



ネットアップの**Astra**と**Cisco Intersight**を活用した**FlexPod**ハイブリッドクラウドを**Red Hat OpenShift**に活用

FlexPod

NetApp
March 13, 2026

目次

ネットアップのAstraとCisco Intersightを活用したFlexPod ハイブリッドクラウドをRed Hat OpenShiftに活用	1
TR-4936：『FlexPod hybrid cloud with NetApp Astra and Cisco Intersight for Red Hat OpenShift』	1
はじめに	1
対象者	1
NetApp Astra Control–主なユースケース	1
解決策コンポーネント	3
FlexPod	3
Astra Control の略	3
Astra Trident	4
ストレージバックエンド	5
NetApp Cloud Volumes ONTAP の略	5
Cloud Central にアクセスできます	6
クラウドマネージャ	6
コネクタ	6
NetApp Cloud Insights の略	6
NetApp Active IQ Unified Manager の略	6
Cisco Intersightの	7
Red Hat OpenShift Container Platform	8
VMware vSphere 7.0	9
ハードウェアおよびソフトウェアのリビジョン	9
インストールと設定	11
FlexPod for OpenShift Container Platform 4ベアメタルインストール	11
AWSにRed Hat OpenShiftを実装しました	12
NetApp Cloud Volumes ONTAP の略	13
OpenShift Container PlatformにAstra Control Centerをインストールします	13
解決策の検証	33
概要	33
リモートバックアップによるアプリケーションのリカバリ	34
まとめ	55
トラブルシューティング	56
追加情報の参照先	56
バージョン履歴	57

ネットアップのAstraとCisco Intersightを活用したFlexPod ハイブリッドクラウドをRed Hat OpenShiftに活用

TR-4936 : 『FlexPod hybrid cloud with NetApp Astra and Cisco Intersight for Red Hat OpenShift』

Abhinav Singhの

はじめに

コンテナ化されたアプリケーションの開発、導入、実行、管理、拡張のための事実上の選択肢となるコンテナとKubernetesは、ビジネスクリティカルなアプリケーションを実行する企業の増加に伴いつつあります。ビジネスクリティカルなアプリケーションは、状態に大きく依存しています。ステートフルアプリケーションには、状態、データ、および設定情報が関連付けられており、ビジネスロジックを実行する前のデータトランザクションに依存します。Kubernetes上で実行されるビジネスクリティカルなアプリケーションには、従来型アプリケーションのような可用性とビジネス継続性の要件が引き続きあります。サービスの停止は、収益の損失、生産性、会社の評判に深刻な影響を与える可能性があります。そのため、Kubernetesワークロードをクラスタ、オンプレミスデータセンター、ハイブリッドクラウド環境内およびクラスタ間で迅速かつ容易に保護、リカバリ、移動することが非常に重要です。企業は、ビジネスをハイブリッドクラウドモデルに移行し、アプリケーションをクラウドネイティブなフォームファクタに刷新するメリットを高く挙げています。

このテクニカルレポートでは、FlexPod コンバージドインフラ解決策 上にNetApp Astra Control CenterとRed Hat OpenShift Container Platformを統合し、Amazon Web Services (AWS) に拡張してハイブリッドクラウドデータセンターを構築しました。使い慣れたことに基づいて作成されています ["FlexPod とRed Hat OpenShift"](#) このドキュメントでは、ネットアップのAstra Control Centerについて説明します。インストール、設定、アプリケーション保護ワークフロー、オンプレミスとクラウド間でのアプリケーション移行から始まります。また、Red Hat OpenShiftで実行されるコンテナ化アプリケーションにNetApp Astra Control Centerを使用する場合の、アプリケーション対応データ管理機能（バックアップとリカバリ、ビジネス継続性など）の利点についても説明します。

次の図に、解決策 の概要を示します。

[エラー：グラフィックイメージがありません]

対象者

このドキュメントは、CTO（最高技術責任者）、アプリケーション開発者、クラウド解決策 アーキテクト、サイト信頼性エンジニア（SRE）、DevOpsエンジニア、ITOps、コンテナ化されたアプリケーションの設計、ホスティング、管理に重点を置いたプロフェッショナルサービスチームなど、読者を想定しています。

NetApp Astra Control–主なユースケース

ネットアップのAstra Controlは、クラウドネイティブなマイクロサービスを利用するお客様にアプリケーション保護を簡易化することを目的としています。

- スナップショットを使用したポイントインタイム (PiT) アプリケーションリプレゼンテーション。Astra Controlを使用すると、Kubernetesで実行されているアプリケーションの構成の詳細や、関連付けられた永

続的ストレージを含む、コンテナ化されたアプリケーションのエンドツーエンドのスナップショットを作成できます。インシデントが発生した場合は、ボタンクリックでアプリケーションを正常な状態に復元できます。

- フルコピー・アプリケーション・バックアップ Astra Controlを使用すると'事前に定義されたスケジュールでフル・アプリケーション・バックアップを実行できますこのスケジュールを使用すると'アプリケーションを同じKubernetesクラスタにリストアしたり'別のKubernetesクラスタにオンデマンドで自動化された方法でリストアしたりできます
- クローンを使用したアプリケーションの移植と移行 Astra Controlを使用すると、アプリケーション全体をKubernetesクラスタ間または同じKubernetesクラスタ内のデータとともにクローニングできます。この機能は、クラスタがどこにあるかに関係なく、Kubernetesクラスタ間でアプリケーションを移植または移行する場合にも役立ちます（クローニング後にソースアプリケーションインスタンスを削除するだけです）。
- アプリケーションの一貫性をカスタマイズできます。Astra Controlを使用すると、実行フックを活用してアプリケーションの休止状態を定義できます。「実行前」と「実行後」のフックをスナップショットおよびバックアップのワークフローにドロップすると、スナップショットまたはバックアップが作成される前に、アプリケーションが独自の方法で休止されます。
- アプリケーションレベルのディザスタリカバリ（DR）を自動化。Astra Controlを使用すると、コンテナ化されたアプリケーション用にBCDR（ビジネス継続性ディザスタリカバリ）計画を設定できます。NetApp SnapMirrorはバックエンドで使用されるため、DRワークフローの実装はすべて自動化されます。

解決策 トポロジ

このセクションでは、解決策 の論理トポロジについて説明します。

次の図は、OpenShift Container Platformクラスタを実行するFlexPod オンプレミス環境と、NetApp Cloud Volumes ONTAP、Cisco Intersight、NetApp Cloud Manager SaaSプラットフォームを使用するAWS上の自己管理型OpenShift Container Platformクラスタからなる解決策 トポロジを示しています。

[エラー：グラフィックイメージがありません]

最初のOpenShift Container PlatformクラスタはFlexPod 上にベアメタルインストールされ、2番目のOpenShift Container PlatformクラスタはFlexPod 上で実行されるVMware vSphereに導入され、3番目のOpenShift Container Platformクラスタはとして導入されます "**プライベートクラスタ**" 自社で管理するインフラとして、AWS上の既存の仮想プライベートクラウド（VPC）に導入できます。

この解決策 では、FlexPod はサイト間VPNを介してAWSに接続されますが、直接接続の実装を使用してハイブリッドクラウドに拡張することもできます。Cisco Intersightは、FlexPod インフラストラクチャコンポーネントの管理に使用されます。

この解決策 では、Astra Control Centerが、FlexPod およびAWSで実行されているOpenShift Container Platformクラスタでホストされているコンテナ化アプリケーションを管理します。FlexPod で実行されているOpenShiftベアメタルインスタンスにAstraコントロールセンターをインストールします。Astra Controlは、マスターノードのkube-APIと通信し、Kubernetesクラスタの変更を継続的に監視します。Kubernetesクラスタに追加した新しいアプリケーションは自動的に検出され、管理用に使用できるようになります。

コンテナ化されたアプリケーションのPiT表現は、Astra Control Centerを使用してスナップショットとしてキャプチャできます。アプリケーションスナップショットは、スケジュールされた保護ポリシーまたはオンデマンドで開始できます。Astraがサポートしているアプリケーションの場合、スナップショットはクラッシュ整合性があります。アプリケーションスナップショットは、永続ボリューム内のアプリケーションデータのスナップショットと、そのアプリケーションに関連付けられているさまざまなKubernetesリソースのアプリケーションメタデータを構成します。

アプリケーションのフルコピーバックアップは、事前定義されたバックアップスケジュールまたはオンデマンドを使用して、Astra Controlを使用して作成できます。オブジェクトストレージは、アプリケーションデータのバックアップを格納するために使用されます。NetApp ONTAP S3、NetApp StorageGRID、および汎用のS3実装をオブジェクトストアとして使用できます。

"次の例は、解決策 コンポーネントです。"

解決策コンポーネント

"Previous : 解決策の概要を示します。"

FlexPod

FlexPod は、仮想化ソリューションと非仮想化ソリューションの両方の統合基盤となるハードウェアとソフトウェアの定義済みセットです。FlexPod には、NetApp ONTAP ストレージ、Cisco Nexus ネットワーク、Cisco MDS ストレージ ネットワーク、Cisco Unified Computing System (Cisco UCS) が含まれています。この設計は、ネットワーク、コンピューティング、ストレージを1つのデータセンターラックに収容できる柔軟性を備えています。また、お客様のデータセンター設計に従って導入することもできます。ポート密度を使用すると、ネットワークコンポーネントは複数の構成に対応できます。

Astra Control の略

Astra Controlは、パブリッククラウドとオンプレミスの両方でホストされるクラウドネイティブアプリケーションに対して、アプリケーション対応のデータ保護サービスを提供します。Astra Controlは、Kubernetesで実行されるコンテナ化されたアプリケーションに、データ保護、ディザスタリカバリ、移行の機能を提供します。

の機能

Astra Control は、Kubernetes アプリケーションデータのライフサイクル管理に不可欠な機能を提供

- 永続的ストレージを自動的に管理
- アプリケーションと整合性のあるオンデマンドのSnapshotとバックアップを作成
- ポリシーベースのスナップショット処理とバックアップ処理を自動化
- ハイブリッドクラウド環境で、アプリケーションと関連データをKubernetesクラスタから別のクラスタに移行する
- 同じKubernetesクラスタまたは別のKubernetesクラスタにアプリケーションをクローニングする
- アプリケーション保護ステータスを視覚化します
- グラフィカルユーザーインターフェイスとすべての保護ワークフローを社内ツールから実装するためのREST APIの完全なリストを提供します。

Astra Controlを使用すると、Kubernetesクラスタで作成された関連リソースの情報を含む、コンテナ化されたアプリケーションを一元的に可視化できます。すべてのクラスタ、すべてのアプリケーション、すべてのクラウド、またはすべてのデータセンターを1つのポータルで表示できます。オンプレミスまたはパブリッククラウドのすべての環境でAstra Control APIを使用して、データ管理ワークフローを実装できます。

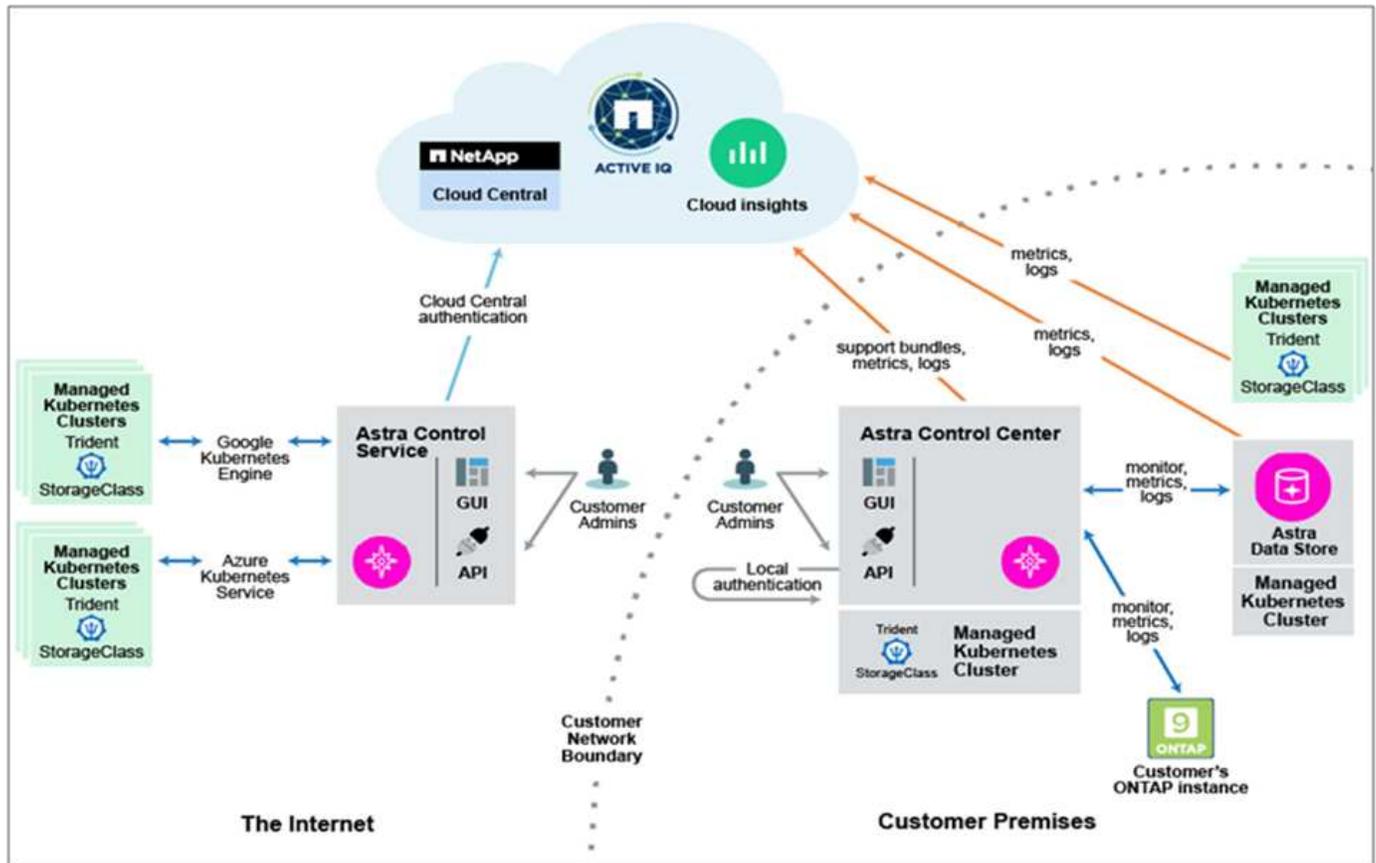
Astra Control消費モデル

Astra Controlには、次の2つの消費モデルがあります。

- * Astra Control Service。ネットアップがホストするフルマネージドサービス。Google Kubernetes Engine (GKE) 、 Azure Kubernetes Service (AKS) でKubernetesクラスタのアプリケーション対応データ管理を実現します。
- * Astra Control Center。*オンプレミスおよびハイブリッドクラウド環境で実行されるKubernetesクラスタのアプリケーション対応データ管理を提供する、自己管理ソフトウェアです。

このテクニカルレポートでは、Kubernetesで実行されるクラウドネイティブアプリケーションを管理するために、Astra Control Centerを活用しています。

次の図は、Astra Controlアーキテクチャを示しています。



Astra Trident

Astra Tridentは、コンテナやKubernetesディストリビューション向けの、完全にサポートされているオープンソースのストレージオーケストレーションツールです。コンテナ化されたアプリケーションの永続性に対する要求を、業界標準のインターフェイス（など）を使用して満たすことができるように、最初から設計されています "CSI (Container Storage Interface) "。Astra Tridentを使用すると、マイクロサービスやコンテナ化されたアプリケーションを利用して、ネットアップのストレージシステムポートフォリオが提供するエンタープライズクラスのストレージサービスを活用できます。

Astra Trident は、 Kubernetes クラスタにポッドとしてデプロイされ、 Kubernetes ワークロードに動的なストレージ オーケストレーション サービスを提供します。これにより、コンテナ化されたアプリケーションは、 NetApp ONTAP (NetApp AFF、 NetApp FAS、 NetApp ONTAP Select、 クラウド、 Amazon FSx for

NetApp ONTAP)、 NetApp Elementソフトウェア (NetApp SolidFire)、 Azure NetApp Filesサービスなど、 NetApp の幅広いポートフォリオから永続ストレージを迅速かつ簡単に利用できるようになります。 FlexPod環境では、 Astra Tridentを使用して、 NetApp AFFおよびFASシステムやCloud Volumes ONTAPなどのONTAPストレージ プラットフォームでホストされているNetApp FlexVolボリュームと LUN によってサポートされるコンテナの永続ボリュームを動的にプロビジョニングおよび管理します。 Trident は、 Astra Control が提供するアプリケーション保護スキームの実装においても重要な役割を果たします。 AstraTridentの詳細については、 "[Astra Tridentのドキュメント](#)"

ストレージバックエンド

Astra Tridentを使用するには、サポートされているストレージバックエンドが必要です。 Tridentバックエンドは、 Tridentとストレージシステムとの関係を定義します。 Tridentは、 そのストレージシステムとの通信方法や、 Tridentがそのシステムからボリュームをプロビジョニングする方法を解説します。 Tridentは、 あるストレージクラスが定義した要件を満たしたストレージプールをバックエンドから自動的に提供します。

- ONTAP AFF と FAS のストレージバックエンド。 ONTAP は、 ストレージソフトウェアおよびハードウェアプラットフォームとして、 コアストレージサービス、 複数のストレージアクセスプロトコルのサポート、 およびネットアップのSnapshotコピーやミラーリングなどのストレージ管理機能を提供します。
- Cloud Volumes ONTAP ストレージバックエンド
- "[Astra データストア](#)" ストレージバックエンド

NetApp Cloud Volumes ONTAP の略

NetApp Cloud Volumes ONTAP は、 ファイルワークロードとブロックワークロードに高度なデータ管理機能を提供するSoftware-Defined Storageです。 Cloud Volumes ONTAP を使用すると、 データ保護、 セキュリティ、 コンプライアンスを強化しながら、 クラウドストレージのコストを最適化し、 アプリケーションのパフォーマンスを向上させることができます。

主なメリットは次のとおりです。

- 組み込みのデータ重複排除、 データ圧縮、 シンプロビジョニング、 クローニングを活用して、 ストレージコストを最小限に抑えます。
- クラウド環境で障害が発生した場合でも、 エンタープライズクラスの信頼性と継続的な運用を確保できます。
- Cloud Volumes ONTAP は、 業界をリードするネットアップのレプリケーションテクノロジーであるSnapMirrorを活用して、 オンプレミスのデータをクラウドにレプリケートします。 これにより、 複数のユースケースでセカンダリコピーを簡単に利用できます。
- また、 Cloud Volumes ONTAP は Cloud Backup Service との統合により、 保護のためのバックアップとリストア機能、 およびクラウドデータの長期アーカイブ機能を提供します。
- アプリケーションをオフラインにすることなく、 ハイパフォーマンスとローパフォーマンスのストレージプールをオンデマンドで切り替えます。
- NetApp SnapCenter を使用してSnapshotコピーの整合性を確保します。
- Cloud Volumes ONTAP は、 データ暗号化をサポートし、 ウィルスやランサムウェアからの保護を提供します。
- クラウドデータセンスとの統合により、 データコンテキストを把握し、 機密データを識別できます。

Cloud Central にアクセスできます

Cloud Centralは、ネットアップのクラウドデータサービスにアクセスして管理するための一元的な場所を提供します。これらのサービスにより、重要なアプリケーションのクラウドでの実行、自動化されたDRサイトの作成、データのバックアップ、複数のクラウド間でのデータの効果的な移行と制御が可能になります。詳細については、[を参照してください "Cloud Centralにアクセスできます。"](#)

クラウドマネージャ

Cloud Managerは、エンタープライズクラスのSaaSベースの管理プラットフォームです。ITエキスパートとクラウドアーキテクトは、ネットアップのクラウドソリューションを使用して、ハイブリッドマルチクラウドインフラを一元管理できます。オンプレミスとクラウドのストレージを表示および管理する一元化されたシステムを提供し、ハイブリッドクラウド、複数のクラウドプロバイダ、アカウントをサポートします。詳細については、[を参照してください "クラウドマネージャ"](#)。

コネクタ

Connectorは、Cloud Managerがパブリッククラウド環境内のリソースとプロセスを管理できるようにするインスタンスです。Cloud Managerのさまざまな機能を使用するには、コネクタが必要です。コネクタは、クラウドまたはオンプレミスネットワークに導入できます。

Connectorは次の場所でサポートされます。

- AWS
- Microsoft Azure
- Google Cloud
- オンプレミス

コネクタの詳細については、[を参照してください "リンクをクリックしてください"](#)

NetApp Cloud Insights の略

ネットアップのクラウドインフラ監視ツールであるCloud Insights を使用すると、Astra Control Centerで管理されるKubernetesクラスタのパフォーマンスと利用率を監視できます。Cloud Insights : ストレージ使用率とワークロードの相関関係を示します。Cloud Insights 接続を Astra コントロールセンターで有効にすると、テレメータの情報が Astra コントロールセンターの UI ページに表示されます。

NetApp Active IQ Unified Manager の略

NetApp Active IQ Unified Manager では、デザインが一新され、直感的に操作できるインターフェイスからONTAP ストレージクラスタを監視できます。コミュニティの情報やAI分析から得た情報を活用できます。運用、パフォーマンス、プロアクティブな分析情報を提供し、ストレージ環境と仮想マシン (VM) で実行される環境を包括的に分析します。ストレージインフラで問題が発生すると、Unified Managerから問題の詳細情報を通知して、ルート原因の特定に役立てることができます。VMダッシュボードにはVMのパフォーマンス統計が表示されるため、VMware vSphereホストからネットワーク経由で最後にストレージへのI/Oパス全体を調査できます。一部のイベントには、問題を修正するための対応策も用意されています。問題が発生したときにEメールやSNMPトラップで通知されるように、イベントにカスタムアラートを設定できます。Active IQ Unified Manager を使用すると、容量や使用状況の傾向を予測して問題が発生する前にプロアクティブに対処することができるため、長期的な問題につながる短期的な事後対処策を実施する必要がなくなり、ユーザのストレージ要件に合わせて計画を立てることができます。

Cisco Intersightの

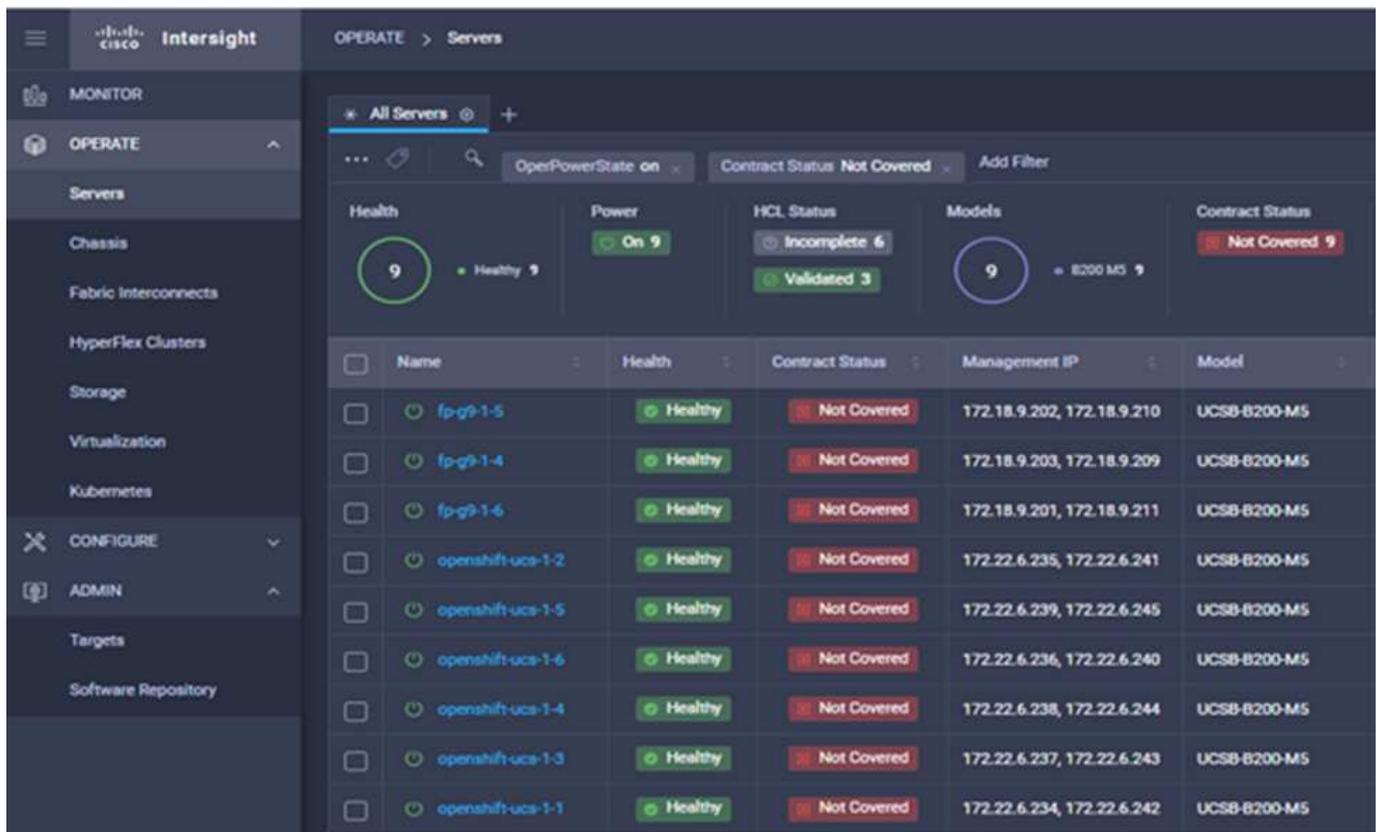
Cisco Intersightは、従来のアプリケーションやクラウドネイティブなインフラに向けて、インテリジェントな自動化、オブザーバビリティ、最適化を実現するSaaSプラットフォームです。このプラットフォームは、ITチームの変化を促進し、ハイブリッドクラウド向けに設計された運用モデルを提供します。

Cisco Intersightには、次のようなメリットがあります。

- *迅速な提供。*俊敏性に優れたソフトウェア開発モデルにより、クラウドまたはお客様のデータセンターからサービスとして提供され、頻繁な更新と継続的な技術革新を実現します。このようにして、お客様は基幹業務の提供を加速することに集中できます。
- *運用の簡素化。*共通のインベントリ、認証、APIを備えた単一のセキュアなSaaS提供ツールを使用して、スタック全体とすべての場所で作業し、チーム間のサイロを排除し、運用を簡素化します。オンプレミスの物理サーバやハイパーバイザーの管理からVM、Kubernetes、サーバレス、自動化、オンプレミスとパブリッククラウドの両方にわたって最適化とコスト管理を実現
- 継続的な最適化。 Cisco Intersightが提供するインテリジェンスを、Cisco TACだけでなくすべてのレイヤで使用して、環境を継続的に最適化します。このインテリジェンスは、推奨される自動化可能なアクションに変換されるため、ワークロードの移動や物理サーバの稼働状態の監視からKubernetesクラスタの自動サイジングまで、あらゆる変更リアルタイムに対応できます。また、コスト削減のために、作業中のパブリッククラウドが推奨されます。

