ネットアップのハイブリッドマルチクラウドと **Red Hat OpenShift**

NetApp Solutions

NetApp
April 10, 2024

This PDF was generated from https://docs.netapp.com/ja-jp/netapp-solutions/rhhc/rhhc-value-prop.html on April 10, 2024. Always check docs.netapp.com for the latest.

目次

ネットアップのハイブリッドマルチクラウドとRed Hat OpenShift Containerワークロード 1
Red Hat OpenShift Containerワークロード向けのネットアップハイブリッドマルチクラウドソリューション 1
Red Hat OpenShift Containerワークロード向けのネットアップハイブリッドマルチクラウドソリューション 14
Red Hat OpenShift Containerワークロード向けのネットアップハイブリッドマルチクラウドソリューション 24
Red Hat OpenShift Containerワークロード向けのネットアップハイブリッドマルチクラウドソリューション 41

ネットアップのハイブリッドマルチクラウドとRed Hat OpenShift Containerワークロード

Red Hat OpenShift Containerワークロード向けのネットアップハイブリッドマルチクラウドソリューション

概要

ネットアップでは、従来型エンタープライズアプリケーションを最新化し、Kubernetesを中心に構築されたコンテナとオーケストレーションプラットフォームを使用して新しいアプリケーションを構築するお客様が大幅に増えています。Red Hat OpenShift Container Platformは、多くのお客様に採用されている例の1つです。

企業内でコンテナを採用するお客様がますます増えています。ネットアップは、ステートフルアプリケーションの永続的ストレージのニーズに加え、データ保護、データセキュリティ、データ移行などの従来のデータ管理のニーズにも応えることができます。しかし、これらのニーズは、さまざまな戦略、ツール、方法を使用して満たしています。

- NetApp ONTAP **ベースのストレージオプションを次に示します。コンテナとKubernetes環境にセキュリティ、データ保護、信頼性、柔軟性を提供します。
  - オンプレミスの自己管理型ストレージ：
- ネットアップファブリック接続ストレージ（FAS ）、ネットアップオールフラッシュFAS アレイ（AFF ）、ネットアップオールSANアレイ（ASA ）、ONTAP Select
  - オンプレミスのプロバイダ管理ストレージ：
- NetApp Keystone が提供するストレージサービス（STaaS）
  - クラウド内の自己管理型ストレージ：
- NetApp Cloud Volumes ONTAP（CVO）は、ハイパースケーラに自己管理型ストレージを提供します
  - クラウド上のプロバイダが管理するストレージ：
- Cloud Volumes Service for Google Cloud（CVS）、Azure NetApp Files （ANF）、Amazon FSx for NetApp ONTAP は、ハイパースケーラにフルマネージドストレージを提供します

ONTAP feature highlights

Storage Administration

- Multi-tenancy
- FlexVol & FlexGroup
- LUN
- Quotas
- ONTAP CLI & API
- System Manager & BlueXP

Performance & Scalability

- FlexCache
- FlexClone
- nconnect, session trunking, multipathing
- Scale-out clusters

Availability & Resilience

- Multi-AZ HA deployment (MetroCluster)
- SnapShot & SnapRestore
- SnapMirror
- SnapMirror Business Continuity
- SnapMirror Cloud

Access Protocols

- NFS –v3, v4, v4.1, v4.2
- SMB – v2, v3
- iSCSI
- Multi-protocol access

Storage Efficiency

- Deduplication & Compression
- Compaction
- Thin provisioning
- Data Tiering (Fabric Pool)

Security & Compliance

- Fpolicy & Vscan
- Active Directory integration
- LDAP & Kerberos
- Certificate based authentication

- NetApp BlueXP **を使用すると、すべてのストレージ資産とデータ資産を単一のコントロールプレーン/インターフェイスから管理できます。

BlueXPを使用して、クラウドストレージ（Cloud Volumes ONTAP やAzure NetApp Files など）の作成と管理、データの移動、保護、分析、オンプレミスやエッジの多くのストレージデバイスの管理を行うことができます。

- NetApp Astra Trident **はCSI準拠のストレージオーケストレーションツールです。上記のさまざまなネットアップストレージオプションを利用して、永続的ストレージをすばやく簡単に利用できます。ネットアップが保守、サポートしているオープンソースのソフトウェアです。

Astra Trident CSI feature highlights

CSI specific

- CSI NetApp® Snapshot™ copies and volume creation from CSI Snapshot copies
- CSI topology
- Volume expansion

Security

- Dynamic-export policy management
- iSCSI initiator-groups dynamic management
- iSCSI bidirectional CHAP

Control

- Storage and performance consumption
- Monitoring
- Volume Import
- Cross Namespace Volume Access

Installation methods

- Binary
- Helm chart
- Operator
- GitOps

Choose your access mode

- RWO (ReadWriteOnce, i.e 1⇔1)
- RWX (*ReadWriteMany*, i.e 1⇔n)
- ROX (*ReadOnlyMany*)
- RWOP (ReadWriteOnce POD)

Choose your protocol

- NFS
- SMB
- iSCSI

ビジネスクリティカルなコンテナワークロードに必要なのは、永続的ボリュームだけではありません。同社のデータ管理要件では、アプリケーションのKubernetesオブジェクトの保護と移行も必要です。

> アプリケーションデータには、ユーザデータに加えてKubernetesオブジェクトが含まれます。例を次に示します。 -ポッド仕様、PVC、デプロイ、サービスなどのKubernetesオブジェクト- 設定マップやシークレットなどのカスタム設定オブジェクト- Snapshotコピー、バックアップ、クローンなどの永続的データ- CRSやCRDなどのカスタムリソース

- NetApp Astra Control **は、フルマネージドと自己管理型の両方のソフトウェアとして提供され、堅牢なアプリケーションデータ管理のためのオーケストレーションを提供します。を参照してください "Astra のドキュメント" を参照してください。

このリファレンスドキュメントでは、NetApp Astra Control Centerを使用して、Red Hat OpenShiftコンテナプラットフォームに導入されたコンテナベースアプリケーションの移行と保護について検証します。また、解決策 では、コンテナプラットフォームを管理するためのRed Hat Advanced Cluster Management（ACM）の導入と使用に関する詳細についても説明しています。また、Astra Trident CSIプロビジョニングツールを使用して、ネットアップストレージとRed Hat OpenShiftコンテナプラットフォームを統合する方法についても詳しく説明します。Astra Control Centerはハブクラスタに導入され、コンテナアプリケーションとその永続的ストレージライフサイクルの管理に使用されます。最後に、Amazon FSx for NetApp ONTAP （FSxN）を永続的ストレージとして使用し、AWS（ROSA）のマネージドRed Hat OpenShiftクラスタ上のコンテナワークロードのレプリケーション、フェイルオーバー、フェイルバックのための解決策 を提供します。

Red Hat OpenShift Containerワークロード向けネットアップハイブリッドマルチクラウドソリューションの価値提案

ほとんどのお客様は、既存のインフラがない状態でKubernetesベースの環境を構築し始めたばかりではありません。おそらく、大規模なVMware環境などで、エンタープライズアプリケーションのほとんどを仮想マシンで実行している従来型のIT環境です。その後、最新のアプリケーション開発チームのニーズを満たすために、小規模なコンテナベースの環境の構築を開始します。これらのイニシアチブは通常、小規模なものから始まり、チームがこれらの新しいテクノロジやスキルを学習し、それらを採用することの多くの利点を認識し始めるにつれて、より普及し始めます。ネットアップなら両方の環境のニーズに対応できるというのは、お客様にとって朗報です。Red Hat OpenShiftを使用したこのハイブリッドマルチクラウド向けソリューションセットは、ネットアップのお客様がインフラや組織全体を刷新することなく、最新のクラウドテクノロジとサービスを採用できるよう支援します。お客様のアプリケーションやデータがオンプレミスでホストされている場合でも、クラウドでホストされている場合でも、仮想マシンで実行されている場合でも、コンテナで実行される場合でも、ネットアップは一貫したデータ管理、保護、セキュリティ、モビリティを提供します。これらの新しいソリューションにより、ネットアップが数十年にわたってオンプレミスのデータセンター環境で提供してきたのと同じ価値を、企業全体のデータホライズン全体で利用できるようになります。ツールの再構築、新しいスキルの習得、新しいチームの構築に多額の投資を行う必要はありません。ネットアップは、お客様がクラウドへの移行のどの段階にいるかにかかわらず、これらのビジネス上の課題を解決できるよう、適切に位置付けられています。

Red Hat OpenShiftを使用したネットアップハイブリッドマルチクラウド：

- ネットアップベースのストレージソリューションでRed Hat OpenShiftを使用する場合に、お客様がデー

タとアプリケーションを管理、保護、保護、保護、移行するための最良の方法を実証する事前検証済みの設計と手法をお客様に提供します。

- VMware環境、ベアメタルインフラ、またはその両方でネットアップストレージを使用してRed Hat OpenShiftを実行しているお客様向けのベストプラクティスを紹介します。
- オンプレミス環境とクラウド環境、およびその両方を使用するハイブリッド環境の両方について、戦略とオプションを説明する。

Red Hat OpenShift Containerワークロード向けのネットアップハイブリッドマルチクラウドのサポート対象ソリューション

解決策 は、OpenShiftコンテナプラットフォーム（OCP）、OpenShift Advanced Cluster Manager（ACM）、NetApp ONTAP 、NetApp BlueXP、NetApp Astra Control Center（ACC）を使用した移行と一元的なデータ保護のテストと検証を行います。

この解決策 では、次のシナリオがネットアップによってテストおよび検証されます。解決策 は、次の特性に基づいて複数のシナリオに分けられます。

- オンプレミス
- クラウド
  - 自己管理型OpenShiftクラスタと自己管理型ネットアップストレージ
  - プロバイダが管理するOpenShiftクラスタとプロバイダが管理するネットアップストレージ

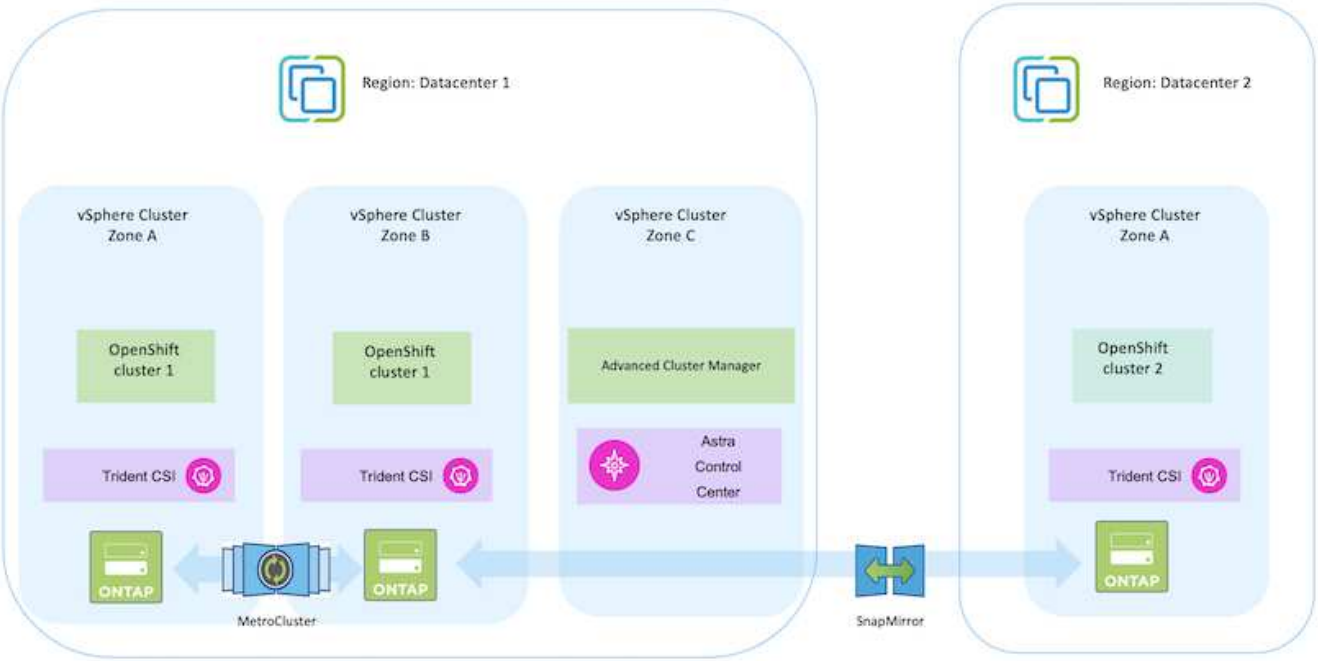
今後、追加のソリューションとユースケースを構築していきます。

シナリオ**1**：**ACC**を使用したオンプレミス環境内でのデータ保護と移行

オンプレミス：自己管理型**OpenShift**クラスタと自己管理型ネットアップストレージ

- ACCを使用して、データ保護のためにSnapshotコピー、バックアップ、リストアを作成します。
- ACCを使用して、コンテナアプリケーションのSnapMirrorレプリケーションを実行します。

シナリオ **1**

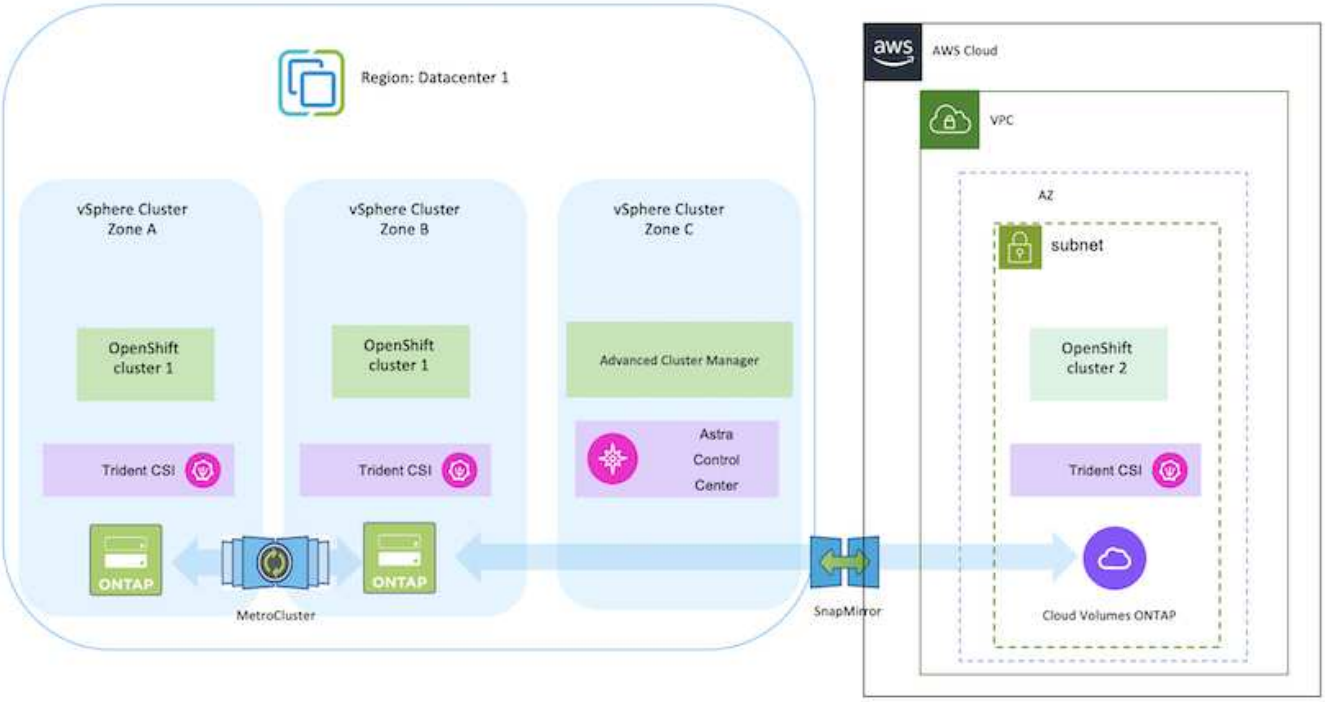


シナリオ**2**：**ACC**を使用したオンプレミス環境から**AWS**環境へのデータ保護と移行

オンプレミス：自己管理型**OpenShift**クラスタと自己管理型ストレージ AWSクラウド：自己管理型OpenShiftクラスタと自己管理型ストレージ**

- ACCを使用して、データ保護のためのバックアップとリストアを実行します。
- ACCを使用して、コンテナアプリケーションのSnapMirrorレプリケーションを実行します。

