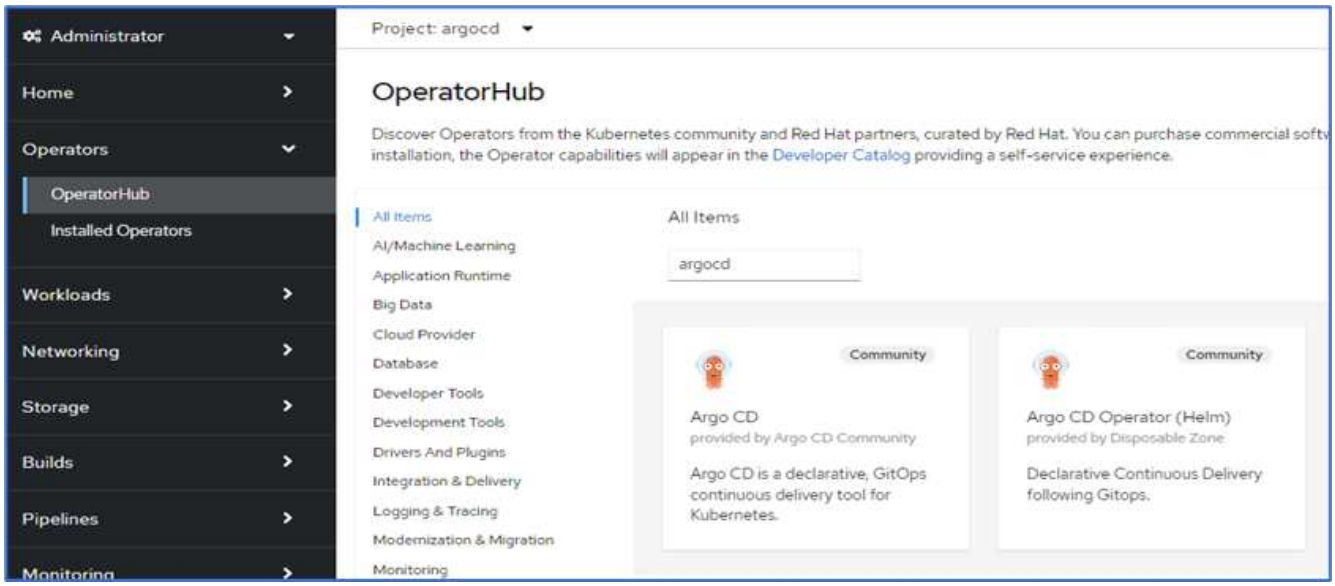
성공적인 애플리케이션 복구를 검증하려면 FlexPod 시스템에서 실행 중인 애플리케이션의 사내 장애를 시뮬레이션하고 원격 백업을 사용하여 클라우드에서 실행 중인 K8s 클러스터로 애플리케이션을 복구합니다.

샘플 응용 프로그램은 데이터베이스에 MySQL을 사용하는 가격 목록 응용 프로그램입니다. 배포를 자동화하기 위해 을 사용했습니다 "argo CD" 도구. argo CD는 선언적 GitOps, 지속적인 Kubernetes 제공 툴입니다.

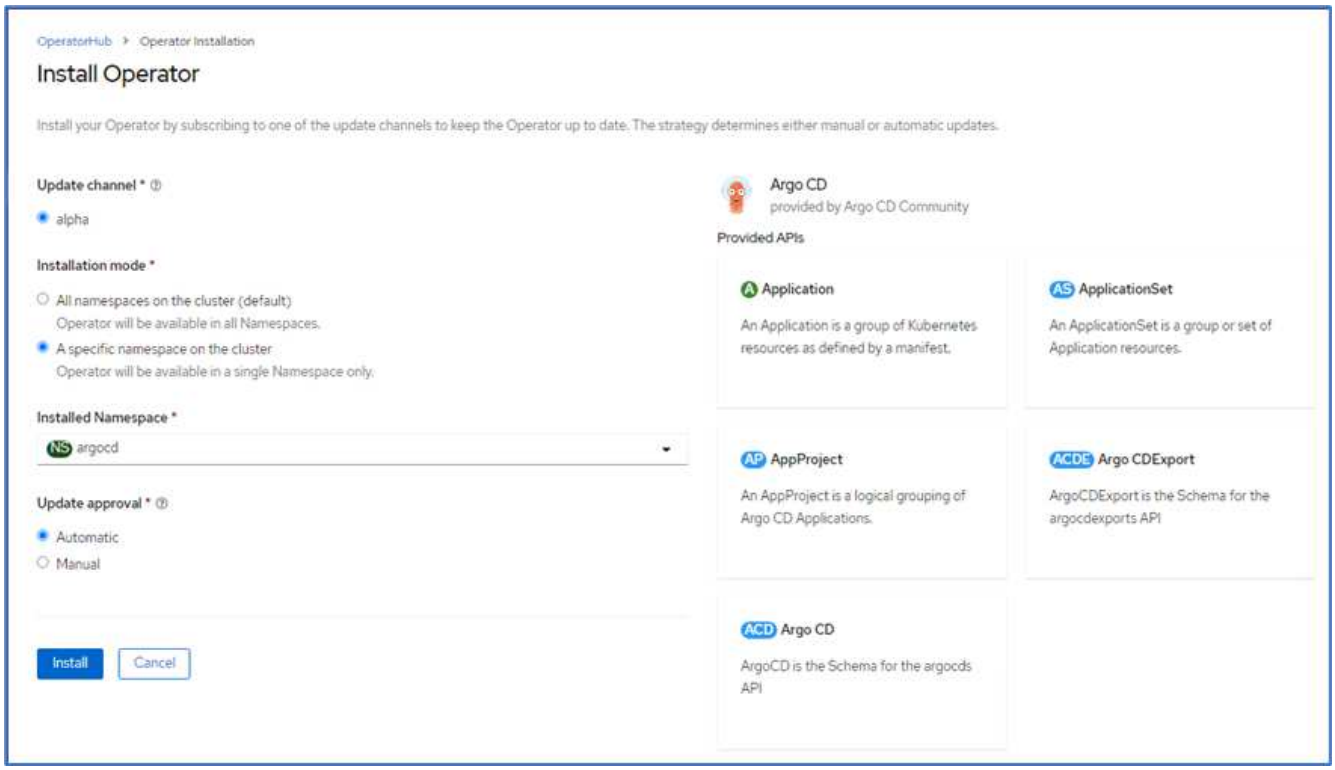
1. 온-프레미스 OpenShift 클러스터에 로그인하여 이름이 argocd인 새 프로젝트를 만듭니다.



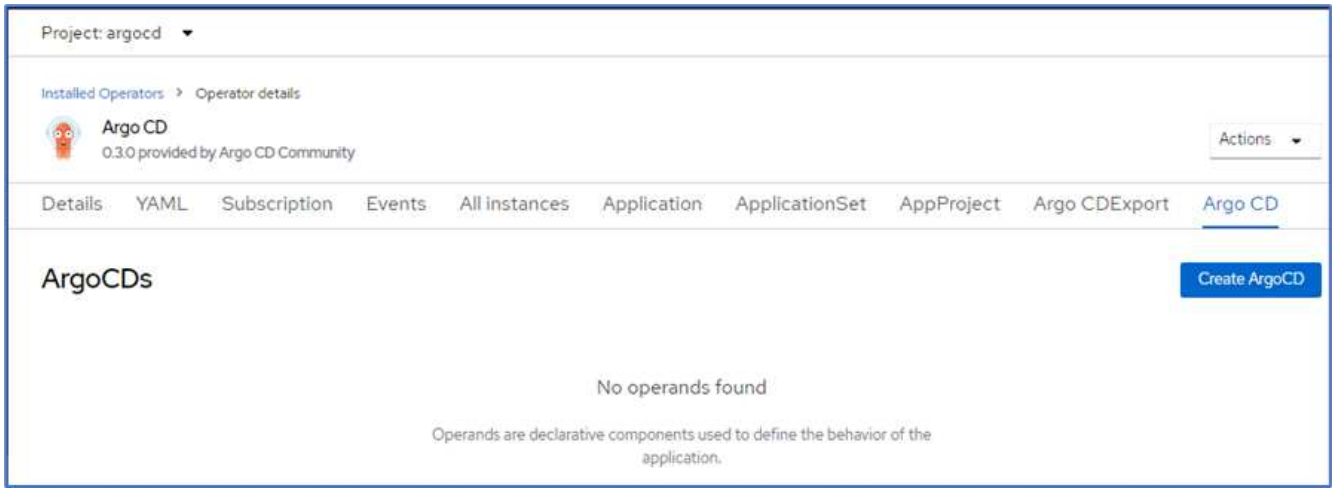
2. OperatorHub에서 argocd를 검색하고 Argo CD 연산자를 선택합니다.



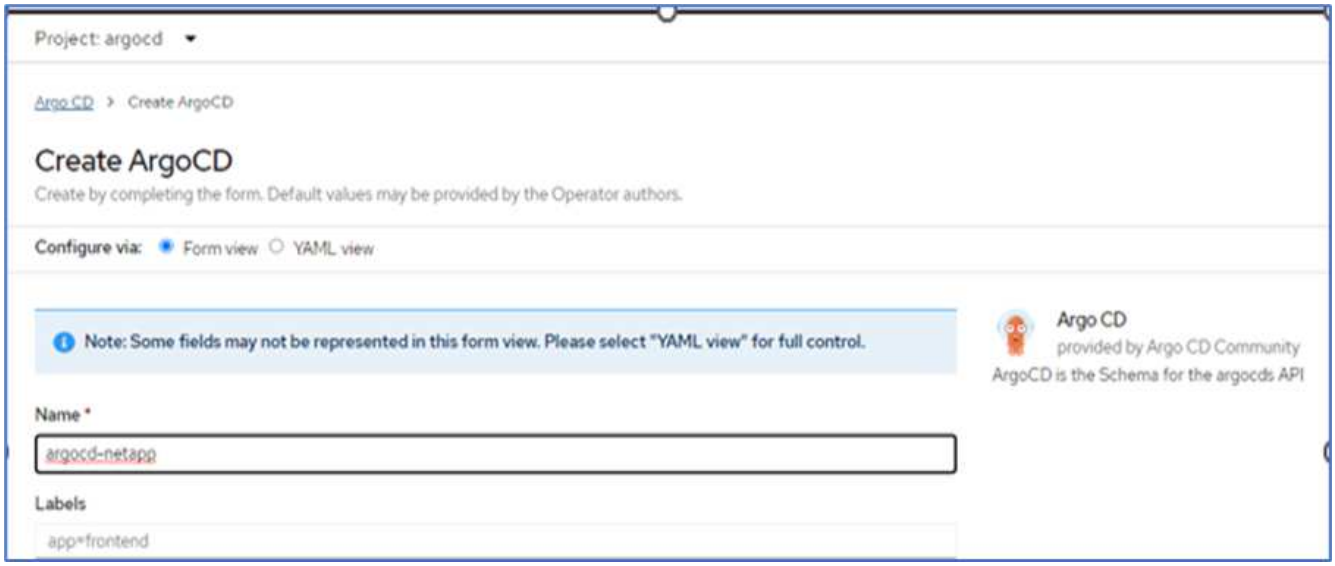
3. 'argocd' 네임스페이스에 연산자를 설치합니다.



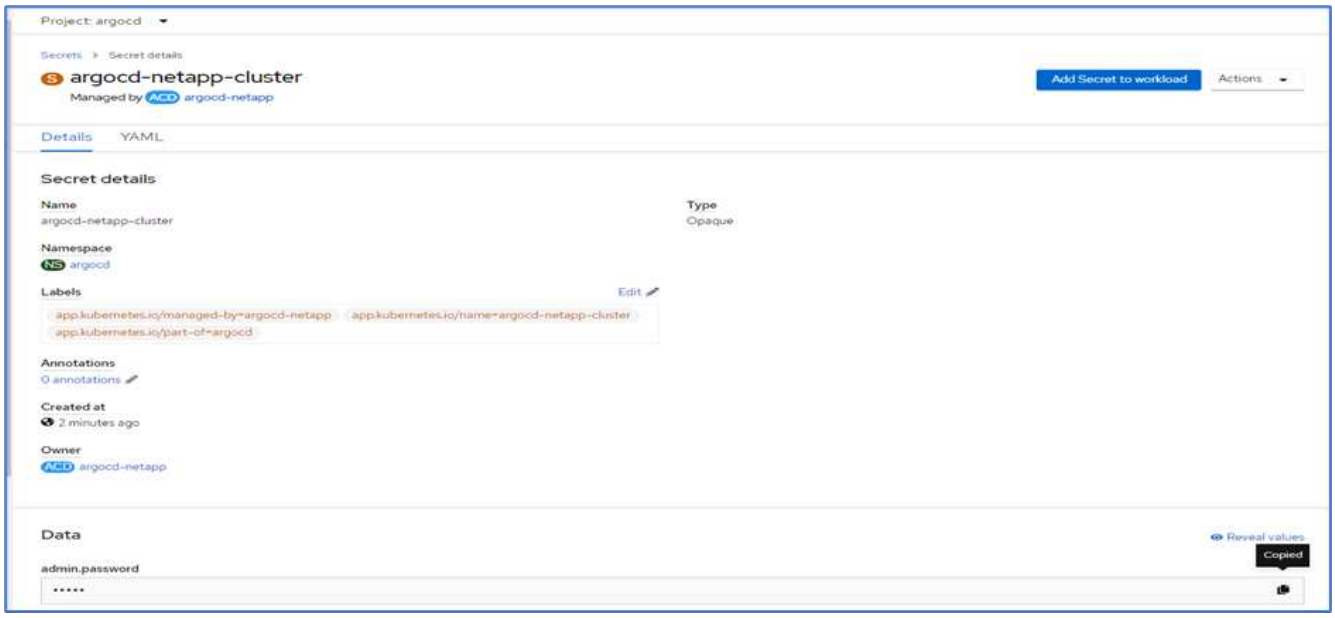
4. 운영자로 이동하여 ArgoCD 생성 을 클릭합니다.