Cisco Intersightには、UCSM Managed Mode (UMM) とIntersight Managed Mode (IMM) という2つの管理操作モードがあります。ファブリックインターコネクトの初期セットアップ中に、ファブリック接続Cisco UCSシステムのネイティブUmmまたはIMMを選択できます。この解決策では、ネイティブUmmが使用されます。

次の図は、Cisco Intersightのダッシュボードを示しています。

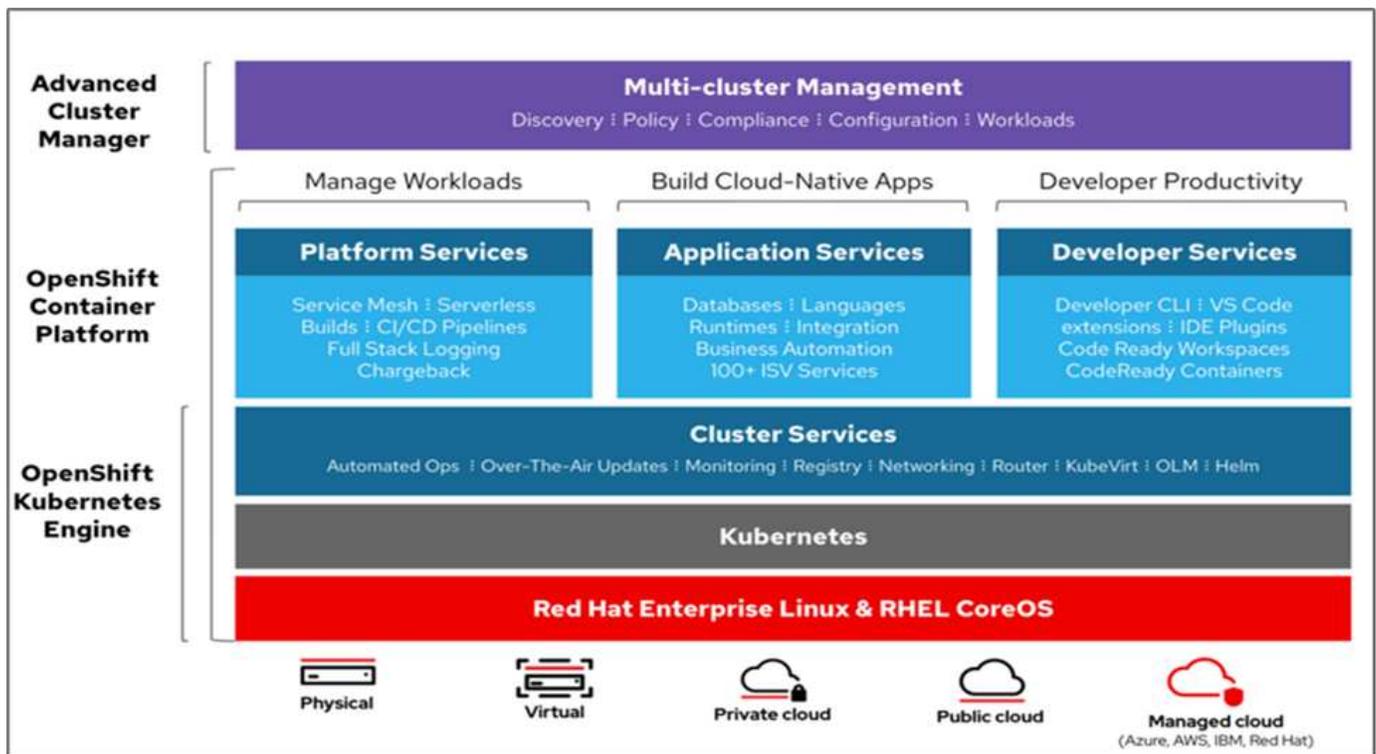


Red Hat OpenShift Container Platform

Red Hat OpenShift Container Platformは、CRI-OとKubernetesを統合し、これらのサービスを管理するためのAPIとWebインターフェイスを提供するコンテナアプリケーションプラットフォームです。CRI-Oは、Kubernetes Container Runtime Interface (CRI) を実装したもので、Open Container Initiative (OCI) 互換のランタイムを使用した実行を可能にします。Kubernetesの実行時にDockerを使用する代わりに、軽量なソリューションです。

OpenShift Container Platformにより、お客様はコンテナを作成および管理できます。コンテナは、オペレーティングシステムや基盤のインフラとは無関係に、それぞれの環境で実行されるスタンドアロンプロセスです。OpenShift Container Platformは、コンテナベースのアプリケーションの開発、導入、管理を支援します。アプリケーションをオンデマンドで作成、変更、および導入できるセルフサービスプラットフォームを提供し、開発とリリースのライフサイクルを短縮します。OpenShift Container Platformには、より小規模で分離されたユニットで構成されるマイクロサービスベースのアーキテクチャがあり、連携して機能します。Kubernetesクラスタ上で実行され、信頼性の高いクラスタキーバリュ型データストアであるetcdに格納されているオブジェクトに関するデータが含まれます。

次の図は、Red Hat OpenShift Containerプラットフォームの概要を示しています。



Kubernetesインフラ

Kubernetesは、OpenShift Container Platform内で、コンテナ化されたアプリケーションを一連のCRI-Oランタイムホスト全体で管理し、導入、メンテナンス、アプリケーション拡張のためのメカニズムを提供します。CRI-Oサービスは、コンテナ化されたアプリケーションをインスタンス化し、実行します。

Kubernetesクラスタは、1つ以上のマスターノードと一連のワーカーノードで構成されます。この解決策 設計には、ハードウェアのハイアベイラビリティ (HA) 機能とソフトウェアスタックが含まれています。Kubernetesクラスタは、3つのマスターノードと最低2つのワーカーノードでHAモードで実行されるように設計されており、クラスタに単一点障害がないようにします。

Red HatコアOS

OpenShift Container Platformは、Red Hat Enterprise Linux CoreOS (RHCOS) を使用します。RHCOSは、CoreOSとRed Hat Atomic Host OSの優れた機能を組み合わせたコンテナ指向のオペレーティングシステムです。RHCOSは、コンテナ化されたアプリケーションをOpenShift Container Platformから実行できるように特別に設計されており、新しいツールと連携して、迅速なインストール、オペレータベースの管理、簡単なアップグレードを実現します。

RHCOSには次の機能があります。

- イグニションは、最初にマシンを起動して構成する際に、OpenShift Container Platformが最初のブートシステム構成として使用するものです。
- Kubernetesネイティブのコンテナランタイム実装であるCRI-Oは、オペレーティングシステムと緊密に統合して、Kubernetes環境を効率的かつ最適化します。CRI-Oには、コンテナの実行、停止、再起動を行う機能があります。これは、OpenShift Container Platform 3で使用されていたDocker Container Engineに完全に代わるものです。
- Kubernetesの主要ノードエージェントであるKubeletはコンテナの起動と監視を担当しています。

VMware vSphere 7.0

VMware vSphereは、大量のインフラ（CPU、ストレージ、ネットワークなどのリソース）をシームレスで汎用性に優れた動的な運用環境として包括的に管理する仮想化プラットフォームです。個々のマシンを管理する従来のオペレーティングシステムとは異なり、VMware vSphereはデータセンター全体のインフラストラクチャを集約して、必要なアプリケーションに迅速かつ動的に割り当てられるリソースを備えた単一の強力なサーバを作成します。

詳細については、を参照してください "[VMware vSphere の場合](#)"。

VMware vSphere vCenterの場合

VMware vCenter Serverでは、1つのコンソールからすべてのホストとVMを統合的に管理でき、クラスタ、ホスト、およびVMのパフォーマンス監視を集約できます。VMware vCenter Serverを使用すると、管理者は、コンピューティングクラスタ、ホスト、VM、ストレージ、ゲストOS、仮想インフラストラクチャのその他の重要なコンポーネントVMware vCenterは、VMware vSphere環境で使用できる豊富な機能を管理します。

ハードウェアおよびソフトウェアのリビジョン

この解決策は、で定義されている、サポートされているバージョンのソフトウェア、ファームウェア、およびハードウェアを実行している任意のFlexPod環境に拡張できます "[NetApp Interoperability Matrix Toolで確認できます](#)" および "[Cisco UCSハードウェア互換性リスト](#)。" OpenShiftクラスタは、VMware vSphereだけでなくベアメタル方式でFlexPodにインストールされます。

複数のOpenShift (k8s) クラスタを管理するために必要なのはAstra Control Centerの1つのインスタンスだけです。各OpenShiftクラスタにはTrident CSIがインストールされています。Astra Control Centerは、このようなOpenShiftクラスタのいずれにもインストールできます。この解決策では、OpenShiftベアメタルクラスタにAstraコントロールセンターをインストールします。

次の表に、OpenShift用のFlexPod ハードウェアおよびソフトウェアのリビジョンを示します。

コンポーネント	プロダクト	バージョン
コンピューティング	Cisco UCSファブリックインターコネクタ6454	4.1 (3c)
	Cisco UCS B200 M5サーバ	4.1 (3c)
ネットワーク	Cisco Nexus 9336C-FX2 NX-OS	9.3 (8)
ストレージ	NetApp AFF A700	9.11.1
	ネットアップアストラコントロールセンター	22.04.0
	NetApp Astra Trident CSIプラグイン	22.04.0
	NetApp Active IQ Unified Managerの略	9.11
ソフトウェア	VMware ESXi nenic イーサネットドライバ	1.0.35.0
	vSphere ESXiの場合	7.0 (U2)
	VMware vCenter Applianceの略	7.0 U2b
	Cisco Intersight Assist仮想アプライアンス	1.0.9-342
	OpenShift Container Platform	4.9
	OpenShift Container Platform マスターノード	RHCOS 4.9
	OpenShift Container Platform Workerノード	RHCOS 4.9

次の表に、AWS上のOpenShift用のソフトウェアバージョンを示します。

コンポーネント	プロダクト	バージョン
コンピューティング	マスターインスタンスタイプ : m5.xlarge	該当なし
	ワーカーインスタンスタイプ : m5.large	該当なし
ネットワーク	Virtual Private Cloud Transit Gatewayの略	該当なし
ストレージ	NetApp Cloud Volumes ONTAP の略	9.11.1
	NetApp Astra Trident CSIプラグイン	22.04.0
ソフトウェア	OpenShift Container Platform	4.9
	OpenShift Container Platform マスターノード	RHCOS 4.9

コンポーネント	プロダクト	バージョン
	OpenShift Container Platform Workerノード	RHCOS 4.9

"次の例：FlexPod for OpenShift Container Platform 4ベアメタルインストール"

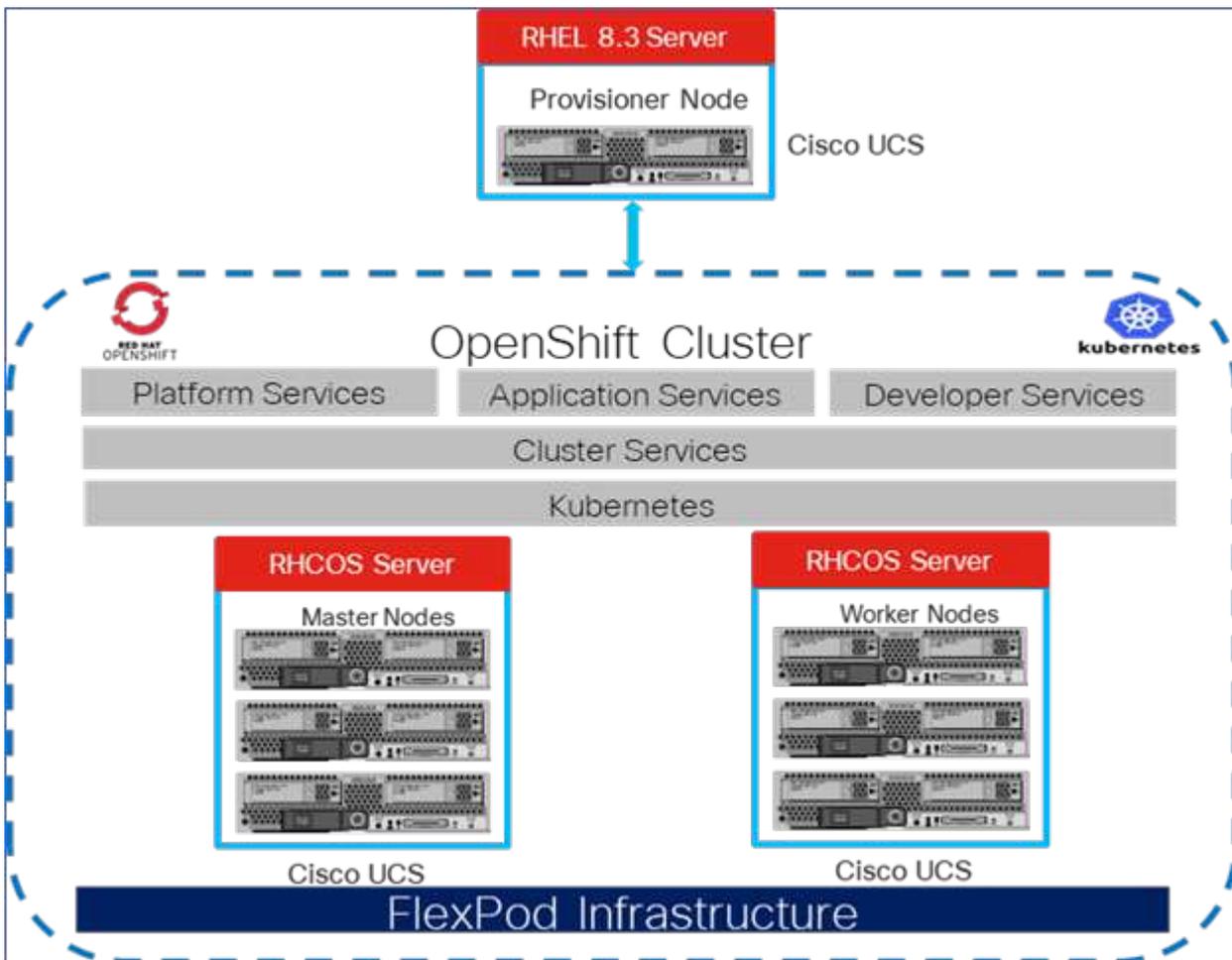
インストールと設定

FlexPod for OpenShift Container Platform 4ベアメタルインストール

"前の図：解決策 コンポーネント。"

FlexPod for OpenShift Container Platform 4のベアメタル設計、導入の詳細、およびNetApp Astra Tridentのインストールと設定については、を参照してください "[FlexPod with OpenShift Cisco Validated Design and Deploymentガイド \(CVD\)](#) "。このCVDでは、Ansibleを使用したFlexPod およびOpenShift Container Platformの導入について説明します。CVDには、ワーカーノード、Astra Tridentインストール、ストレージバックエンド、ストレージクラス構成の準備に関する詳細も記載されています。この構成は、Astra Control Centerの導入と構成を行うためのいくつかの前提条件です。

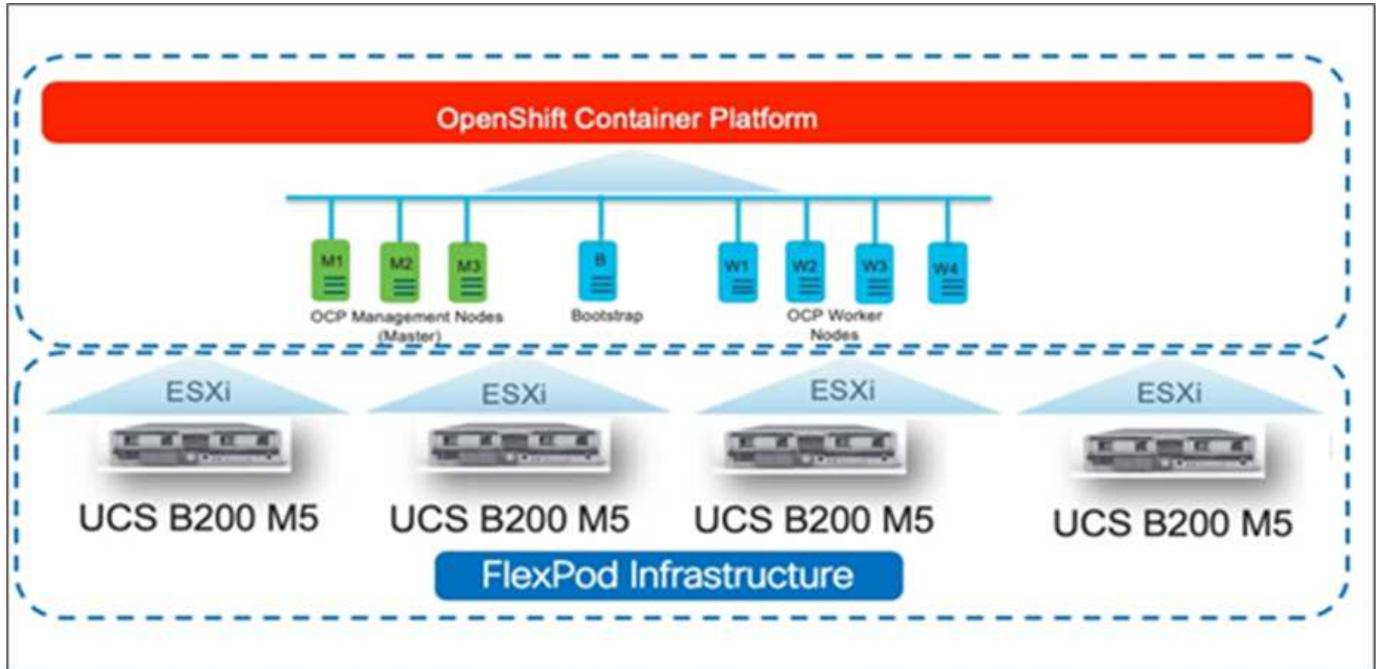
次の図は、FlexPod 上のOpenShift Container Platform 4ベアメタルを示しています。



VMware環境に実装されたOpenShift Container Platform 4用FlexPod

VMware vSphereを実行しているFlexPod にRed Hat OpenShift Container Platform 4を導入する方法については、を参照してください "[OpenShift Container Platform 4のFlexPod データセンター](#)".

次の図は、vSphere上のOpenShift Container Platform 4のFlexPod を示しています。



"次の例は、AWSでRed Hat OpenShiftを実装したものです。"

AWSにRed Hat OpenShiftを実装しました

"従来：FlexPod for OpenShift Container Platform 4ベアメタルインストール"

DRサイトとしてAWSに実装された、独立した自己管理OpenShift Container Platform 4クラスターです。マスターノードとワーカーノードは、3つのアベイラビリティゾーンにまたがって配置されるため、高可用性が実現します。

Instances (6) Info							
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
```

OpenShiftはとして導入されます **"プライベートクラスタ"** AWS上の既存のVPCに接続できます。プライベートOpenShift Container Platformクラスタは外部エンドポイントを公開しないため、内部ネットワークからのみアクセスでき、インターネットには表示されません。シングルノードのNetApp Cloud Volumes ONTAPは、NetApp Cloud Managerを使用して導入されます。これにより、TridentからAstraにバックエンドとしてストレージが提供されます。

AWSへのOpenShiftのインストールの詳細については、を参照してください **"OpenShiftのドキュメント"**。

"次のステップ：NetApp Cloud Volumes ONTAP"

NetApp Cloud Volumes ONTAP の略

"以前は、AWSでRed Hat OpenShiftを利用していました。"

NetApp Cloud Volumes ONTAP インスタンスはAWSに導入され、Astra Tridentのバックエンドストレージとして機能します。Cloud Volumes ONTAP 作業環境を追加する前に、コネクタを配置する必要があります。コネクタを配置せずにCloud Volumes ONTAP の最初の作業環境を作成するかどうかを確認するメッセージが表示されます。AWSにコネクタを導入するには、を参照してください **"コネクタを作成します"**。

AWSにCloud Volumes ONTAP を導入する手順については、を参照してください **"AWSでのクイックスタート"**。

Cloud Volumes ONTAP を導入したら、Astra Tridentをインストールし、OpenShift Container Platformクラスタでストレージバックエンドとスナップショットクラスを設定できます。

"次は、OpenShift Container PlatformにAstra Control Centerをインストールする方法です。"

OpenShift Container PlatformにAstra Control Centerをインストールします

"Previous：NetApp Cloud Volumes ONTAP の略。"

FlexPod で実行されているOpenShiftクラスタ、またはCloud Volumes ONTAP ストレージバックエンドを使用するAWSにAstraコントロールセンターをインストールできます。この解決策では、OpenShiftベアメタルクラスタにAstraコントロールセンターを導入します。

Astra Control Centerは、説明されている標準的なプロセスを使用してインストールできます **"こちらをご覧ください"** または、Red Hat OpenShift OperatorHubから入手してください。Astra Control Operatorは、Red Hat

認定オペレータです。この解決策では、AstraコントロールセンターはRed Hat OperatorHubを使用してインストールされます。

環境要件

- Astra Control Centerは複数のKubernetesディストリビューションをサポートします。Red Hat OpenShiftでは、Red Hat OpenShift Container Platform 4.8または4.9がサポートされます。
- Astra Control Centerでは、環境およびエンドユーザーのアプリケーションリソース要件に加えて、次のリソースが必要です。

コンポーネント	要件
ストレージバックエンドの容量	500GB以上の容量があります
ワーカーノード	少なくとも3つのワーカーノードがあり、それぞれ4つのCPUコアと12GBのRAMが搭載されています
Fully Qualified Domain Name (FQDN；完全修飾ドメイン名) アドレス	Astra Control Center の FQDN アドレス
Astra Trident	Astra Trident 21.04 以降がインストールおよび設定されている
入力コントローラまたはロードバランサ	入力コントローラでURLまたはロードバランサを使用してAstra Control Centerを公開し、FQDNに解決されるIPアドレスを提供するように設定します

- 既存のプライベートイメージレジストリが必要です。このレジストリには、Astra Control Centerビルドイメージをプッシュできます。イメージをアップロードするイメージレジストリのURLを指定する必要があります。



一部のイメージは特定のワークフローの実行中にプルされ、必要に応じてコンテナが作成および破棄されます。

- Astra Control Center を使用するには、ストレージクラスを作成してデフォルトのストレージクラスとして設定する必要があります。Astra Control Center は、Astra Trident が提供する次の ONTAP ドライバをサポートしています。
 - ONTAP - NAS
 - ONTAP-NAS-flexgroup
 - ONTAP - SAN
 - ONTAP - SAN - 経済性



導入したOpenShiftクラスタにAstra Tridentがインストールされ、ONTAP バックエンドで設定されているとします。また、デフォルトのストレージクラスも定義されています。

- OpenShift環境でアプリケーションクローニングを行う場合、Astra Control CenterはOpenShiftでボリュームをマウントし、ファイルの所有権を変更できるようにする必要があります。これらの処理を許可するようにONTAP エクスポートポリシーを変更するには、次のコマンドを実行します。

```
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
```



管理対象のコンピューティングリソースとして2つ目のOpenShift運用環境を追加するには、Astra Tridentボリュームスナップショット機能が有効になっていることを確認します。Tridentを使用してボリュームSnapshotを有効にし、テストする方法については、を参照してください "[Astra Tridentの手順](#)"。

- A "[VolumeSnapClass](#)" アプリケーションの管理元であるすべてのKubernetesクラスタで設定する必要があります。Astra Control CenterがインストールされているKubernetesクラスタも含めることができます。Astra Control Centerでは、実行中のKubernetesクラスタ上のアプリケーションを管理できます。

アプリケーション管理の要件

- **ライセンス。** Astra Control Centerを使用してアプリケーションを管理するには、Astra Control Centerライセンスが必要です。
- ***名前空間。** *名前空間は、Astra Control Centerによってアプリケーションとして管理できる最大のエンティティです。既存のネームスペース内のアプリケーションラベルとカスタムラベルに基づいてコンポーネントを除外し、リソースのサブセットをアプリケーションとして管理できます。
- *** StorageClass。** * StorageClassが明示的に設定されたアプリケーションをインストールし、アプリケーションのクローンを作成する必要がある場合、クローン処理のターゲットクラスタに最初に指定されたStorageClassが必要です。明示的にStorageClassを設定したアプリケーションを、同じストレージクラスを持たないクラスタにクローニングすると失敗します。
- *** Kubernetesのリソース。** * Astra ControlではキャプチャされないKubernetesリソースを使用するアプリケーションには、アプリケーションデータの完全な管理機能が備わっていない可能性があります。Astra Controlでは、次のKubernetesリソースをキャプチャできます。

Kubernetesのリソース		
クラスタロール	ClusterRoleBinding	ConfigMap
CustomResourceDefinition の場合	CustomResource の場合	cronjob
デモンセット (DemonSet)	HorizontalPodAutoscaler のように表示されます	入力
DeploymentConfig	MutingWebhook	PersistentVolumeClaim のように表示され
ポッド	PodDisruptionBudget (予算の廃止)	PodTemplate
ネットワークポリシー	ReplicaSet	ロール
RoleBinding です	ルート	秘密
検証 Webhook		

OpenShift OperatorHub を使用して Astra Control Center をインストールします

次の手順は、Red Hat OperatorHubを使用してAstraコントロールセンターをインストールします。この解決策では、FlexPod上で動作するベアメタルOpenShiftクラスターにAstraコントロールセンターをインストールします。

1. から Astra Control Center バンドル（「Astra - control-center-[version].tar.gz」）をダウンロードします ["NetApp Support Site"](#)。
2. からAstra Control Centerの証明書とキーの.zipファイルをダウンロードします ["NetApp Support Site"](#)。
3. バンドルの署名を確認します。

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

4. Astraの画像を抽出します。

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

5. Astra ディレクトリに移動します。

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

6. イメージをローカルレジストリに追加します。

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

7. 適切なスクリプトを使用して、イメージをロードし、イメージにタグを付け、ローカルレジストリにプッシュします。

Docker の場合：

```

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

```

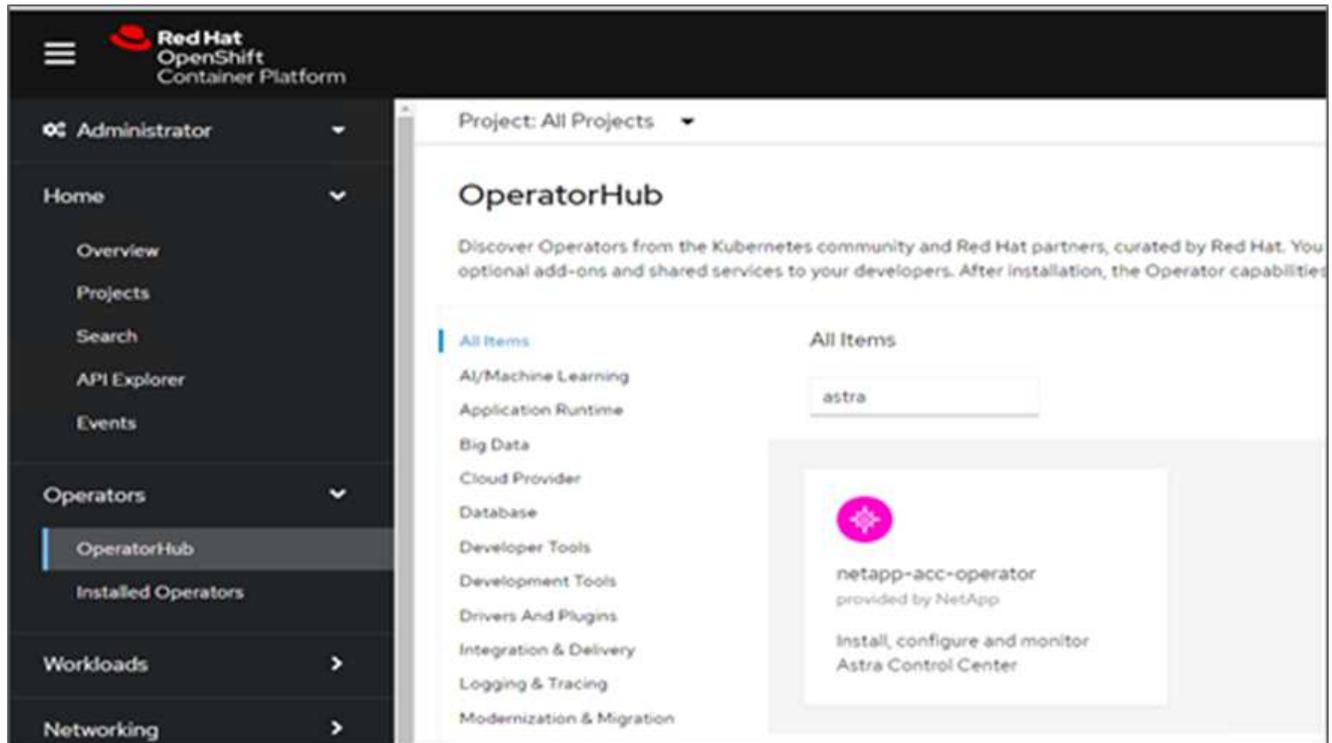
Podman の場合：