シナリオ **2**

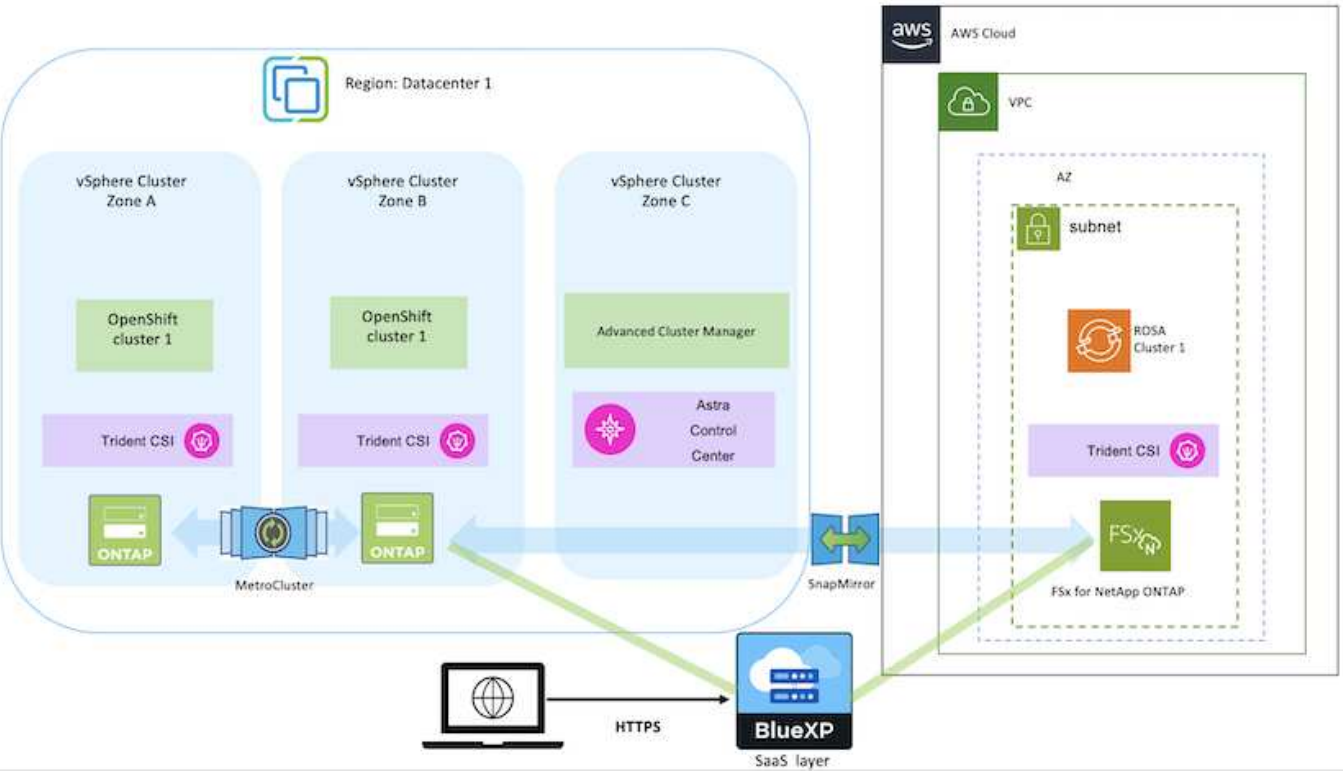


シナリオ**3**：オンプレミス環境から**AWS**環境へのデータ保護と移行

オンプレミス：自己管理型**OpenShift**クラスタと自己管理型ストレージ AWSクラウド：プロバイダ管理型OpenShiftクラスタ（ROSA）とプロバイダ管理型ストレージ（FSxN）**

- BlueXPを使用して永続ボリュームのレプリケーション（FSxN）を実行
- OpenShift GitOpsを使用して、アプリケーションメタデータを再作成します。

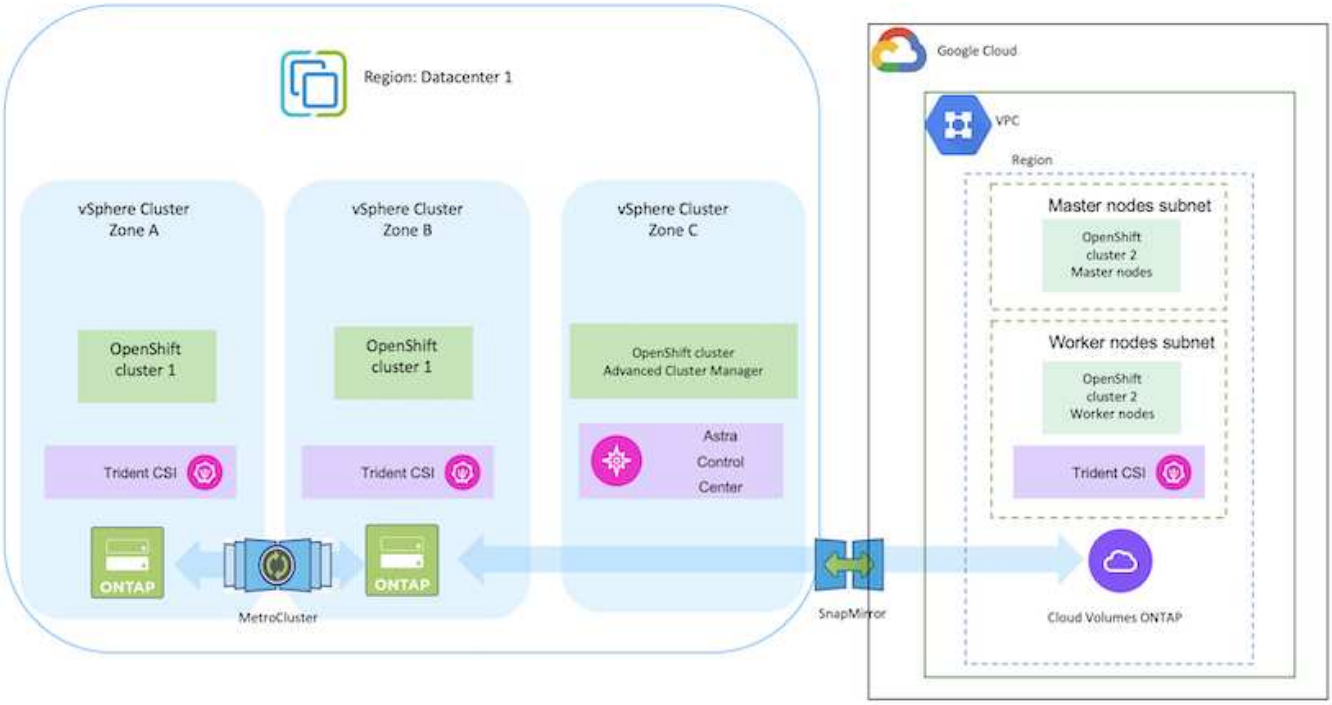
シナリオ**3**



シナリオ**4**：**ACC**を使用したオンプレミス環境から**GCP**環境へのデータ保護と移行

オンプレミス：自己管理型**OpenShift**クラスタと自己管理型ストレージ
Google Cloud：自己管理型**OpenShift**クラスタと自己管理型ストレージ

- ACCを使用して、データ保護のためのバックアップとリストアを実行します。
- ACCを使用して、コンテナアプリケーションのSnapMirrorレプリケーションを実行します。

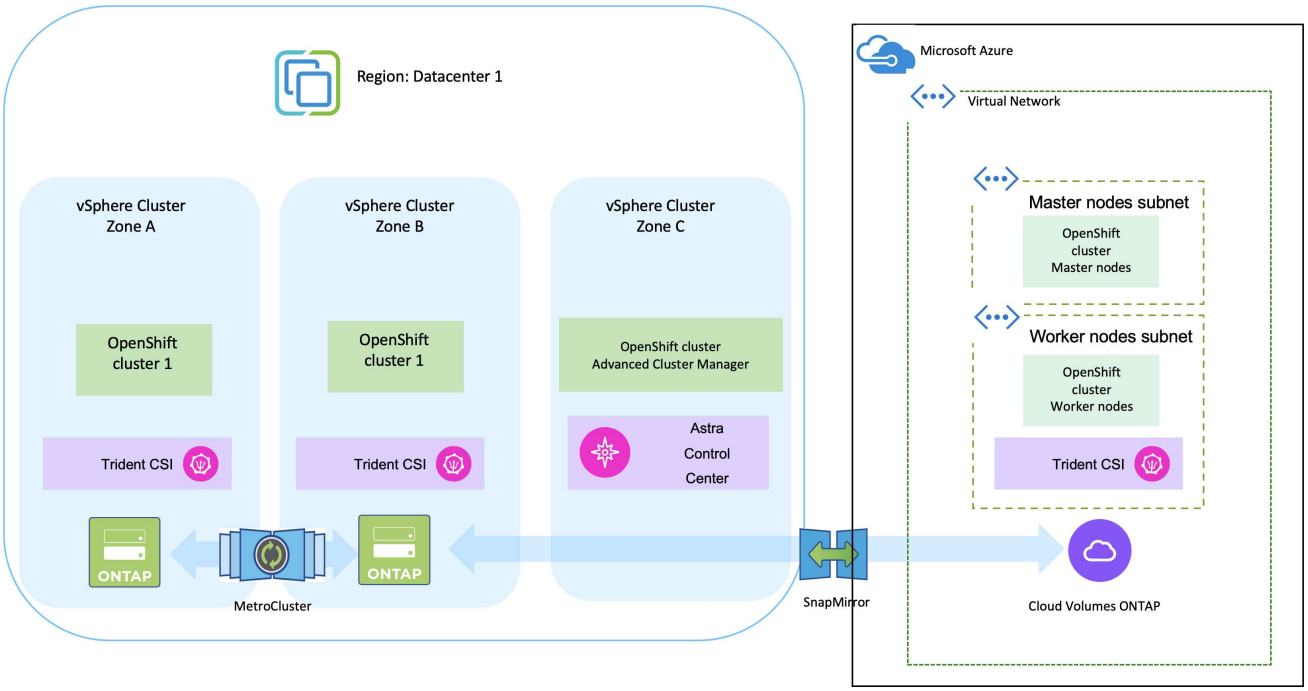


MetroCluster 構成でONTAP を使用する場合の考慮事項については、を参照してください "こちらをご覧ください"。

シナリオ**5**：**ACC**を使用したオンプレミス環境から**Azure**環境へのデータ保護と移行

オンプレミス：自己管理型**OpenShift**クラスタと自己管理型ストレージ
Azureクラウド：自己管理型**OpenShift**クラスタと自己管理型ストレージ

- ACCを使用して、データ保護のためのバックアップとリストアを実行します。
- ACCを使用して、コンテナアプリケーションのSnapMirrorレプリケーションを実行します。



MetroCluster 構成でONTAP を使用する場合の考慮事項については、を参照してください "こちらをご覧ください"。

解決策 検証で使用されるさまざまなコンポーネントのバージョン

解決策 は、OpenShiftコンテナプラットフォーム、OpenShift Advanced Cluster Manager、NetApp ONTAP 、NetApp Astra Control Centerを使用した移行と一元的なデータ保護のテストと検証を行います。

解決策のシナリオ1、2、3は、次の表に示すバージョンを使用して検証されました。

* コンポーネント *	* バージョン *
* VMware *	vSphere Clientバージョン8.0.0.10200 VMware ESXi、8.0.0、20842819

ハブクラスタ	OpenShift 4.11.34
ソースクラスタとデスティネーションクラスタ	オンプレミスとAWSでのOpenShift 4.12.9
* NetApp Astra Trident *	Tridentサーバとクライアント23.04.0
* NetApp Astra Control Center *	ACC 22.11.0-82
* NetApp ONTAP *	ONTAP 9.12.1
* AWS FSx for NetApp ONTAP *	シングルAZ

解決策のシナリオ4は、次の表に示すバージョンを使用して検証されました。

* コンポーネント *	* バージョン *
* VMware *	vSphere Clientバージョン8.0.2.00000 VMware ESXi、8.0.2、22380479
ハブクラスタ	OpenShift 4.13.13
ソースクラスタとデスティネーションクラスタ	OpenShift 4.13.12 オンプレミスとGoogle Cloud
* NetApp Astra Trident *	Tridentサーバおよびクライアント23.07.0
* NetApp Astra Control Center *	ACC 23.07.0-25
* NetApp ONTAP *	ONTAP 9.12.1
* Cloud Volumes ONTAP *	シングルAZ、シングルノード、9.14.0