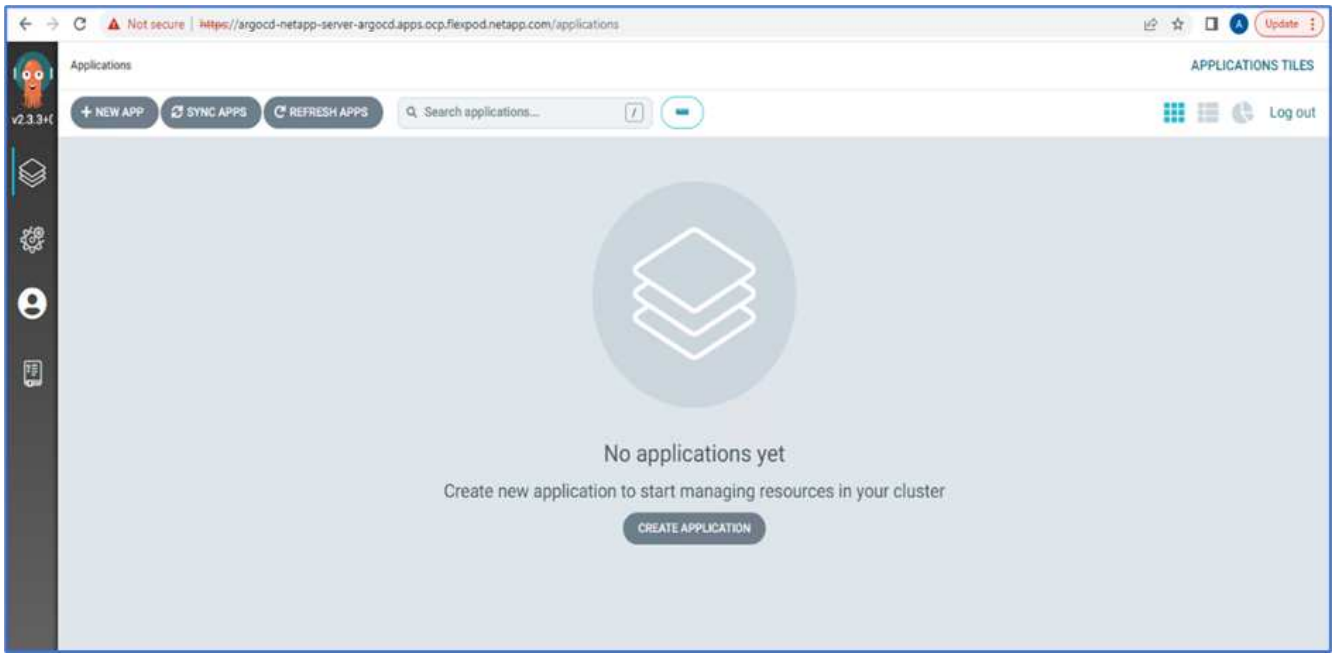
5. 아르고 CD 인스턴스를 "argocd" 프로젝트에 배포하려면 이름을 입력하고 만들기 를 클릭합니다.



6. Argo CD에 로그인하려면 기본 사용자는 admin이고 암호는 "argocd-NetApp-cluster"라는 이름의 비밀 파일에 있어야 합니다.



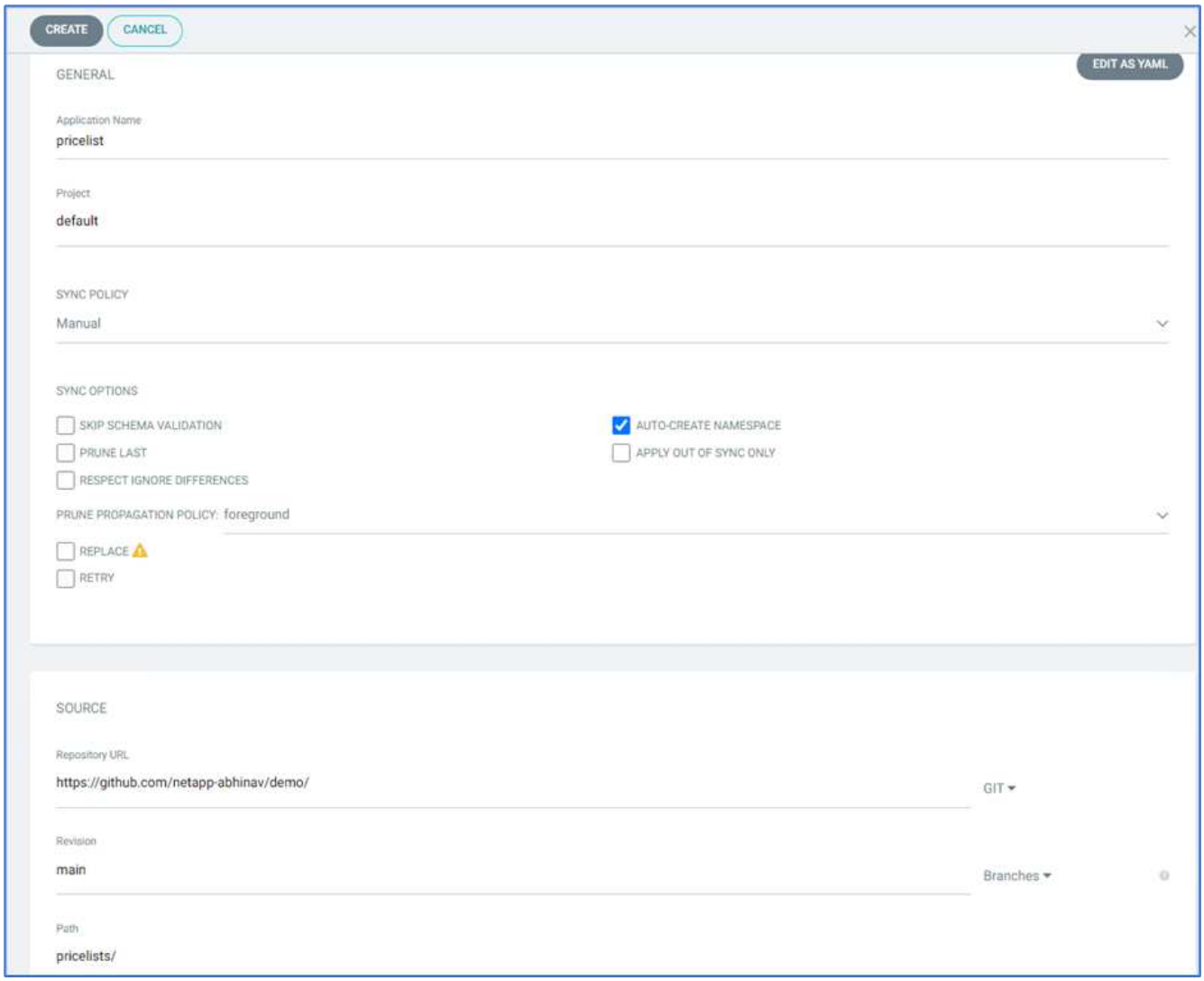
7. 측면 메뉴에서 루트 > 위치를 선택하고 "argocd" 루트의 URL을 클릭합니다. 사용자 이름과 암호를 입력합니다.



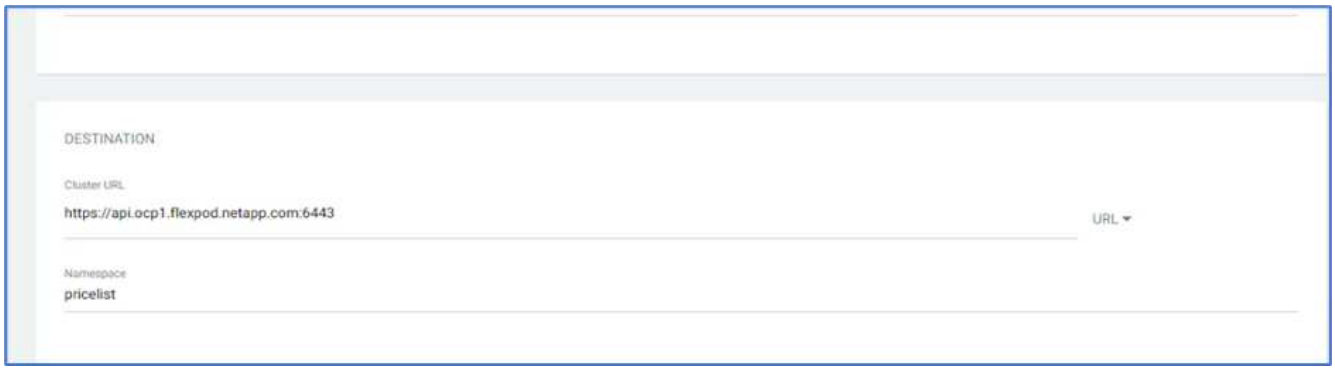
8. CLI를 통해 Argo CD에 온-프레미스 OpenShift 클러스터를 추가합니다.

```
####Login to Argo CD####
abhinav3@abhinav-ansible$ argocd-linux-amd64 login argocd-netapp-server-
argocd.apps.ocp.flexpod.netapp.com --insecure
Username: admin
Password:
'admin:login' logged in successfully
Context'argocd-netapp-server-argocd.apps.ocp.flexpod.netapp.com' updated
####List the On-Premises OpenShift cluster####
abhinav3@abhinav-ansible$ argocd-linux-amd64 cluster add
ERRO[0000] Choose a context name from:
CURRENT  NAME
CLUSTER              SERVER
*        default/api-ocp-flexpod-netapp-com:6443/abhinav3
api-ocp-flexpod-netapp-com:6443
https://api.ocp.flexpod.netapp.com:6443
         default/api-ocp1-flexpod-netapp-com:6443/abhinav3
api-ocp1-flexpod-netapp-com:6443
https://api.ocp1.flexpod.netapp.com:6443
####Add On-Premises OpenShift cluster###
abhinav3@abhinav-ansible$ argocd-linux-amd64 cluster add default/api-
ocp1-flexpod-netapp-com:6443/abhinav3
WARNING: This will create a service account `argocd-manager` on the
cluster referenced by context `default/api-ocp1-flexpod-netapp-
com:6443/abhinav3` with full cluster level admin privileges. Do you want
to continue [y/N]? y
INFO[0002] ServiceAccount "argocd-manager" already exists in namespace
"kube-system"
INFO[0002] ClusterRole "argocd-manager-role" updated
INFO[0002] ClusterRoleBinding "argocd-manager-role-binding" updated
Cluster 'https://api.ocp1.flexpod.netapp.com:6443' added
```

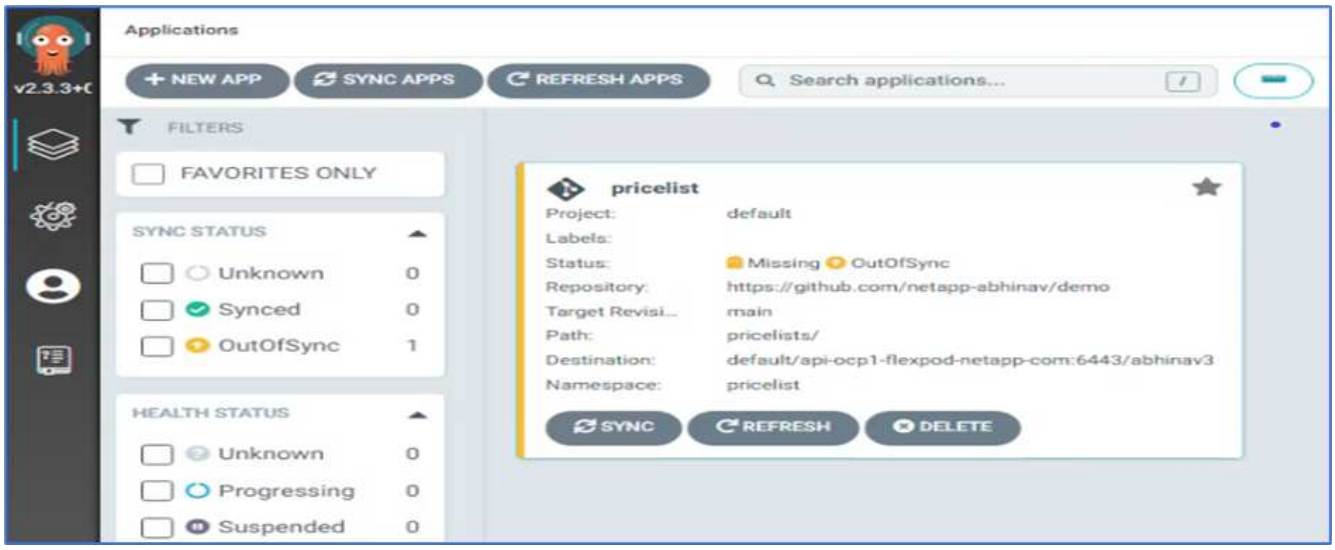
9. ArgoCD UI에서 새 앱 을 클릭하고 앱 이름 및 코드 리포지토리에 대한 세부 정보를 입력합니다.