```

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

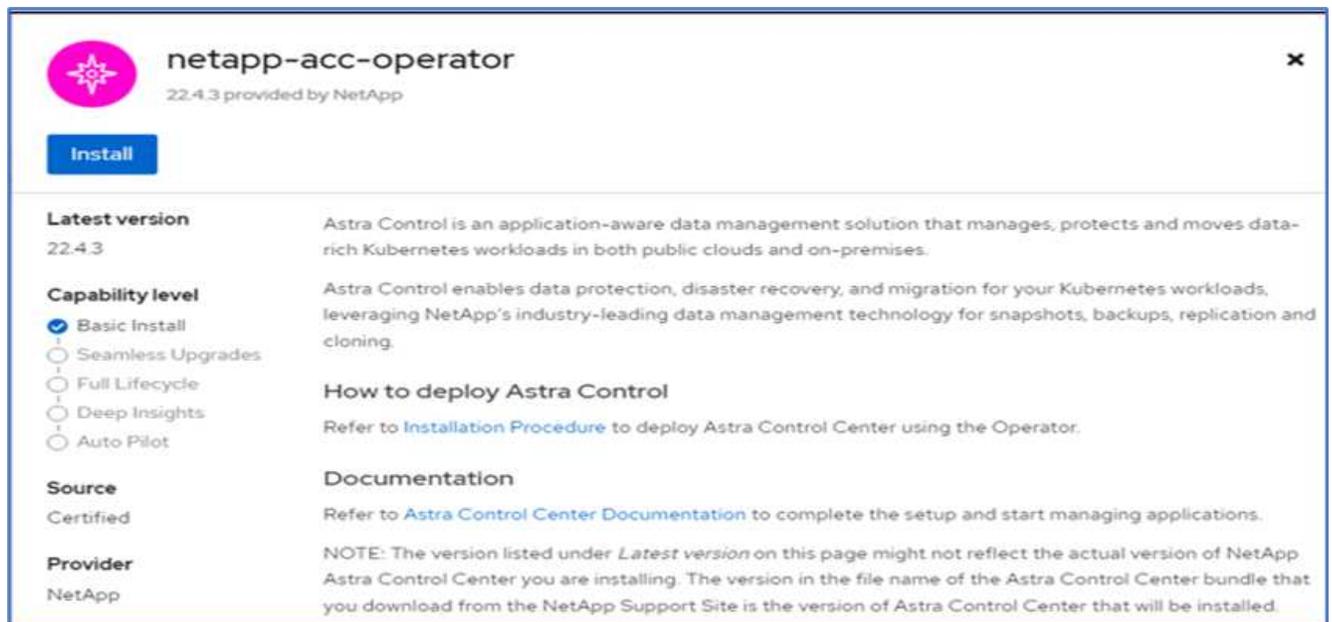
```

8. ベアメタルOpenShiftクラスタのWebコンソールにログインします。サイドメニューから、[演算子]>[演算子ハブ]を選択します。「stra」と入力して、「NetApp-acc-operator」のリストを表示します。

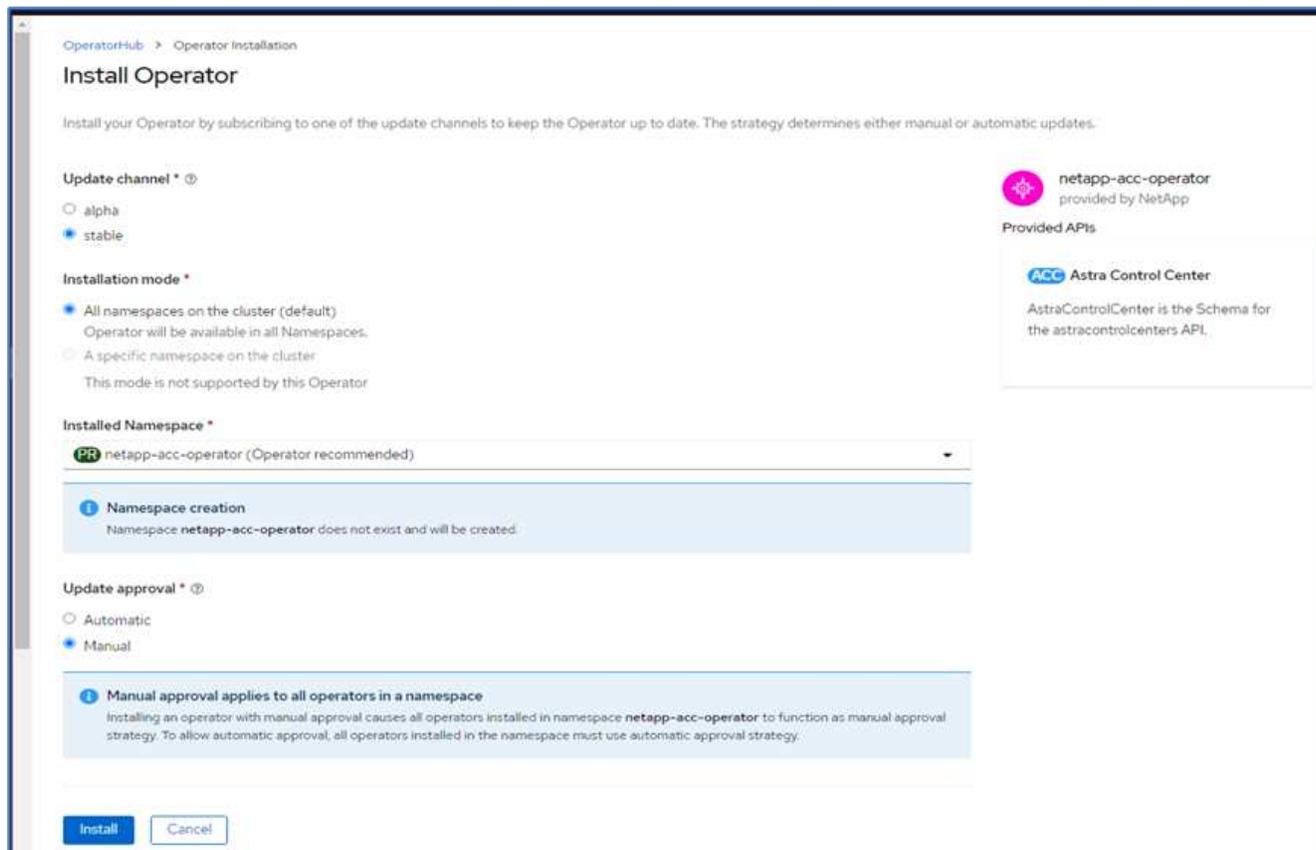


「NetApp-acc-operator」は、Red Hat OpenShift Operatorの認定を受けたもので、OperatorHubカタログの下にリストされています。

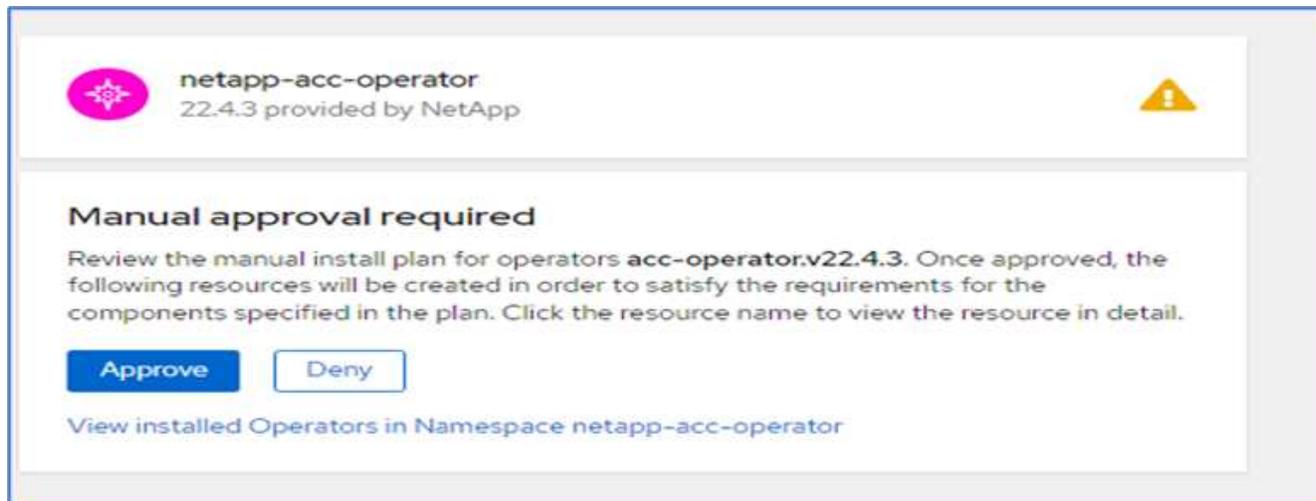
9. 「NetApp-acc-operator」を選択し、「Install」をクリックします。



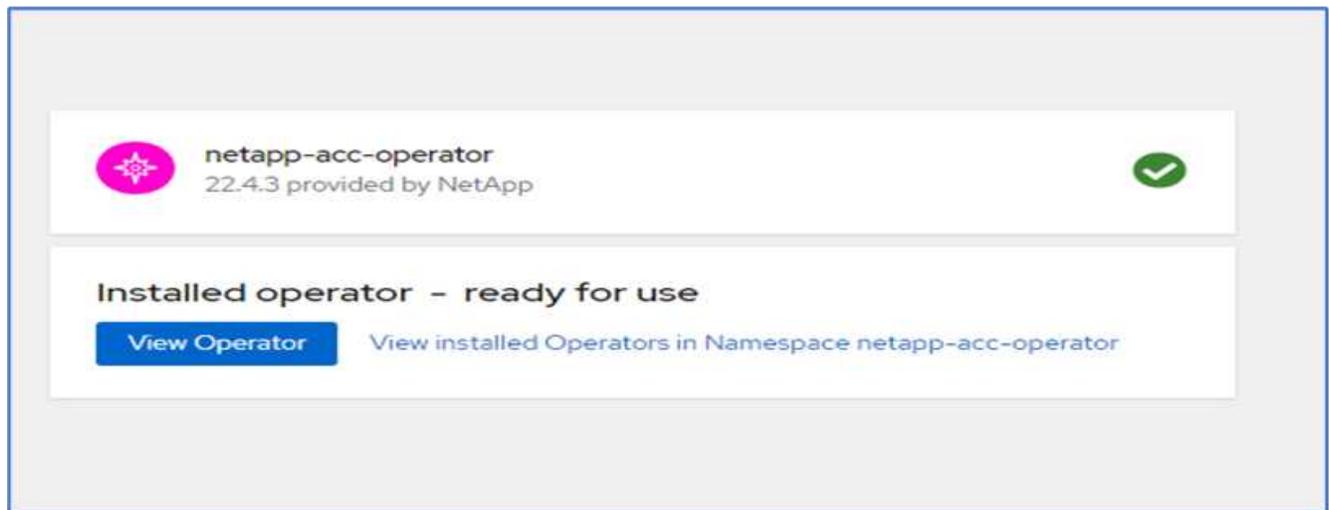
10. 適切なオプションを選択し、[インストール]をクリックします。



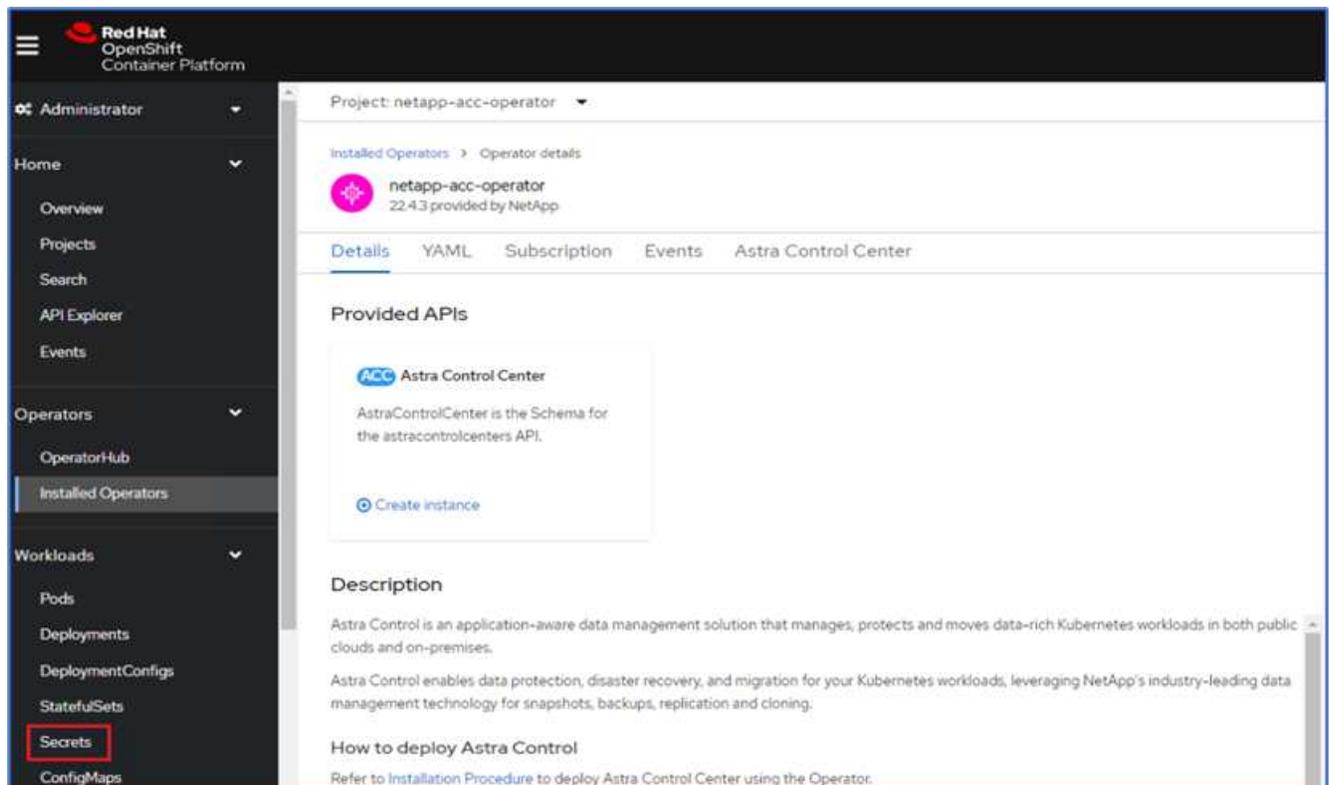
11. インストールを承認し、オペレータがインストールされるまで待ちます。



12. この段階で、オペレータは正常にインストールされ、使用可能な状態になります。View Operator（オペレータの表示）をクリックして、Astra Control Centerのインストールを開始します。



13. Astra Control Centerをインストールする前に、事前にプッシュしたDockerレジストリからAstraイメージをダウンロードするプルシークレットを作成します。



14. Astra Control CenterのイメージをDocker private repoから取得するには、NetApp-acc-operator'ネームスペースにシークレットを作成します。このシークレット名は、後の手順でAstra Control Center YAMLマニフェストに表示されます。

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

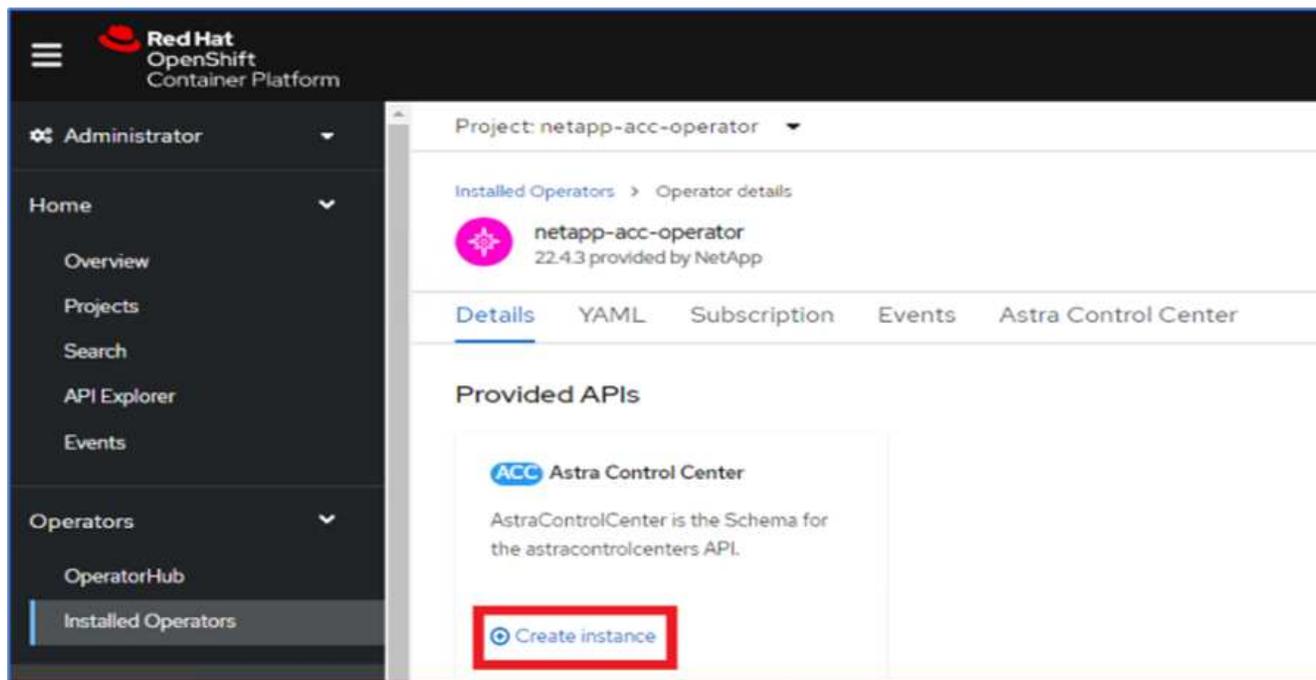
Username *

Password *

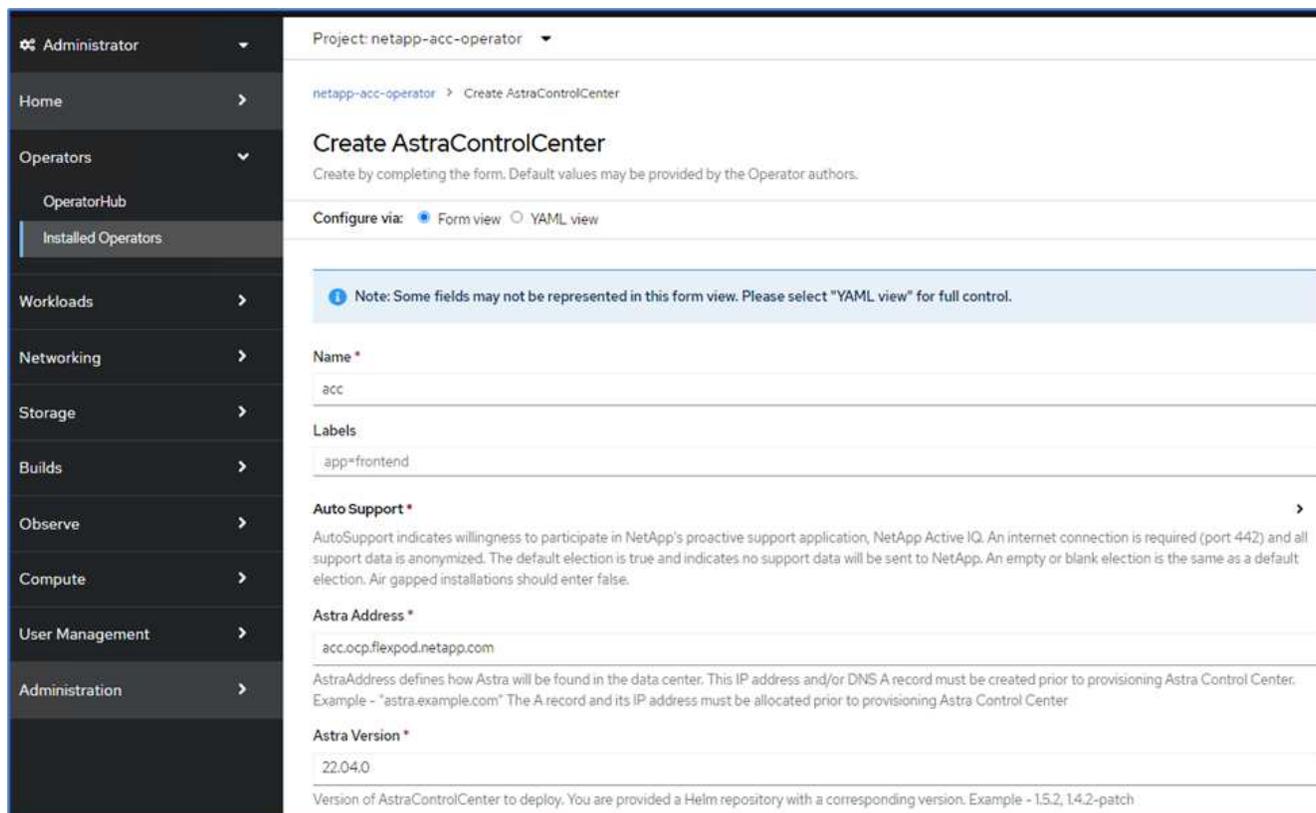
Email

[+ Add credentials](#)

15. サイドメニューから、[演算子]>[インストールされた演算子]を選択し、[提供されたAPI]セクションの下にある[インスタンスの作成]をクリックします。



16. Create AstraControlCenterフォームに入力します名前、Astraアドレス、Astraバージョンを入力します。



[Astra Address]で、Astra Control CenterのFQDNアドレスを入力します。このアドレスは、Astra Control CenterのWebコンソールにアクセスするために使用されます。FQDNは、到達可能なIPネットワークにも解決される必要があり、DNSで設定する必要があります。

17. アカウント名、Eメールアドレス、管理者の姓を入力し、デフォルトのボリューム再利用ポリシーをその

そのまま使用します。ロードバランサを使用している場合は、入力タイプを「AccTraefik」に設定します。それ以外の場合は、「Ingress Controller」で「Generic」を選択します。イメージレジストリで、コンテナイメージのレジストリパスとシークレットを入力します。

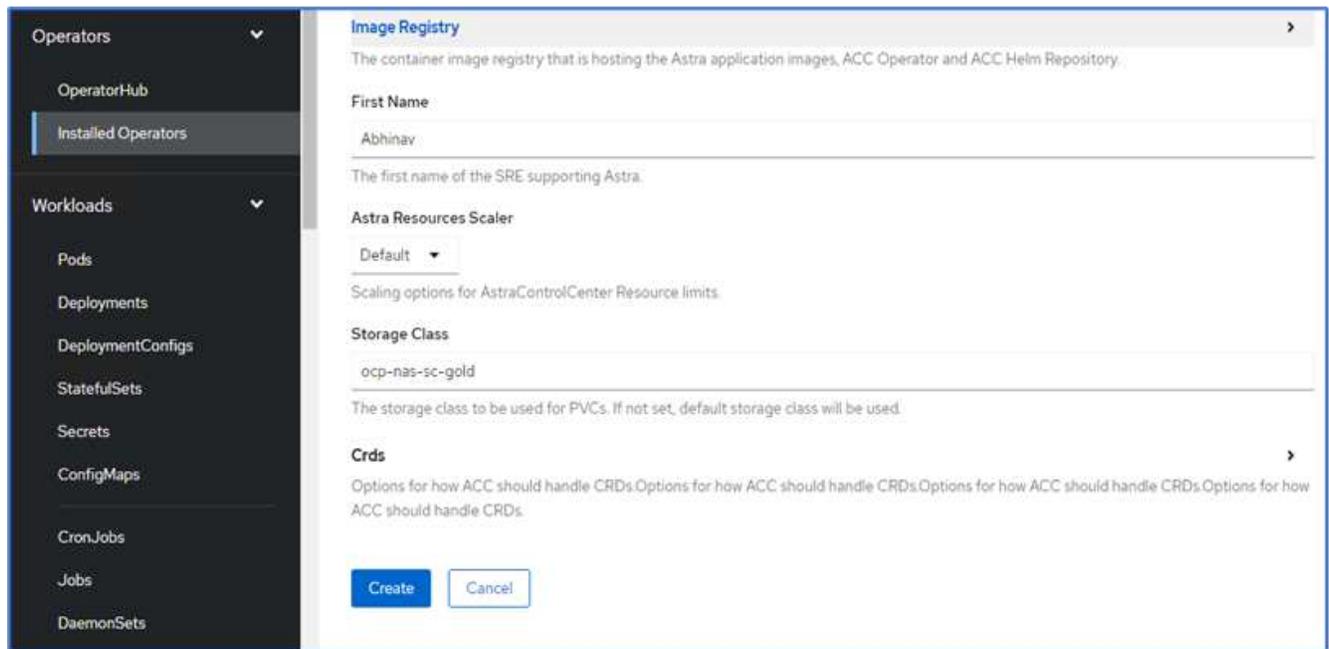
The screenshot shows the configuration page for the 'netapp-acc-operator' project. The left sidebar contains navigation options: Administrator, Home, Operators (OperatorHub, Installed Operators), Workloads, Networking, Storage, Builds, Observe, Compute, User Management, and Administration. The main content area includes the following fields:

- 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**: [Redacted] (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.)

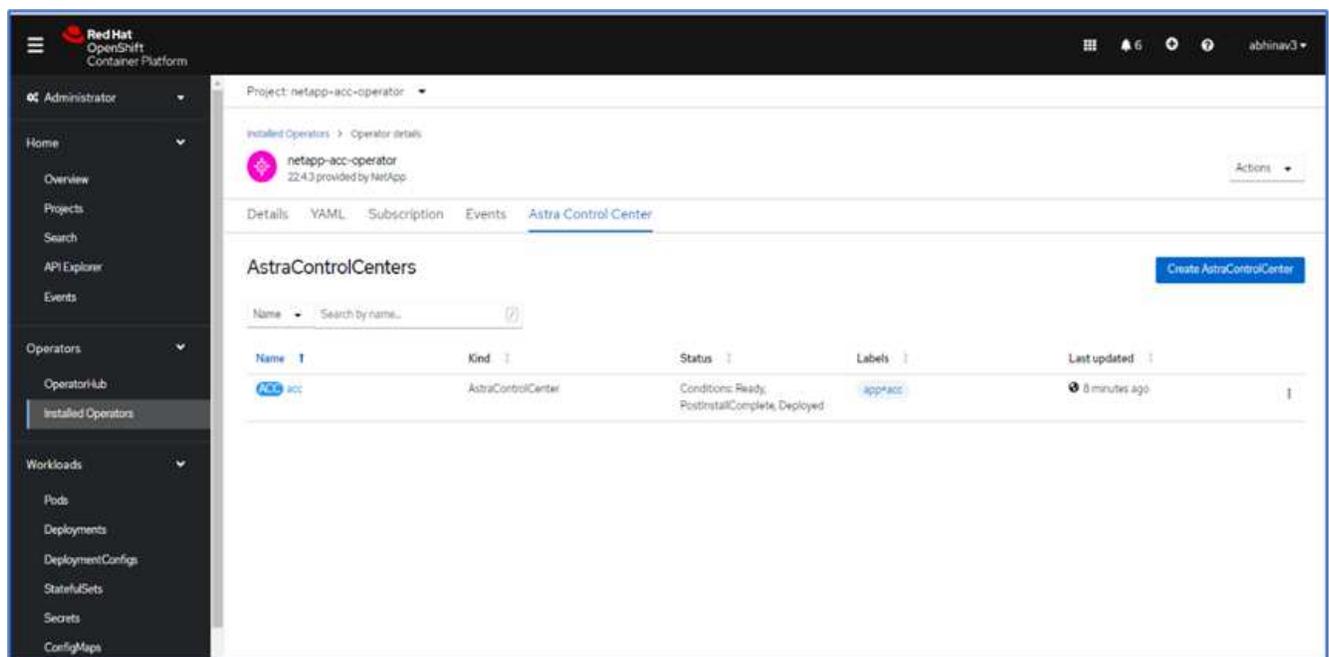


この解決策では、MetalLBロードバランサが使用されます。したがって、入力タイプはAccTraefikです。これにより、Astra Control Center traefikゲートウェイが、LoadBalancerタイプのKubernetesサービスとして公開されます。

18. 管理者の名を入力し、リソースの拡張を設定して、ストレージクラスを指定します。Create をクリックします。



Astra Control Centerインスタンスのステータスは、[Deploying]から[Ready]に変わります。



- すべてのシステムコンポーネントが正常にインストールされ、すべてのポッドが実行されていることを確認します。