解決策のシナリオ5は、次の表に示すバージョンを使用して検証されました。

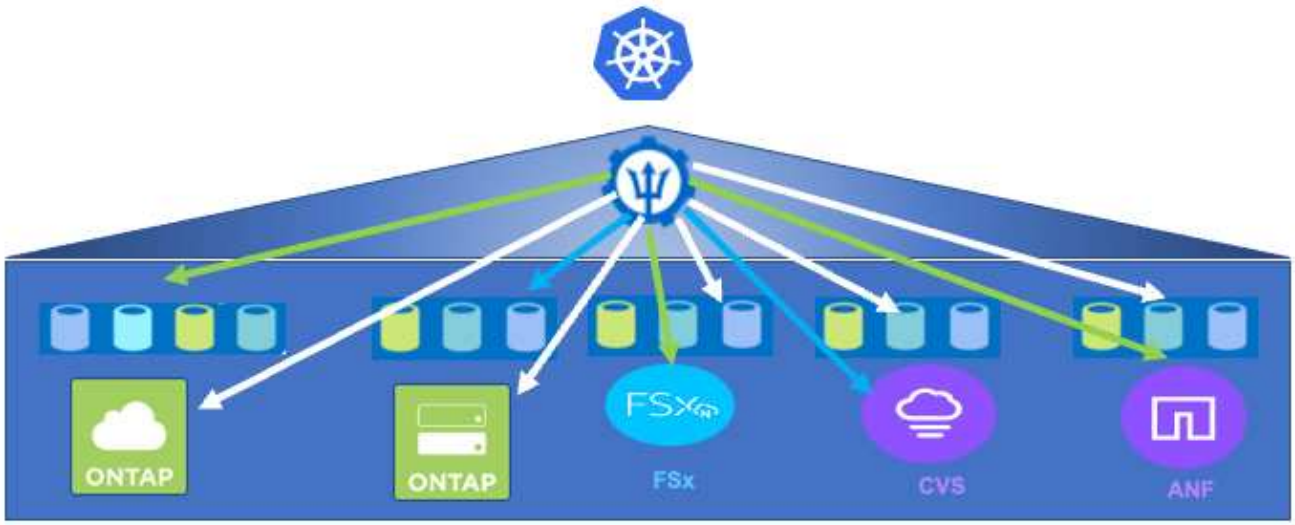
* コンポーネント *	* バージョン *
* VMware *	vSphere Clientバージョン8.0.2.00000 VMware ESXi、8.0.2、22380479
ソースクラスタとデスティネーションクラスタ	OpenShift 4.13.25 オンプレミスとAzure
* NetApp Astra Trident *	Tridentサーバとクライアント、Astra Controlプロビジョニングツール23.10.0
* NetApp Astra Control Center *	ACC 23.10
* NetApp ONTAP *	ONTAP 9.12.1
* Cloud Volumes ONTAP *	シングルAZ、シングルノード、9.14.0

Red Hat OpenShift Containersとのネットアップストレージ統合がサポートされています

Red Hat Open Shiftコンテナをvmwareで実行する場合でも、ハイパースケーラで実行す

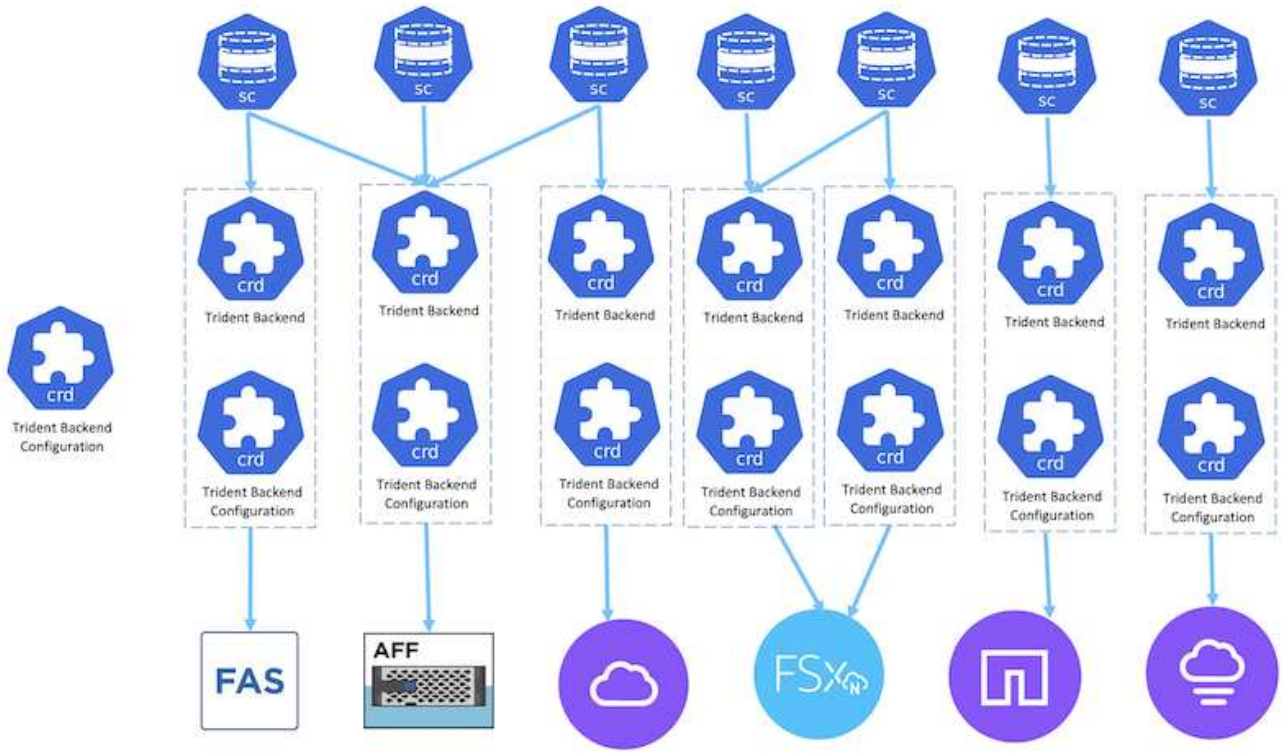
る場合でも、NetApp Astra Tridentは、サポートするさまざまなタイプのバックエンドネットアップストレージのCSIプロビジョニングツールとして使用できます。

次の図は、NetApp Astra Tridentを使用してOpenShiftクラスタと統合できるバックエンドのネットアップストレージを示しています。

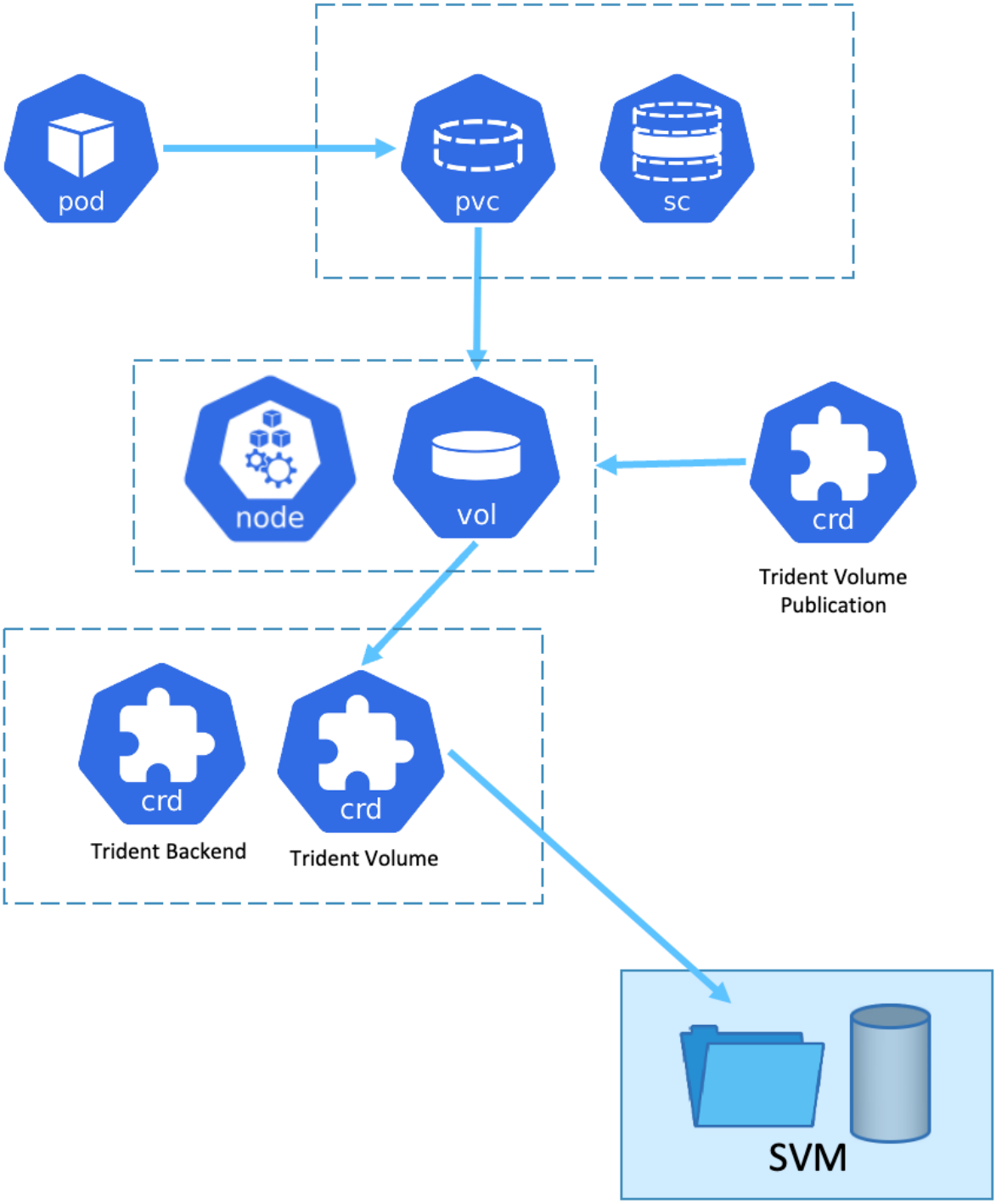


ONTAP Storage Virtual Machine（SVM）はセキュアマルチテナンシーを提供します。単一のOpenShiftクラスタは、単一のSVMまたは複数のSVMに接続することも、複数のONTAP クラスタに接続することもできます。ストレージクラスは、パラメータまたはラベルに基づいてバックエンドストレージをフィルタリングします。ストレージ管理者は、Tridentバックエンド構成を使用してストレージシステムに接続するためのパラメータを定義します。接続が正常に確立されると、Tridentバックエンドが作成され、ストレージクラスでフィルタできる情報が入力されます。

ストレージクラスとバックエンドの関係を次に示します。



アプリケーション所有者がストレージクラスを使用して永続ボリュームを要求します。バックエンドストレージはストレージクラスでフィルタリングされます。ポッドとバックエンドストレージの関係を以下に示します。



Container Storage Interface（CSI）オプション

vSphere環境では、VMware CSIドライバやAstra Trident CSIを選択してONTAP と統合できます。VMware CSIでは永続ボリュームがローカルSCSIディスクとして使用され、Tridentではネットワークが使用されます。VMware CSIはONTAP でのRWXアクセスモードをサポートしていないため、RWXモードが必要な場合は、アプリケーションでTrident CSIを使用する必要があります。FCベースの導入ではVMware CSIが推奨され、SnapMirror Business Continuity（SMBC）によってゾーンレベルの高可用性が実現されます。

VMware CSIがサポートします

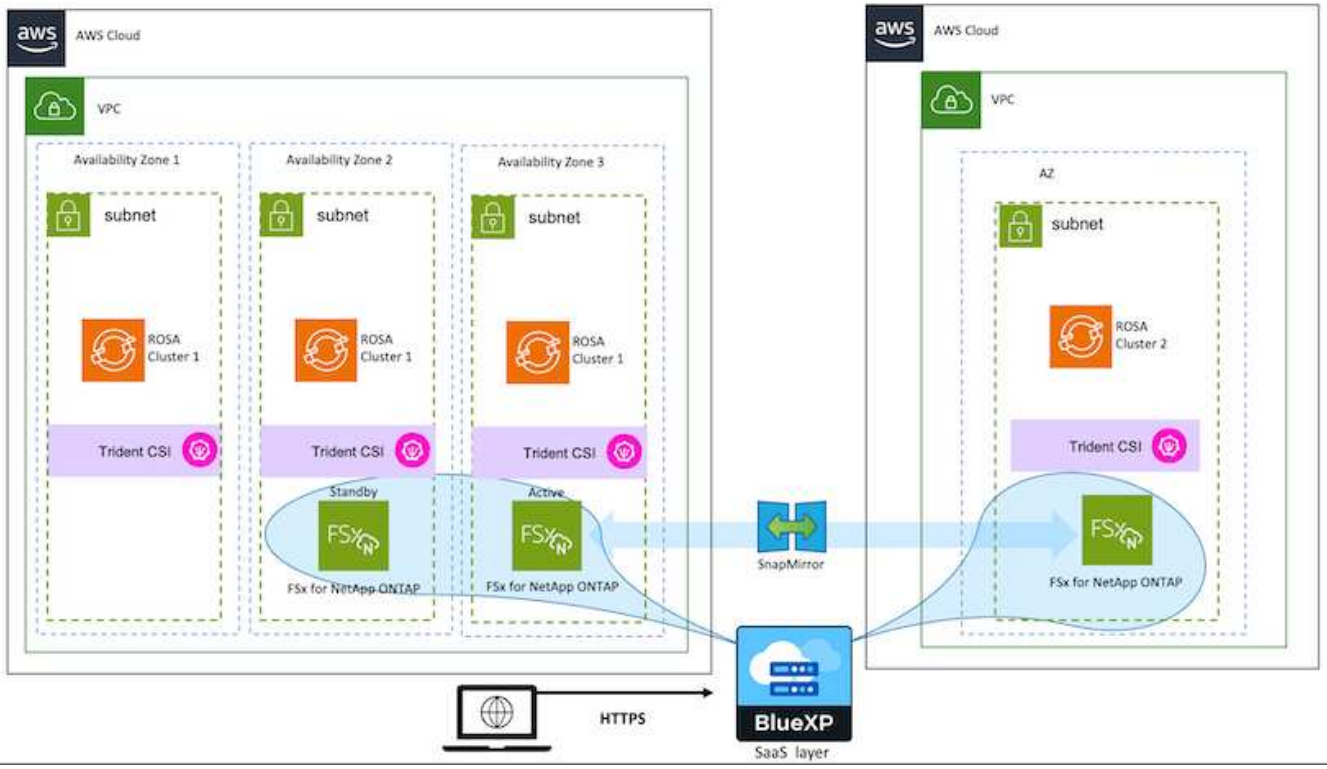
- ブロックベースのコアデータストア（FC、FCoE、iSCSI、NVMeoF）
- コアファイルベースのデータストア（NFS v3、v4）
- VVolデータストア（ブロックとファイル）

Tridentには、**ONTAP** をサポートするために次のドライバがあります

- ONTAP-SAN（専用ボリューム）
- ONTAP-SANの経済性（共有ボリューム）
- ONTAP-NAS（専用ボリューム）
- ONTAP-NASの経済性（共有ボリューム）
- ontap-nas-flexgroup（専用の大規模ボリューム）

ONTAP は、VMware CSIとAstra Trident CSIのどちらについても、NFSとマルチパスのnconnect、セッショントランキング、Kerberosなど、ブロックプロトコルのCHAP認証などをサポートします。