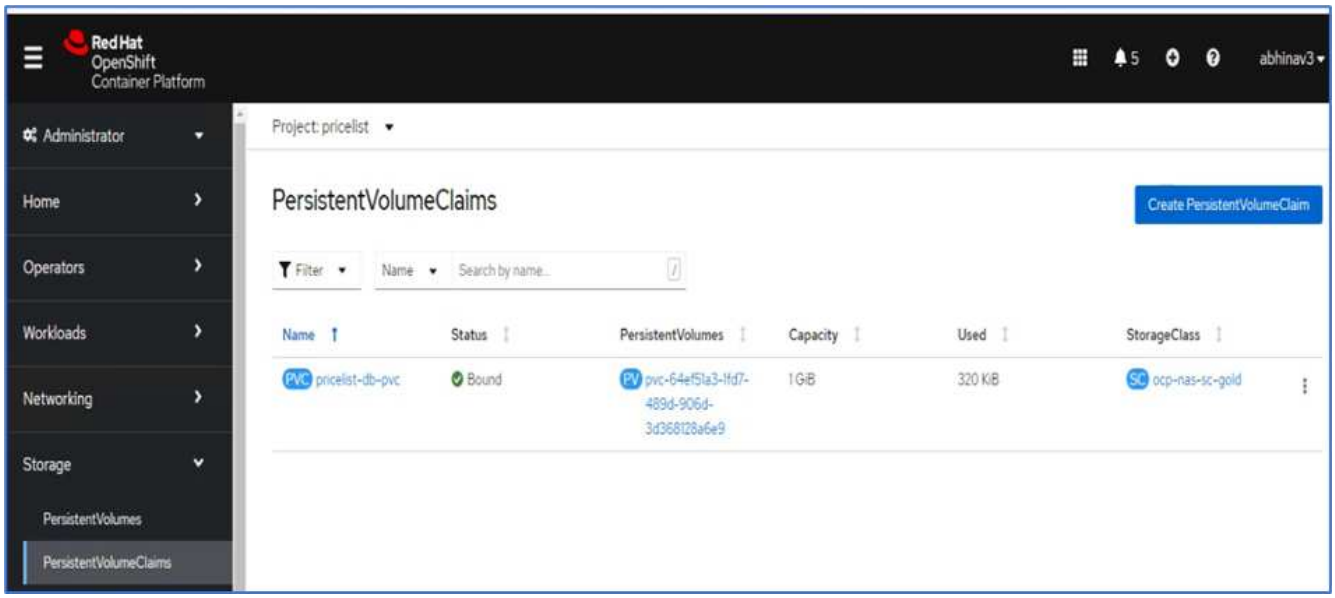
10. 네임스페이스와 함께 앱을 배포할 OpenShift 클러스터를 입력합니다.



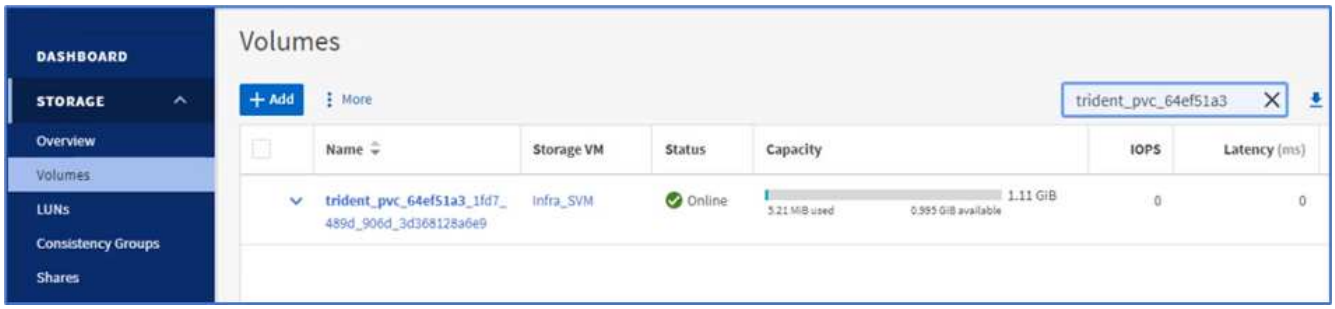
11. 온-프레미스 OpenShift 클러스터에 앱을 배포하려면 동기화 를 클릭합니다.



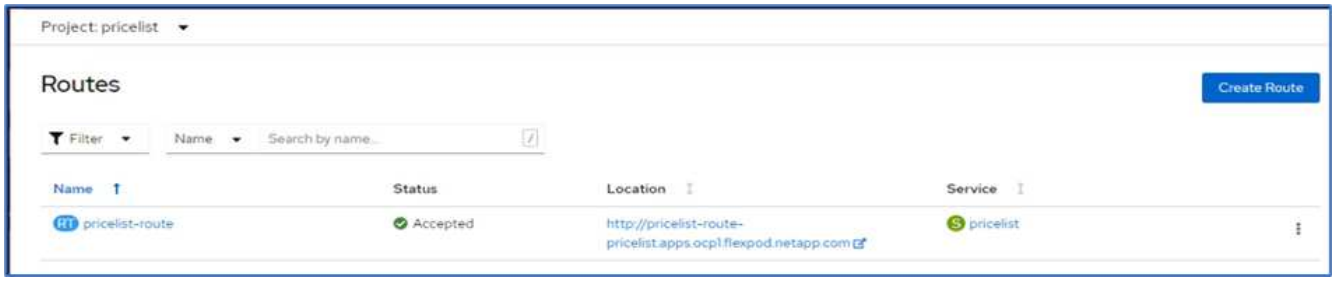
12. OpenShift Container Platform 콘솔에서 Project Pricelist로 이동한 후 Storage에서 PVC의 이름과 크기를 확인합니다.



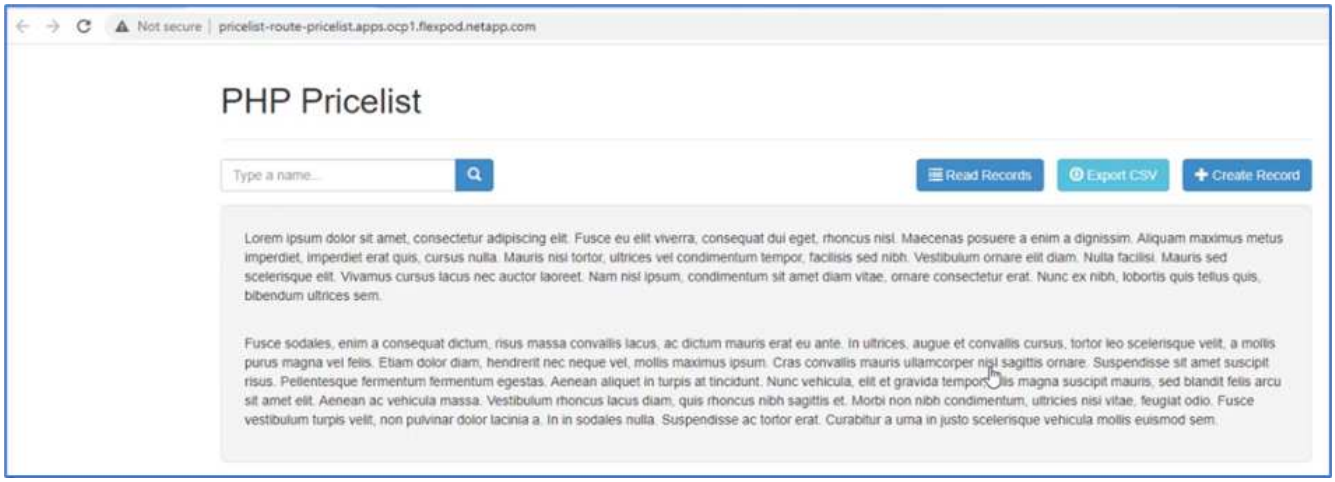
13. System Manager에 로그인하여 PVC를 확인합니다.



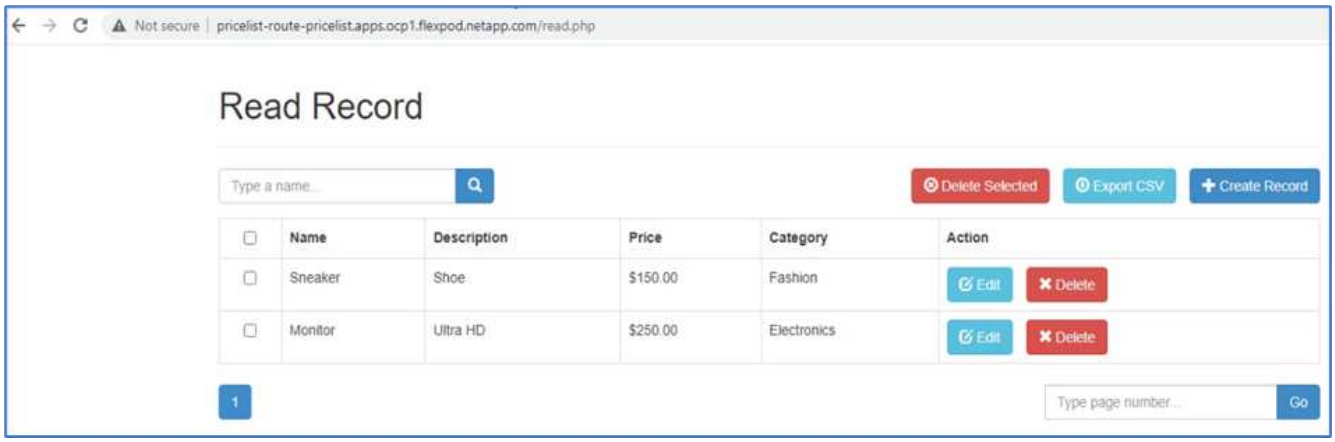
14. Pod가 실행된 후 측면 메뉴에서 네트워킹 > 경로를 선택하고 위치 에서 URL을 클릭합니다.



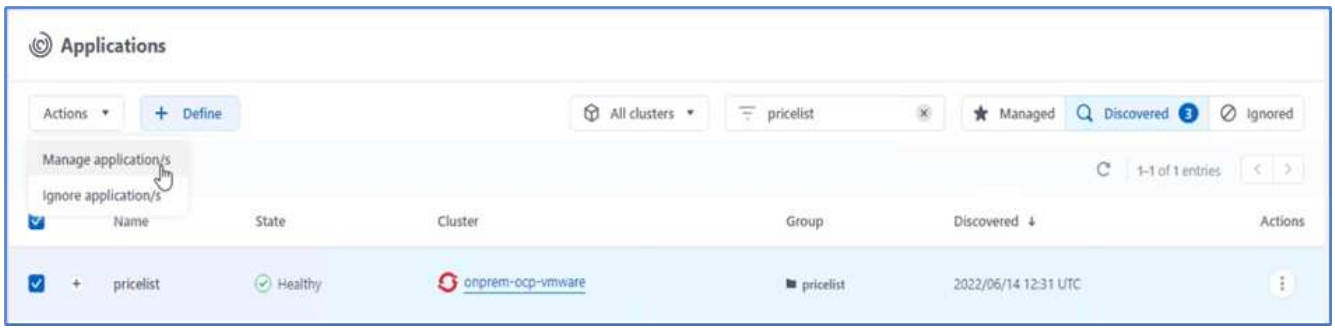
15. 가격 목록 앱 홈 페이지가 표시됩니다.



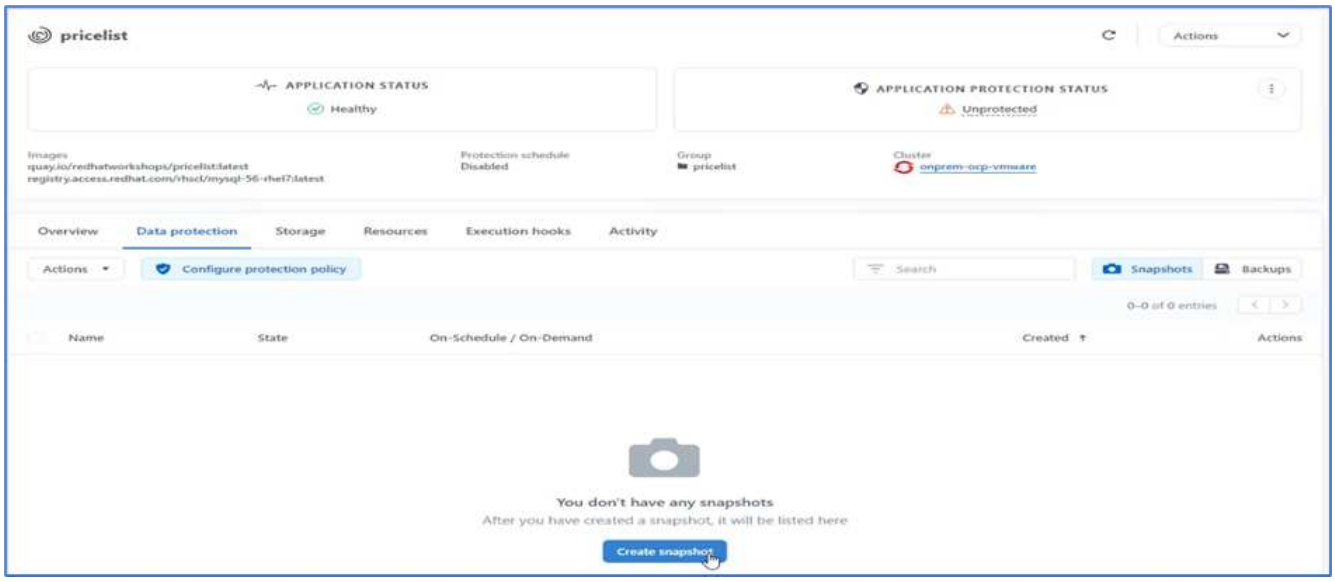
16. 웹 페이지에서 몇 개의 레코드를 만듭니다.