```

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 6m4s	1/1	Running	0
api-token-authentication-4q5lj 5m26s	1/1	Running	0
api-token-authentication-pzptd 5m27s	1/1	Running	0
api-token-authentication-tbtg6 5m27s	1/1	Running	0
asup-669df8d49-qps54 5m26s	1/1	Running	0
authentication-5867c5f56f-dnpp2 3m54s	1/1	Running	0
bucket-service-85495bc475-5zcc5 5m55s	1/1	Running	0
cert-manager-67f486bbc6-txhh6 9m5s	1/1	Running	0
cert-manager-cainjector-75959db744-4l5p5 9m6s	1/1	Running	0
cert-manager-webhook-765556b869-g6wdf 9m6s	1/1	Running	0
cloud-extension-5d595f85f-txrfl 5m27s	1/1	Running	0
cloud-insights-service-674649567b-5s4wd 5m49s	1/1	Running	0
composite-compute-6b58d48c69-46vhc 6m11s	1/1	Running	0
composite-volume-6d447fd959-chnrt 5m27s	1/1	Running	0
credentials-66668f8ddd-8qc5b 7m20s	1/1	Running	0
entitlement-fd6fc5c58-wxnmh 6m20s	1/1	Running	0
features-756bbb7c7c-rgcrm 5m26s	1/1	Running	0
fluent-bit-ds-278pg 3m35s	1/1	Running	0
fluent-bit-ds-5pqc6 3m35s	1/1	Running	0
fluent-bit-ds-8l7cq 3m35s	1/1	Running	0
fluent-bit-ds-9qbft 3m35s	1/1	Running	0
fluent-bit-ds-nj475 3m35s	1/1	Running	0
fluent-bit-ds-x9pd8 3m35s	1/1	Running	0

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 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-vcb4j 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

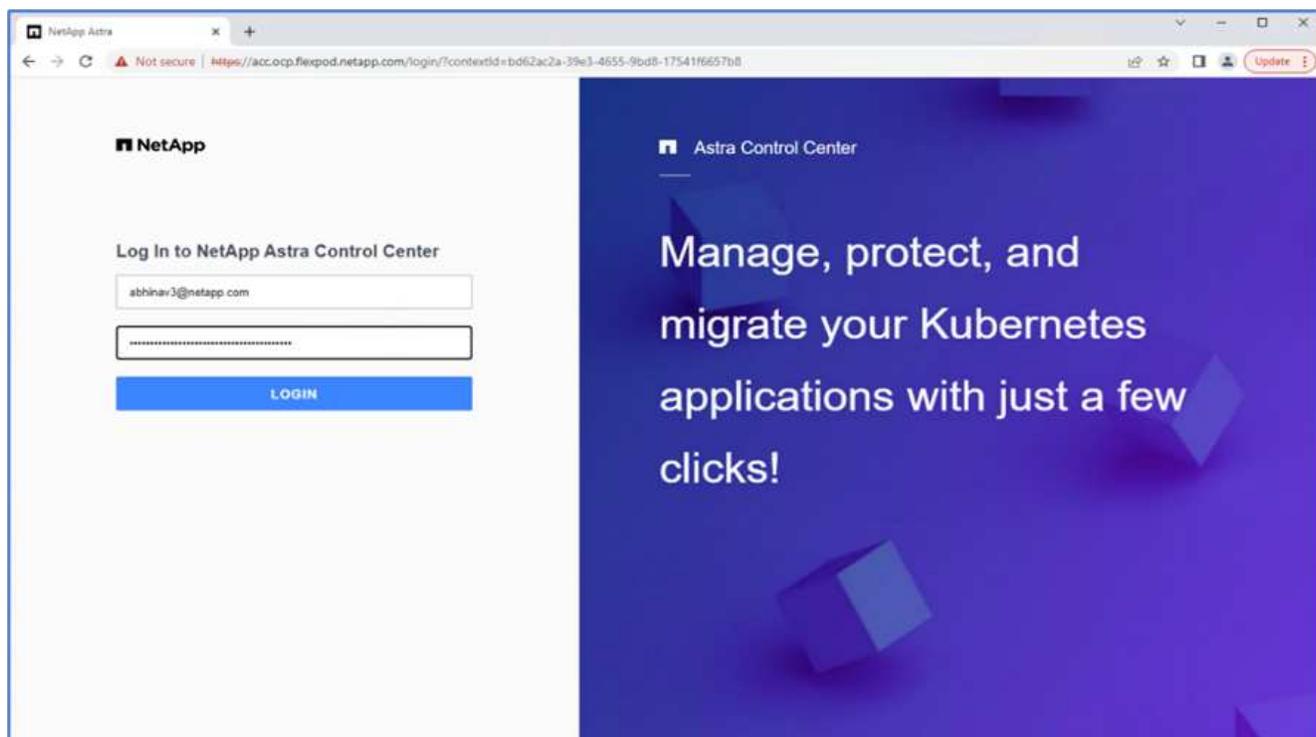


各ポッドのステータスが「Running」である必要があります。システムのポッドが導入されるまでに数分かかることがあります。

- すべてのポッドが実行中の場合は、次のコマンドを実行して1回限りのパスワードを取得します。出力のYAMLバージョンで、「status.deploymentState」フィールドで展開された値を確認し、「status.uuid」値をコピーします。パスワードは「ACC-」で、その後にUUID値が続きます。（ACC-[UUID]）。

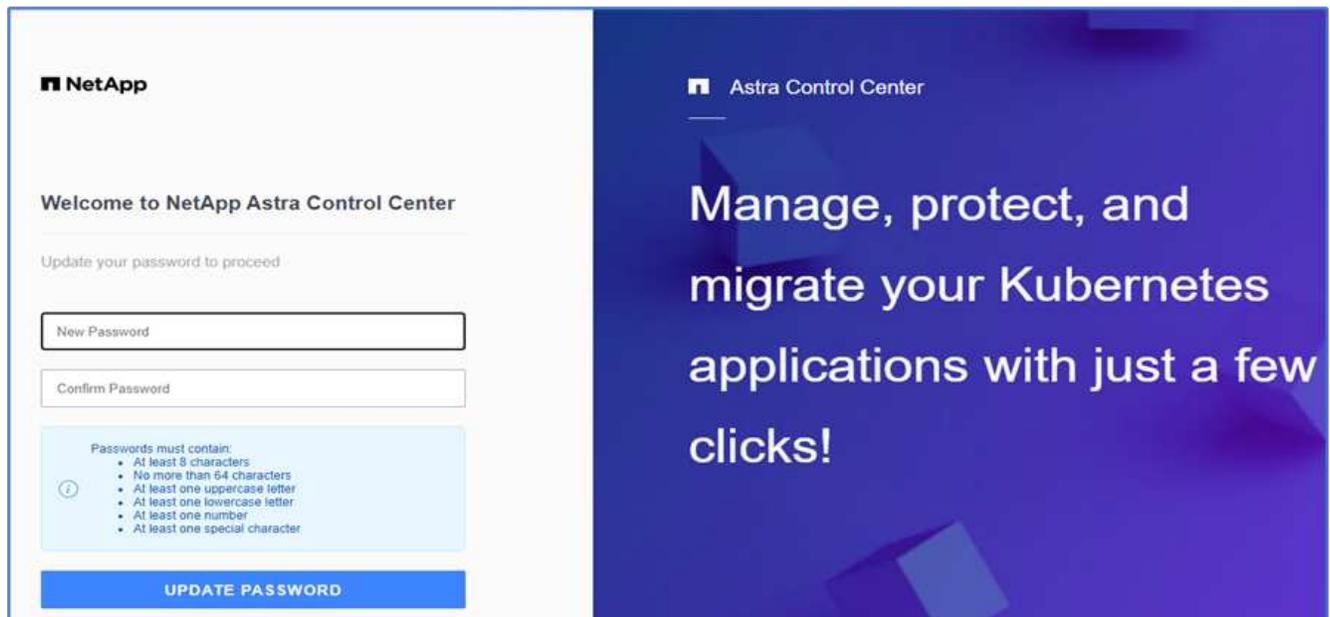
```
root@abhinav-ansible# oc get acc -o yaml -n netapp-acc-operator
```

- ブラウザで、指定したFQDNを使用してURLに移動します。
- デフォルトのユーザ名（インストール時に指定したEメールアドレス）とワンタイムパスワードACC-[UUID]を使用してログインします。



誤ったパスワードを3回入力すると、管理者アカウントは15分間ロックされます。

- パスワードを変更して次に進みます。

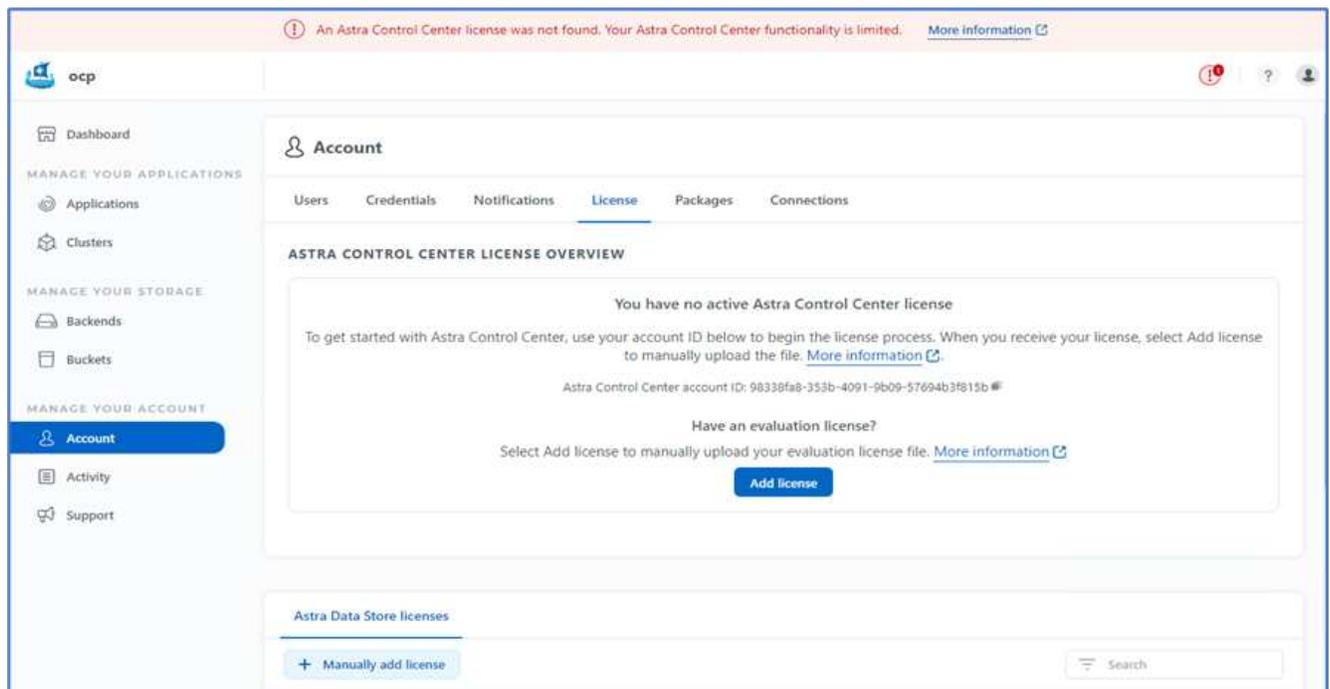


Astra Control Centerのインストールの詳細については、を参照してください "[Astra Control Centerのインストールの概要](#)" ページ

Astra Control Center をセットアップします

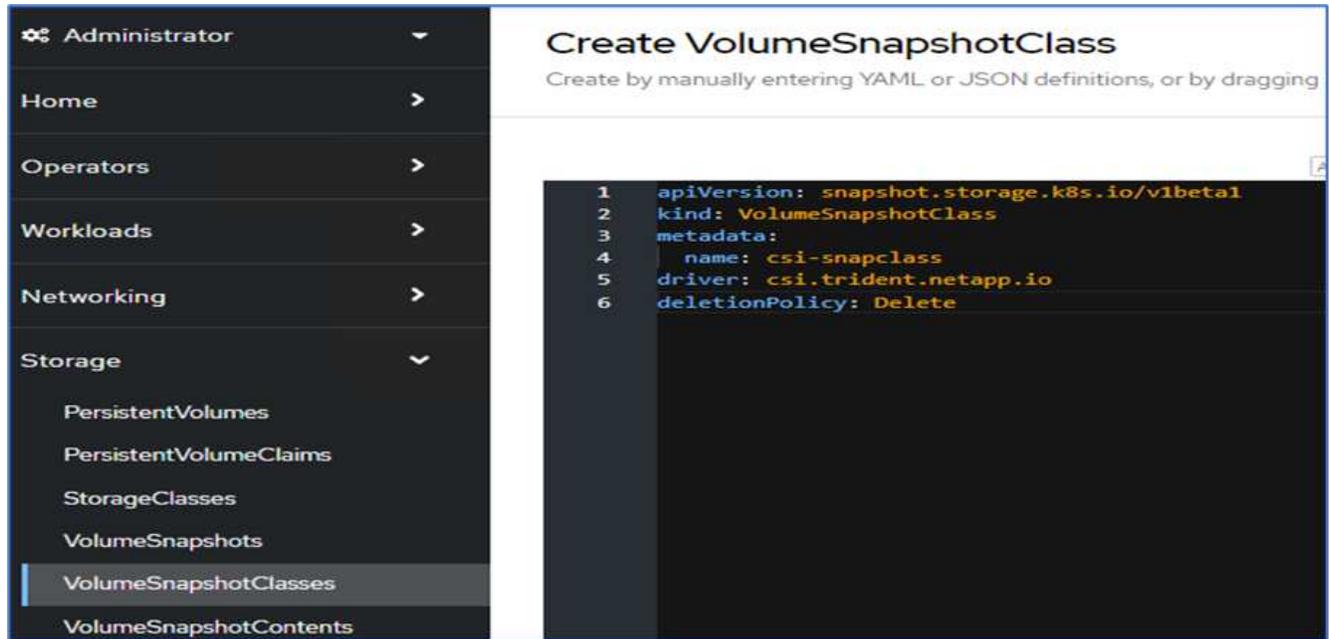
Astra Control Centerをインストールしたら、UIにログインし、ライセンスのアップロード、クラスタの追加、ストレージの管理、バケットの追加を行います。

1. [アカウント]の下のホームページで、[ライセンス]タブに移動し、[ライセンスの追加]を選択してAstraライセンスをアップロードします。

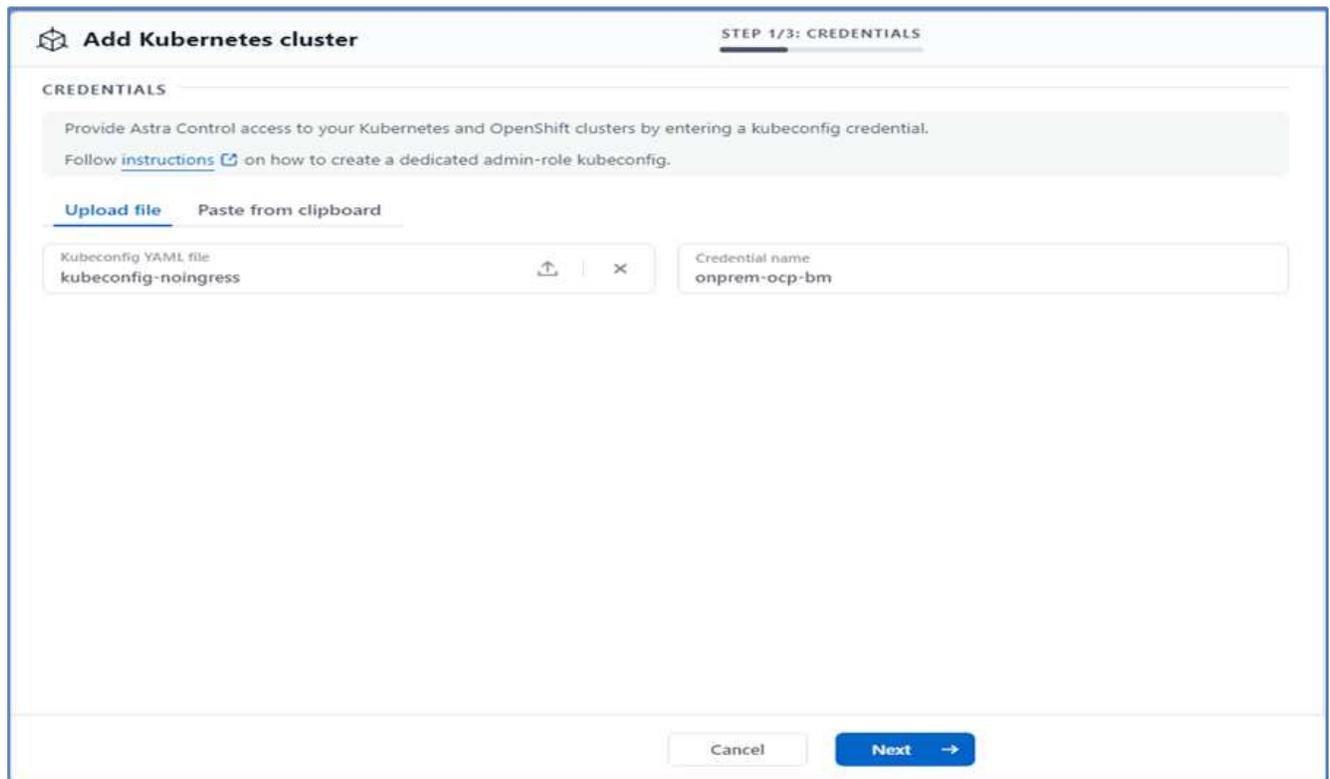


2. OpenShiftクラスタを追加する前に、OpenShift WebコンソールからAstra Tridentボリュームスナップショットクラスを作成します。Volumeスナップショット・クラスには'csi.trident.netapp.io'ドライバが設定さ

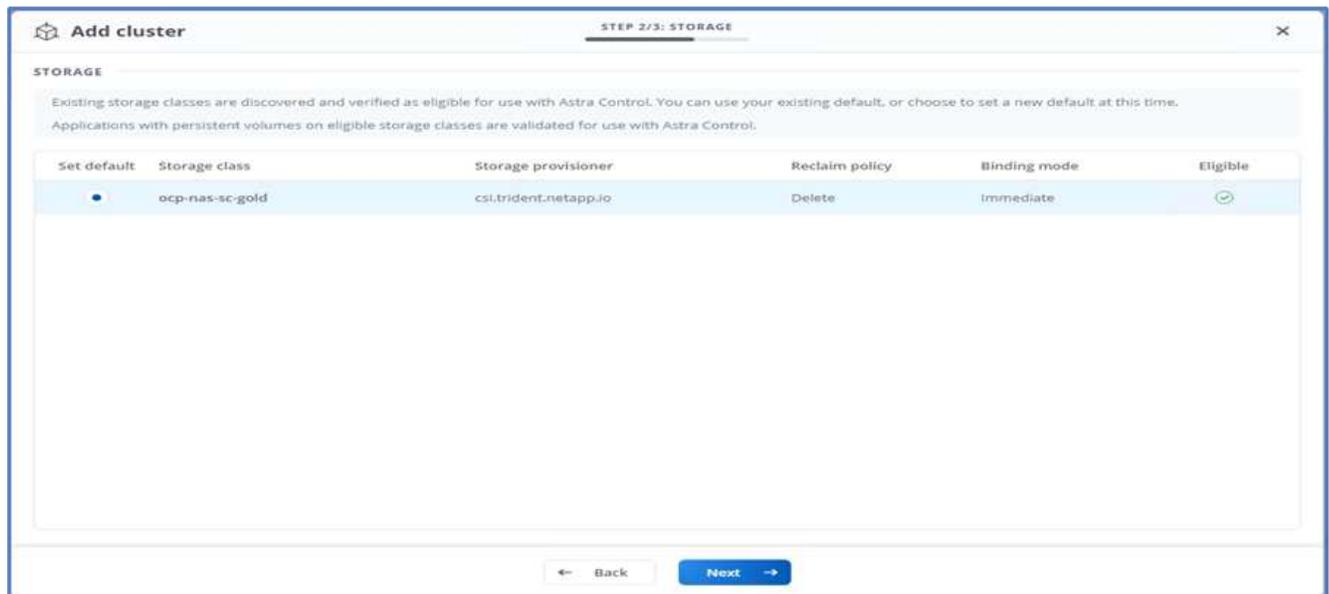
れています



3. Kubernetesクラスタを追加するには、ホームページでクラスタに移動し、Kubernetesクラスタを追加をクリックします。次に、クラスタの「kubeconfig」ファイルをアップロードし、クレデンシャル名を指定します。次へをクリックします。



4. 既存のストレージクラスは自動的に検出されます。デフォルトのストレージクラスを選択し、Next（次へ）をクリックし、Add cluster（クラスタの追加）をクリックします。

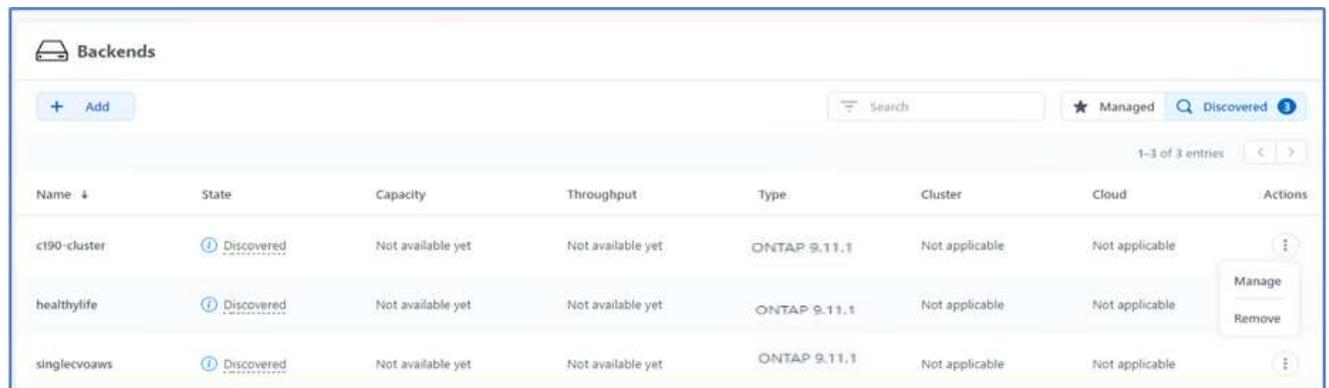


5. クラスタは数分で追加されます。OpenShift Container Platform クラスタを追加するには、手順1~4を繰り返します。



管理対象のコンピューティングリソースとしてOpenShift運用環境を追加するには、Astra Tridentを実行してください "[VolumeSnapshotClassオブジェクト](#)" が定義されている。

6. ストレージを管理するには、バックエンドに移動し、管理するバックエンドに対する処理の下にある3つのドットをクリックします。[管理]をクリックします



7. ONTAP の資格情報を入力し、[次へ]をクリックします。情報を確認し、[管理]をクリックします。バックエンドは次の例のようになります。

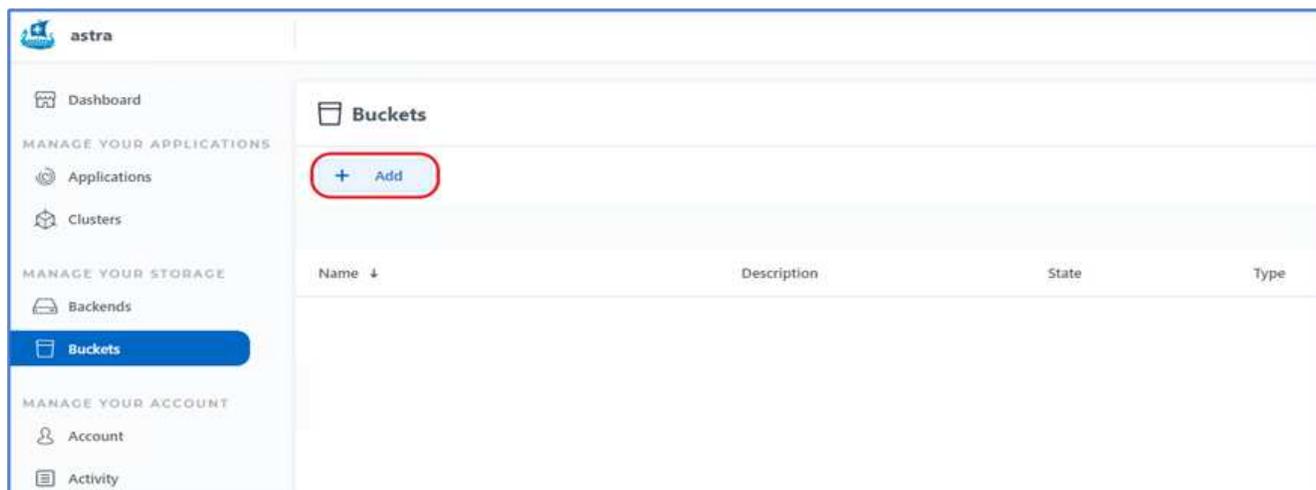
Backends

+ Add ★ Managed

1-3 of 3 entries < >

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 Controlにバケットを追加するには、バケットを選択して追加をクリックします。



9. バケットタイプを選択し、バケット名、S3サーバ名、またはIPアドレスとS3クレデンシャルを指定します。更新をクリックします。

Edit bucket

STORAGE BUCKET

Edit the access details of your existing object store bucket.