AWSでは、FSx for NetApp ONTAP （FSxN）を単一のアベイラビリティゾーン（AZ）または複数のAZに導入できます。高可用性を必要とする本番ワークロードに対しては、複数のAZを使用することでゾーンレベルのフォールトトレランスが実現し、NVMe読み取りキャッシュも単一のAZよりも優れています。詳細については、を参照してください "AWSパフォーマンスのガイドライン"。
ディザスタリカバリサイトのコストを削減するために、単一のAZ FSx ONTAP を利用できます。



FSx ONTAP でサポートされるSVMの数については、を参照してください "FSx ONTAP Storage Virtual Machineの管理"

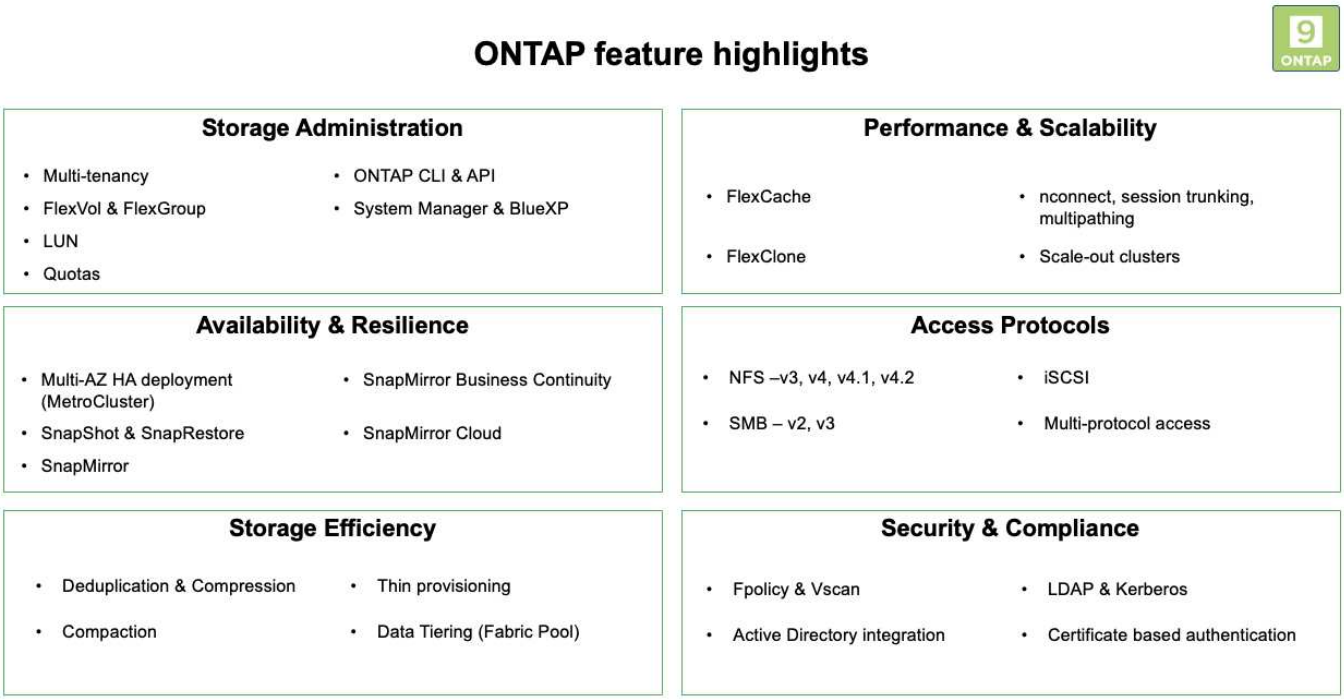
Red Hat OpenShift Containerワークロード向けのネットアップハイブリッドマルチクラウドソリューション

概要

ネットアップでは、従来型エンタープライズアプリケーションを最新化し、Kubernetesを中心に構築されたコンテナとオーケストレーションプラットフォームを使用して新しいアプリケーションを構築するお客様が大幅に増えています。Red Hat OpenShift Container Platformは、多くのお客様に採用されている例の1つです。

企業内でコンテナを採用するお客様がますます増えています。ネットアップは、ステートフルアプリケーションの永続的ストレージのニーズに加え、データ保護、データセキュリティ、データ移行などの従来のデータ管理のニーズにも応えることができます。しかし、これらのニーズは、さまざまな戦略、ツール、方法を使用して満たしています。

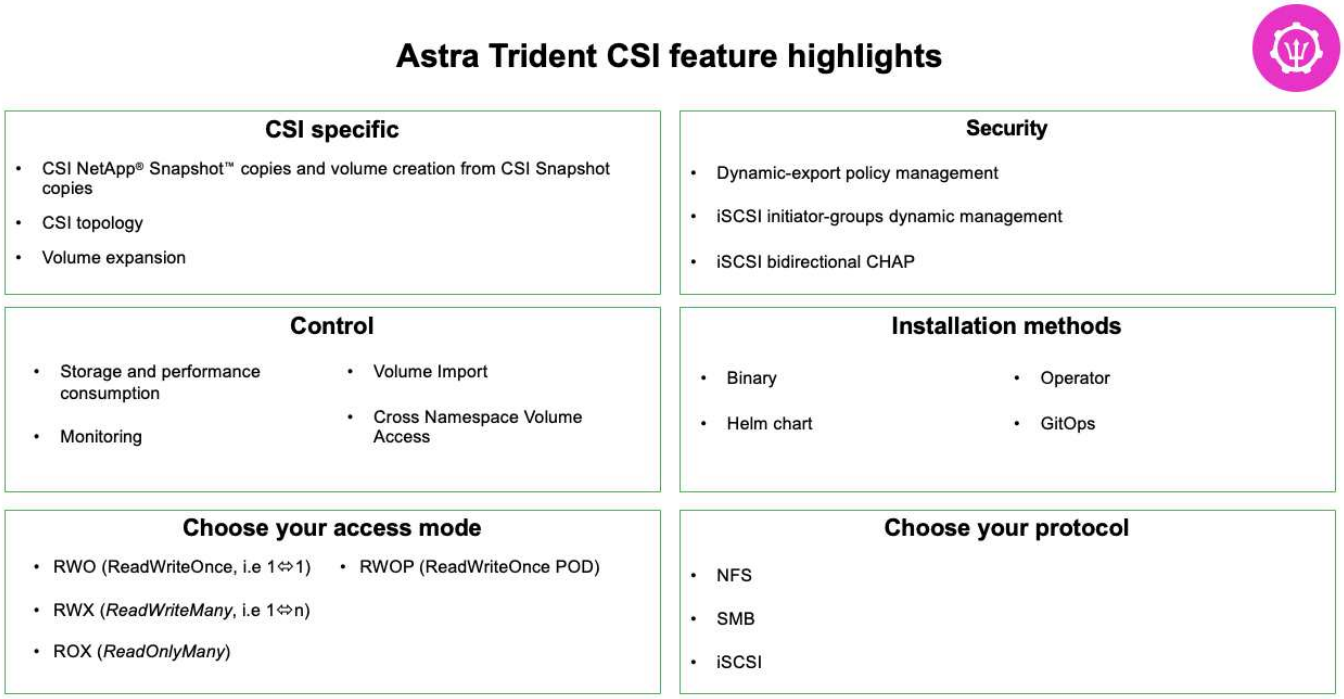
- NetApp ONTAP **ベースのストレージオプションを次に示します。コンテナとKubernetes環境にセキュリティ、データ保護、信頼性、柔軟性を提供します。
  - オンプレミスの自己管理型ストレージ：
- ネットアップファブリック接続ストレージ（FAS ）、ネットアップオールフラッシュFAS アレイ（AFF ）、ネットアップオールSANアレイ（ASA ）、ONTAP Select
  - オンプレミスのプロバイダ管理ストレージ：
- NetApp Keystone が提供するストレージサービス（STaaS）
  - クラウド内の自己管理型ストレージ：
- NetApp Cloud Volumes ONTAP（CVO）は、ハイパースケーラに自己管理型ストレージを提供します
  - クラウド上のプロバイダが管理するストレージ：
- Cloud Volumes Service for Google Cloud（CVS）、Azure NetApp Files （ANF）、Amazon FSx for NetApp ONTAP は、ハイパースケーラにフルマネージドストレージを提供します



- NetApp BlueXP **を使用すると、すべてのストレージ資産とデータ資産を単一のコントロールプレーン/インターフェイスから管理できます。

BlueXPを使用して、クラウドストレージ（Cloud Volumes ONTAP やAzure NetApp Files など）の作成と管理、データの移動、保護、分析、オンプレミスやエッジの多くのストレージデバイスの管理を行うことができます。

- NetApp Astra Trident **はCSI準拠のストレージオーケストレーションツールです。上記のさまざまなネットアップストレージオプションを利用して、永続的ストレージをすばやく簡単に利用できます。ネットアップが保守、サポートしているオープンソースのソフトウェアです。



ビジネスクリティカルなコンテナワークロードに必要なのは、永続的ボリュームだけではありません。同社のデータ管理要件では、アプリケーションのKubernetesオブジェクトの保護と移行も必要です。

> アプリケーションデータには、ユーザデータに加えてKubernetesオブジェクトが含まれます。例を次に示します。 -ポッド仕様、PVC、デプロイ、サービスなどのKubernetesオブジェクト- 設定マップやシークレットなどのカスタム設定オブジェクト- Snapshotコピー、バックアップ、クローンなどの永続的データ- CRSやCRDなどのカスタムリソース

- NetApp Astra Control **は、フルマネージドと自己管理型の両方のソフトウェアとして提供され、堅牢なアプリケーションデータ管理のためのオーケストレーションを提供します。を参照してください "Astra のドキュメント" を参照してください。

このリファレンスドキュメントでは、NetApp Astra Control Centerを使用して、Red Hat OpenShiftコンテナプラットフォームに導入されたコンテナベースアプリケーションの移行と保護について検証します。また、解決策 では、コンテナプラットフォームを管理するためのRed Hat Advanced Cluster Management（ACM）の導入と使用に関する詳細についても説明しています。また、Astra Trident CSIプロビジョニングツールを使用して、ネットアップストレージとRed Hat OpenShiftコンテナプラットフォームを統合する方法についても詳しく説明します。Astra Control Centerはハブクラスタに導入され、コンテナアプリケーションとその永続的ストレージライフサイクルの管理に使用されます。最後に、Amazon FSx for NetApp ONTAP （FSxN）を永続的ストレージとして使用し、AWS（ROSA）のマネージドRed Hat OpenShiftクラスタ上のコンテナワーク

ロードのレプリケーション、フェイルオーバー、フェイルバックのための解決策 を提供します。

VMware上でのNetApp解決策 とRed Hat OpenShift Containerプラットフォームのワークロード

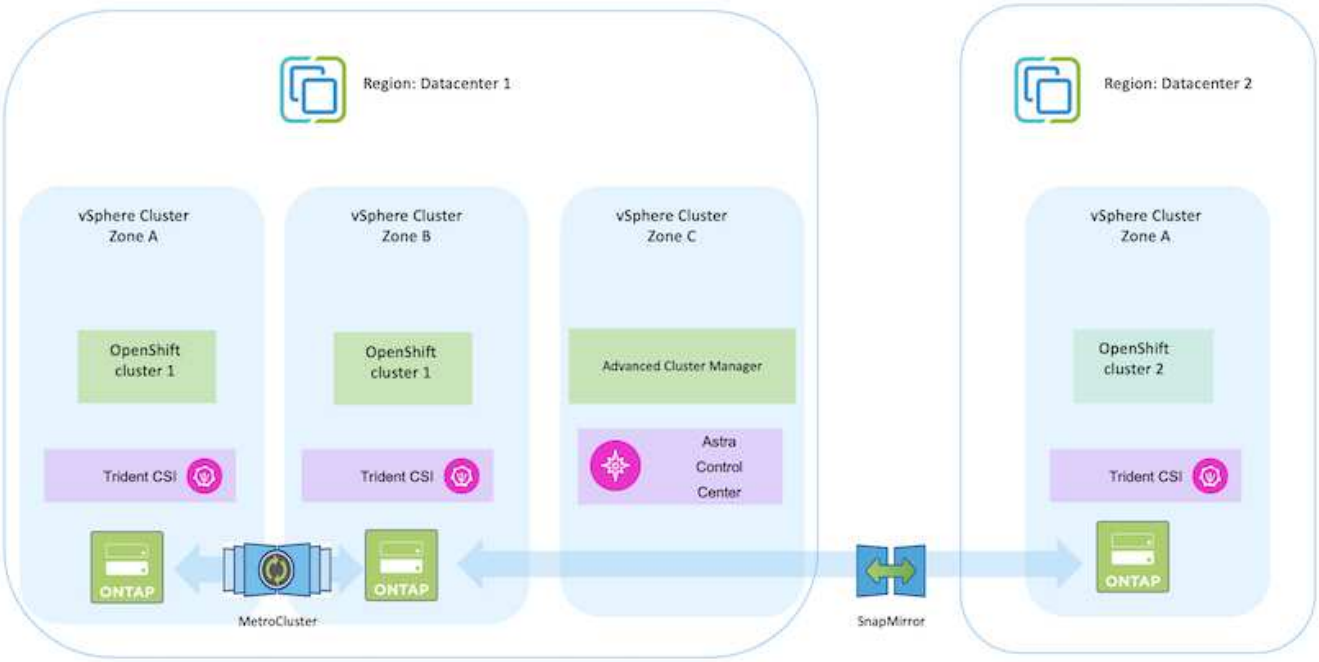
最新のコンテナ化されたアプリケーションをプライベートデータセンターのインフラで実行する必要がある場合は、実行できます。コンテナワークロードを導入するための本番環境向け環境を成功させるためには、Red Hat OpenShiftコンテナプラットフォーム（OCP）の計画と導入が必要です。OCPクラスタは、VMwareまたはベアメタルに導入できます。

NetApp ONTAP ストレージは、コンテナ導入にデータ保護、信頼性、柔軟性を提供します。Astra Tridentは、お客様のステートフルアプリケーション向けに永続的ONTAP ストレージを利用するための動的ストレージプロビジョニングツールとして機能します。Astra Control Centerを使用すると、データ保護、移行、ビジネス継続性など、ステートフルアプリケーションに求められる多くのデータ管理要件をオーケストレーションできます。

VMware vSphereでは、NetApp ONTAP toolsがvCenterプラグインを提供し、データストアのプロビジョニングに使用できます。タグを適用し、OpenShiftでノードの設定とデータを格納するために使用します。NVMeベースのストレージは、低レイテンシと高パフォーマンスを実現します。

この解決策 では、Astra Control Centerを使用したコンテナワークロードのデータ保護と移行について詳しく説明します。この解決策 では、オンプレミス環境内のvSphere上のRed Hat OpenShiftクラスタにコンテナワークロードが導入されます。注：今後、ベアメタル上のOpenShiftクラスタ上のコンテナワークロード向けに解決策 を提供する予定です。