17. 이 앱은 Astra Control Center에서 발견됩니다. 앱을 관리하려면 응용 프로그램 > 검색으로 이동하여 가격 목록 앱을 선택하고 작업 에서 응용 프로그램 관리 를 클릭합니다.

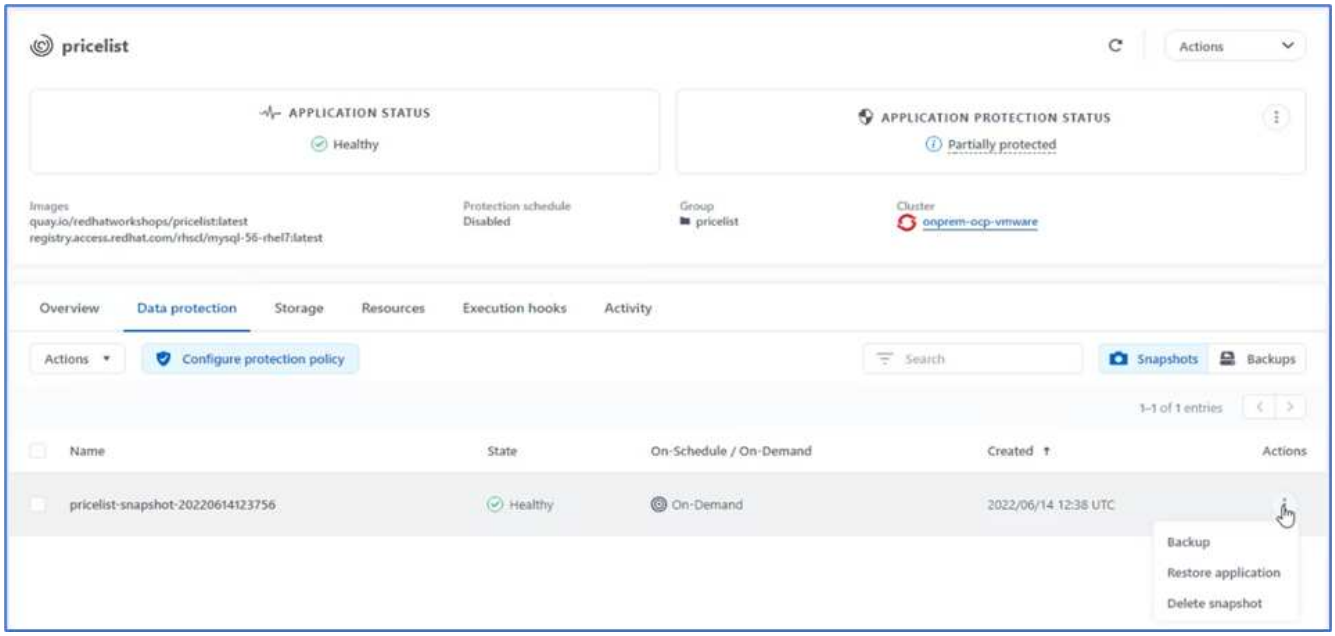


18. 가격 목록 앱을 클릭하고 데이터 보호 를 선택합니다. 이때 스냅샷 또는 백업이 없어야 합니다. 스냅샷 생성 을 클릭하여 필요 시 스냅샷을 생성합니다.

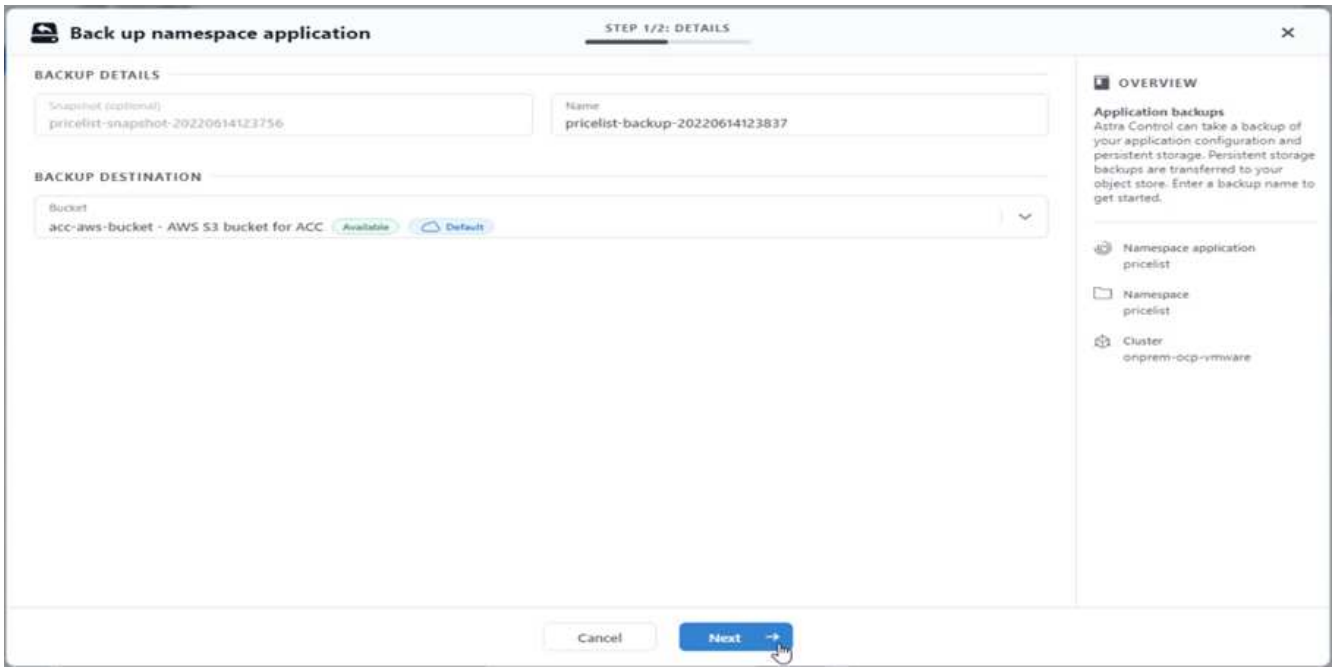


NetApp Astra Control Center는 주문형 스냅샷과 예약된 스냅샷 및 백업을 모두 지원합니다.

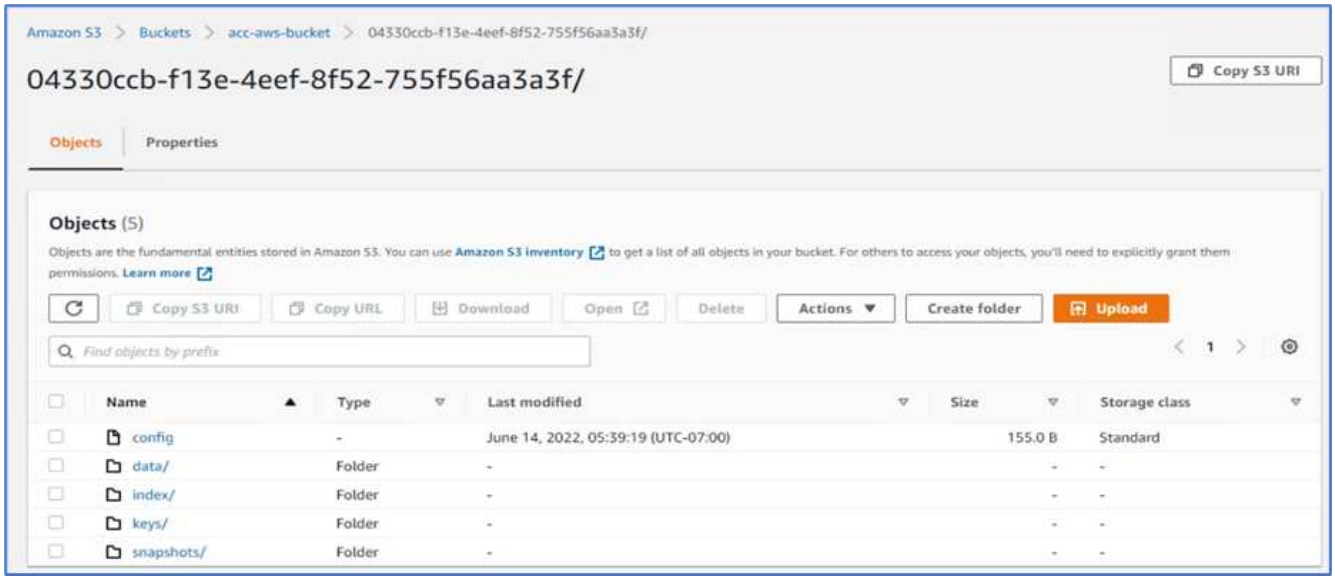
19. 스냅샷이 생성되고 상태가 양호하면 해당 스냅샷을 사용하여 원격 백업을 생성합니다. 이 백업은 S3 버킷에 저장됩니다.



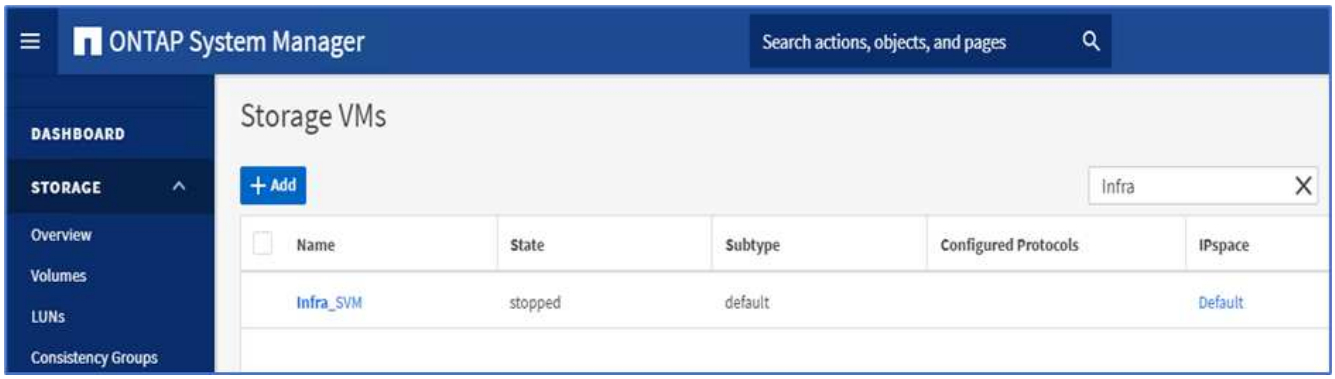
20. AWS S3 버킷을 선택하고 백업 작업을 시작합니다.



21. 백업 작업에서는 AWS S3 버킷에 여러 오브젝트가 포함된 폴더를 생성해야 합니다.



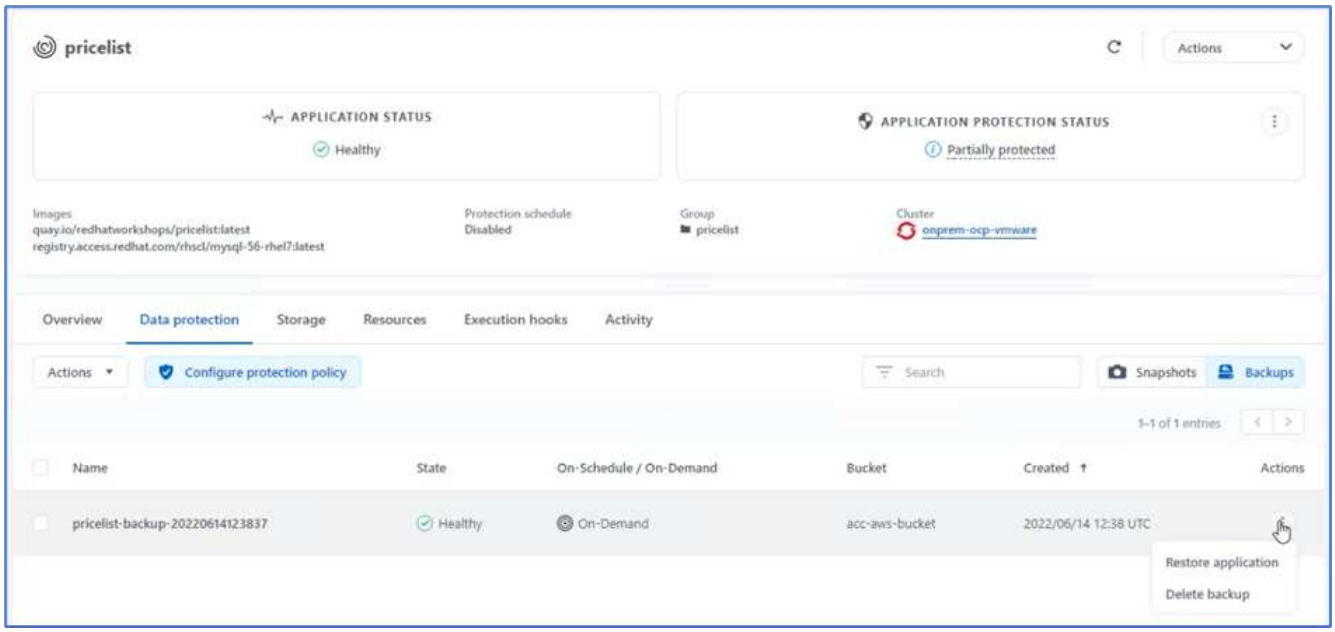
22. 원격 백업이 완료되면 PV의 백업 볼륨을 호스팅하는 SVM(스토리지 가상 머신)을 중지하여 온프레미스 재해 상황을 시뮬레이션합니다.