Type:

Existing bucket name:

Description (optional):

S3 server name or IP address:

Make this bucket the default bucket for this cloud

SELECT CREDENTIALS

Astra Control requires S3 access credentials with the roles necessary to facilitate Kubernetes application data management.

[Add](#) [Use existing](#)

Access ID: Secret key:

Credential name:

EDITING STORAGE BUCKETS

Edit your existing object store bucket. If the selected bucket is not currently defined as the default bucket for the cloud, you can replace the currently defined default bucket. Read more in [Storage buckets](#).



この解決策では、AWS S3バケットとONTAP S3バケットの両方が使用されま
す。StorageGRIDを使用することもできます。

バケットは正常な状態である必要があります。

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

アプリケーション対応データ管理用のAstra Control CenterへのKubernetesクラスタ登録の一部として、Astra Controlは、ロールバインドとネットアップ監視名前スペースを自動的に作成し、アプリケーションポッドとワーカーノードから指標とログを収集します。サポートされているONTAPベースのストレージクラスのいずれかをデフォルトにします。

お先にどうぞ ["Astra Control 管理にクラスタを追加"](#)では、クラスターにアプリケーションをインストールし（Astra Controlの外部）、Astra Controlの[アプリ]ページに移動して、アプリケーションとそのリソースを管理できます。Astraを使用したアプリケーションの管理の詳細については、を参照してください ["アプリケーション管理の要件"](#)。

["次：解決策の検証の概要"](#)

解決策の検証

概要

["前のレポート：OpenShift Container PlatformにAstra Control Centerをインストールしました。"](#)

このセクションでは、いくつかのユースケースで解決策を復習します。

- リモートバックアップから、クラウドで実行されている別のOpenShiftクラスタへのステートフルアプリケーションのリストア。
- OpenShiftクラスタ内の同じ名前スペースへのステートフルアプリケーションのリストア。
- あるFlexPod システム（OpenShift Container Platformベアメタル）から別のFlexPod システム（VMware上のOpenShift Container Platform）にクローニングすることでアプリケーションを移動できます。

特に、この解決策で検証されるのはユースケースが少ないことがわかります。この検証は、Astra Control Centerの全機能を表しているわけではありません。

["Next：リモートバックアップを使用したアプリケーションのリカバリ。"](#)

リモートバックアップによるアプリケーションのリカバリ

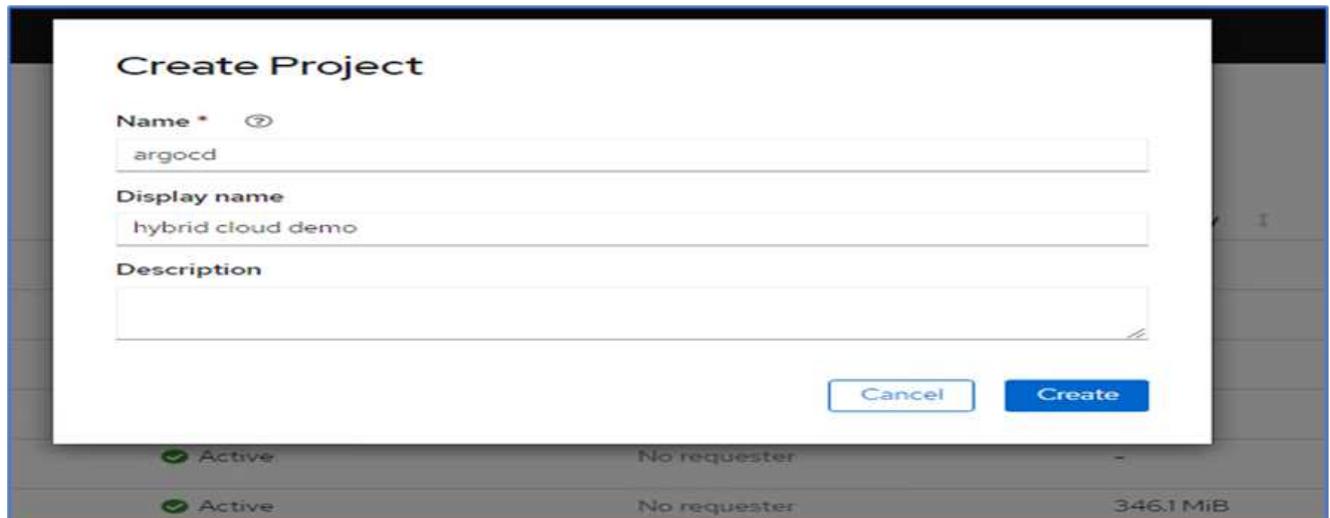
"Previous: 解決策 の検証の概要を示します。"

Astraでは、アプリケーションと整合性のあるフルバックアップを作成できます。このバックアップを使用すると、アプリケーションのデータを使用して、オンプレミスのデータセンターやパブリッククラウドで実行されている別のKubernetesクラスタにリストアできます。

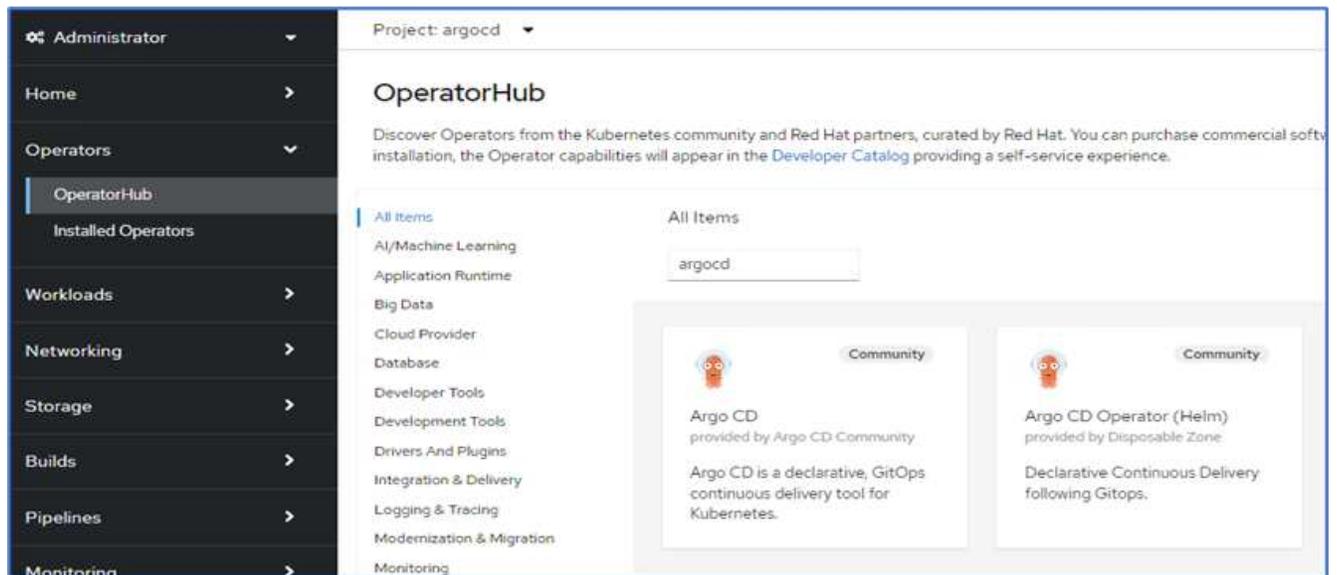
アプリケーションのリカバリが正常に行われるかどうかを検証するには、FlexPod システムで実行されているアプリケーションのオンプレミス障害をシミュレートし、リモートバックアップを使用してクラウドで実行されているKubernetesクラスタにアプリケーションをリストアします。

サンプルアプリケーションは、データベースにMySQLを使用する価格表アプリケーションです。導入を自動化するために、を使用しました "Argo CD" ツール。Argo CDは、Kubernetes向けの宣言型、GitOps、継続的デリバリーツールです。

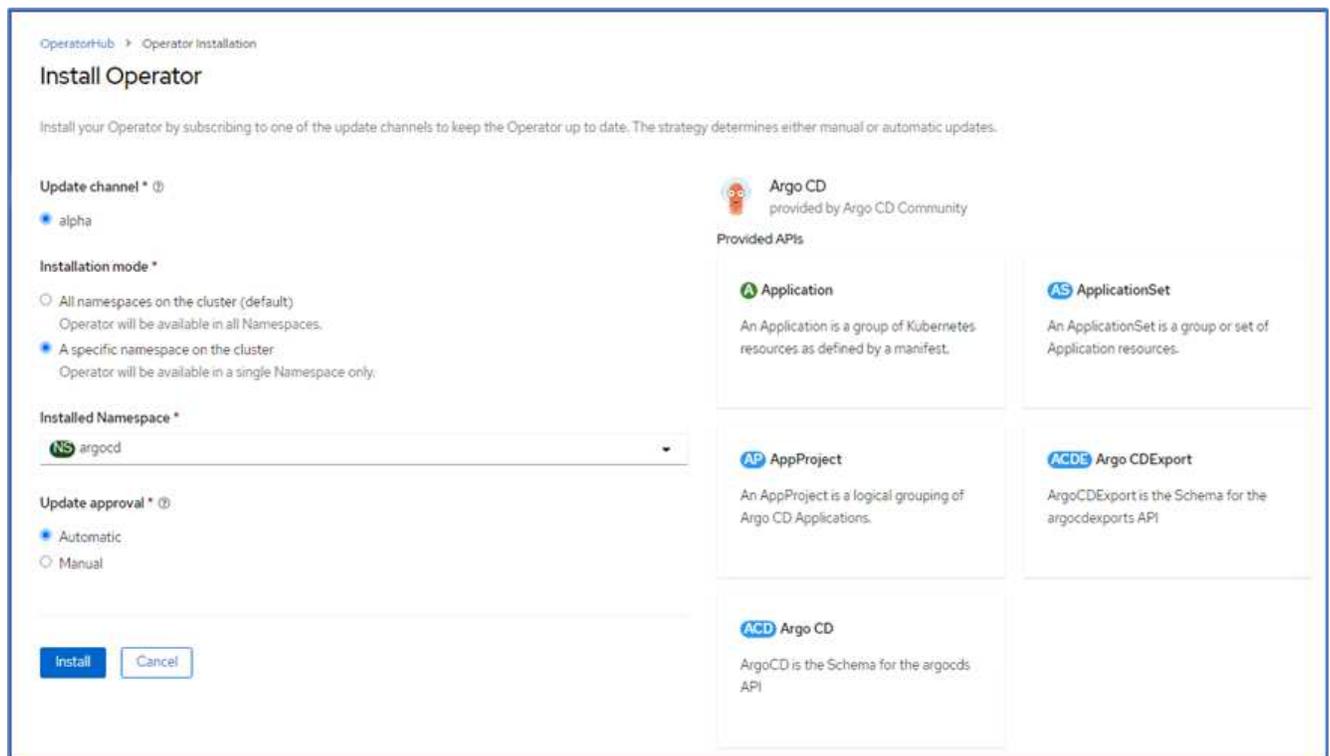
1. オンプレミスOpenShiftクラスタにログインし、「argocd」という名前の新しいプロジェクトを作成します。



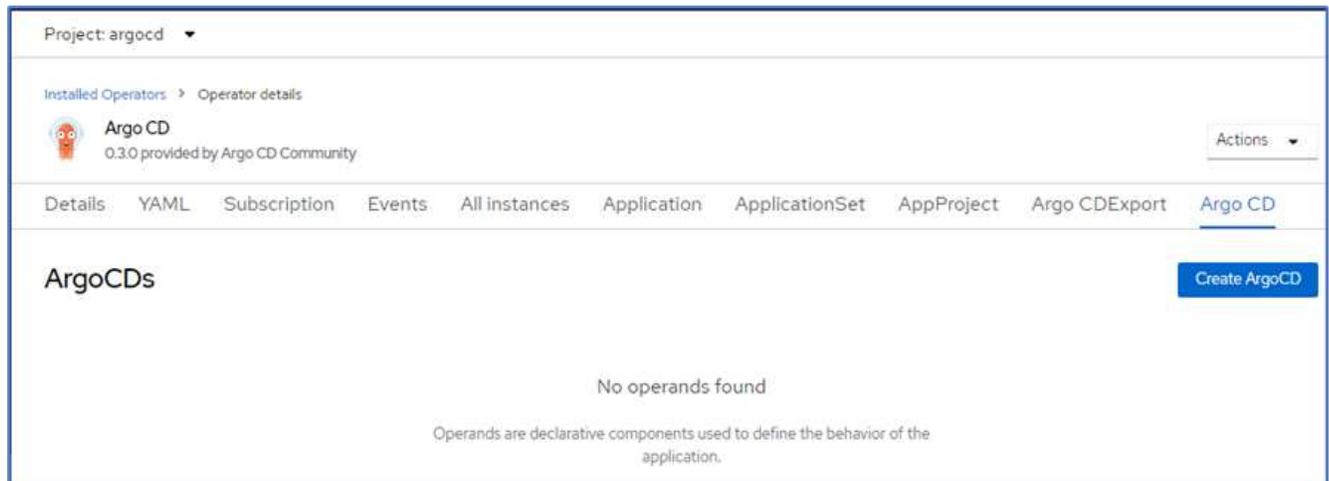
2. OperatorHubで'argocd'を検索し'Argo CD operator'を選択します



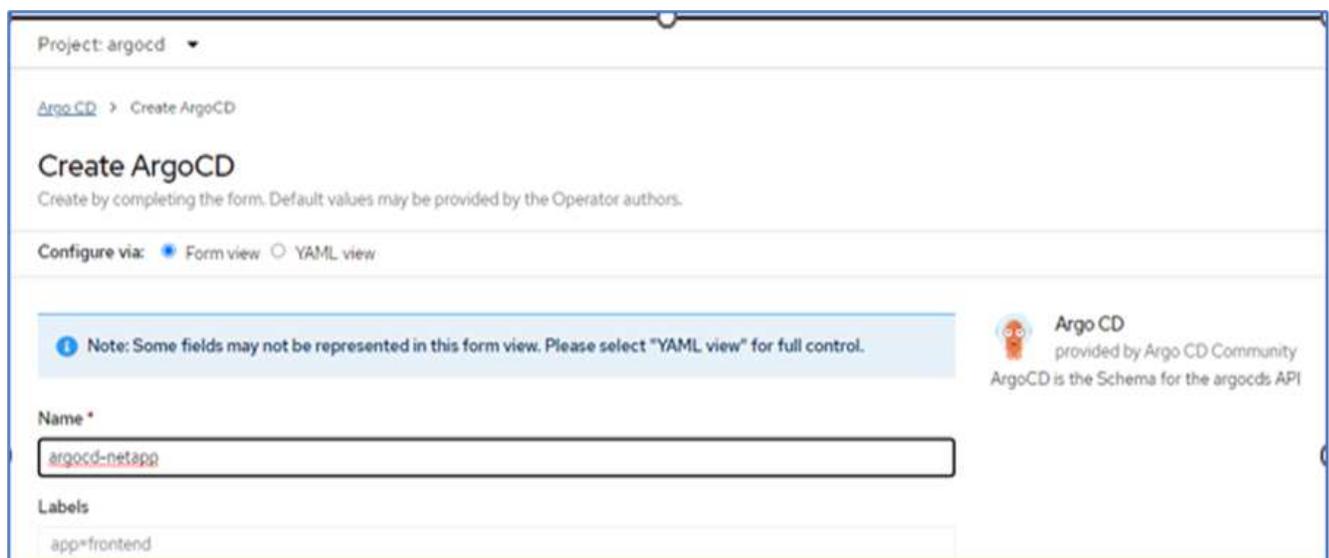
3. 「argocd」名前空間に演算子をインストールします。



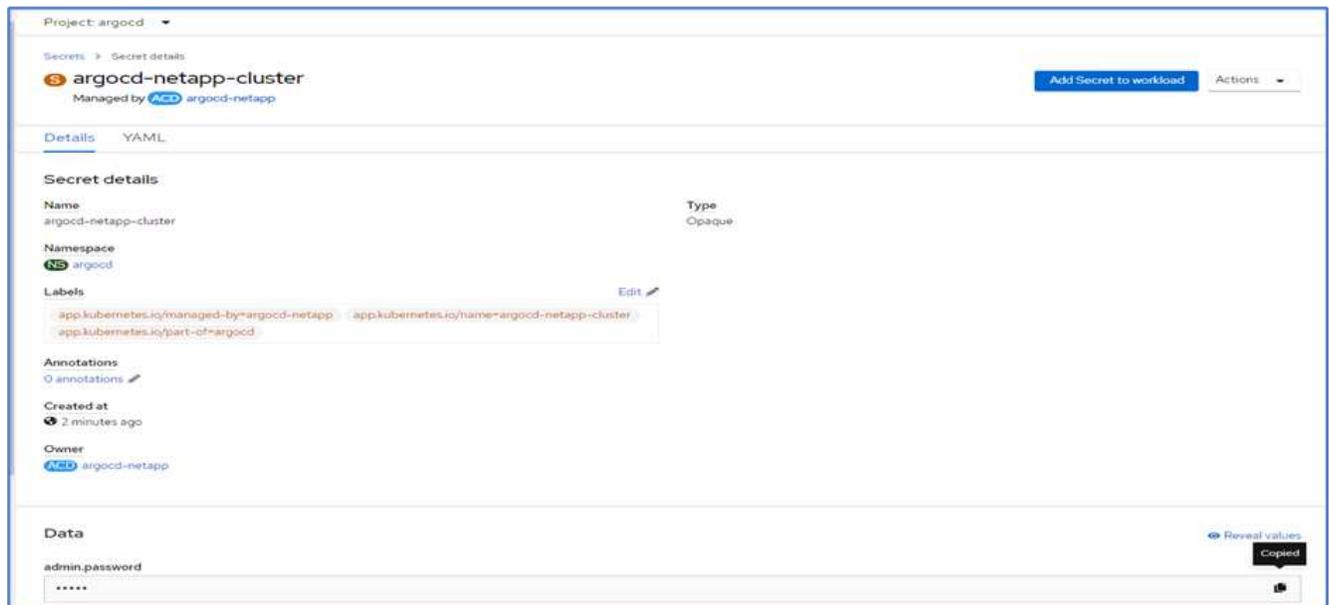
4. オペレータに移動し、Create ArgCDをクリックします。



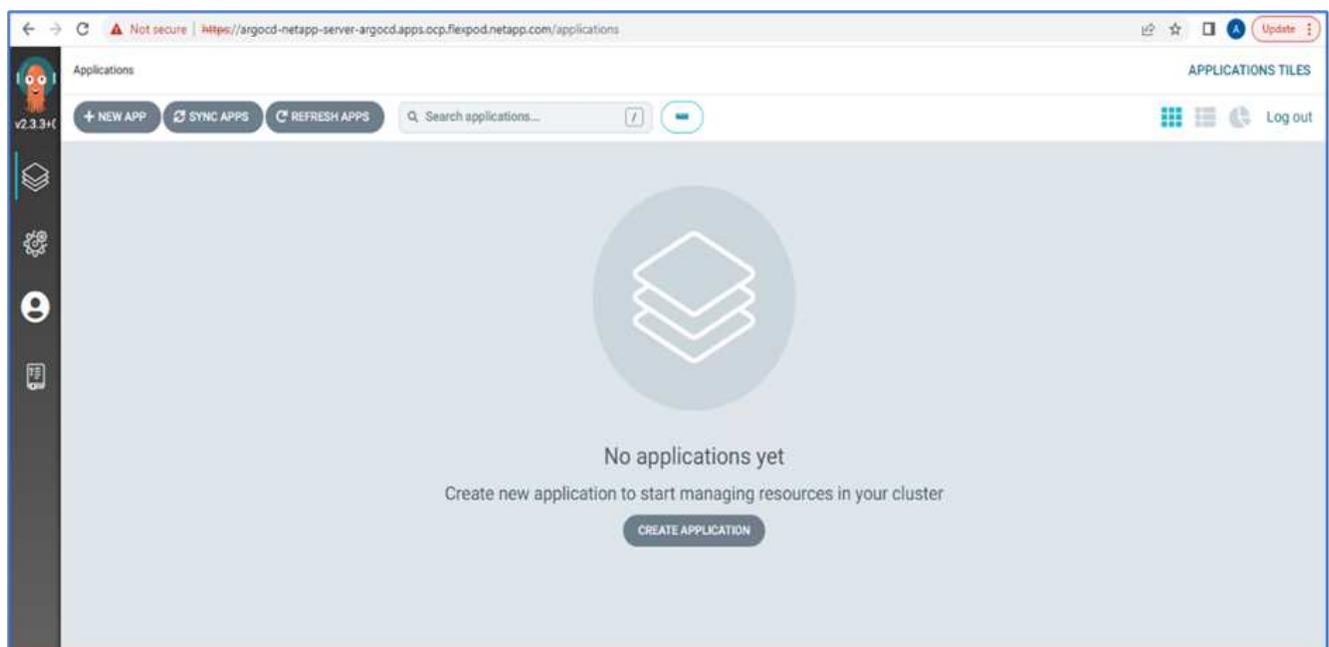
- Argo CDインスタンスを'argocd'プロジェクトに配備するには'名前を指定してCreateをクリックします



- Argo CDにログインするには、デフォルトのユーザはadminで、パスワードは「argocd -NetApp-cluster」という名前のシークレットファイルに含まれています。



7. サイド・メニューから'ルート>ロケーション'を選択し'argocd'ルートのURLをクリックしますユーザ名とパスワードを入力します。



8. CLIを使用して、Argo CDにオンプレミスOpenShiftクラスタを追加します。

```

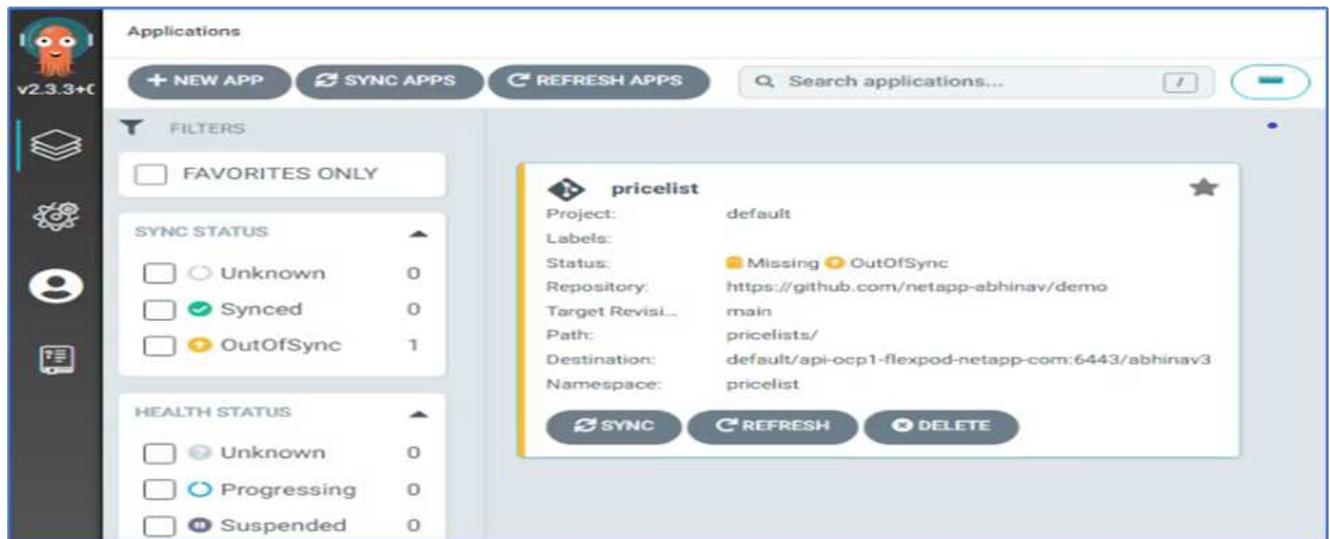
####Login to Argo CD####
abhinav3@abhinav-ansible$ argocd-linux-amd64 login argocd-netapp-server-
argocd.apps.ocp.flexpod.netapp.com --insecure
Username: admin
Password:
'admin:login' logged in successfully
Context'argocd-netapp-server-argocd.apps.ocp.flexpod.netapp.com' updated
####List the On-Premises 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

```

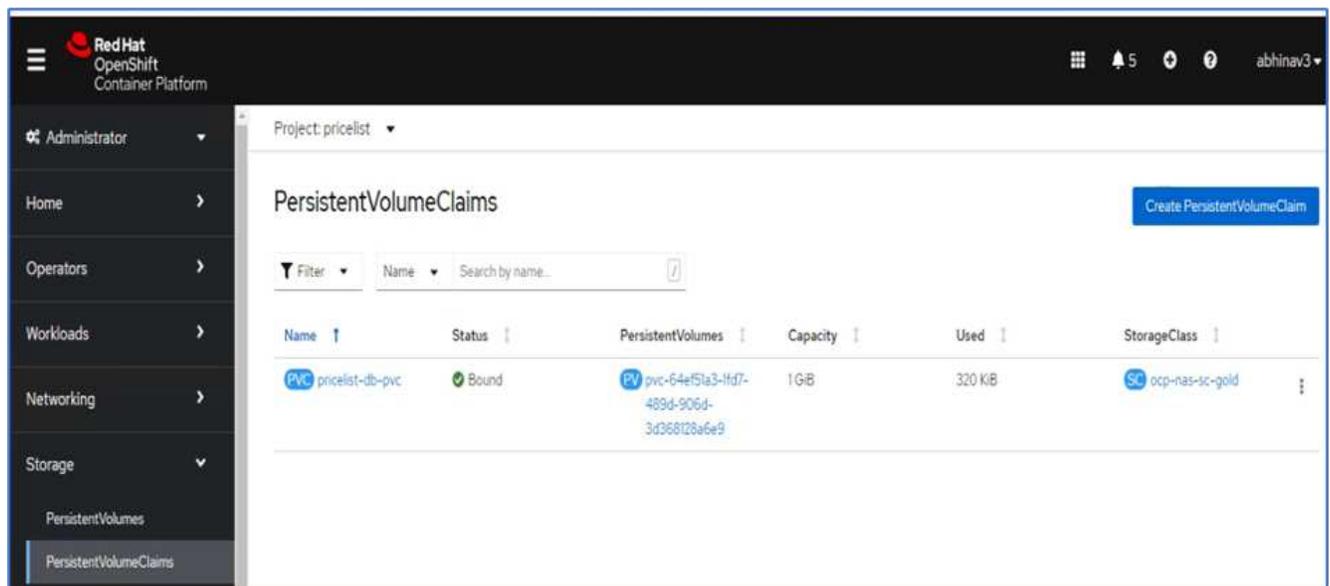
9. Argocd UIで、[新しいアプリ]をクリックし、アプリ名とコードリポジトリの詳細を入力します。