Astra Control Centerを使用した**OpenShift**コンテナワークロード向けのデータ保護と移行の解決策

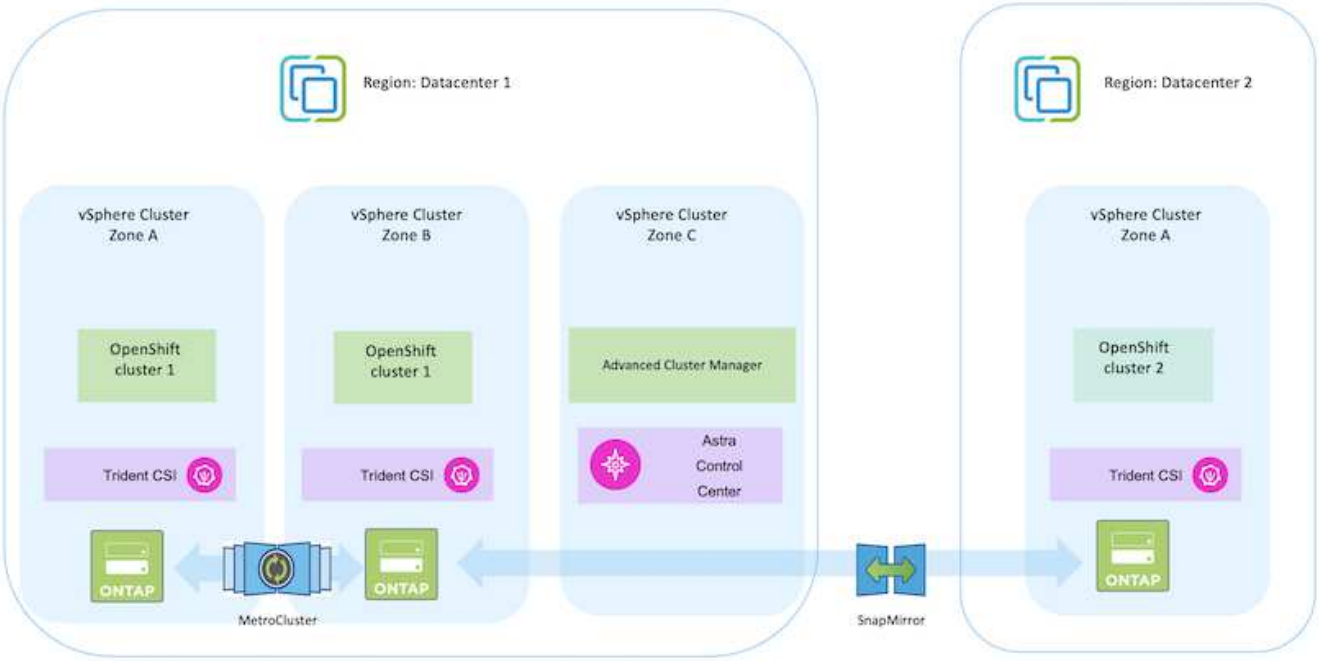


VMwareにRed Hat OpenShift Containerプラットフォームを導入して設定します

このセクションでは、OpenShiftクラスタをセットアップおよび管理し、クラスタ上でステートフルアプリケーションを管理する方法の大まかなワークフローについて説明します。このスライドでは、NetApp ONTAP ストレージアレイとAstra Tridentを使用して永続ボリュームを提供する方法を示しています。ステートフルアプリケーションに対してデータ保護と移行のアクティビティを実行するためのAstra Control Centerの使用方法について詳しく説明します。

> Red Hat OpenShift Containerプラットフォームクラスタは、いくつかの方法で導入できます。このセットアップの概要概要 には、使用した具体的な方法のドキュメントへのリンクが記載されています。その他の方法については、に記載されている関連リンクを参照してください "リソースセクション"。

次の図は、データセンターのVMwareに導入されたクラスタを示しています。



セットアッププロセスは、次の手順に分けることができます。

CentOS VMを導入、設定

- VMware vSphere環境に導入されます。
- このVMは、NetApp Astra TridentやNetApp Astra Control Center for the解決策 など、一部のコンポーネントの導入に使用されます。
- このVMにはインストール時にrootユーザが設定されます。

VMware vSphere（ハブクラスタ）に**OpenShift Container Platform**クラスタを導入して設定

の手順を参照してください "支援された展開" OCPクラスタを導入する方法。

> 次の点に注意してください。-インストーラに提供するsshの公開鍵と秘密鍵を作成します。これらのキーは、必要に応じてマスターノードとワーカーノードにログインするために使用されます。-アシスタントインストーラからインストーラプログラムをダウンロードします。このプログラムを使用して、VMware vSphere環境でマスターノードとワーカーノード用に作成したVMをブートします。- VMには、CPU、メモリ、およびハードディスクの最小要件が必要です。（のvm createコマンドを参照してください "これ" この情報を提供するマスターノードとワーカーノードのページ）-すべてのVMでdiskUUIDを有効にする必要があります。-マスター用に最低3ノード、ワーカー用に3ノードを作成します。-インストーラによって検出されたら、VMware vSphere統合トグルボタンをオンにします。

ハブクラスタに**Advanced Cluster Management**をインストールします

これは、ハブクラスタのAdvanced Cluster Management Operatorを使用してインストールします。手順を参照してください "こちらをご覧ください"。

ハブクラスタに内部**Red Hat Quay**レジストリをインストールします。

- Astraイメージをプッシュするには内部レジストリが必要です。Quay内部レジストリは、HubクラスタのOperatorを使用してインストールされます。
- 手順を参照してください "こちらをご覧ください"

2つの**OCP**クラスタ（ソースとデスティネーション）を追加でインストール

- 追加のクラスタは、ハブクラスタのACMを使用して展開できます。
- 手順を参照してください "こちらをご覧ください"。

NetApp ONTAP ストレージの設定

- VMware環境のOCP VMに接続されたONTAP クラスタをインストールします。
- SVMを作成
- SVMのストレージにアクセスするようにNASデータLIFを設定します。

OCPクラスタに**NetApp Trident**をインストール

- ハブ、ソース、デスティネーションの3つのクラスタすべてにNetApp Tridentをインストール
- 手順を参照してください "こちらをご覧ください"。
- ONTAP-NAS用のストレージバックエンドを作成
- ONTAP-NAS用のストレージクラスを作成
- 手順を参照してください "こちらをご覧ください"。

NetApp Astra Control Centerをインストール

- NetApp Astra Control Centerは、ハブクラスタでAstra Operatorを使用してインストールします。
- 手順を参照してください "こちらをご覧ください"。

覚えておくべきポイント：*サポートサイトからNetApp Astra Control Centerのイメージをダウンロード* イメージを内部レジストリにプッシュします。*こちらの手順を参照してください。

ソースクラスタにアプリケーションを導入します

OpenShift GitOpsを使用してアプリケーションを導入します。（例：Postgres、Ghost）

ソースクラスタとデスティネーションクラスタを**Astra Control Center**に追加

Astra Controlの管理にクラスタを追加したら、（Astra Control以外の）クラスタにアプリケーションをインストールし、Astra Controlの[Applications]ページに移動してアプリケーションとそのリソースを定義できます。を参照してください "Astra Control Centerのアプリケーションの管理セクションを開始します"。

次の手順では、Astra Control Centerを使用して、ソースクラスタからデスティネーションクラスタへのデータ保護とデータ移行を行います。

Astraを使用したデータ保護

このページには、Astra Control Center（ACC）を使用してVMware vSphereで実行されるRed Hat OpenShift Containerベースのアプリケーションのデータ保護オプションが表示されます。

ユーザがRed Hat OpenShiftを使用してアプリケーションを最新化する過程で、偶発的な削除やその他の人的エラーからユーザを保護するためのデータ保護戦略を策定する必要があります。多くの場合、データを管理から保護するために、規制やコンプライアンスの目的で保護戦略が必要になります。

データ保護の要件は、ポイントインタイムコピーへのリバートから別の障害ドメインへの自動フェイルオーバーまで、人手を介さずにさまざまです。多くのお客様がONTAP をKubernetesアプリケーションに最適なストレージプラットフォームとして選択しています。その理由は、マルチテナンシー、マルチプロトコル、ハイパフォーマンスと容量のサービス、マルチサイト環境のレプリケーションとキャッシュ、セキュリティと柔軟性などの豊富な機能があるからです。

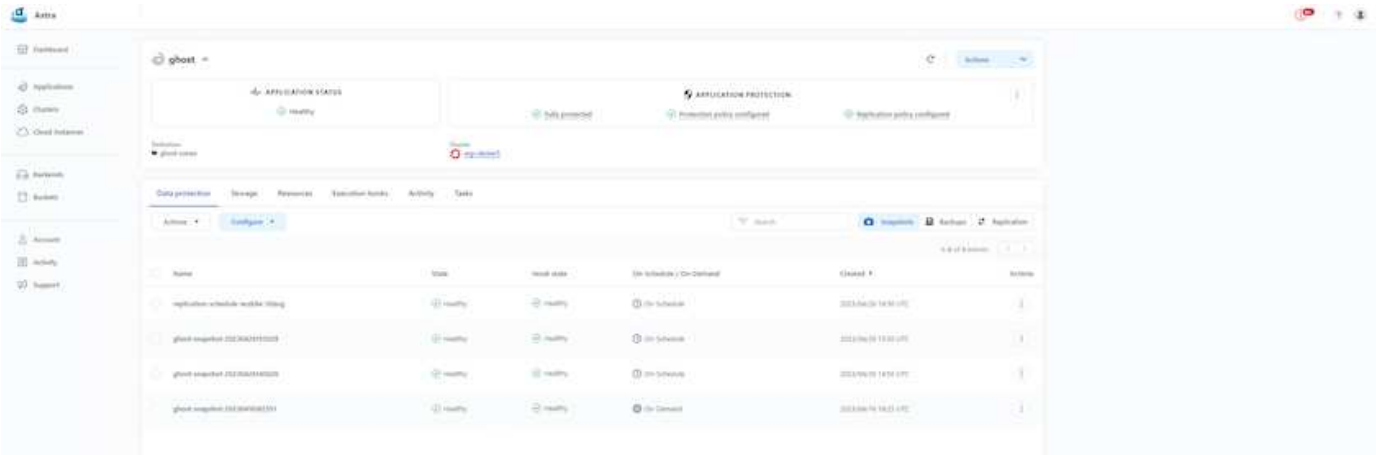
ONTAP のデータ保護は、アドホックまたはポリシー制御の-スナップショット*-バックアップおよびリストアを使用して実現できます

Snapshotコピーとバックアップのどちらも、次のタイプのデータを保護します。-アプリケーションの状態を表すアプリケーションメタデータ-アプリケーションに関連付けられた永続的データボリューム-アプリケーションに属するリソースアーティファクト

ACCを使用したスナップショット

SnapshotとACCを使用して、データのポイントインタイムコピーをキャプチャできます。保護ポリシーでは、保持するSnapshotコピーの数を定義します。最小スケジュールオプションは毎時です。オンデマンドで手動のSnapshotコピーをいつでも、スケジュールされたSnapshotコピーよりも短い間隔で作成できます。Snapshotコピーは、アプリケーションと同じプロビジョニングされたボリュームに格納されます。

ACCでスナップショットを設定しています

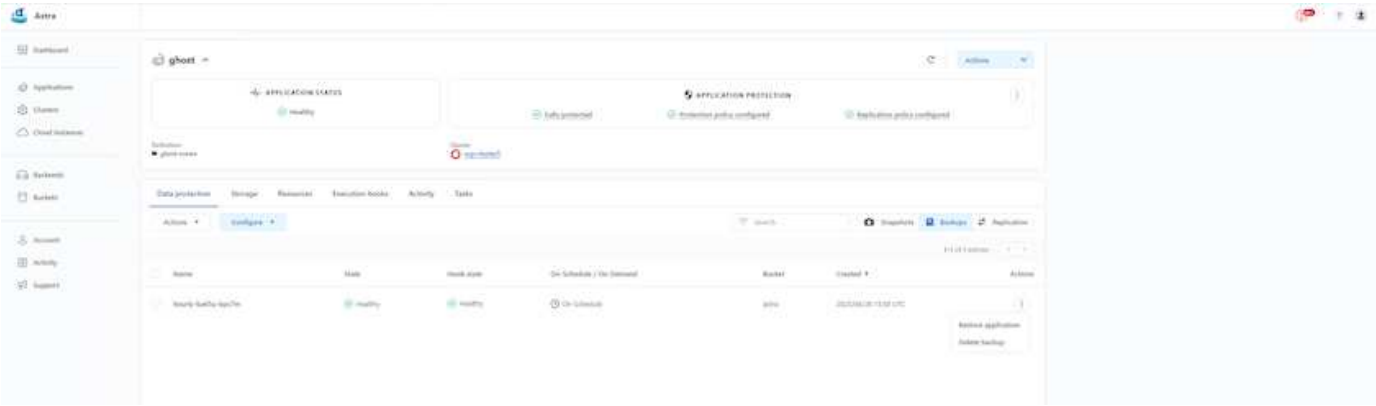


ACCを使用したバックアップと復元

バックアップはSnapshotに基づいています。ACCはCSIを使用してSnapshotコピーを作成し、ポイントインタイムSnapshotコピーを使用してバックアップを実行できます。バックアップは外部のオブジェクトストア（別の場所にあるONTAP S3を含むs3互換）に格納されます。スケジュールされたバックアップの保護ポリシーと保持するバックアップバージョンの数を設定できます。最小RPOは1時間です。