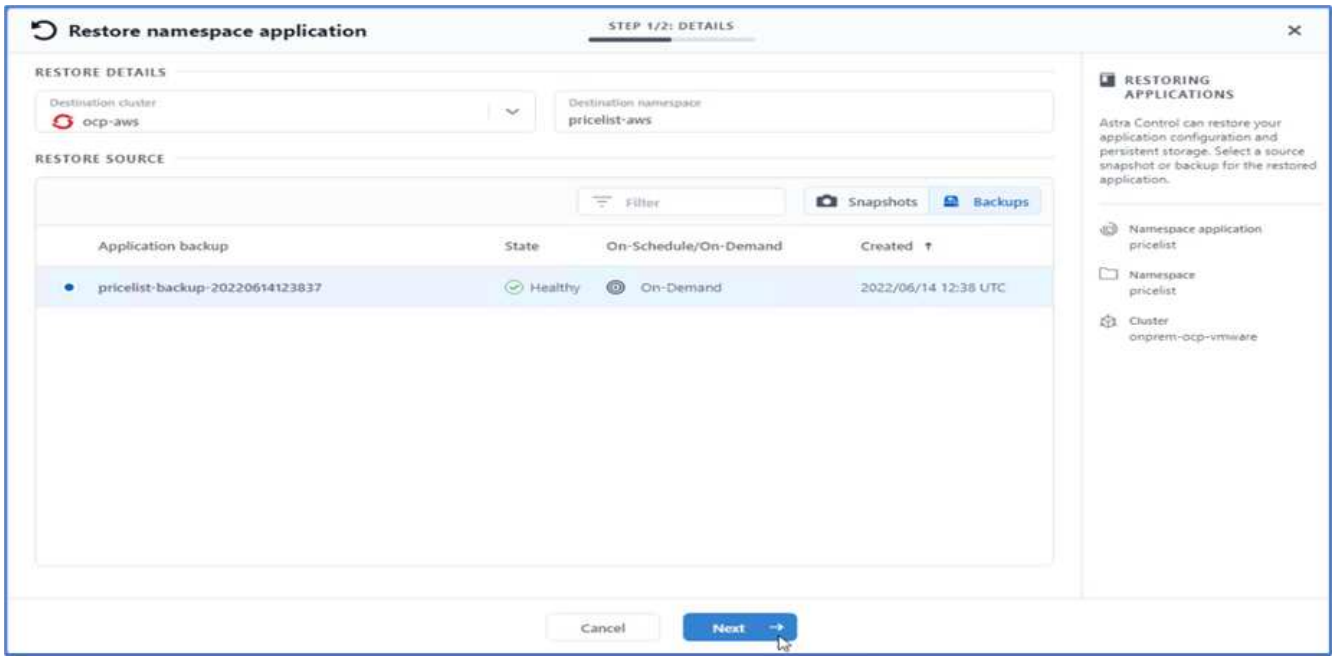
23. 웹 페이지를 새로 고쳐 중단을 확인합니다. 웹 페이지를 사용할 수 없습니다.

예상대로 웹 사이트가 다운되었습니다. 이제 Astra를 사용하여 AWS에서 실행되는 OpenShift 클러스터로 원격 백업에서 앱을 빠르게 복구하겠습니다.

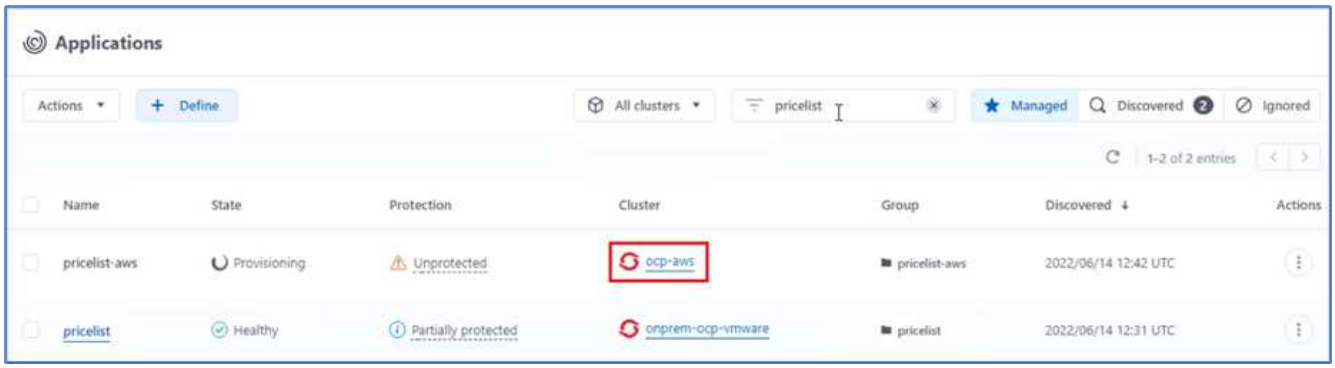
24. Astra Control Center에서 Pricelist 앱을 클릭하고 Data Protection > Backups를 선택합니다. 백업을 선택하고 작업 에서 응용 프로그램 복원 을 클릭합니다.



25. 대상 클러스터로 OCP-AWS를 선택하고 이름 공간에 이름을 지정합니다. 필요 시 백업, 다음, 복원을 차례로 클릭합니다.

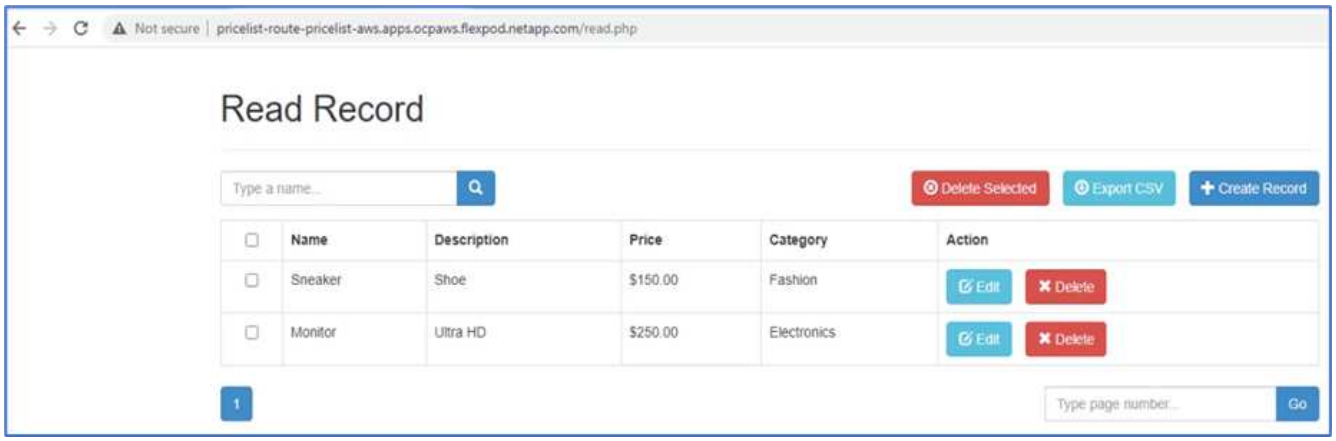


26. AWS에서 실행되는 OpenShift 클러스터에서 'pricelist-app'이라는 이름의 새 앱이 표시됩니다.



27. OpenShift 웹 콘솔에서도 동일하게 확인합니다.

28. Pricelist-AWS 프로젝트 아래의 모든 포드가 실행 중인 후 루트 로 이동한 다음 URL을 클릭하여 웹 페이지를 시작합니다.



이 프로세스는 Pricelist 애플리케이션이 성공적으로 복원되었으며 Astra Control Center의 도움을 받아 AWS에서 원활하게 실행되는 OpenShift 클러스터에서 데이터 무결성이 유지되었는지 확인합니다.

DevTest를 위한 스냅샷 복사본 및 애플리케이션 이동성으로 데이터 보호

이 사용 사례는 다음 섹션에 설명된 대로 두 부분으로 구성됩니다.

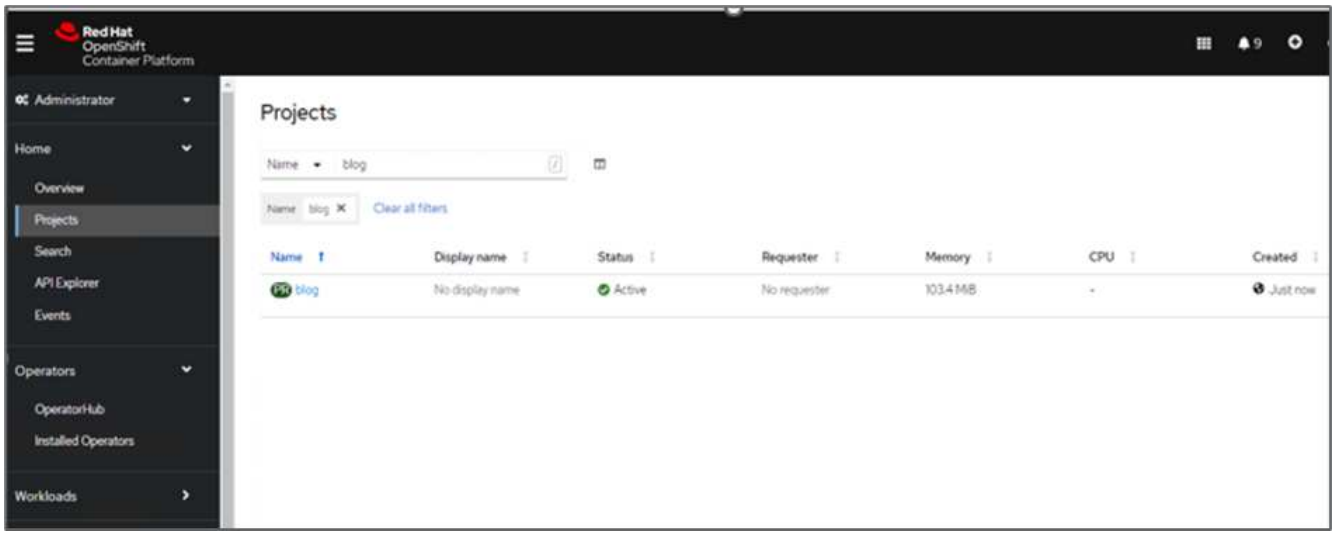
파트 **1**

Astra Control Center를 사용하면 로컬 데이터 보호를 위해 애플리케이션 인식 스냅샷을 생성할 수 있습니다. 실수로 데이터를 삭제하거나 손상된 경우 이전에 기록된 스냅샷을 사용하여 응용 프로그램 및 관련 데이터를 알려진 양호한 상태로 되돌릴 수 있습니다.

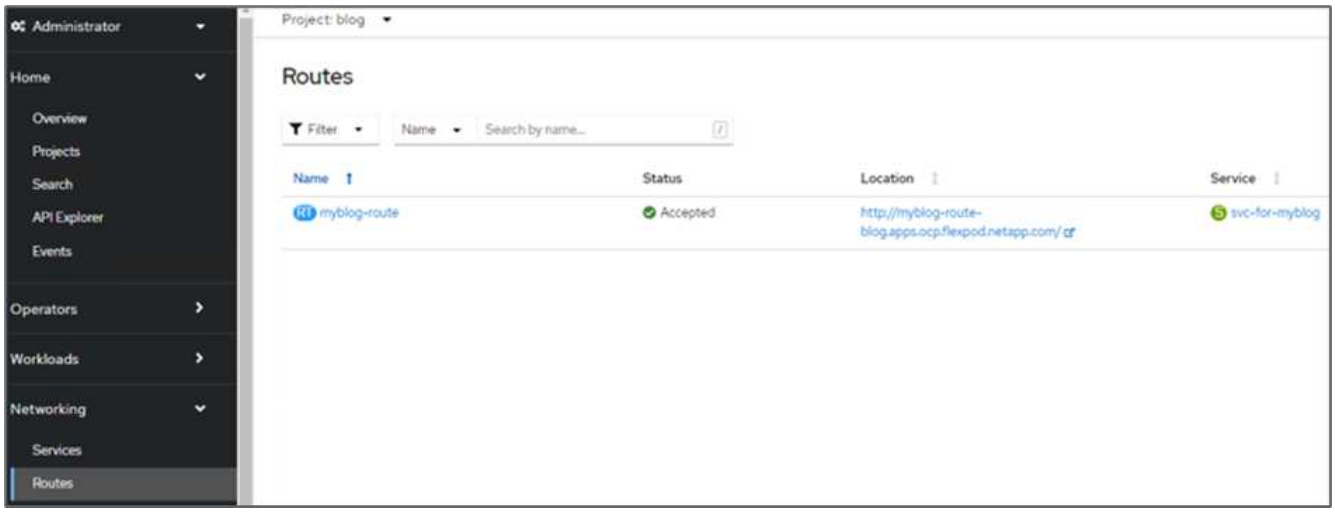
이 시나리오에서 개발 및 테스트(DevTest) 팀은 Ghost 블로그 응용 프로그램인 샘플 상태 저장 응용 프로그램(블로그 사이트)을 배포하고 일부 콘텐츠를 추가하며 앱을 사용 가능한 최신 버전으로 업그레이드합니다. Ghost 응용 프로그램은 SQLite를 데이터베이스에 사용합니다. 애플리케이션을 업그레이드하기 전에 Astra Control Center를 사용하여 데이터 보호를 위한 스냅샷(필요 시)을 생성합니다. 자세한 단계는 다음과 같습니다.