10. ネームスペースとともにアプリケーションを導入するOpenShiftクラスタを入力します。

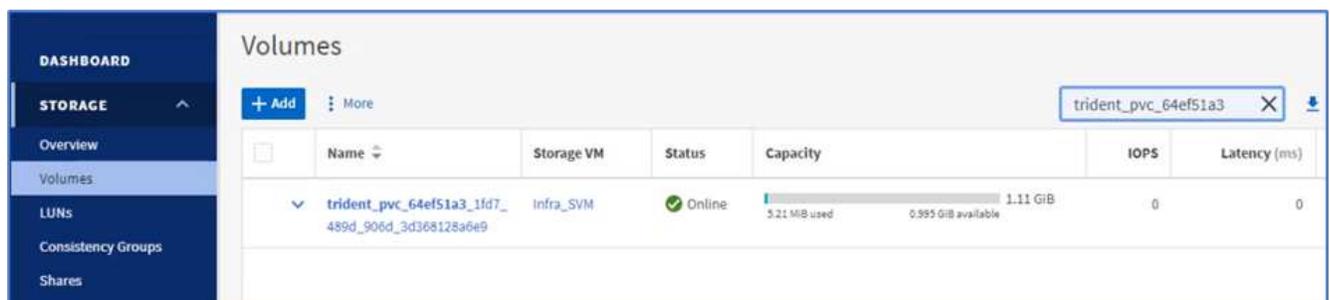
11. オンプレミスOpenShiftクラスタにアプリを導入するには、[同期]をクリックします。



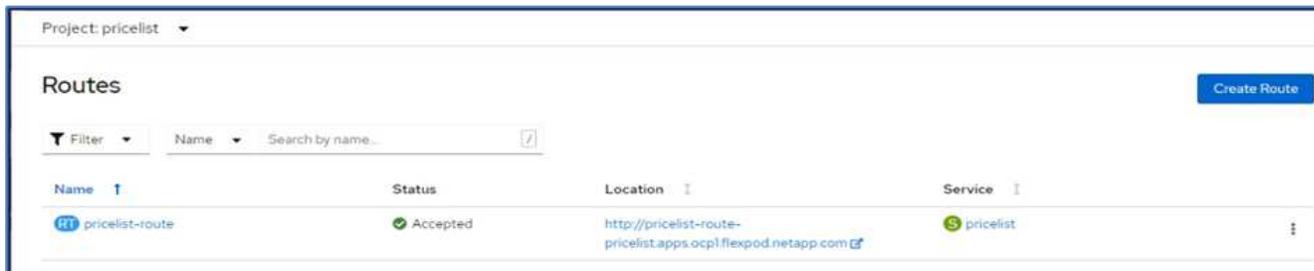
- OpenShift Container Platformコンソールで、プロジェクト価格表に移動し、ストレージでPVCの名前とサイズを確認します。



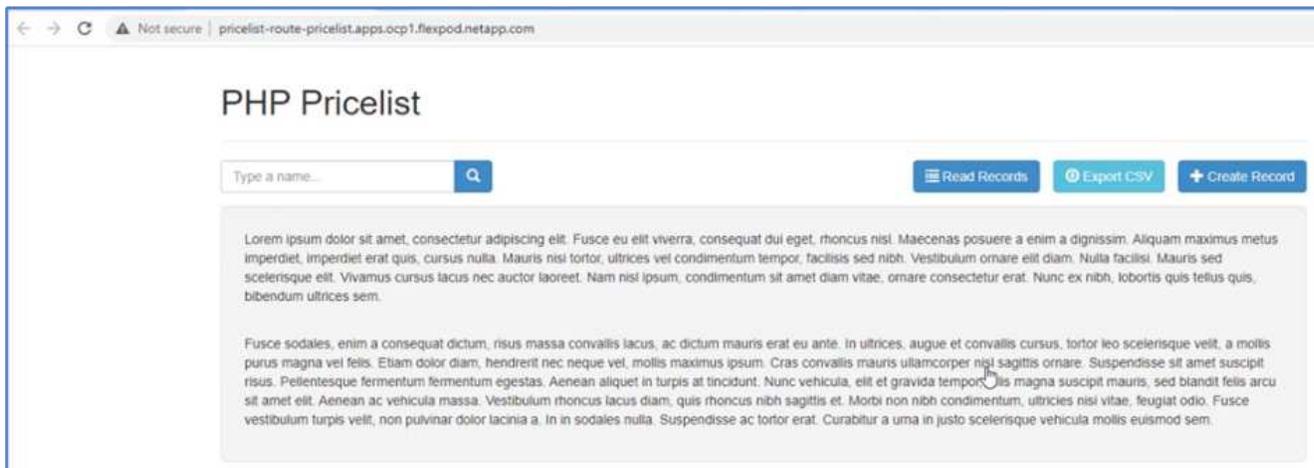
- System Managerにログインし、PVCを確認します。



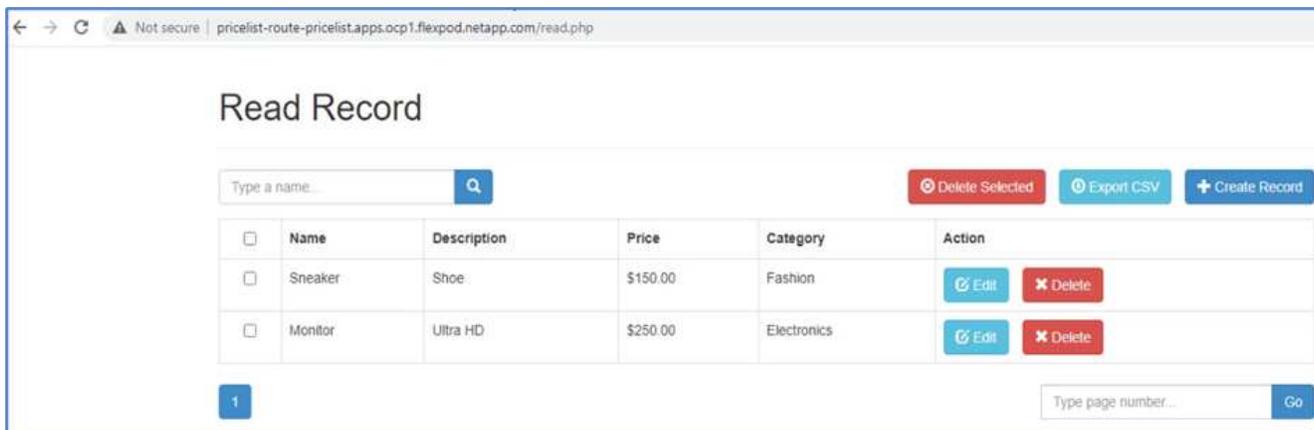
- ポッドが実行されたら、サイドメニューからネットワーキング／ルートを選択し、「場所」の下のURLをクリックします。



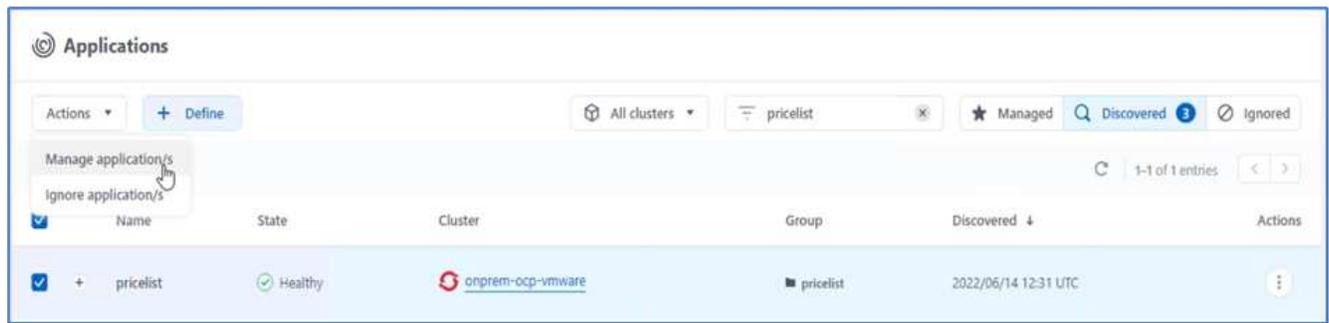
15. 価格表アプリのホームページが表示されます。



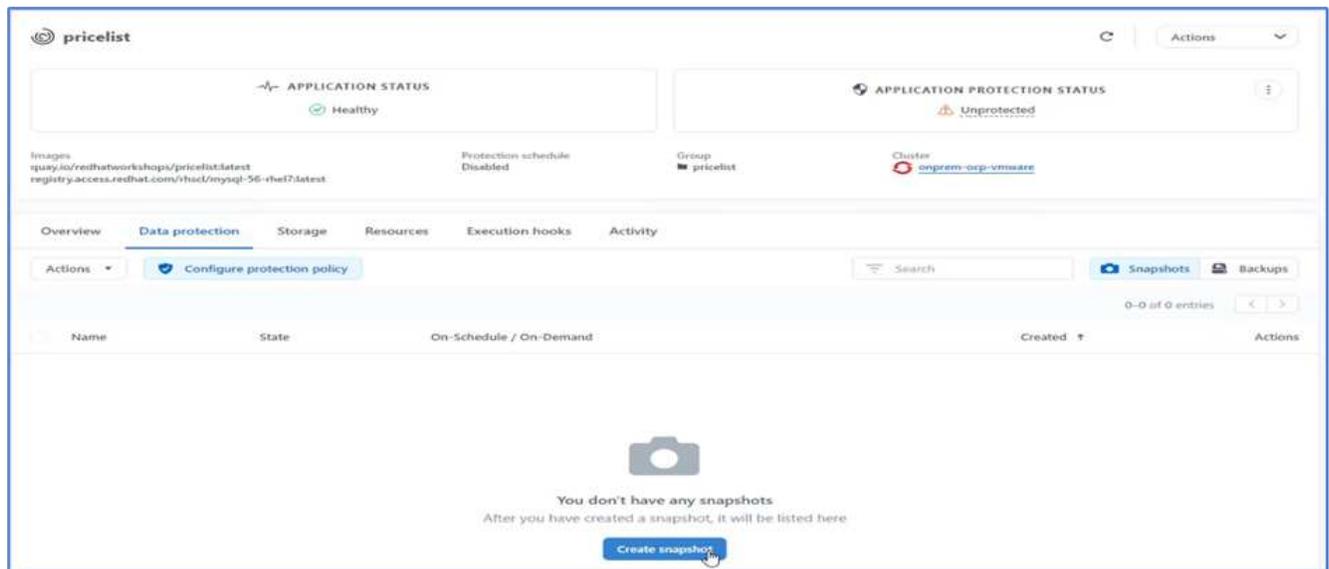
16. Webページにレコードをいくつか作成します。



17. アプリケーションはAstra Control Centerで検出されます。アプリを管理するには、[アプリケーション]>[検出済み]に移動し、価格表アプリを選択して、[アクション]の[アプリケーションの管理]をクリックします。

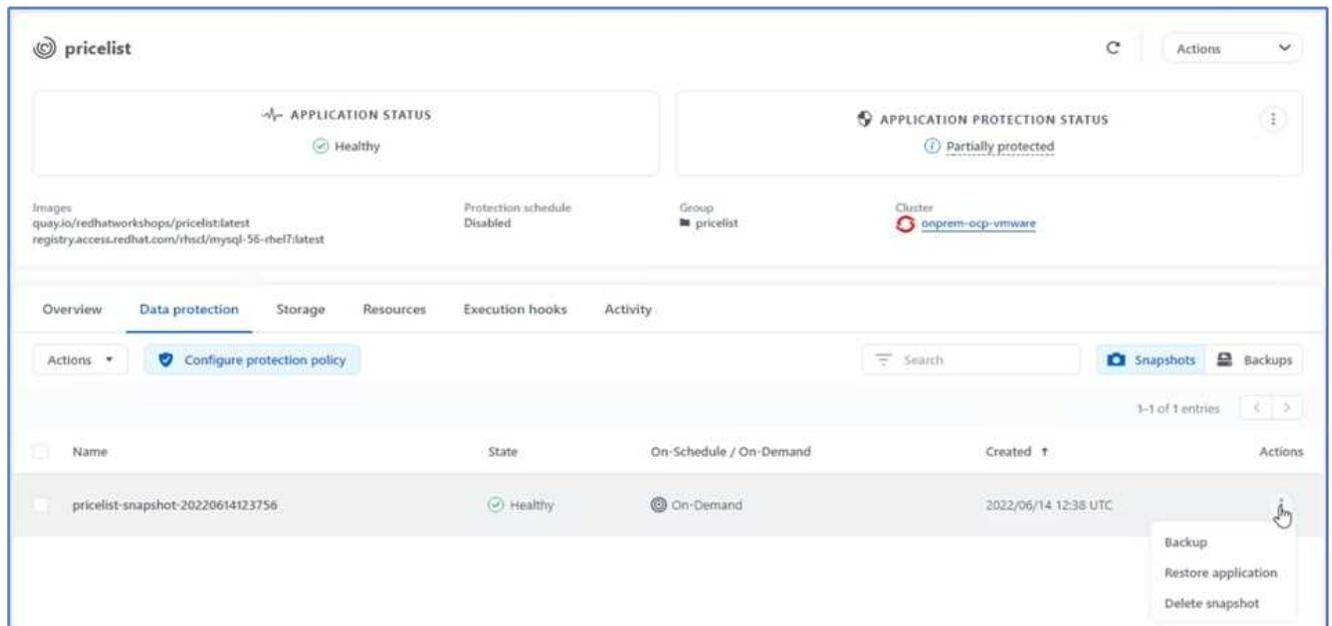


18. 価格表アプリをクリックし、[データ保護]を選択します。この時点では、Snapshotやバックアップは作成されていません。スナップショットの作成をクリックして、オンデマンドスナップショットを作成します。

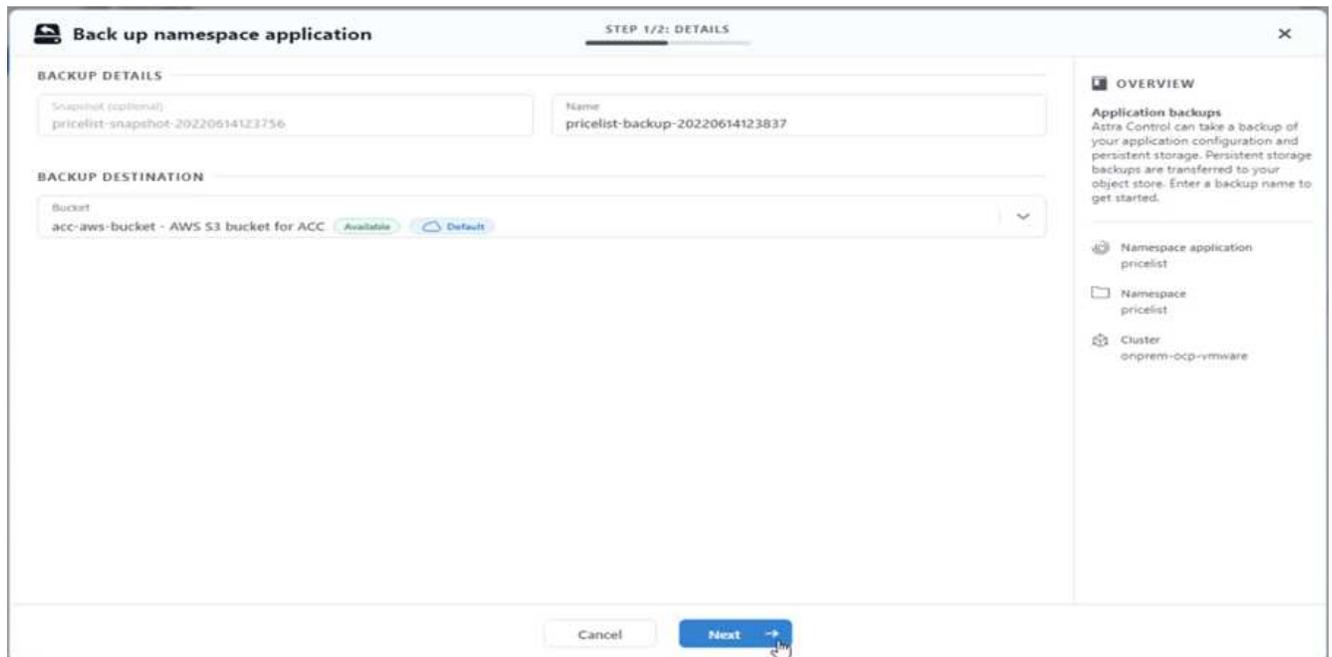


NetApp Astra Control Centerは、オンデマンドおよびスケジュールされたスナップショットとバックアップの両方をサポートします。

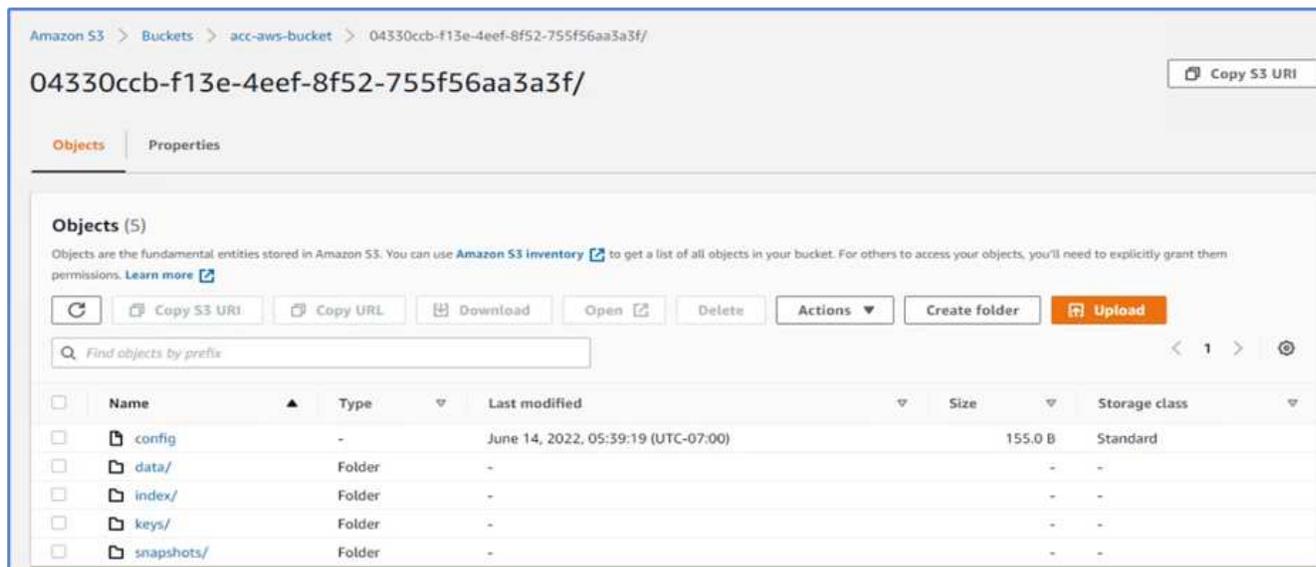
19. スナップショットが作成され、状態が正常になったら、そのスナップショットを使用してリモートバックアップを作成します。このバックアップはS3バケットに格納されます。



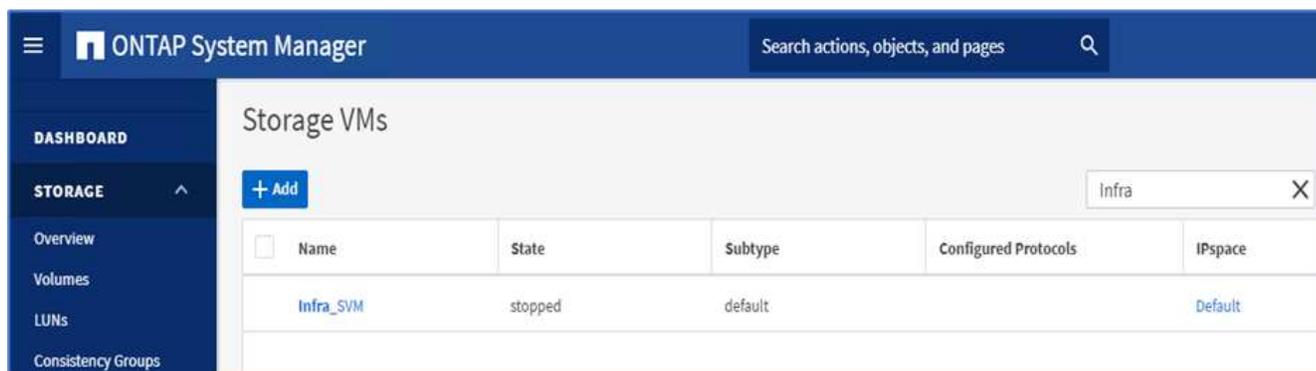
20. AWS S3バケットを選択してバックアップ処理を開始します。



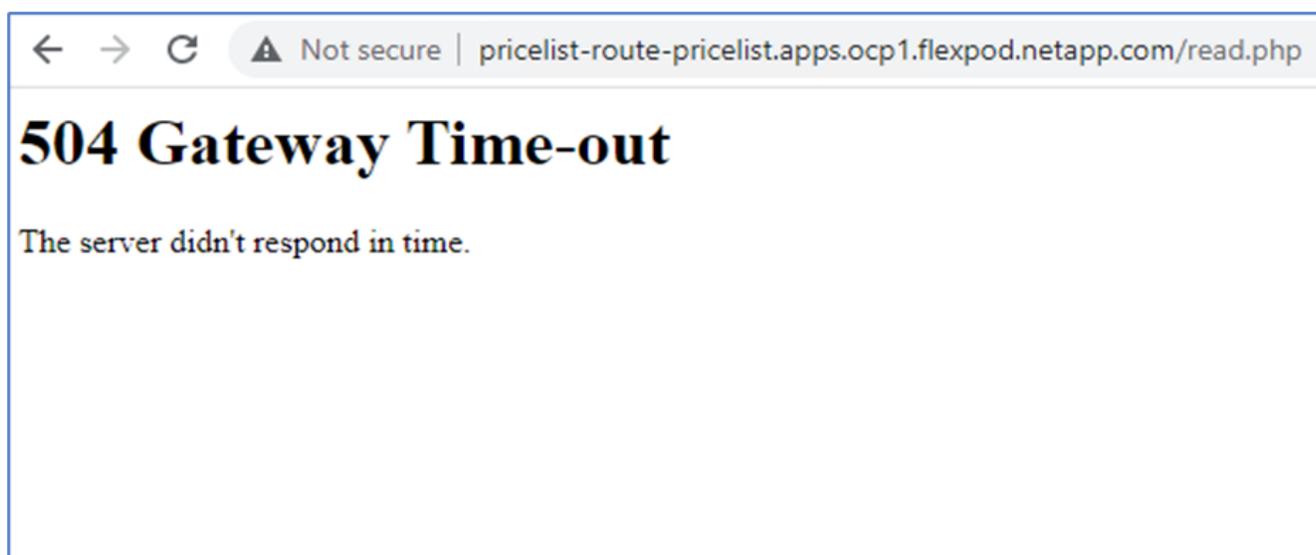
21. バックアップ処理では、AWS S3バケットに複数のオブジェクトを含むフォルダを作成する必要があります。



22. リモートバックアップが完了したら、PVの元のボリュームをホストするStorage Virtual Machine (SVM)を停止して、オンプレミスでの災害をシミュレートします。

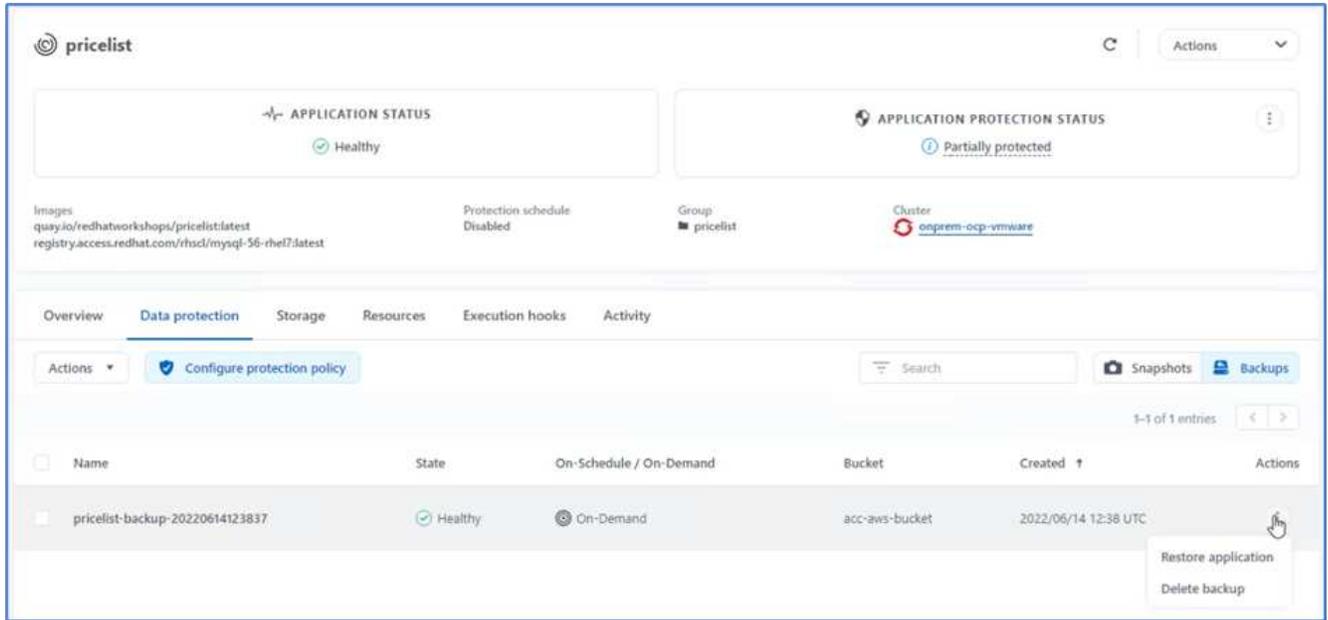


23. Webページを更新してシステム停止を確認します。Webページは使用できません。

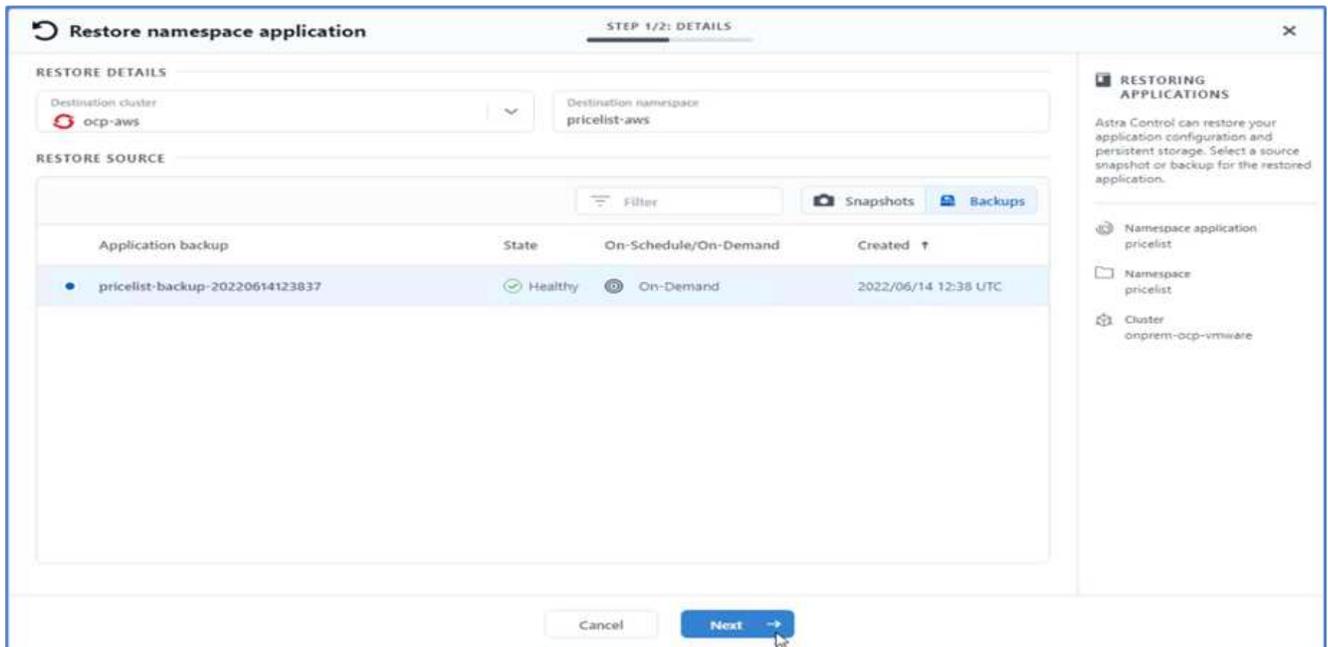


ウェブサイトは予想どおりに停止しているので、AWSで実行されているOpenShiftクラスタにAstraを使用して、リモートバックアップからアプリケーションを迅速にリカバリしてみましょう。