ACCを使用したバックアップからのアプリケーションのリストア

ACCは、バックアップが格納されているS3バケットからアプリケーションをリストアします。



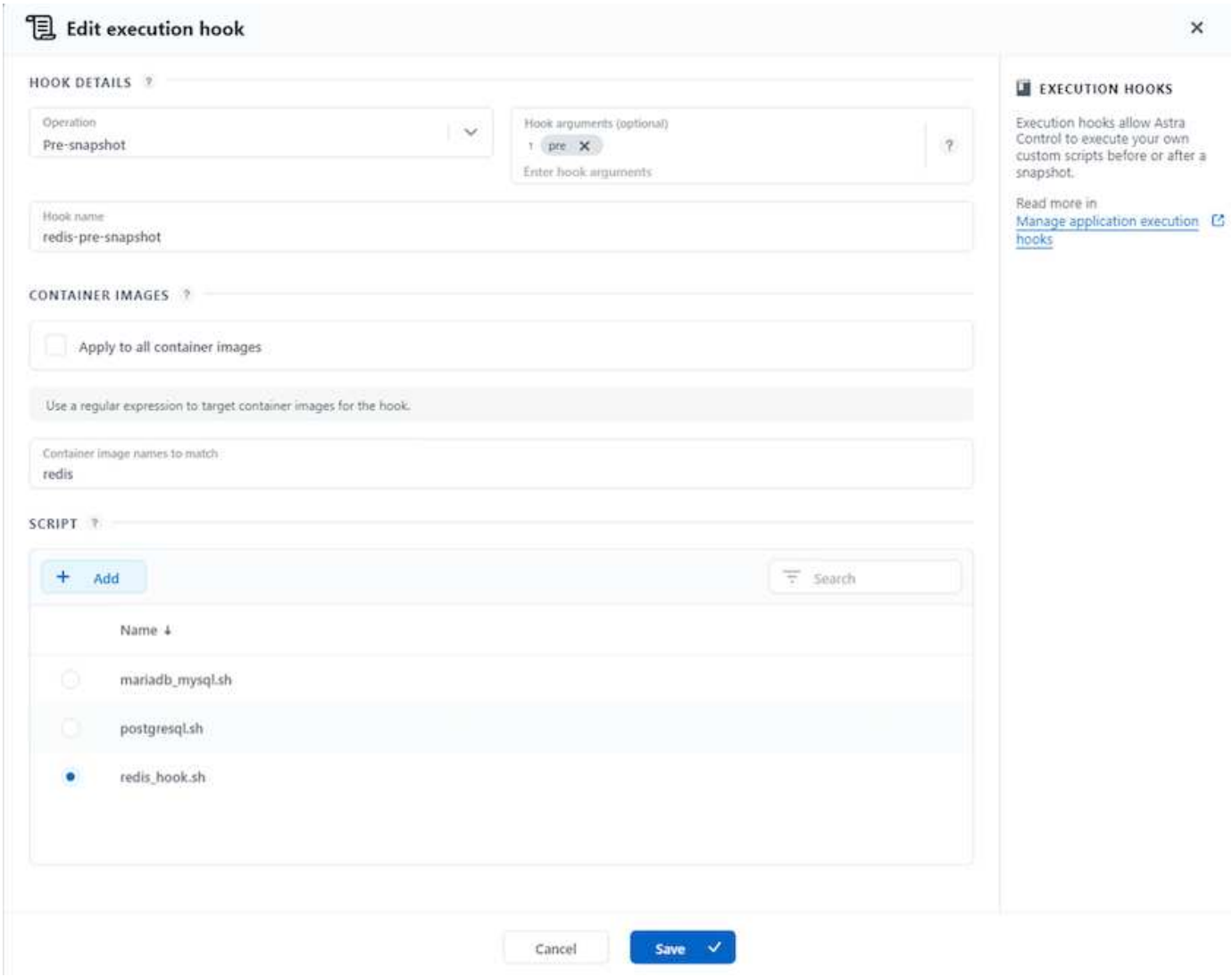
アプリケーション固有の実行フック

さらに、実行フックは、管理対象アプリのデータ保護操作と組み合わせて実行するように構成することができます。ストレージアレイレベルのデータ保護機能を使用できますが、バックアップとリストアでアプリケーションとの整合性を確保するために追加の手順が必要になることがよくあります。アプリケーション固有の追加手順は次のとおりです。- Snapshotコピーの作成前または作成後。-バックアップの作成前または作成後。- Snapshotコピーまたはバックアップからリストアしたあと。

Astra Controlでは、実行フックと呼ばれるカスタムスクリプトとしてコード化されたアプリケーション固有の手順を実行できます。

"NetApp Verda GitHubプロジェクト" 一般的なクラウドネイティブアプリケーションの実行フックを提供し、アプリケーションを簡単に保護し、堅牢で、オーケストレーションを容易にします。リポジトリにないアプリケーションに十分な情報がある場合は、そのプロジェクトに貢献してください。

redisアプリケーションの**pre-Snapshot**用のサンプル実行フック。



ACCを使用したレプリケーション

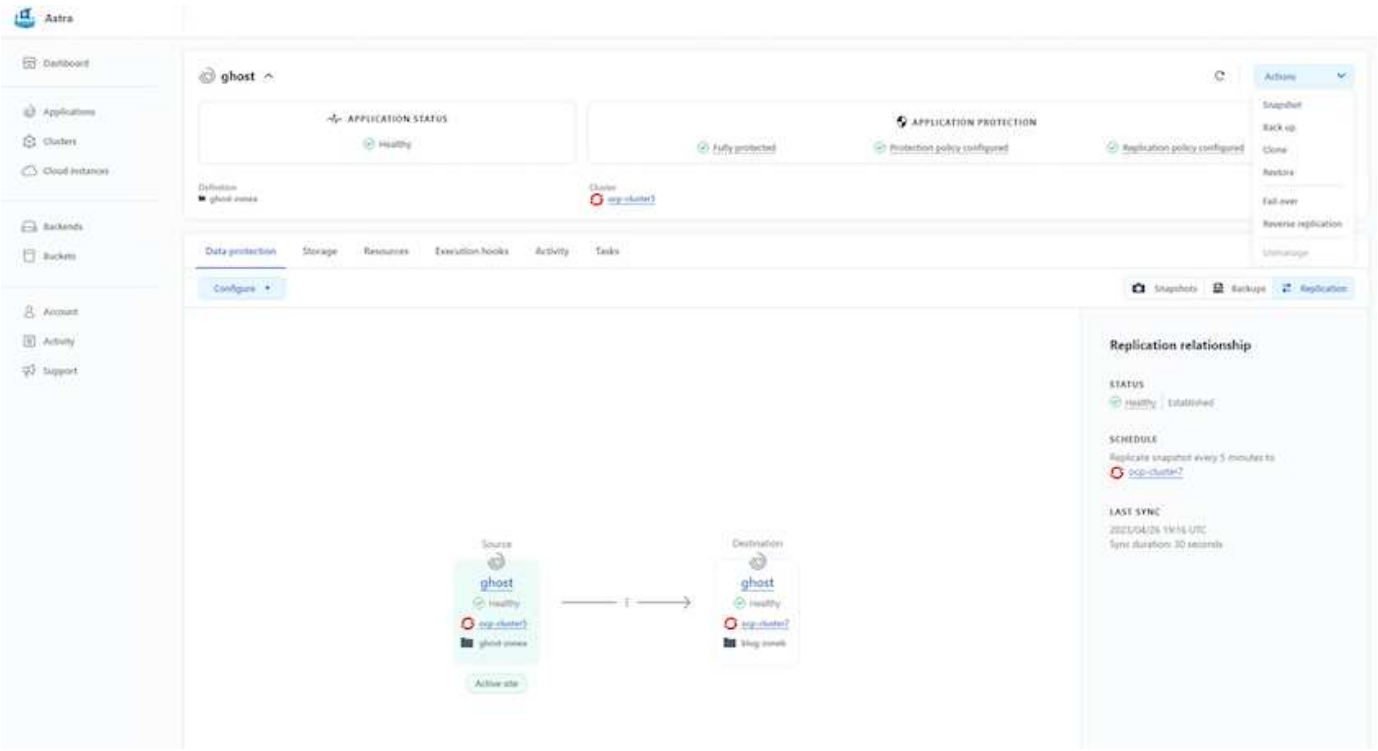
リージョンを保護する場合や、RPOとRTOの低い解決策 を実現する場合は、別のサイト（できれば別のリージョン）で実行されている別のKubernetesインスタンスにアプリケーションをレプリケートできます。ACC

は、最短5分でRPOを実現するONTAP 非同期SnapMirrorを利用します。レプリケーションはONTAP にレプリケートすることで実行され、フェイルオーバーによってデスティネーションクラスタにKubernetesリソースが作成されます。

> レプリケーションは、バックアップがS3に保存され、S3からリストアが実行されるバックアップとリストアとは異なります。2種類のデータ保護の違いの詳細については、https://docs.netapp.com/us-en/astra-control-center/concepts/data-protection.html#replication-to-a-remote-cluster[here]を参照してください。

を参照してください "こちらをご覧ください" SnapMirrorのセットアップ手順を参照してください。

ACCを使用した**SnapMirror**



> SANエコノミーおよびNASエコノミーのストレージドライバは、レプリケーション機能をサポートしていません。を参照してください "こちらをご覧ください" を参照してください。

デモビデオ：

"Astra Control Centerを使用したディザスタリカバリのデモビデオ"

Astra Control Centerによるデータ保護

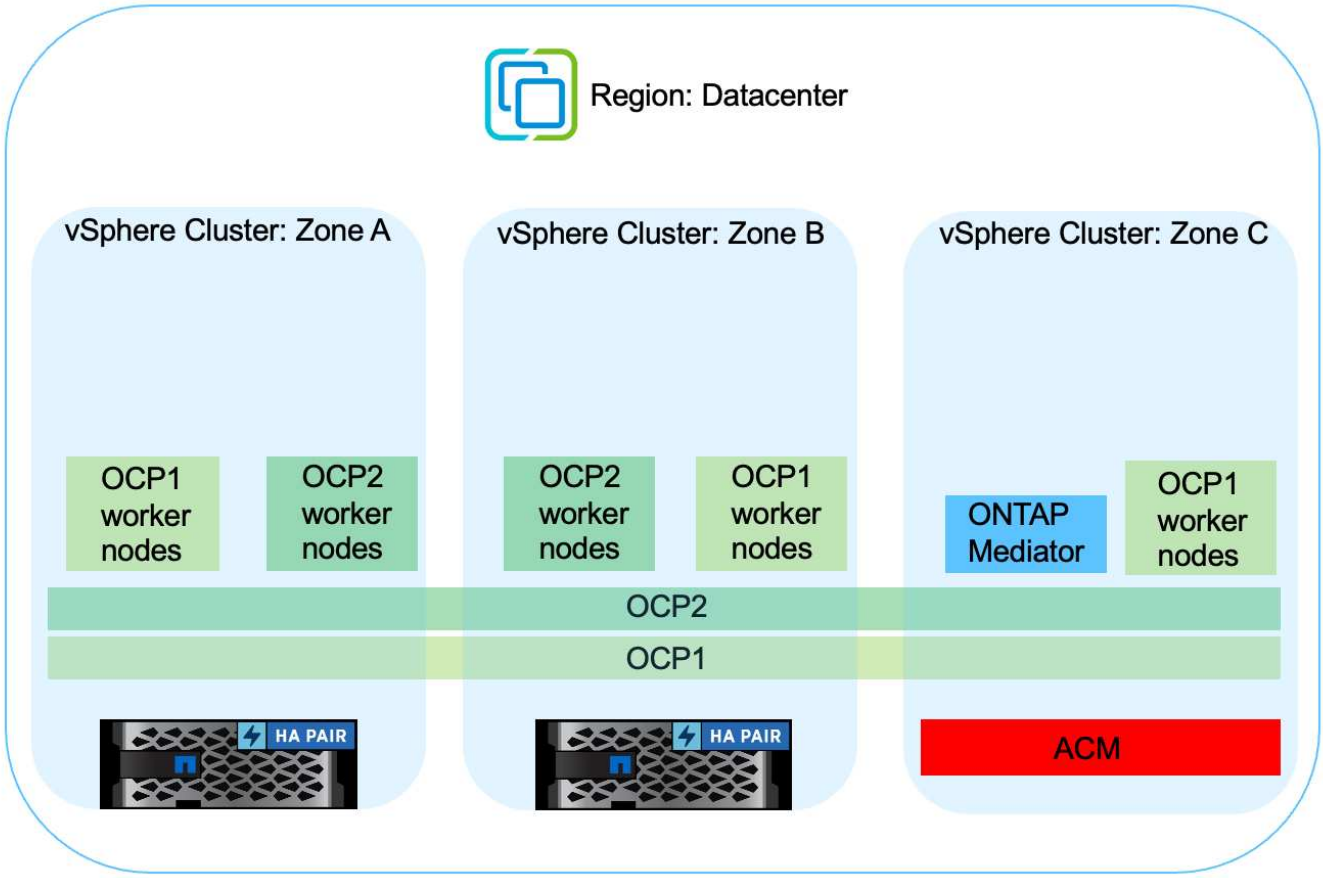
MetroCluster によるビジネス継続性

ONTAP 用のハードウェアプラットフォームのほとんどは、デバイス障害から保護するための高可用性機能を備えており、ダイヤスターリカバリを実行する必要はありません。しかし、火災やその他の災害からデータを保護し、RPOゼロとRTOを低く抑えてビジネスを継続するためには、多くの場合MetroCluster 解決策 が使用されます。

現在ONTAP システムをお持ちのお客様は、ゾーンレベルのディザスタリカバリを実現するために、距離の制

限内にサポート対象のONTAP システムを追加することで、MetroCluster に拡張できます。CSI（コンテナストレージインターフェイス）であるAstra Tridentは、MetroCluster 構成のほか、Cloud Volumes ONTAP 、Azure NetApp Files 、AWS FSx for NetApp ONTAP などの他のオプションを含むNetApp ONTAP をサポートしています Astra Tridentには、ONTAP 向けに5つのストレージドライバオプションが用意されていますが、いずれもMetroCluster 構成でサポートされています。を参照してください "こちらをご覧ください" Astra TridentでサポートされるONTAP ストレージドライバの詳細については、を参照してください。

MetroCluster 解決策 には、両方のフォールトドメインから同じネットワークアドレスにアクセスするためのレイヤ2ネットワーク拡張または機能が必要です。MetroCluster を設定すると、MetroCluster SVM内のすべてのボリュームが保護され、SyncMirror （RPOゼロ）のメリットが得られるため、解決策 はアプリケーション所有者に対して透過的に実行されます。



Tridentバックエンド構成（TBC）の場合は、MetroCluster 構成を使用する際にデータLIFとSVMを指定しないでください。管理LIF用のSVM管理IPを指定し、vsadminロールのクレデンシャルを使用してください。

Astra Control Centerのデータ保護機能の詳細を確認できます "こちらをご覧ください"

Astra Control Centerを使用したデータ移行

このページには、Astra Control Center（ACC）を使用したRed Hat OpenShiftクラスタ上のコンテナワークロードのデータ移行オプションが表示されます。

Kubernetesアプリケーションは、多くの場合、ある環境から別の環境に移動する必要があります。アプリケーションを永続的データと一緒に移行する場合は、NetApp ACCを使用できます。

異なる**Kubernetes**環境間でのデータ移行

ACCは、Google Anthos、Red Hat OpenShift、Tanzu Kubernetes Grid、Rancher Kubernetes Engine、Upstream Kubernetes、 など 詳細については、を参照してください "こちらをご覧ください"。