1. 샘플 블로깅 앱을 배포하고 ArgoCD에서 동기화합니다.

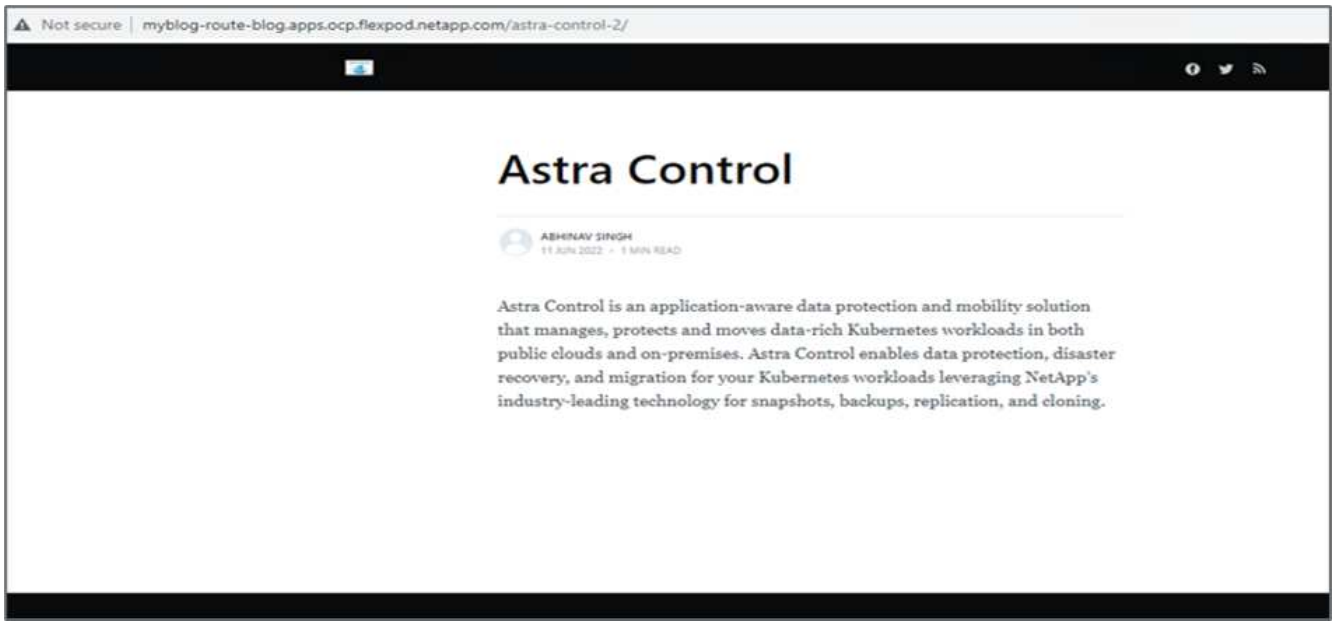
2. 첫 번째 OpenShift 클러스터에 로그인한 후 Project로 이동하여 검색 표시줄에 Blog를 입력합니다.



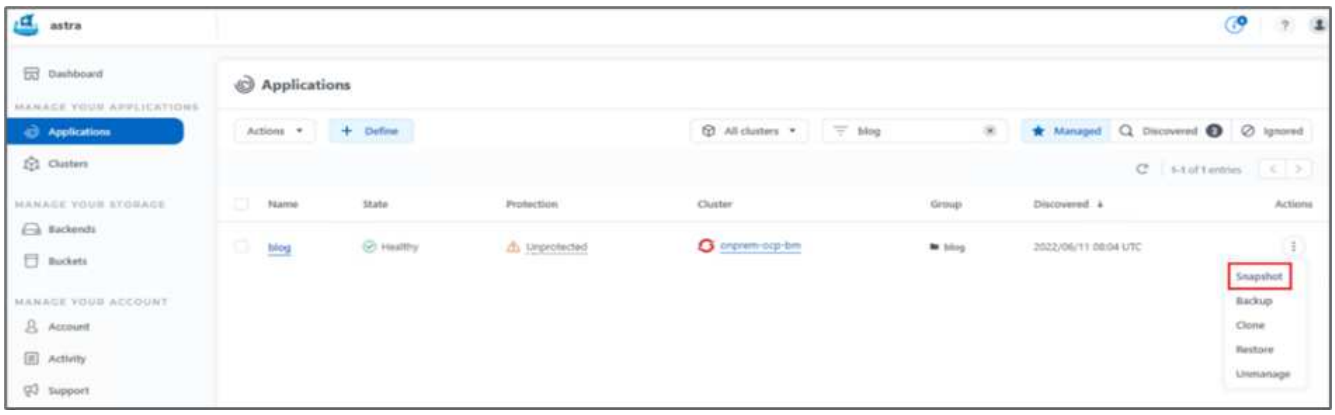
3. 측면 메뉴에서 네트워킹 > 경로를 선택하고 URL을 클릭합니다.



4. 블로그 홈 페이지가 표시됩니다. 일부 콘텐츠를 블로그 사이트에 추가하고 게시합니다.



5. Astra Control Center로 이동합니다. 먼저 검색된 탭에서 앱을 관리하고 스냅샷 복사본을 만듭니다.

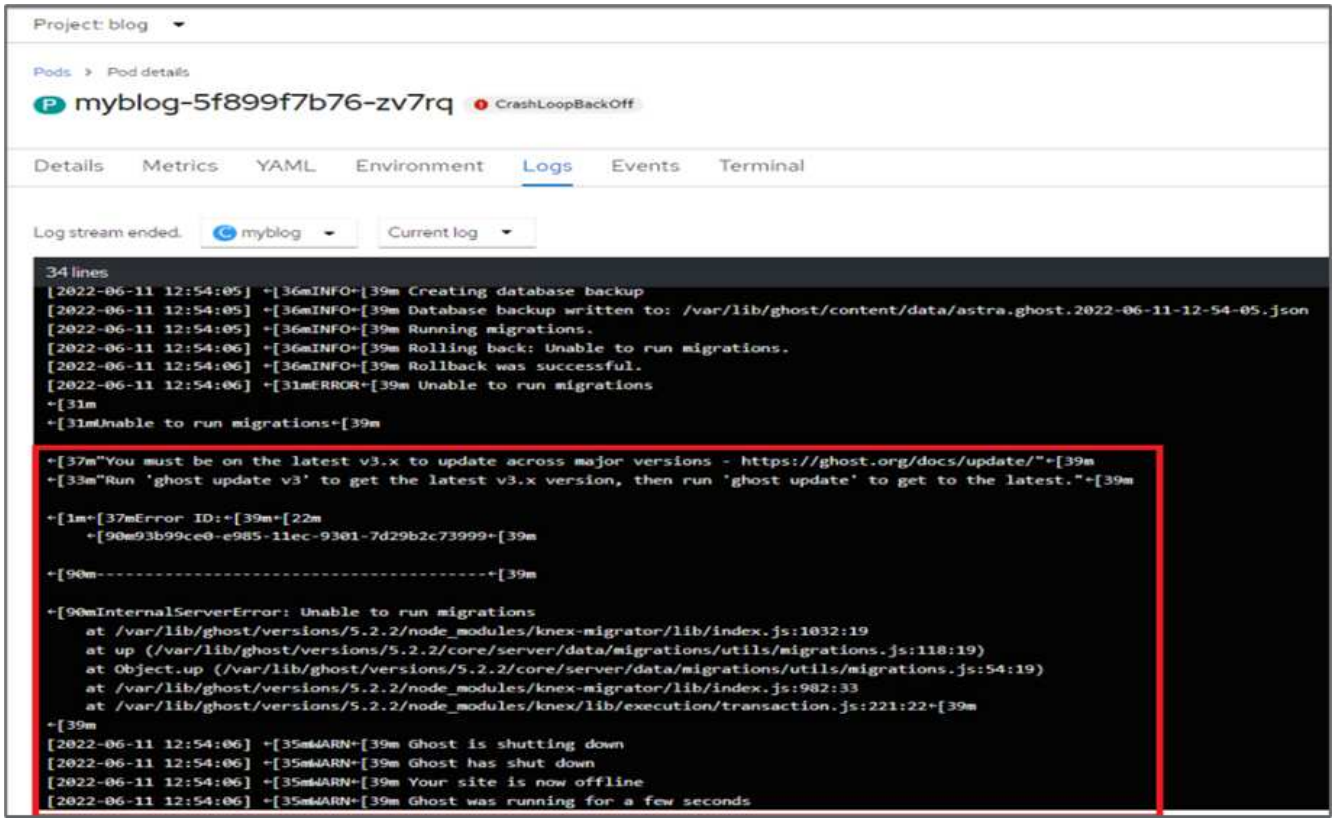


스냅샷, 백업 또는 둘 모두를 정의된 일정에 따라 생성하여 앱을 보호할 수도 있습니다. 자세한 내용은 을 참조하십시오 "스냅샷 및 백업으로 애플리케이션 보호".

6. 주문형 스냅샷이 생성된 후 앱을 최신 버전으로 업그레이드하십시오. 현재 이미지 버전은 고스트: 3.6-알파인(Ghost: 3.6-알파인)이며 타겟 버전은 유령: 최신이다. 앱을 업그레이드하려면 Git 리포지토리로 직접 변경하고 Argo CD에 동기화하십시오.

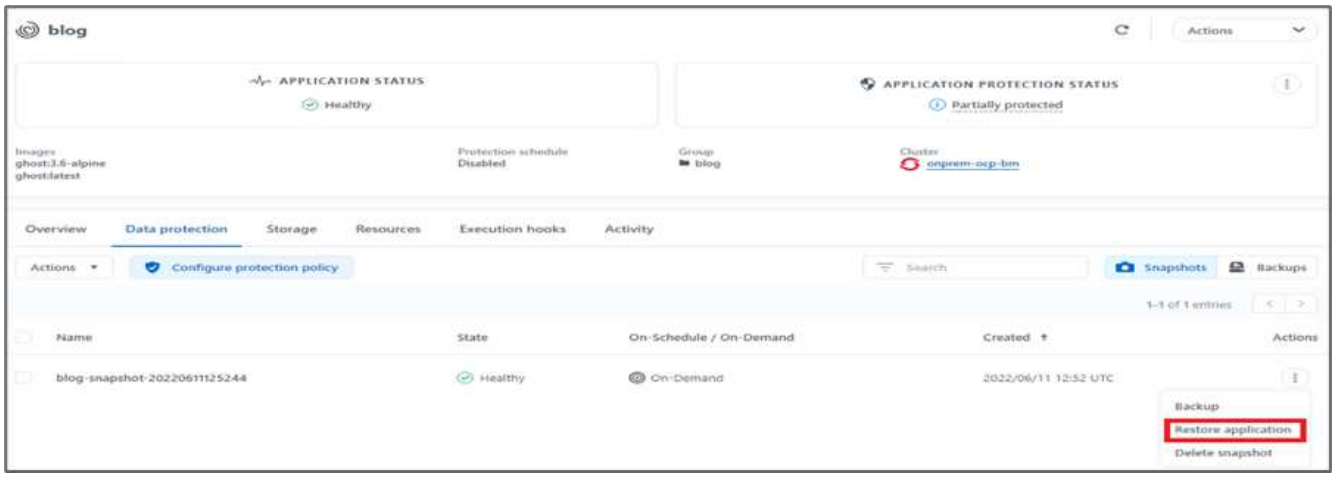
```
spec:
  containers:
  - name: myblog
    image: ghost:latest
    imagePullPolicy: Always
    ports:
    - containerPort: 2368
```

7. 블로그 사이트가 다운되고 전체 응용 프로그램이 손상되었기 때문에 최신 버전으로 직접 업그레이드할 수 없습니다.

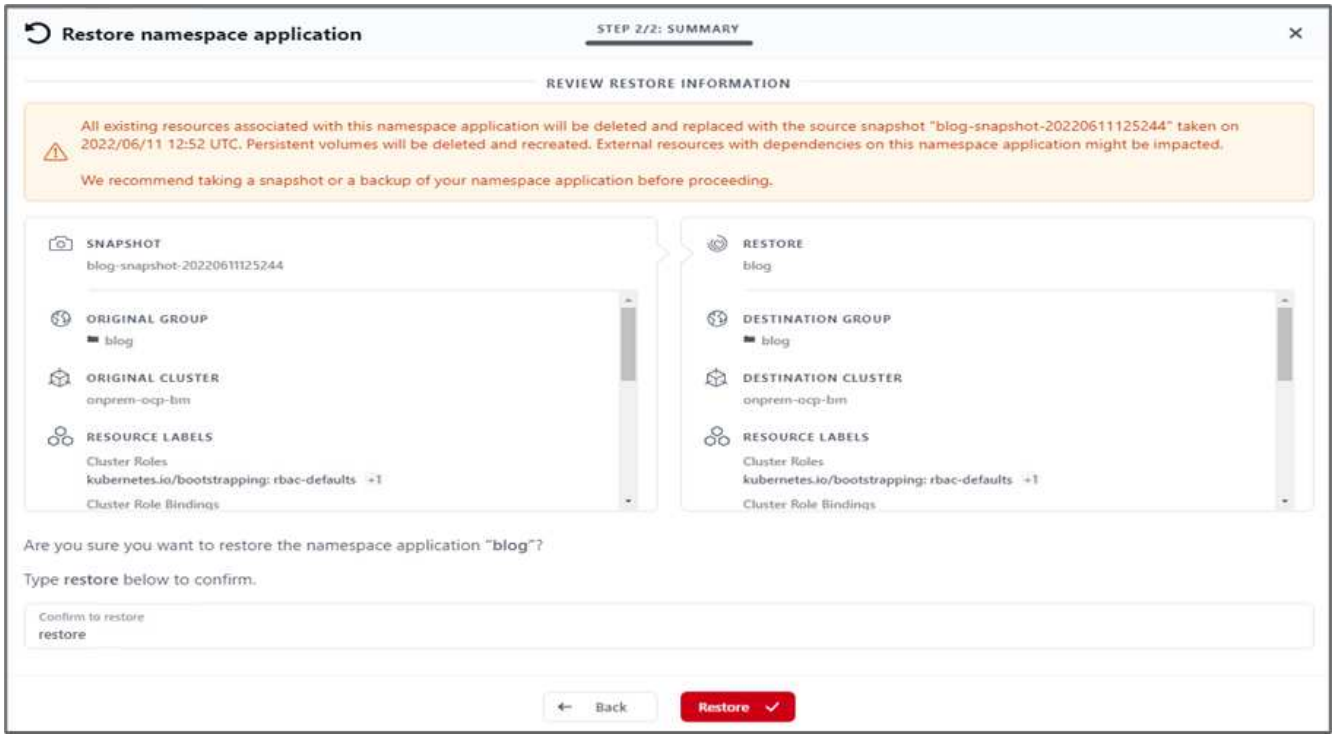


8. 블로그 사이트를 사용할 수 없음을 확인하려면 URL을 새로 고치십시오.