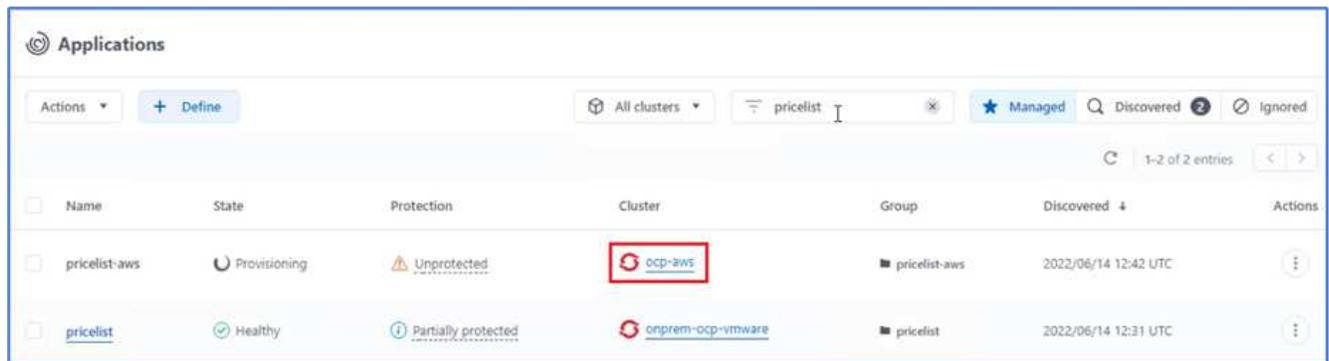
24. Astra Control Centerで、価格表アプリをクリックし、[データ保護]>[バックアップ]を選択します。バックアップを選択し、[操作]の下の[アプリケーションの復元]をクリックします。



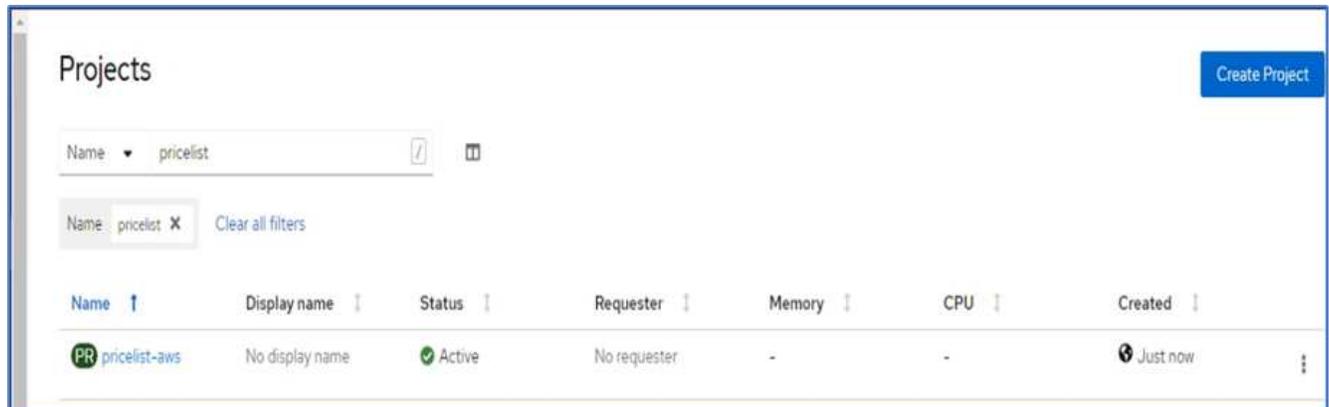
25. デスティネーションクラスタとして「OCP-AWS」を選択し、ネームスペースに名前を付けます。[オンデマンドバックアップ]、[次へ]、[復元]の順にクリックします。



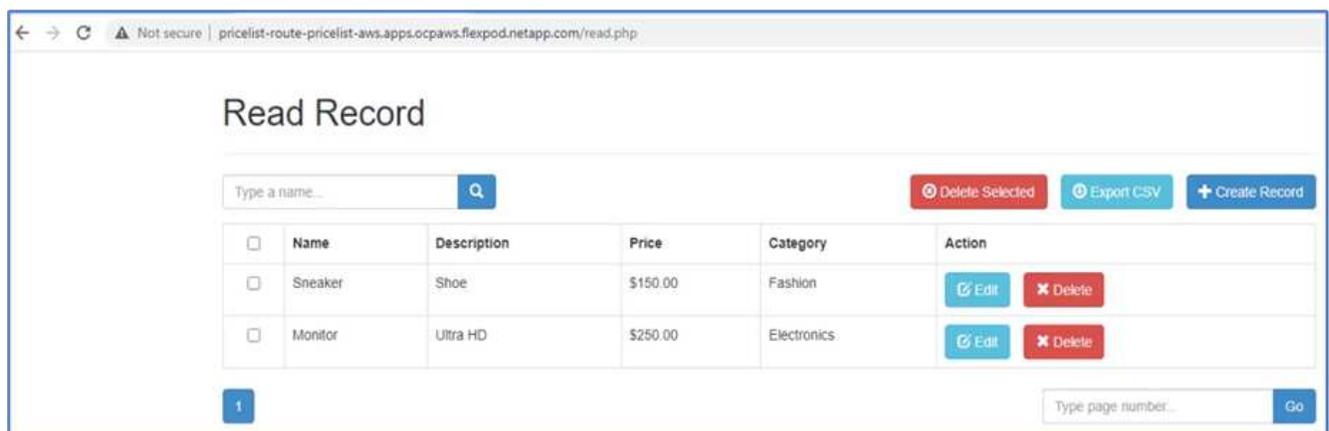
26. 「pricelist-app」という名前の新しいアプリケーションは、AWSで実行されるOpenShiftクラスタでプロビジョニングされます。



27. OpenShift Webコンソールで同じことを確認します。



28. 「pricelist -aws」プロジェクトの下ポッドがすべて実行されたら、「Routes」に移動し、URLをクリックしてWebページを起動します。



このプロセスでは、貴重なアプリケーションが正常に復元され、Astra Control Centerを利用してAWS上でシームレスに実行されるOpenShiftクラスターでデータの整合性が維持されていることを検証します。

SnapshotコピーとDevTestのアプリケーション移動によるデータ保護

この使用事例は、次のセクションで説明する2つの部分で構成されています。

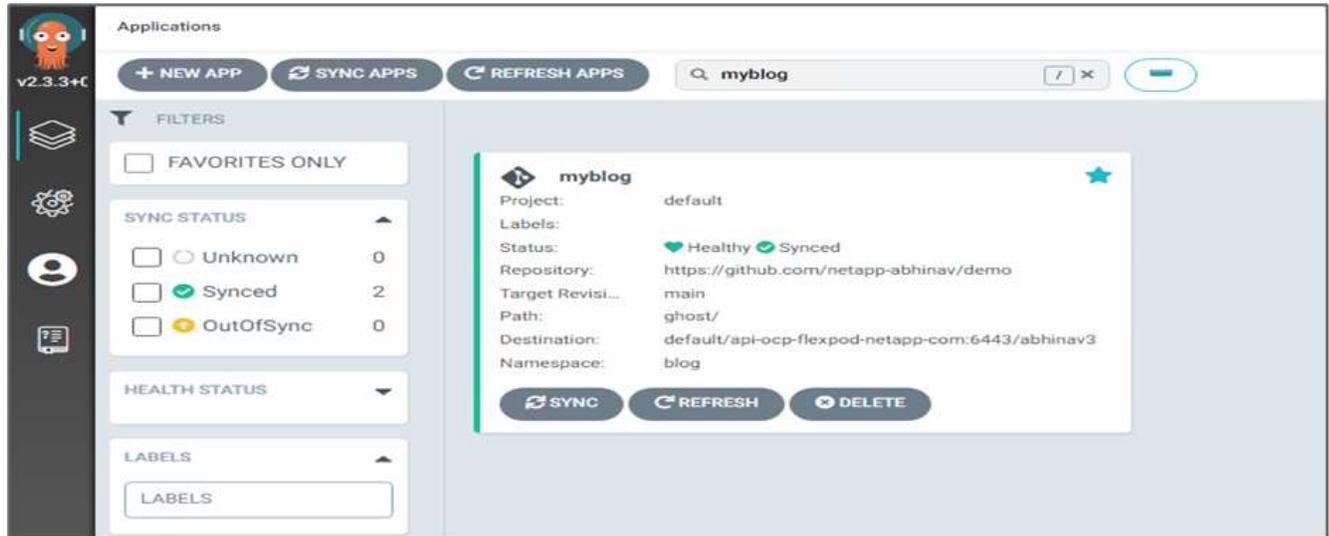
パート1

Astra Control Centerを使用すると、アプリケーション対応のスナップショットを作成してローカルデータを

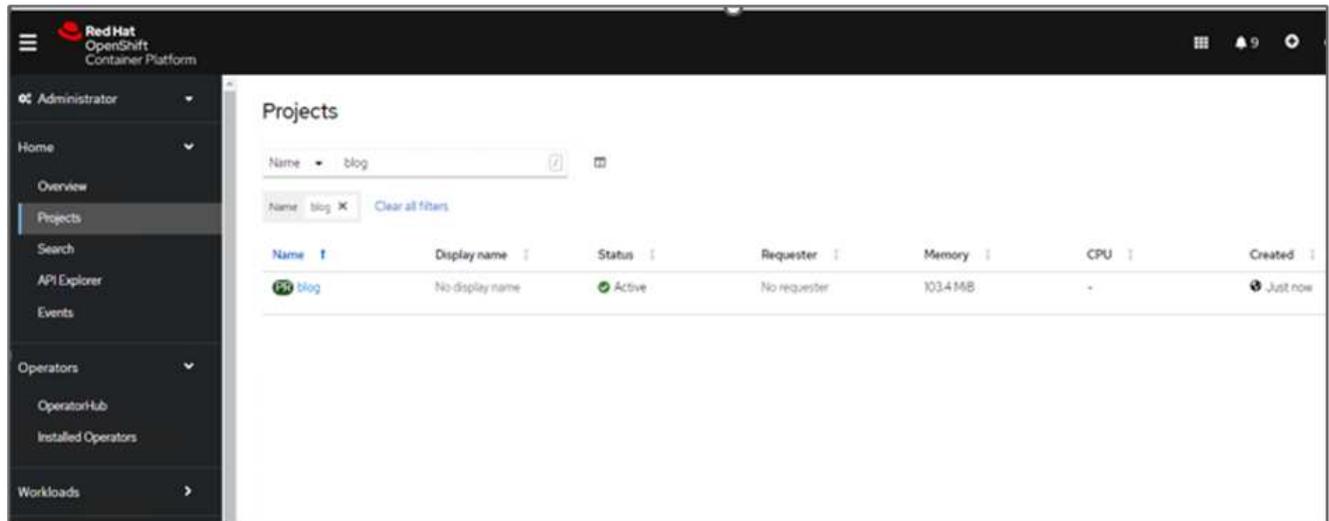
保護できます。データを誤って削除したり破損したりした場合は、以前に記録したスナップショットを使用して、アプリケーションおよび関連データを既知の正常な状態に戻すことができます。

このシナリオでは、開発とテスト（DevTest）チームが、Ghostブログアプリケーションであるサンプルのステートフルアプリケーション（ブログサイト）を導入し、コンテンツを追加し、アプリケーションを最新バージョンにアップグレードします。Ghostアプリケーションでは、データベースにSQLiteを使用します。アプリケーションをアップグレードする前に、Astra Control Centerを使用してスナップショット（オンデマンド）を作成し、データを保護します。詳細な手順は次のとおりです。

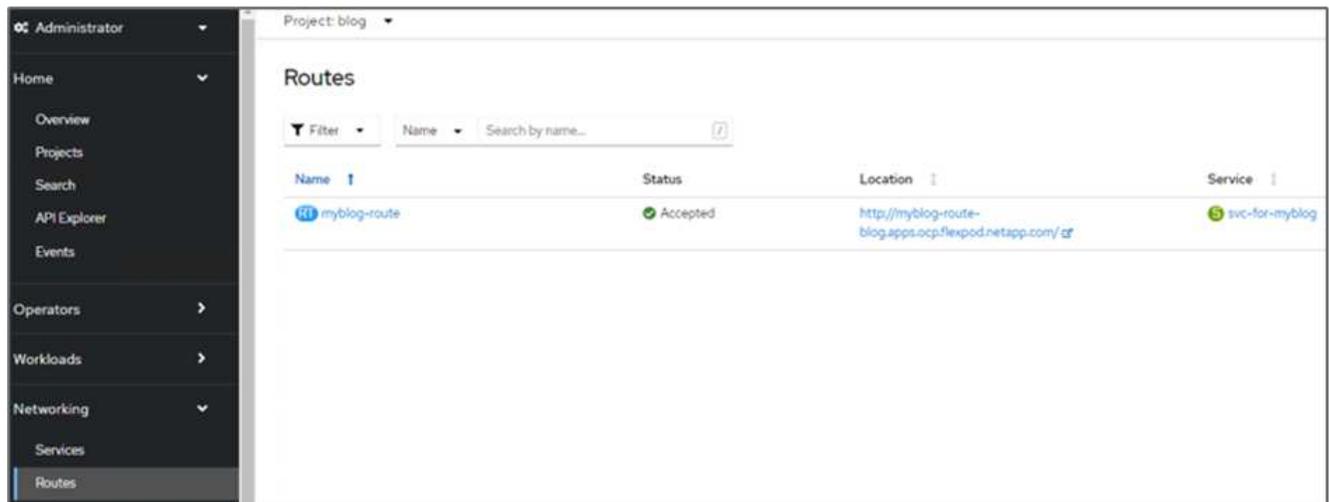
1. サンプルブログアプリをデプロイし、ArgoCDから同期します。



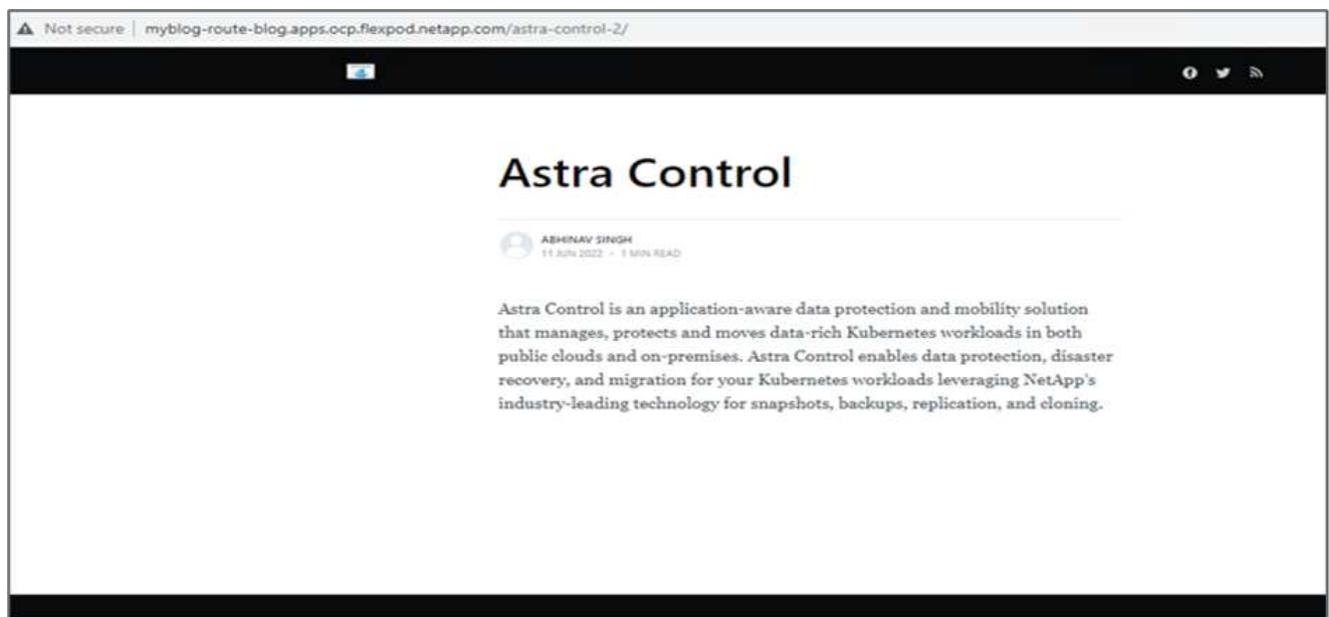
2. 最初のOpenShiftクラスタにログインし、Projectに移動して、検索バーにBlogと入力します。



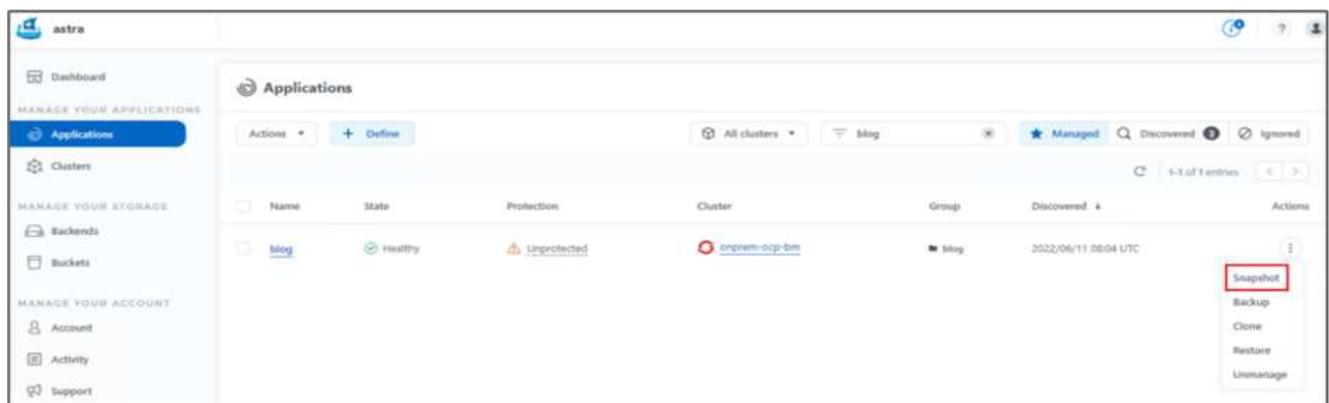
3. サイドメニューから、[Networking]>[Routes]の順に選択し、URLをクリックします。



4. ブログのホームページが表示されます。ブログサイトにコンテンツを追加して公開します。



5. Astra Control Centerにアクセスします。最初に検出タブからアプリケーションを管理してから、Snapshotコピーを作成します。





定義したスケジュールでスナップショット、バックアップ、またはその両方を作成することで、アプリケーションを保護することもできます。詳細については、を参照してください "Snapshot とバックアップでアプリケーションを保護"。

6. オンデマンドスナップショットが正常に作成されたら、アプリケーションを最新バージョンにアップグレードします。現在のイメージのバージョンは「ghost:3.6 -アルパイン」で、ターゲットのバージョンは「ghost:latest」です。アプリをアップグレードするには、Gitリポジトリに直接変更を加え、Argo CDに同期します。

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

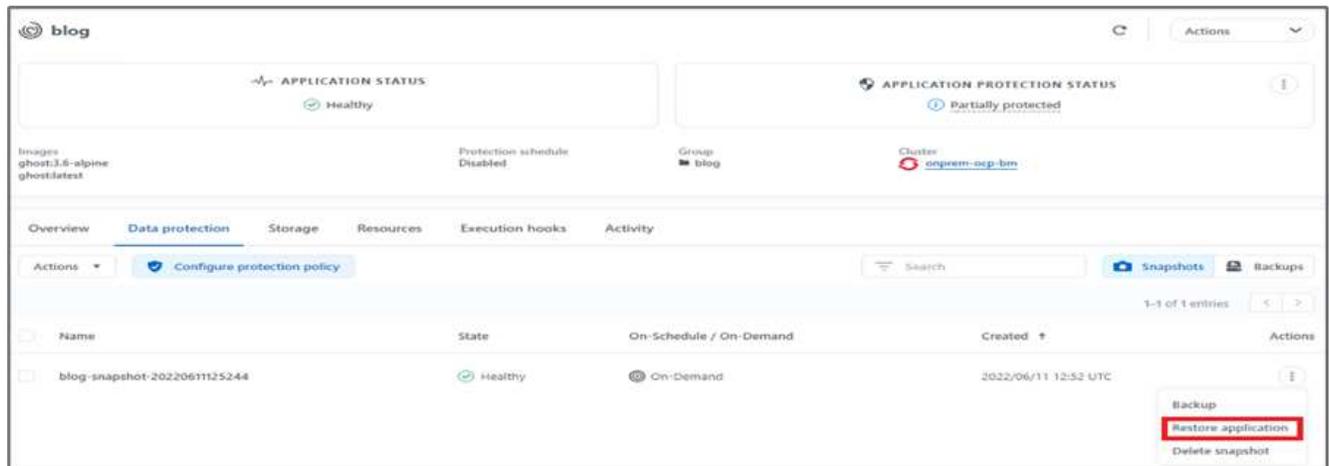
7. ブログサイトがダウンし、アプリケーション全体が破損しているために、最新バージョンへの直接アップグレードがサポートされていないことがわかります。

```
Project: blog
Pods > Pod details
myblog-5f899f7b76-zv7rq CrashLoopBackOff
Details Metrics YAML Environment Logs Events Terminal
Log stream ended. myblog Current log
34 lines
[2022-06-11 12:54:05] +[36mINFO+[39m Creating database backup
[2022-06-11 12:54:05] +[36mINFO+[39m Database backup written to: /var/lib/ghost/content/data/astra.ghost.2022-06-11-12-54-05.json
[2022-06-11 12:54:05] +[36mINFO+[39m Running migrations.
[2022-06-11 12:54:06] +[36mINFO+[39m Rolling back: Unable to run migrations.
[2022-06-11 12:54:06] +[36mINFO+[39m Rollback was successful.
[2022-06-11 12:54:06] +[31mERROR+[39m Unable to run migrations
+[31m
+[31mUnable to run migrations+[39m
+[37m>You must be on the latest v3.x to update across major versions - https://ghost.org/docs/update/"+[39m
+[33m"Run 'ghost update v3' to get the latest v3.x version, then run 'ghost update' to get to the latest."+[39m
+[1m+[37mError ID:+[39m+[22m
  +[90m93b99ce0-e985-11ec-9301-7d29b2c73999+[39m
+[90m-----+[39m
+[90mInternalServerError: Unable to run migrations
  at /var/lib/ghost/versions/5.2.2/node_modules/knex-migrator/lib/index.js:1032:19
  at up (/var/lib/ghost/versions/5.2.2/core/server/data/migrations/utils/migrations.js:118:19)
  at Object.up (/var/lib/ghost/versions/5.2.2/core/server/data/migrations/utils/migrations.js:54:19)
  at /var/lib/ghost/versions/5.2.2/node_modules/knex-migrator/lib/index.js:982:33
  at /var/lib/ghost/versions/5.2.2/node_modules/knex/lib/execution/transaction.js:221:22+[39m
+[39m
[2022-06-11 12:54:06] +[35mWARN+[39m Ghost is shutting down
[2022-06-11 12:54:06] +[35mWARN+[39m Ghost has shut down
[2022-06-11 12:54:06] +[35mWARN+[39m Your site is now offline
[2022-06-11 12:54:06] +[35mWARN+[39m Ghost was running for a few seconds
```