アプリケーションをあるクラスタから別のクラスタに移行するには、ACCの次の機能のいずれかを使用できます。

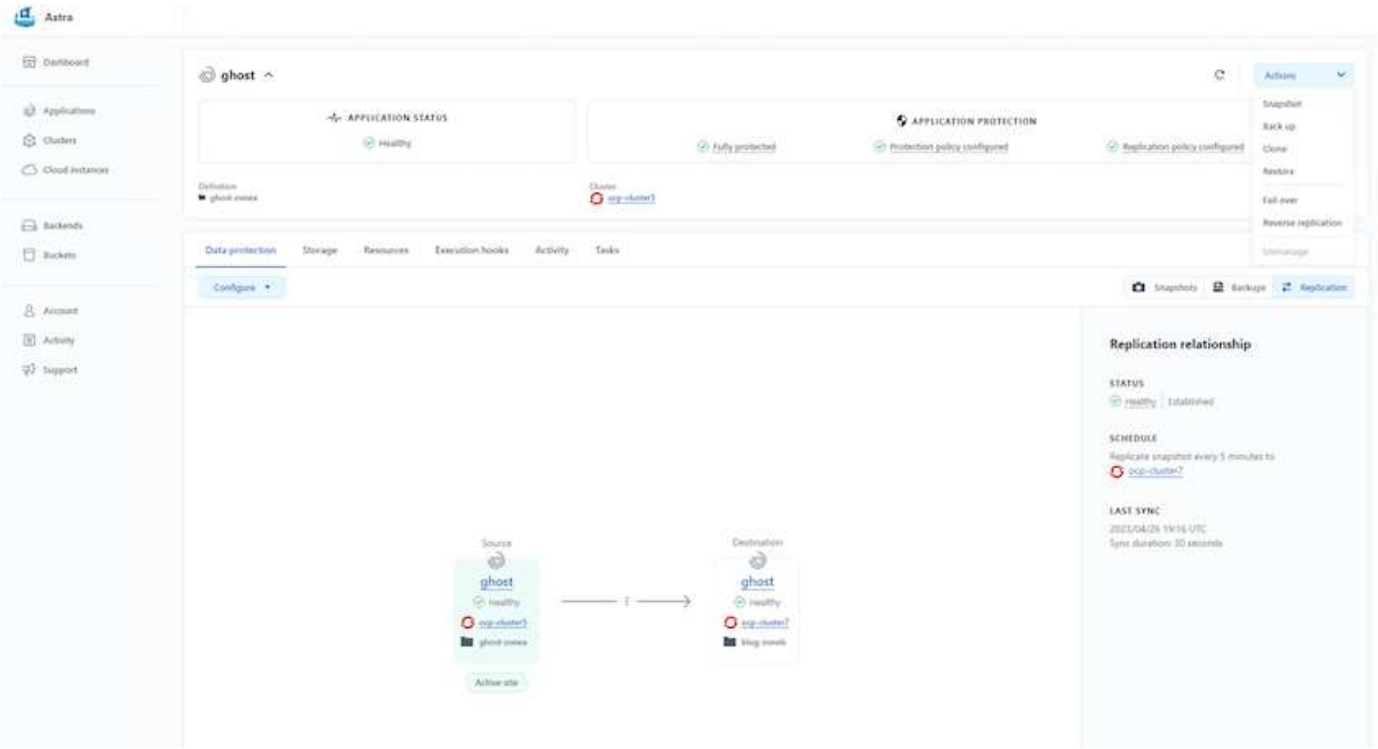
- レプリケーション
- バックアップとリストア
- クローン

を参照してください "データ保護セクション" レプリケーションおよびバックアップとリストアオプションの場合。

を参照してください "こちらをご覧ください" クローン作成の詳細については、を参照してください。

> Astraレプリケーション機能は、Trident Container Storage Interface（CSI）でのみサポートされます。ただし、NASエコノミードライバとSANエコノミードライバでは、レプリケーションはサポートされていません。

ACCを使用したデータ複製の実行



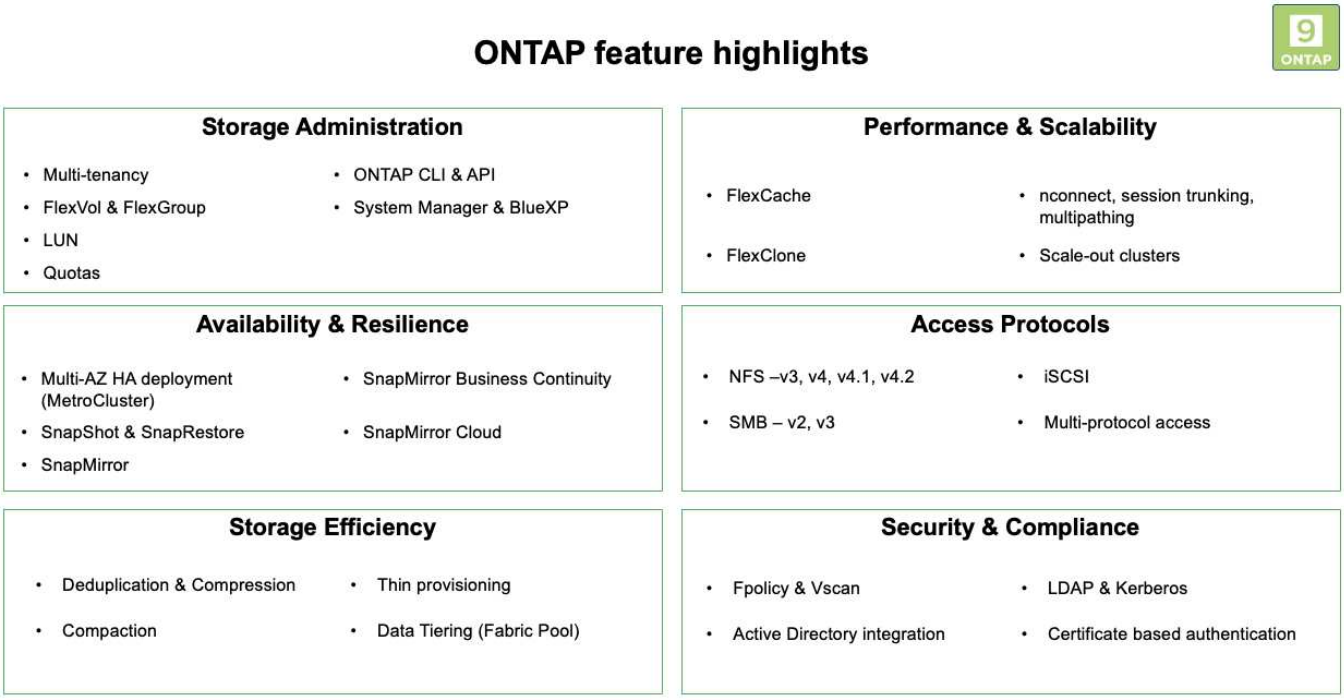
Red Hat OpenShift Containerワークロード向けのネットアップハイブリッドマルチクラウドソリューション

概要

ネットアップでは、従来型エンタープライズアプリケーションを最新化し、Kubernetesを中心に構築されたコンテナとオーケストレーションプラットフォームを使用して新しいアプリケーションを構築するお客様が大幅に増えています。Red Hat OpenShift Container Platformは、多くのお客様に採用されている例の1つです。

企業内でコンテナを採用するお客様がますます増えています。ネットアップは、ステートフルアプリケーションの永続的ストレージのニーズに加え、データ保護、データセキュリティ、データ移行などの従来のデータ管理のニーズにも応えることができます。しかし、これらのニーズは、さまざまな戦略、ツール、方法を使用して満たしています。

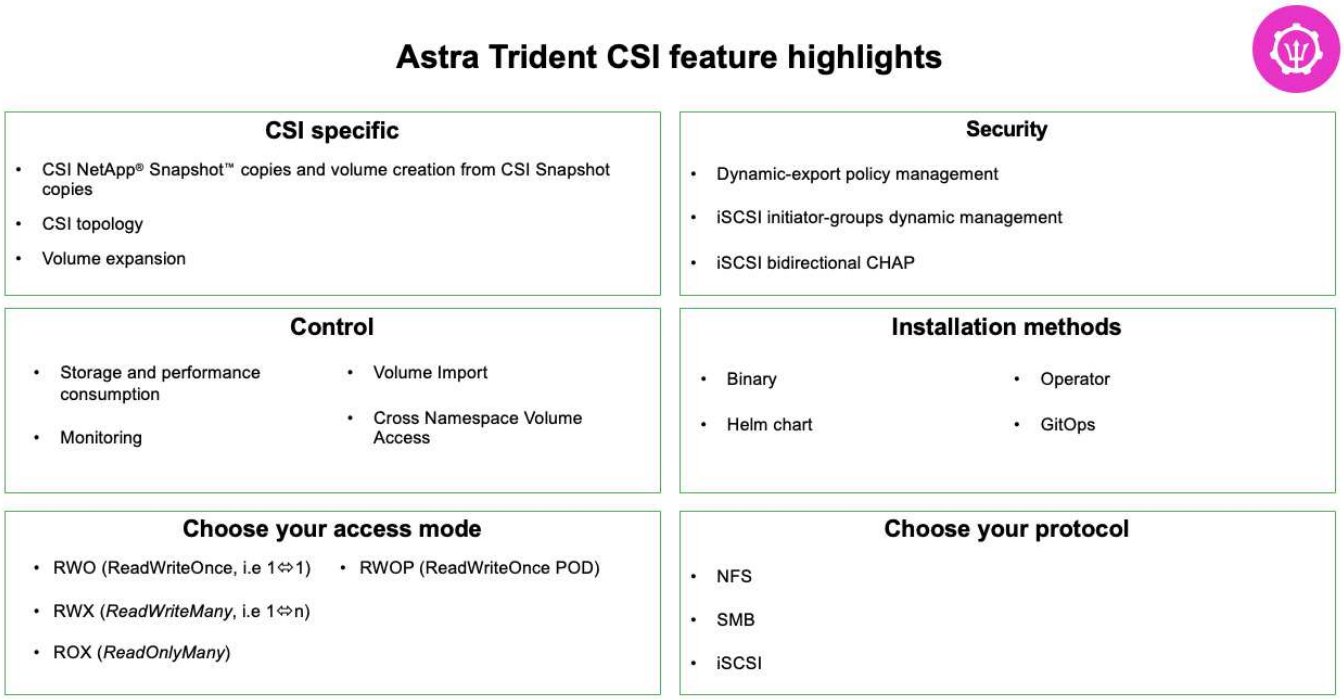
- NetApp ONTAP **ベースのストレージオプションを次に示します。コンテナとKubernetes環境にセキュリティ、データ保護、信頼性、柔軟性を提供します。
  - オンプレミスの自己管理型ストレージ：
- ネットアップファブリック接続ストレージ（FAS ）、ネットアップオールフラッシュFAS アレイ（AFF ）、ネットアップオールSANアレイ（ASA ）、ONTAP Select
  - オンプレミスのプロバイダ管理ストレージ：
- NetApp Keystone が提供するストレージサービス（STaaS）
  - クラウド内の自己管理型ストレージ：
- NetApp Cloud Volumes ONTAP（CVO）は、ハイパースケーラに自己管理型ストレージを提供します
  - クラウド上のプロバイダが管理するストレージ：
- Cloud Volumes Service for Google Cloud（CVS）、Azure NetApp Files （ANF）、Amazon FSx for NetApp ONTAP は、ハイパースケーラにフルマネージドストレージを提供します



- NetApp BlueXP **を使用すると、すべてのストレージ資産とデータ資産を単一のコントロールプレーン/インターフェイスから管理できます。

BlueXPを使用して、クラウドストレージ（Cloud Volumes ONTAP やAzure NetApp Files など）の作成と管理、データの移動、保護、分析、オンプレミスやエッジの多くのストレージデバイスの管理を行うことができます。

- NetApp Astra Trident **はCSI準拠のストレージオーケストレーションツールです。上記のさまざまなネットアップストレージオプションを利用して、永続的ストレージをすばやく簡単に利用できます。ネットアップが保守、サポートしているオープンソースのソフトウェアです。



ビジネスクリティカルなコンテナワークロードに必要なのは、永続的ボリュームだけではありません。同社のデータ管理要件では、アプリケーションのKubernetesオブジェクトの保護と移行も必要です。

> アプリケーションデータには、ユーザデータに加えてKubernetesオブジェクトが含まれます。例を次に示します。 -ポッド仕様、PVC、デプロイ、サービスなどのKubernetesオブジェクト- 設定マップやシークレットなどのカスタム設定オブジェクト- Snapshotコピー、バックアップ、クローンなどの永続的データ- CRSやCRDなどのカスタムリソース

- NetApp Astra Control **は、フルマネージドと自己管理型の両方のソフトウェアとして提供され、堅牢なアプリケーションデータ管理のためのオーケストレーションを提供します。を参照してください "Astra のドキュメント" を参照してください。

このリファレンスドキュメントでは、NetApp Astra Control Centerを使用して、Red Hat OpenShiftコンテナプラットフォームに導入されたコンテナベースアプリケーションの移行と保護について検証します。また、解決策 では、コンテナプラットフォームを管理するためのRed Hat Advanced Cluster Management（ACM）の導入と使用に関する詳細についても説明しています。また、Astra Trident CSIプロビジョニングツールを使用して、ネットアップストレージとRed Hat OpenShiftコンテナプラットフォームを統合する方法についても詳しく説明します。Astra Control Centerはハブクラスタに導入され、コンテナアプリケーションとその永続的ストレージライフサイクルの管理に使用されます。最後に、Amazon FSx for NetApp ONTAP （FSxN）を永続的ストレージとして使用し、AWS（ROSA）のマネージドRed Hat OpenShiftクラスタ上のコンテナワークロードのレプリケーション、フェイルオーバー、フェイルバックのための解決策 を提供します。

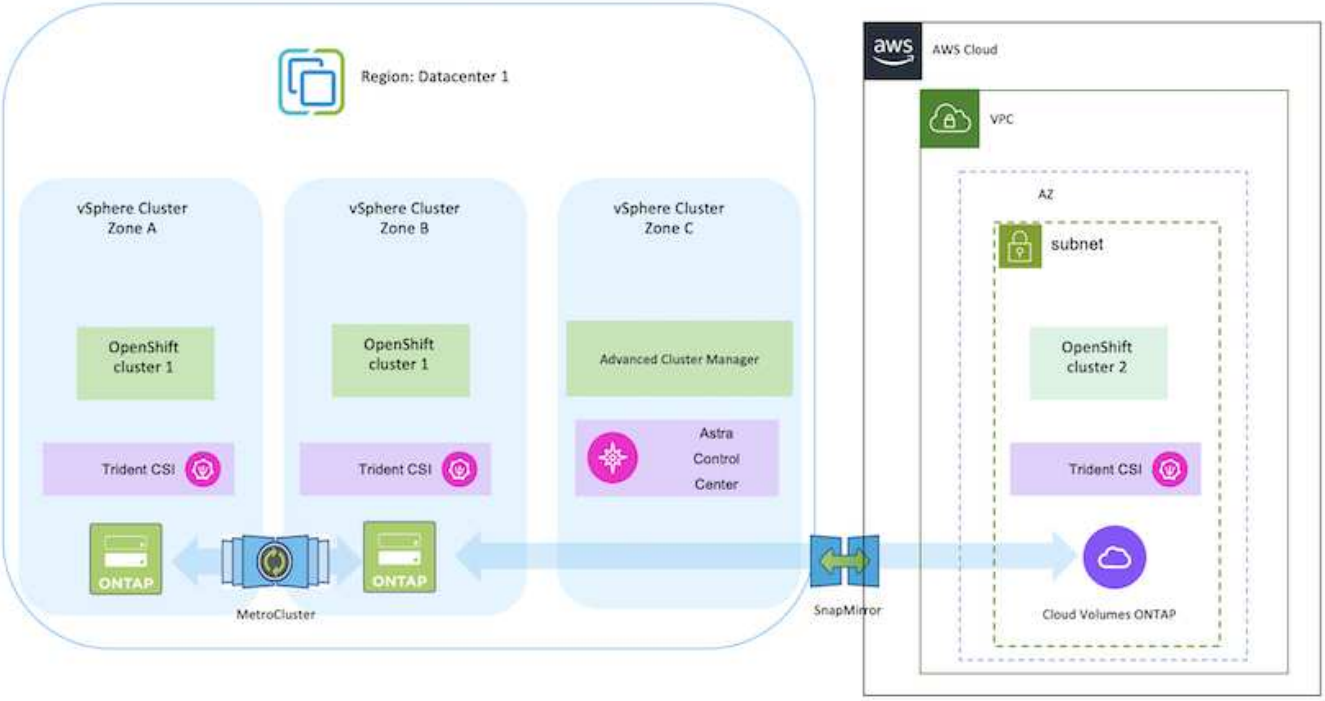
NetApp解決策 とRed Hat OpenShift Containerプラットフォームのワークロードをハイブリッドクラウドで運用