9. 스냅샷에서 앱을 복구합니다.

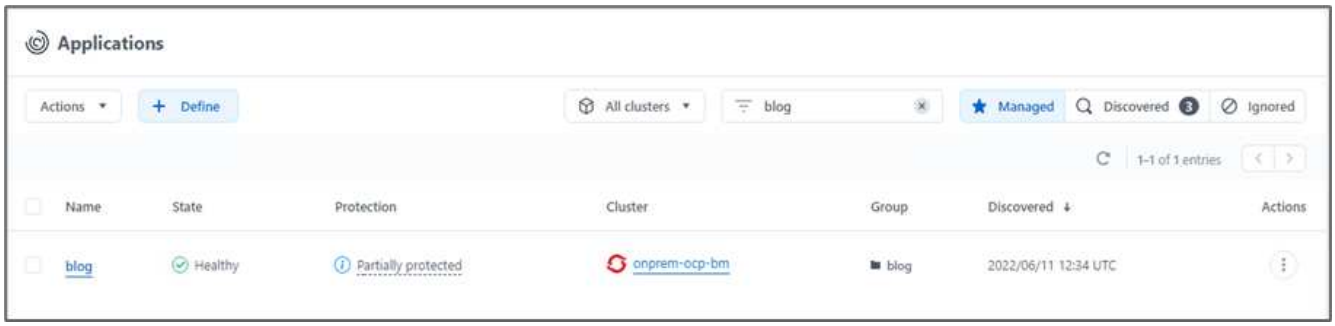


10. 앱은 동일한 OpenShift 클러스터에서 복원됩니다.

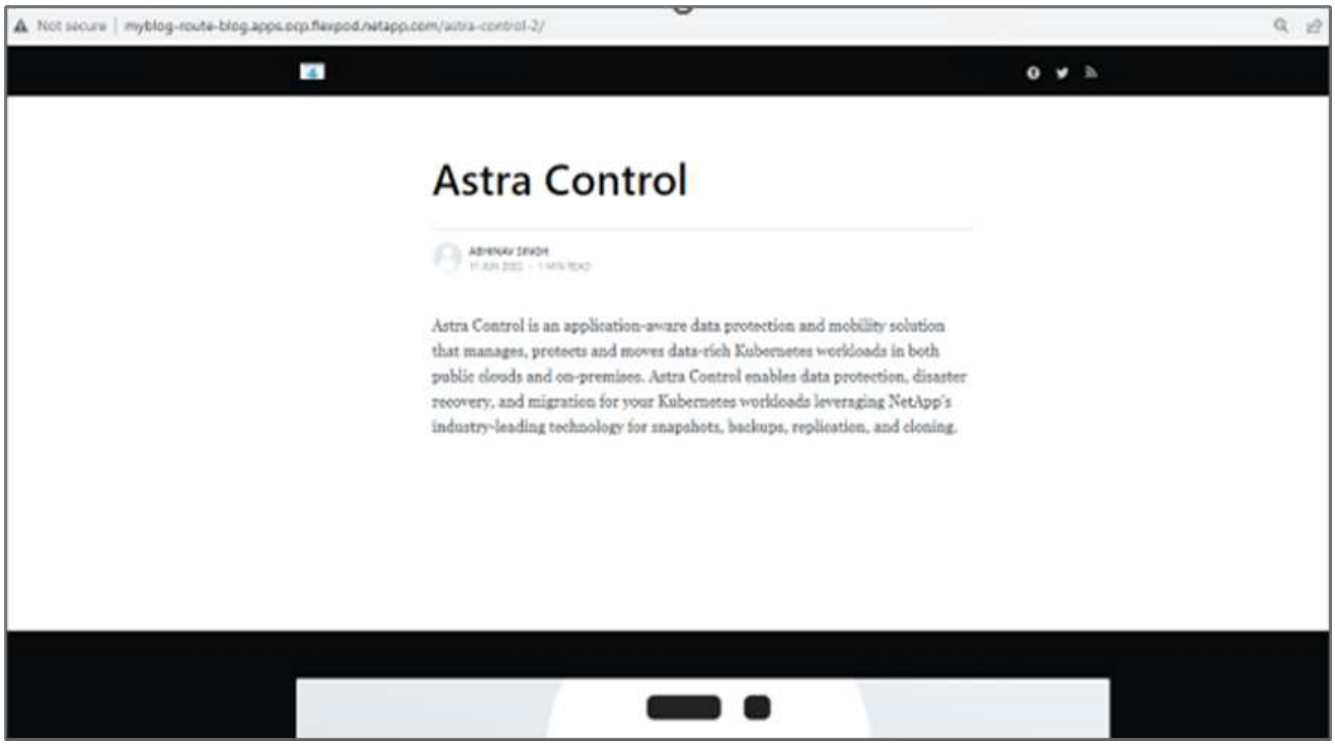


11. 앱 복원 프로세스가 즉시 시작됩니다.

12. 몇 분 후 사용 가능한 스냅샷에서 앱이 성공적으로 복구됩니다.



13. 웹 페이지를 사용할 수 있는지 확인하려면 URL을 새로 고치십시오.

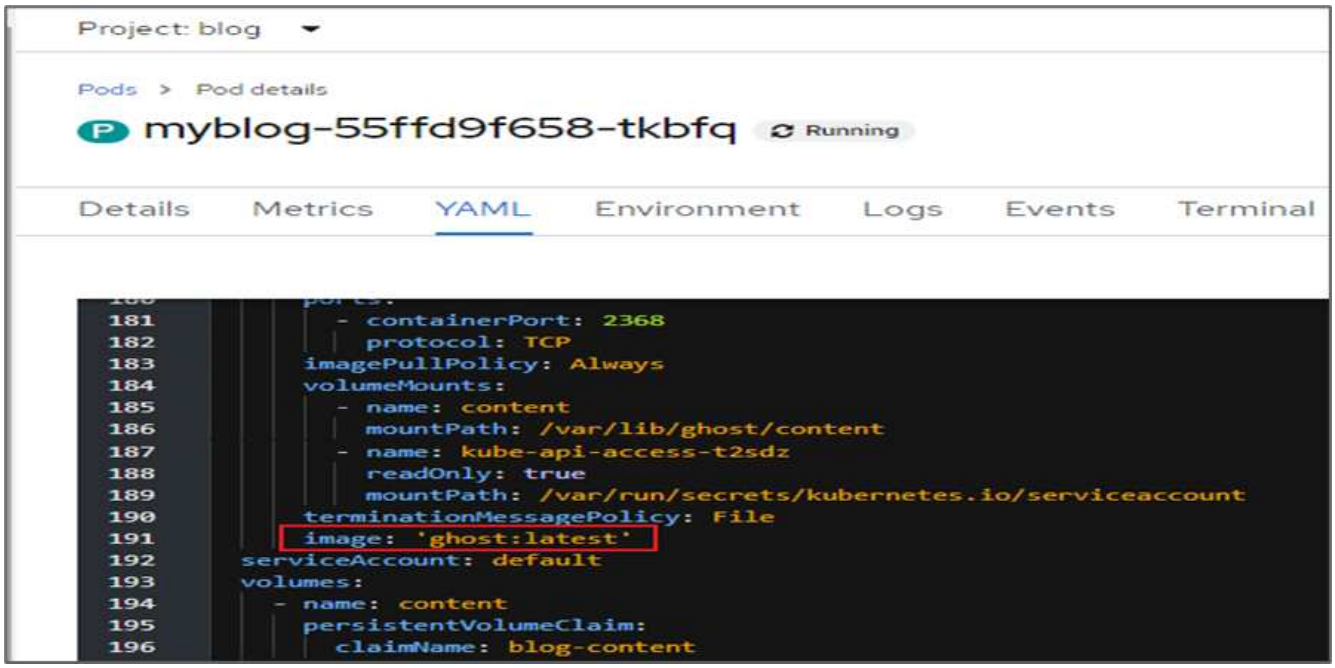


DevTest 팀은 Astra Control Center를 통해 스냅샷을 사용하여 블로그 사이트 앱 및 관련 데이터를 성공적으로 복구할 수 있습니다.

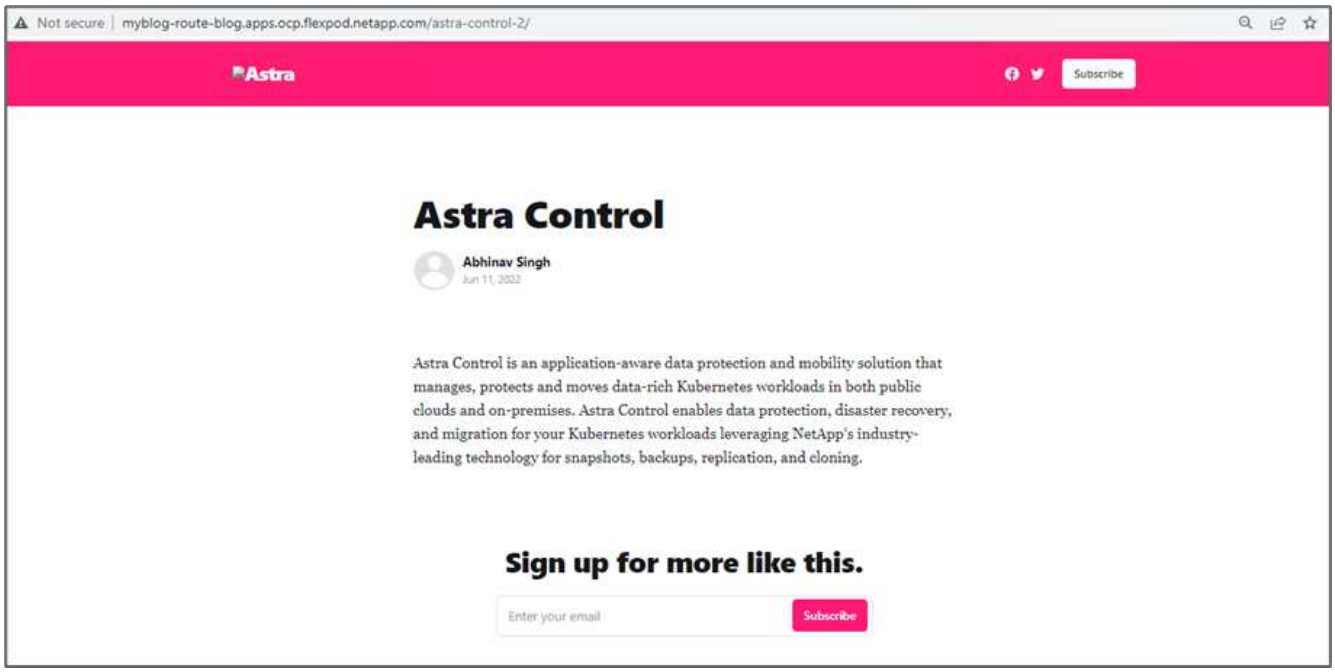
2부

Astra Control Center를 사용하면 클러스터 위치에 관계없이 온프레미스 또는 클라우드의 전체 애플리케이션과 함께 Kubernetes 클러스터 간에 데이터를 이동할 수 있습니다.