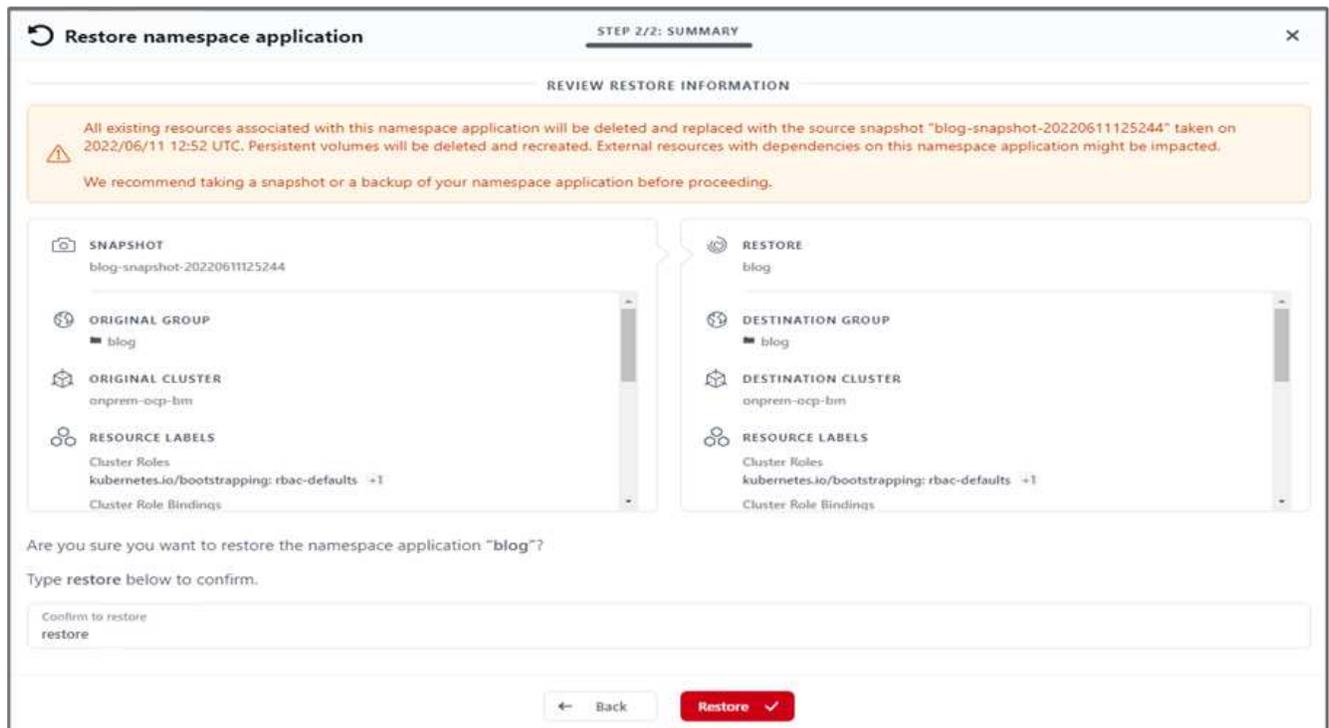
8. ブログサイトが利用できないことを確認するには、URLを更新します。



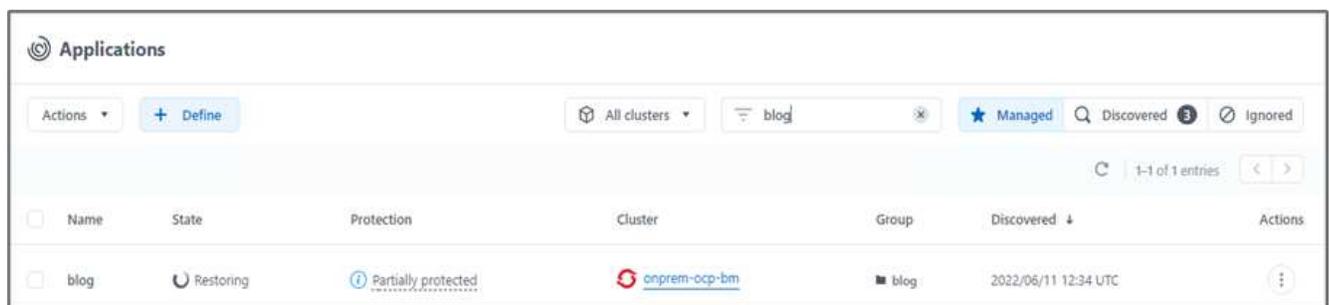
9. スナップショットからアプリケーションを復元します。



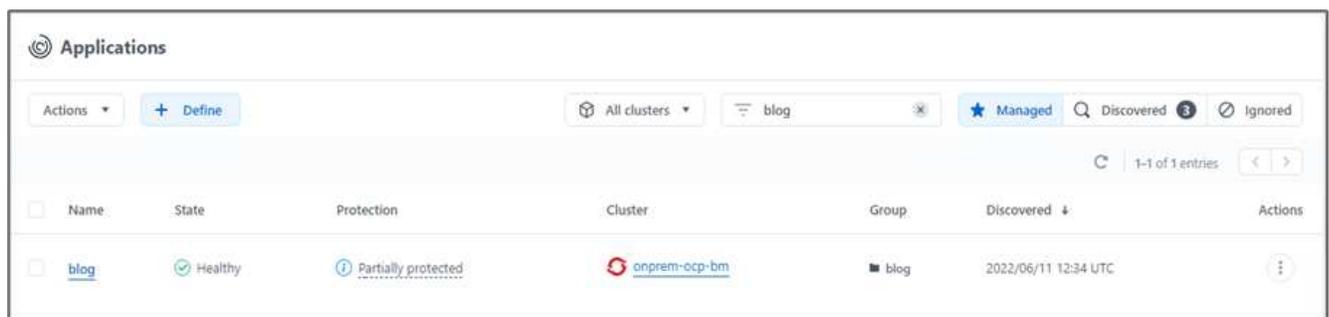
10. アプリケーションは同じOpenShiftクラスタにリストアされます。



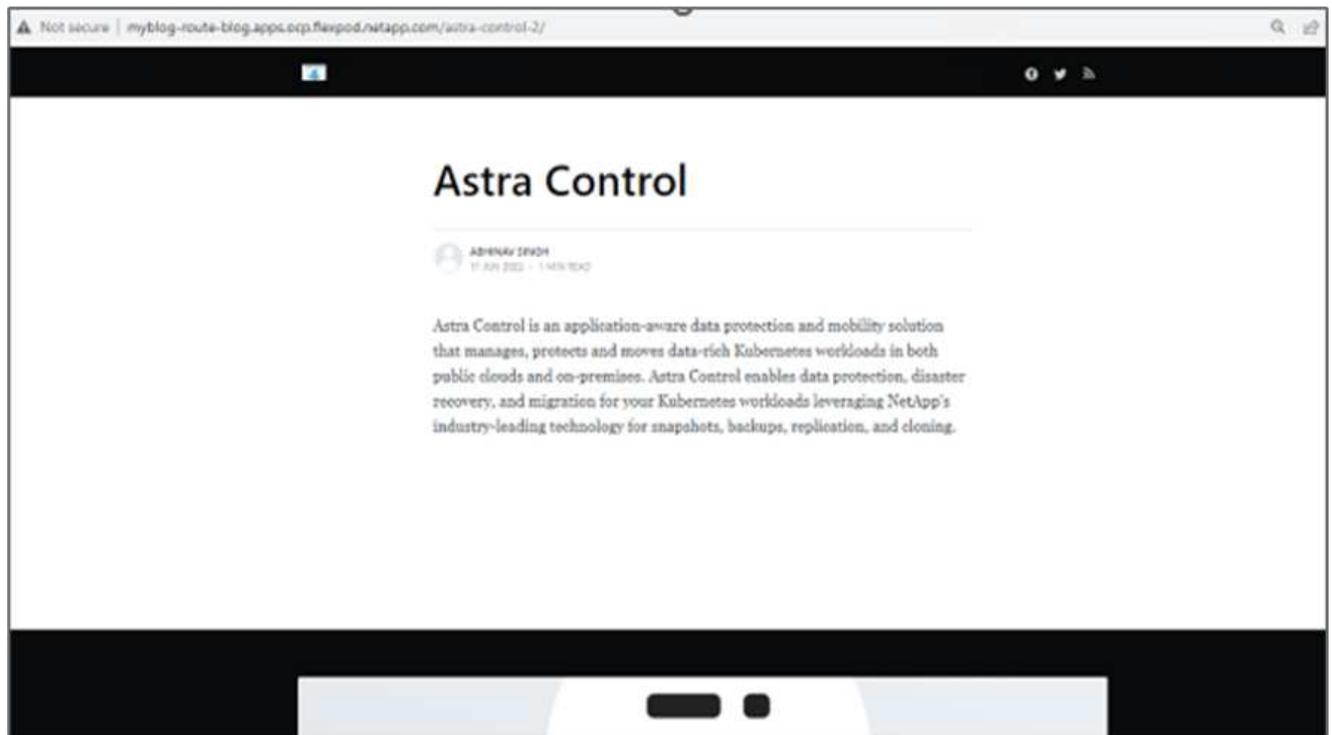
11. アプリケーションのリストアプロセスがただちに開始されます。



12. 数分後に、使用可能なスナップショットからアプリケーションが正常にリストアされます。



13. Webページが表示されるかどうかを確認するには、URLを更新します。



DevTestチームは、Astra Control Centerを活用して、ブログサイトアプリとその関連データをスナップショットを使用して正常にリカバリできます。

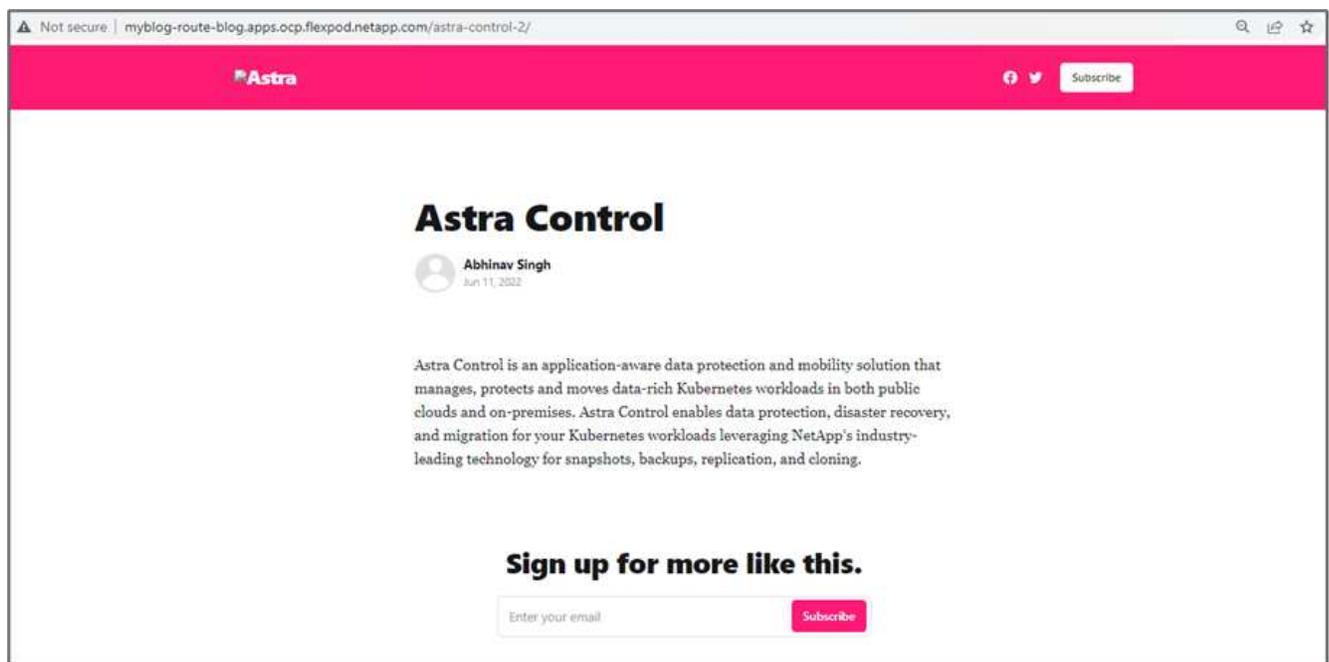
パート2

Astra Control Centerを使用すると、クラウド上またはオンプレミスで、クラウド上のどの場所にあるかに関係なく、アプリケーション全体をKubernetesクラスター間でデータとともに移動できます。

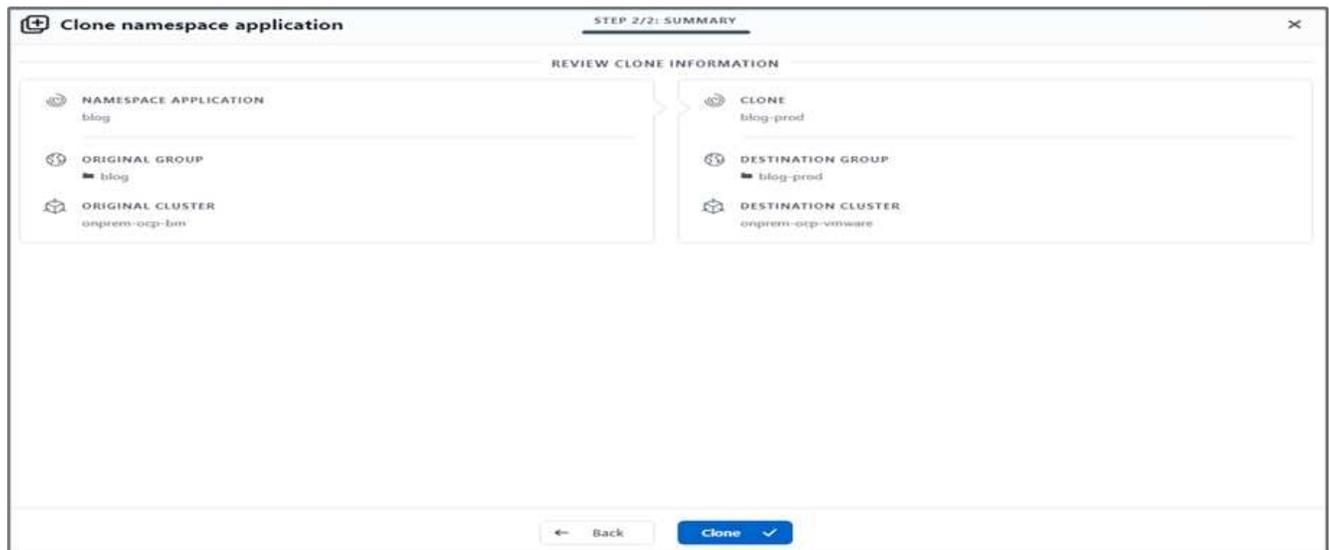
1. DevTestチームは、アプリケーションを最初にサポートされているバージョン（「ゴースト-4.6-アルプス」）にアップグレードしてから、最終バージョン（「ゴースト-最新」）にアップグレードして、本番環境を準備します。その後、別のFlexPod システムで実行されている本番環境のOpenShiftクラスターにクローニングされているアプリケーションをアップグレードします。
2. この時点で、アプリケーションが最新バージョンにアップグレードされ、本番環境のクラスターにクローニングできる状態になります。

```
Project: blog
Pods > Pod details
P myblog-55ffd9f658-tkbfq Running
Details Metrics YAML Environment Logs Events Terminal
180
181     - containerPort: 2368
182       protocol: TCP
183     imagePullPolicy: Always
184     volumeMounts:
185       - name: content
186         mountPath: /var/lib/ghost/content
187       - name: kube-api-access-t2sdz
188         readOnly: true
189         mountPath: /var/run/secrets/kubernetes.io/serviceaccount
190     terminationMessagePolicy: File
191     image: 'ghost:latest'
192     serviceAccount: default
193   volumes:
194     - name: content
195       persistentVolumeClaim:
196         claimName: blog-content
```

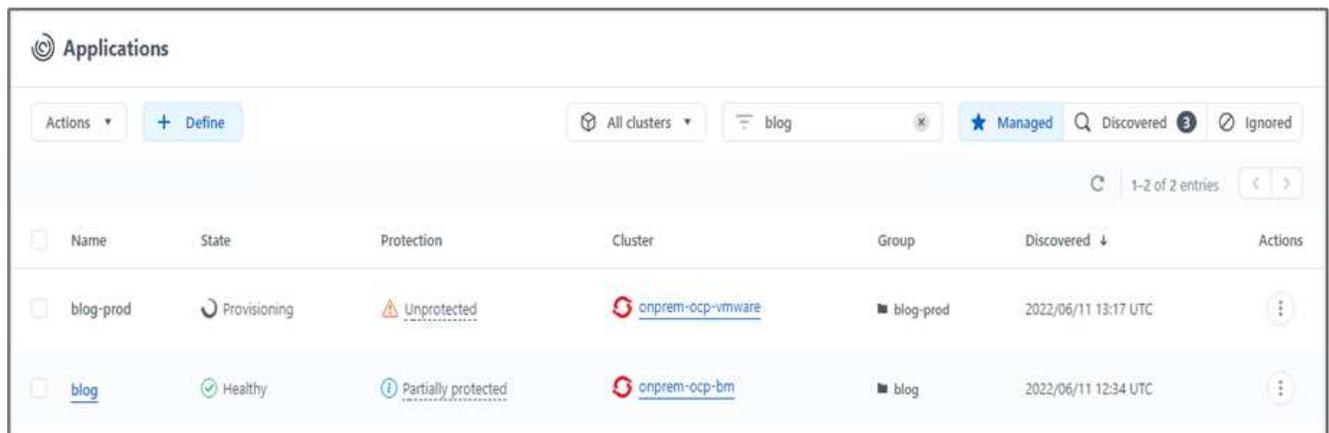
3. 新しいテーマを確認するには、ブログサイトを更新します。



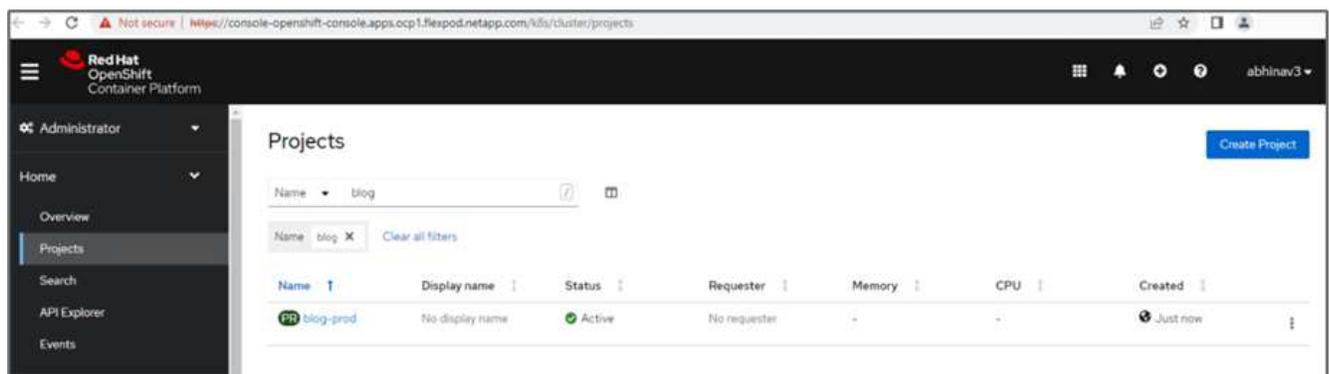
4. Astra Control Centerから、VMware vSphereで実行されている他の本番環境OpenShiftクラスタにアプリケーションをクローニングします。



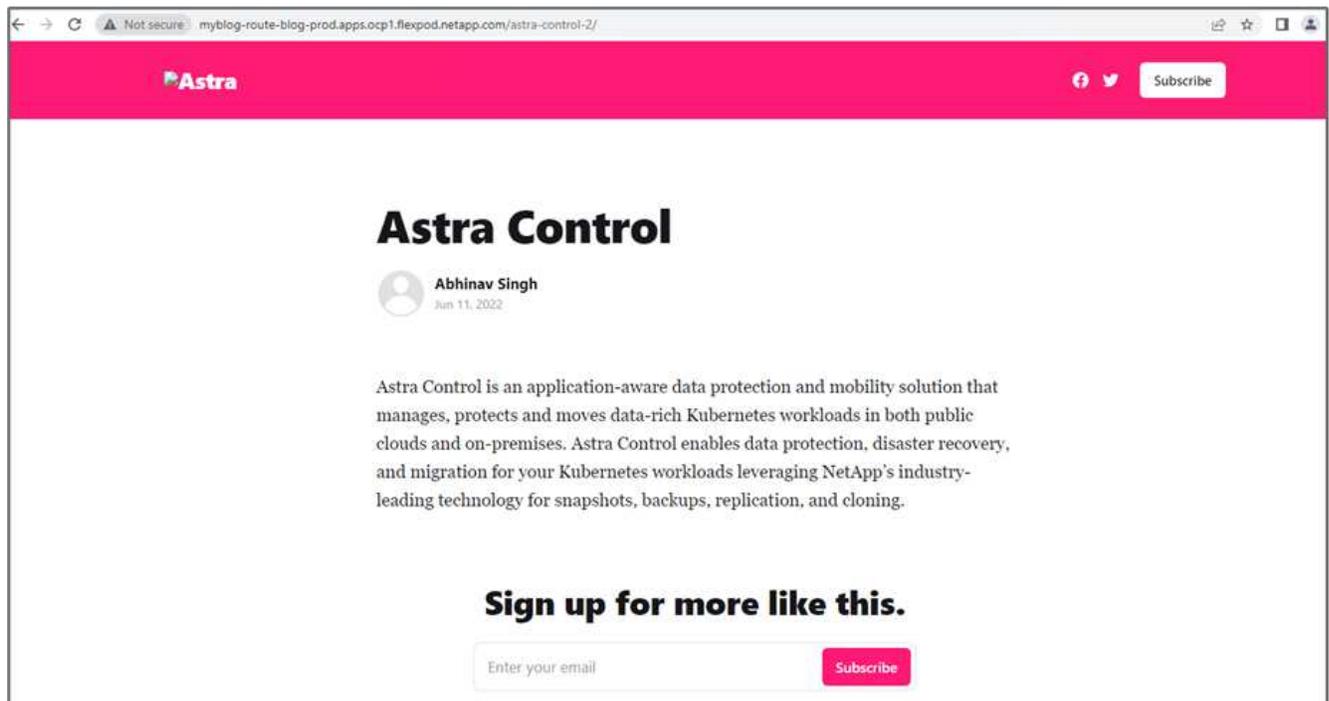
これで、本番環境のOpenShiftクラスタで新しいアプリケーションクローンがプロビジョニングされます。



5. 本番環境のOpenShiftクラスタにログインし、プロジェクトブログを検索します。



6. サイドメニューから、Networking > Routesを選択し、Locationの下のURLをクリックします。同じホームページとコンテンツが表示されます。



これでAstra Control Center解決策 の検証は終了です。Kubernetesクラスタが配置されている場所に関係なく、アプリケーション全体とそのデータを1つのKubernetesクラスタから別のクラスタにクローニングできるようになりました。

"次は終わりです"

まとめ

"Previous : リモートバックアップを使用したアプリケーションのリカバリ。"

この解決策 では、 ネットアップのAstraポートフォリオを使用して、FlexPod とAWSで実行されるコンテナ化アプリケーション向けの保護計画を実装しました。 ネットアップのAstra Control CenterとAstra Tridentは、 Cloud Volumes ONTAP 、 Red Hat OpenShift、 FlexPod インフラとともに、 この解決策 のコアコンポーネントを形成しました。

Snapshotをキャプチャしてアプリケーションの保護を実証し、クラウド環境とオンプレミス環境で実行されているKubernetesクラスタ間でアプリケーションをリストアするフルコピーバックアップを実行しました。

また、 Kubernetesクラスタ間でアプリケーションのクローニングを実演し、お客様が希望する場所で選択したKubernetesクラスタにアプリケーションを移行できるようにする方法についても説明しました。

FlexPod は絶えず進化しているため、お客様はアプリケーションやビジネス提供プロセスを最新化できます。この解決策 を使用することで、 FlexPod のお客様は、 解決策 のコストを低く抑えながら、短期またはフルタイムのDRプランを作成できる場所としてパブリッククラウドを使用して、クラウドネイティブアプリケーション向けのBCDRプランを自信を持って構築できます。

Astra Controlを使用すると、クラスタの配置場所に関係なく、アプリケーション全体をKubernetesクラスタ間でデータとともに移動できます。また、クラウドネイティブアプリケーションの導入、運用、保護を高速化するのにも役立ちます。

トラブルシューティング

トラブルシューティングのガイダンスについては、を参照してください "[オンラインドキュメント](#)"。

追加情報の参照先

このドキュメントに記載されている情報の詳細については、以下のドキュメントや Web サイトを参照してください。

- FlexPod ホームページ

["https://www.flexpod.com"](https://www.flexpod.com)

- FlexPod のシスコ検証済み設計および導入ガイド

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

- Ansibleを使用して、VMwareのコードとしてInfrastructureを使用したFlexPod の導入

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

- Ansibleを使用したRed Hat OpenShift Bare Metalのコードとしてのインフラを使用したFlexPod 導入

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

- Cisco UCS ハードウェアおよびソフトウェア相互運用性ツール

["http://www.cisco.com/web/techdoc/ucs/interoperability/matrix/matrix.html"](http://www.cisco.com/web/techdoc/ucs/interoperability/matrix/matrix.html)

- Cisco Intersightのデータシート

["https://intersight.com/help/saas/home"](https://intersight.com/help/saas/home)

- ネットアップAstraのドキュメント

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

- ネットアップアストラコントロールセンター

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

- ネットアップアストラ Trident

["https://docs.netapp.com/us-en/trident/index.html"](https://docs.netapp.com/us-en/trident/index.html)

- NetApp Cloud Manager の略

["https://docs.netapp.com/us-en/occm/concept_overview.html"](https://docs.netapp.com/us-en/occm/concept_overview.html)

- NetApp Cloud Volumes ONTAP の略

["https://docs.netapp.com/us-en/occm/task_getting_started_aws.html"](https://docs.netapp.com/us-en/occm/task_getting_started_aws.html)

- Red Hat OpenShift のサービスです

["https://www.openshift.com/"](https://www.openshift.com/)

- NetApp Interoperability Matrix Tool で確認できます

["http://support.netapp.com/matrix/"](http://support.netapp.com/matrix/)

バージョン履歴

バージョン	日付	ドキュメントのバージョン履歴
バージョン 1.0 以降	2022年7月	ACC 22.04.0用リリース。

著作権に関する情報

Copyright © 2026 NetApp, Inc. All Rights Reserved. Printed in the U.S.このドキュメントは著作権によって保護されています。著作権所有者の書面による事前承諾がある場合を除き、画像媒体、電子媒体、および写真複写、記録媒体、テープ媒体、電子検索システムへの組み込みを含む機械媒体など、いかなる形式および方法による複製も禁止します。

ネットアップの著作物から派生したソフトウェアは、次に示す使用許諾条項および免責条項の対象となります。

このソフトウェアは、ネットアップによって「現状のまま」提供されています。ネットアップは明示的な保証、または商品性および特定目的に対する適合性の暗示的保証を含み、かつこれに限定されないいかなる暗示的な保証も行いません。ネットアップは、代替品または代替サービスの調達、使用不能、データ損失、利益損失、業務中断を含み、かつこれに限定されない、このソフトウェアの使用により生じたすべての直接的損害、間接的損害、偶発的損害、特別損害、懲罰的損害、必然的損害の発生に対して、損失の発生の可能性が通知されていたとしても、その発生理由、根拠とする責任論、契約の有無、厳格責任、不法行為（過失またはそうでない場合を含む）にかかわらず、一切の責任を負いません。

ネットアップは、ここに記載されているすべての製品に対する変更を随時、予告なく行う権利を保有します。ネットアップによる明示的な書面による合意がある場合を除き、ここに記載されている製品の使用により生じる責任および義務に対して、ネットアップは責任を負いません。この製品の使用または購入は、ネットアップの特許権、商標権、または他の知的所有権に基づくライセンスの供与とはみなされません。

このマニュアルに記載されている製品は、1つ以上の米国特許、その他の国の特許、および出願中の特許によって保護されている場合があります。

権利の制限について：政府による使用、複製、開示は、DFARS 252.227-7013（2014年2月）およびFAR 5252.227-19（2007年12月）のRights in Technical Data -Noncommercial Items（技術データ - 非商用品目に関する諸権利）条項の(b)(3)項、に規定された制限が適用されます。

本書に含まれるデータは商用製品および/または商用サービス（FAR 2.101の定義に基づく）に関係し、データの所有権はNetApp, Inc.にあります。本契約に基づき提供されるすべてのネットアップの技術データおよびコンピュータソフトウェアは、商用目的であり、私費のみで開発されたものです。米国政府は本データに対し、非独占的かつ移転およびサブライセンス不可で、全世界を対象とする取り消し不能の制限付き使用权を有し、本データの提供の根拠となった米国政府契約に関連し、当該契約の裏付けとする場合にのみ本データを使用できます。前述の場合を除き、NetApp, Inc.の書面による許可を事前に得ることなく、本データを使用、開示、転載、改変するほか、上演または展示することはできません。国防総省にかかる米国政府のデータ使用权については、DFARS 252.227-7015(b)項（2014年2月）で定められた権利のみが認められます。

商標に関する情報

NetApp、NetAppのロゴ、<http://www.netapp.com/TM>に記載されているマークは、NetApp, Inc.の商標です。その他の会社名と製品名は、それを所有する各社の商標である場合があります。