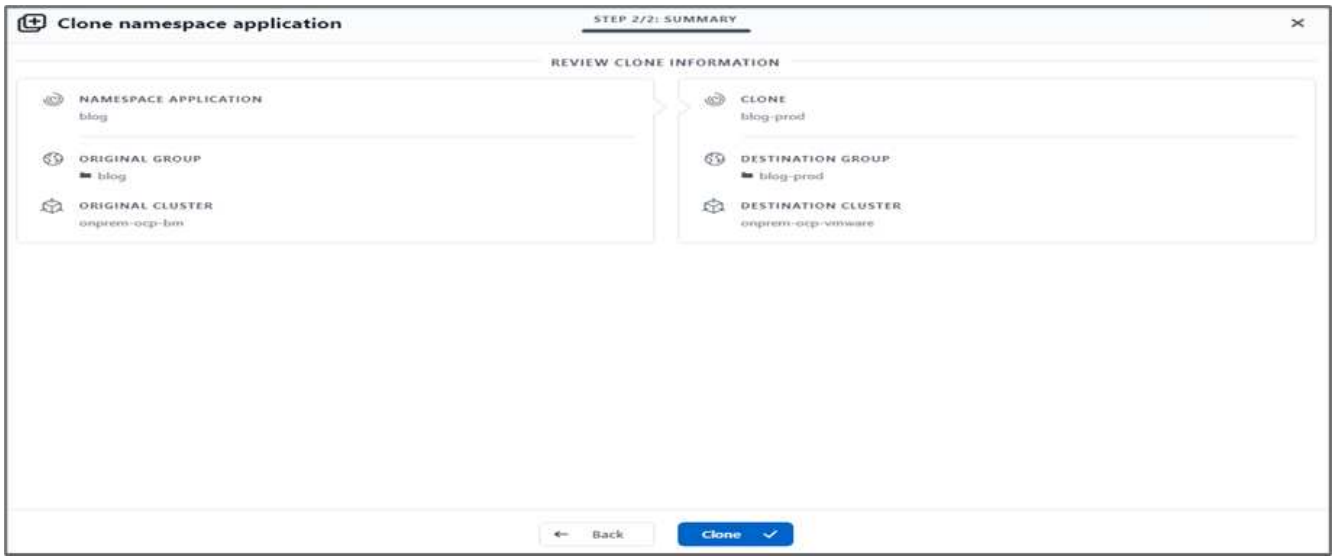
1. DevTest 팀은 처음에 앱을 지원되는 버전('Ghost-4.6-Alpine')으로 업그레이드한 후 최종 버전('Ghost-latest')으로 업그레이드하여 프로덕션을 준비합니다. 그런 다음 다른 FlexPod 시스템에서 실행되는 프로덕션 OpenShift 클러스터에 클론 복제된 앱을 업그레이드합니다.
2. 이때 앱이 최신 버전으로 업그레이드되어 운영 클러스터에 클론 복제할 준비가 됩니다.



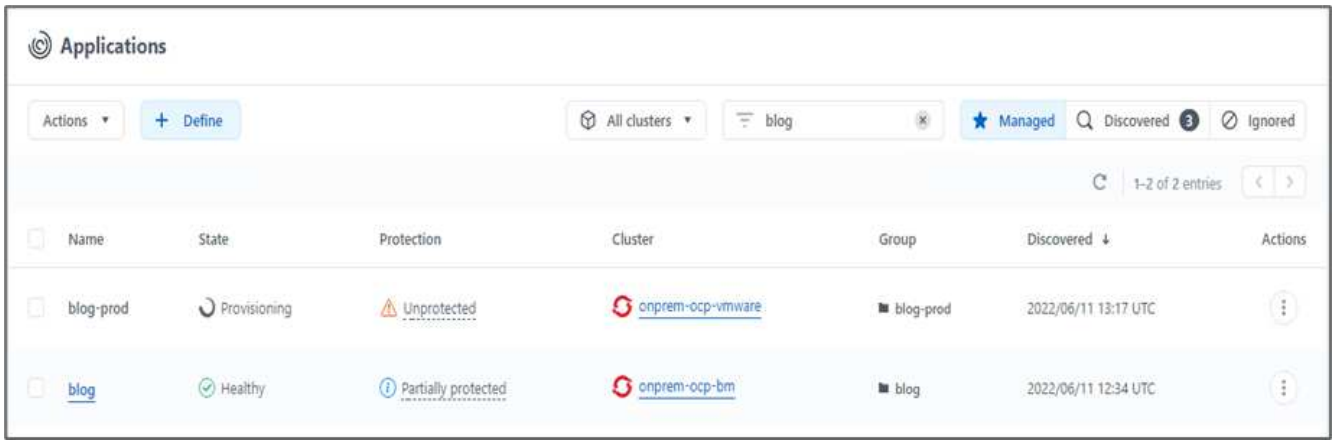
3. 새 테마를 확인하려면 블로그 사이트를 새로 고칩니다.



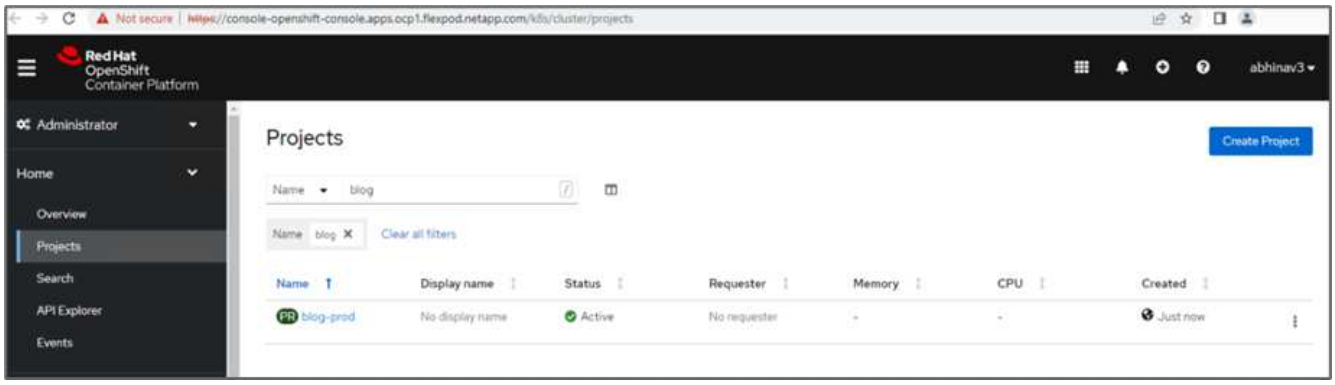
4. Astra Control Center에서 VMware vSphere에서 실행 중인 다른 운영 OpenShift 클러스터로 앱을 복제합니다.



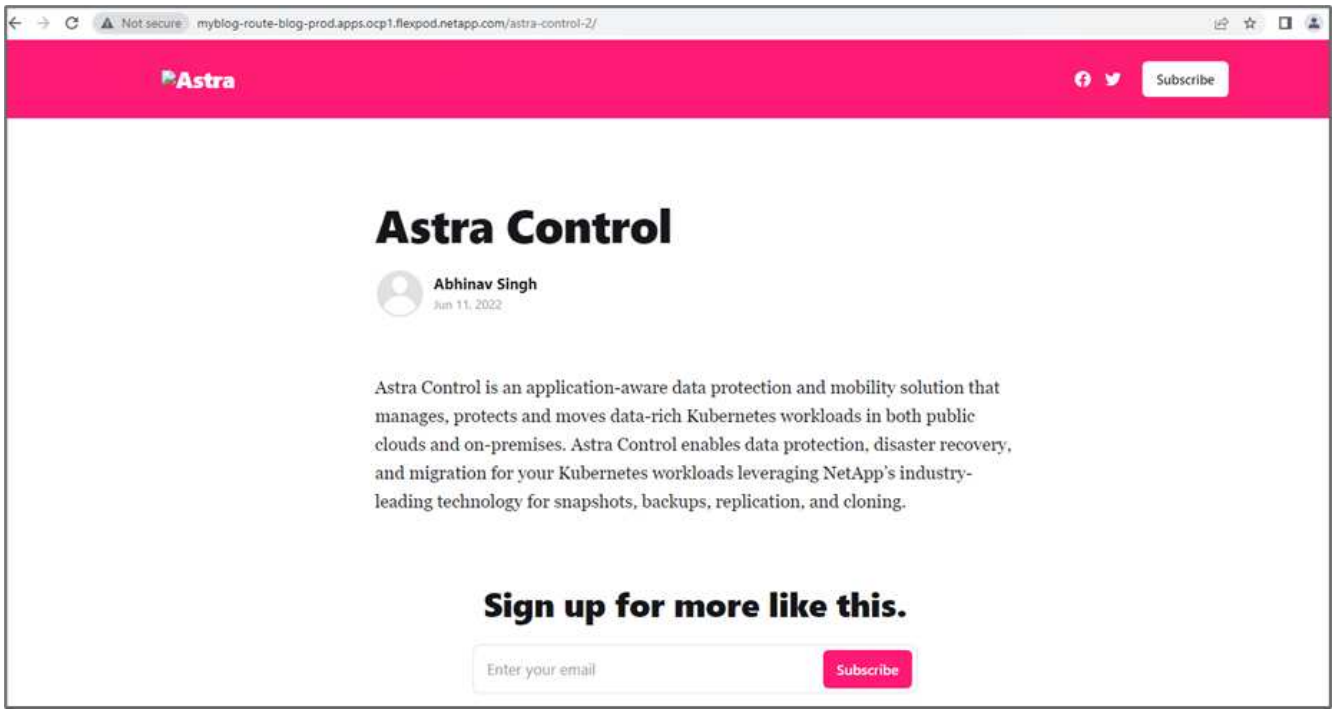
이제 운영 OpenShift 클러스터에서 새 애플리케이션 클론이 프로비저닝됩니다.



5. Production OpenShift 클러스터에 로그인하여 프로젝트 블로그를 검색합니다.



6. 측면 메뉴에서 네트워킹 > 경로를 선택하고 위치 아래에서 URL을 클릭합니다. 콘텐츠가 포함된 동일한 홈페이지가 표시됩니다.



이것으로 Astra Control Center 솔루션 검증을 마치겠습니다. 이제 Kubernetes 클러스터의 위치에 관계없이 전체 애플리케이션과 해당 데이터를 Kubernetes 클러스터 간에 복제할 수 있습니다.

"다음: 결론."

결론

"이전: 원격 백업을 사용한 애플리케이션 복구."

이 솔루션에서는 NetApp Astra 포트폴리오를 사용하여 FlexPod 및 AWS에서 실행되는 컨테이너식 애플리케이션에 대한 보호 계획을 구현했습니다. NetApp Astra Control Center 및 Astra Trident와 Cloud Volumes ONTAP, Red Hat OpenShift 및 FlexPod 인프라와 함께 이 솔루션의 핵심 구성요소를 구성했습니다.

NetApp은 스냅샷 캡처를 통해 애플리케이션 보호를 입증했으며 전체 복사본 백업을 실행하여 클라우드 및 사내 환경에서 실행되는 서로 다른 K8s 클러스터 간에 애플리케이션을 복원합니다.

또한, K8s 클러스터 전체에 애플리케이션을 클론 복제함으로써 고객이 원하는 위치에서 원하는 K8s 클러스터로 애플리케이션을 마이그레이션할 수 있도록 했습니다.

FlexPod은 고객이 애플리케이션 및 비즈니스 제공 프로세스를 현대화할 수 있도록 지속적으로 진화해 왔습니다. FlexPod 고객은 이 솔루션을 사용하여 임시 또는 전체 시간 DR 계획을 위한 위치로 퍼블릭 클라우드를 사용하는 클라우드 네이티브 애플리케이션에 대한 BCDR 계획을 자신 있게 구축하는 동시에 솔루션 비용을 절감할 수 있습니다.

Astra Control을 사용하면 클러스터의 위치에 관계없이 전체 애플리케이션과 함께 Kubernetes 클러스터 간에 데이터를 이동할 수 있습니다. 또한 클라우드 네이티브 애플리케이션을 더 빠르게 구축, 운영, 보호할 수 있습니다.

문제 해결

문제 해결 지침은 을 참조하십시오 "온라인 설명서".

추가 정보를 찾을 수 있는 위치

이 문서에 설명된 정보에 대해 자세히 알아보려면 다음 문서 및/또는 웹 사이트를 검토하십시오.

- FlexPod 홈 페이지

  "https://www.flexpod.com"

- FlexPod용 Cisco Validated Design 및 구축 가이드

  "https://www.cisco.com/c/en/us/solutions/design-zone/data-center-design-guides/flexpod-design-guides.html"

- Ansible을 사용하여 VMware용 코드로 인프라를 포함하는 FlexPod 구축

  "https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/UCS_CVDs/flexpod_m6_esxi7u2.html#AnsibleAutomationWorkflowandSolutionDeployment"

- Ansible을 사용한 Red Hat OpenShift 베어 메탈 의 코드로서 인프라를 포함하는 FlexPod 구축

  "https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/UCS_CVDs/flexpod_iac_redhat_openshift.html"

- Cisco UCS 하드웨어 및 소프트웨어 상호 운용성 툴

  "http://www.cisco.com/web/techdoc/ucs/interoperability/matrix/matrix.html"

- Cisco Intersight 데이터시트

  "https://intersight.com/help/saas/home"

- NetApp Astra 문서

  "https://docs.netapp.com/us-en/astra-control-center/index.html"

- NetApp Astra Control Center를 참조하십시오

  "https://docs.netapp.com/us-en/astra-control-center/index.html"

- NetApp Astra Trident

  "https://docs.netapp.com/us-en/trident/index.html"

- NetApp 클라우드 관리자

  "https://docs.netapp.com/us-en/occm/concept_overview.html"

- NetApp Cloud Volumes ONTAP를 참조하십시오

  "https://docs.netapp.com/us-en/occm/task_getting_started_aws.html"

- Red Hat OpenShift

  "https://www.openshift.com/"

- NetApp 상호 운용성 매트릭스 툴

  "http://support.netapp.com/matrix/"

버전 기록

버전	날짜	문서 버전 기록
버전 1.0	2022년 7월	ACC 22.04.0 해제

저작권 정보

Copyright © 2024 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 라이센스와 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 라이센스도 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 라이센스를 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 라이센스는 DFARS 조항 252.227-7015(b)(2014년 2월)에 명시된 권한으로 제한됩니다.

상표 정보

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