お客様は、一部のワークロードまたはすべてのワークロードをデータセンターからクラウドに移行する準備が整った時点で、モダナイゼーションに移行する可能性があります。お客様は、さまざまな理由から、クラウドで自己管理型OpenShiftコンテナと自己管理型ネットアップストレージを使用することができます。データセンターからコンテナワークロードを移行するための本番環境向け環境を成功させるには、Red Hat OpenShift コンテナプラットフォーム（OCP）をクラウドに計画して導入する必要があります。OCPクラスタは、データセンターのVMwareまたはベアメタルに導入し、クラウド環境のAWS、Azure、Google Cloudに導入できます。

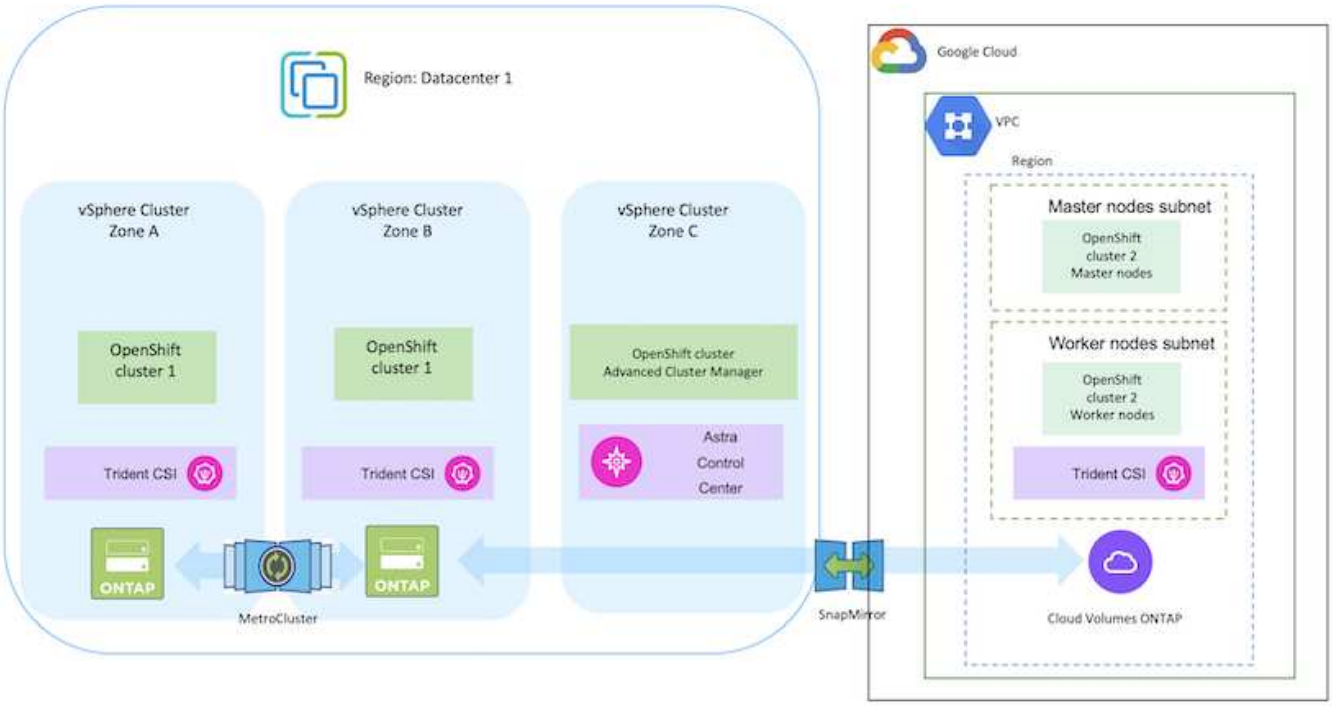
NetApp Cloud Volumes ONTAP ストレージは、AWS、Azure、Google Cloudでのコンテナ導入にデータ保護、信頼性、柔軟性を提供します。Astra Tridentは、お客様のステートフルアプリケーション向けに永続的Cloud Volumes ONTAP ストレージを利用するための動的ストレージプロビジョニングツールとして機能します。Astra Control Centerを使用すると、データ保護、移行、ビジネス継続性など、ステートフルアプリケーションに求められる多くのデータ管理要件をオーケストレーションできます。

Astra Control Centerを使用したハイブリッドクラウドでの**OpenShift**コンテナワークロード向けのデータ保護と移行解決策

オンプレミスとAWS

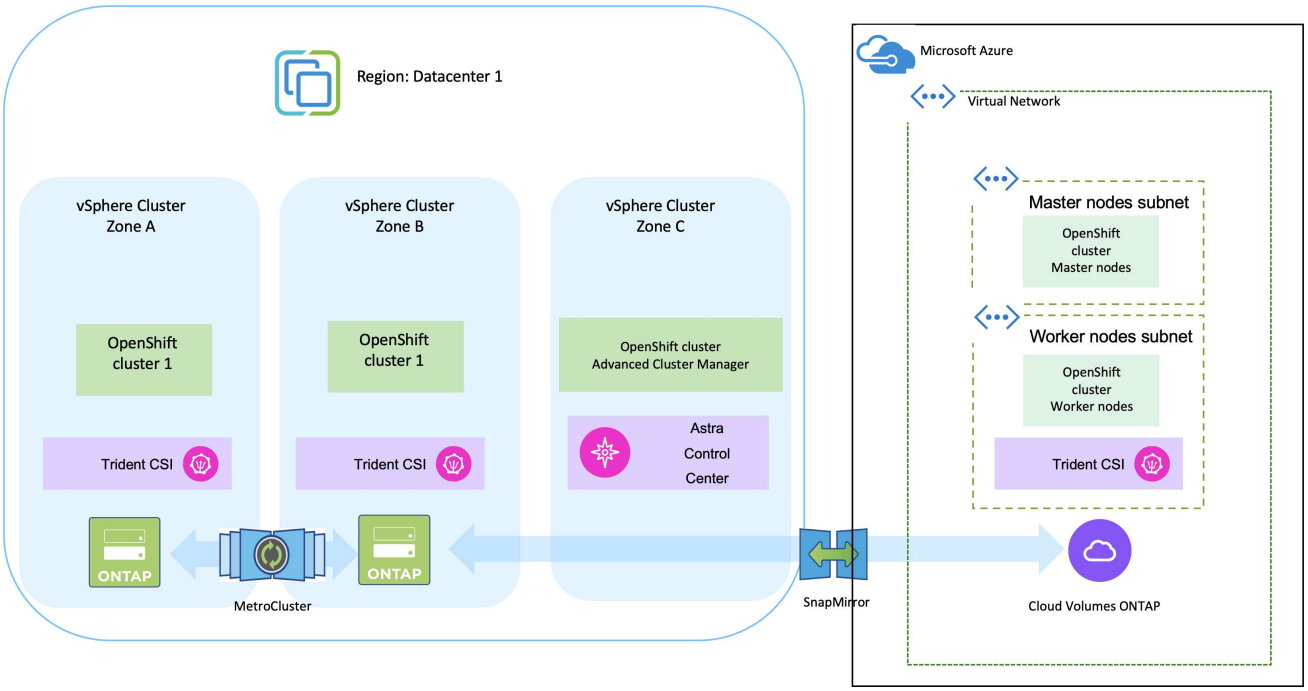


オンプレミスとGoogle Cloud



オンプレミスとAzureクラウド

[redacted]



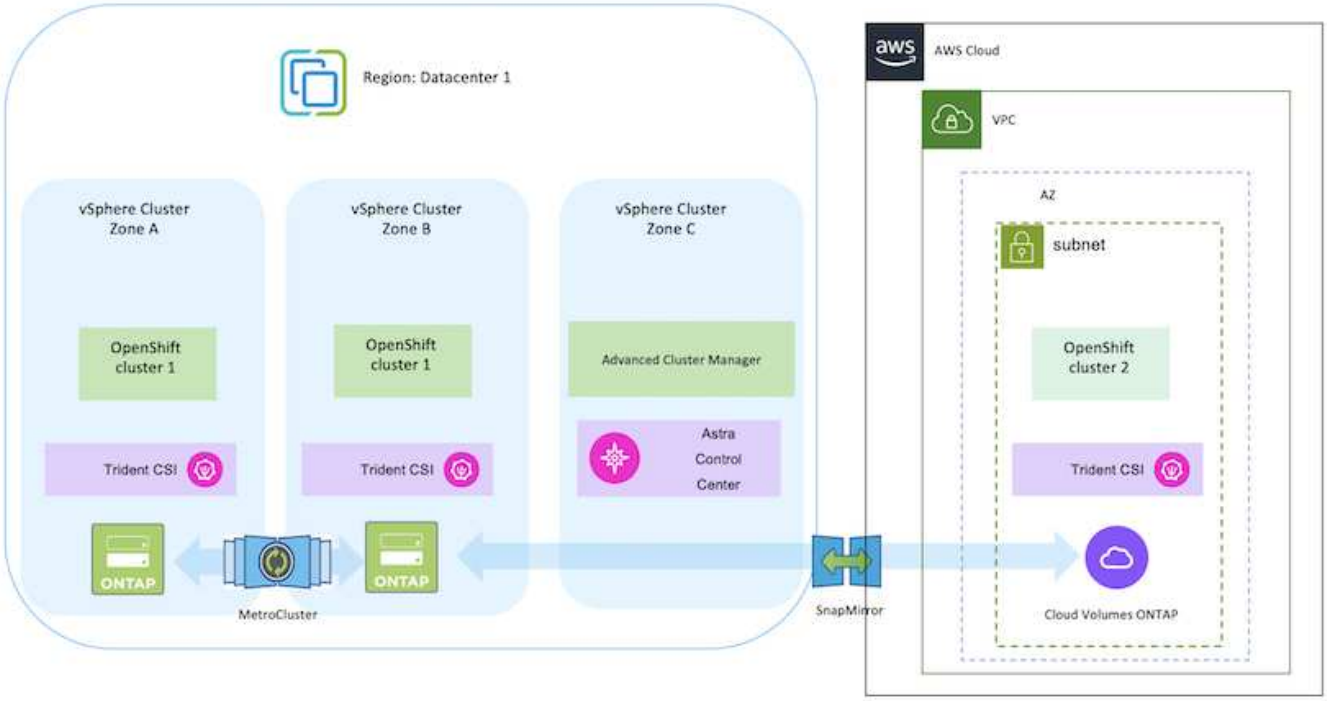
[redacted]

AWSにRed Hat OpenShift Containerプラットフォームを導入して設定します

このセクションでは、AWSでOpenShiftクラスタをセットアップおよび管理し、それらにステートフルアプリケーションを導入する方法の大まかなワークフローについて説明します。このスライドでは、Astra Tridentを使用してNetApp Cloud Volumes ONTAP ストレージを使用し、永続ボリュームを提供する方法を示しています。ステートフルアプリケーションに対してデータ保護と移行のアクティビティを実行するためのAstra Control Centerの使用方法について詳しく説明します。

> Red Hat OpenShift Containerプラットフォームクラスタを AWSに導入する方法はいくつかあります。このセットアップの概要概要 には、使用した具体的な方法のドキュメントへのリンクが記載されています。その他の方法については、に記載されている関連リンクを参照してください "リソースセクション"。

次の図は、AWSに導入され、VPNを使用してデータセンターに接続されたクラスタを示しています。



セットアッププロセスは、次の手順に分けることができます。

Advanced Cluster Managementから**AWS**に**OCP**クラスタをインストールします。

- サイト間VPN接続（pfsenseを使用）を使用してVPCを作成し、オンプレミスネットワークに接続します。
- オンプレミスネットワークはインターネットに接続されています。
- 3つの異なるAZに3つのプライベートサブネットを作成します。
- VPC用にRoute 53プライベートホストゾーンとDNSリゾルバを作成します。

Advanced Cluster Management（ACM）ウィザードを使用して、AWSにOpenShiftクラスタを作成します。手順を参照してください "こちらをご覧ください"。

AWSでは、OpenShift Hybrid Cloudコンソールからクラスタを作成することもできます。を参照してください "こちらをご覧ください" 手順については、を参照し

ACMを使用してクラスタを作成する場合は、フォームビューで詳細を入力した後でYAMLファイルを編集してインストールをカスタマイズできます。クラスタが作成されたら、トラブルシューティングや追加の手動設定のために、クラスタのノードにSSHログインできます。インストール時に指定したsshキーとユーザ名coreを使用してログインします。

BlueXPを使用して**AWS**に**Cloud Volumes ONTAP** を導入

- オンプレミスのVMware環境にコネクタをインストールします。手順を参照してください "こちらをご覧ください"。
- コネクタを使用してAWSにCVOインスタンスを導入します。手順を参照してください "こちらをご覧ください"。

コネクタはクラウド環境にも設置できます。を参照してください "こちらをご覧ください" 追加情報 の場合。

OCPクラスタに**Astra Trident**をインストール

- Helmを使用してTrident Operatorを導入します。手順を参照してください "こちらをご覧ください"
- バックエンドとストレージクラスを作成手順を参照してください "こちらをご覧ください"。

AWSの**OCP**クラスタを**Astra Control Center**に追加します。

AWSのOCPクラスタをAstra Control Centerに追加します。

マルチゾーンアーキテクチャに**Trident**の**CSI**トポロジ機能を使用

今日のクラウドプロバイダは、Kubernetes / OpenShiftのクラスタ管理者がゾーンベースのクラスタのノードを生成できるようにしています。ノードは、リージョンによって異なるアベイラビリティゾーンに配置することも、リージョンによって配置することもできます。マルチゾーンアーキテクチャでワークロード用のボリュームをプロビジョニングするために、 Astra Trident は CSI トポロジを使用します。CSI トポロジ機能を使用すると、領域およびアベイラビリティゾーンに基づいて、ボリュームへのアクセスをノードのサブセットに制

限できます。を参照してください "こちらをご覧ください" を参照してください。

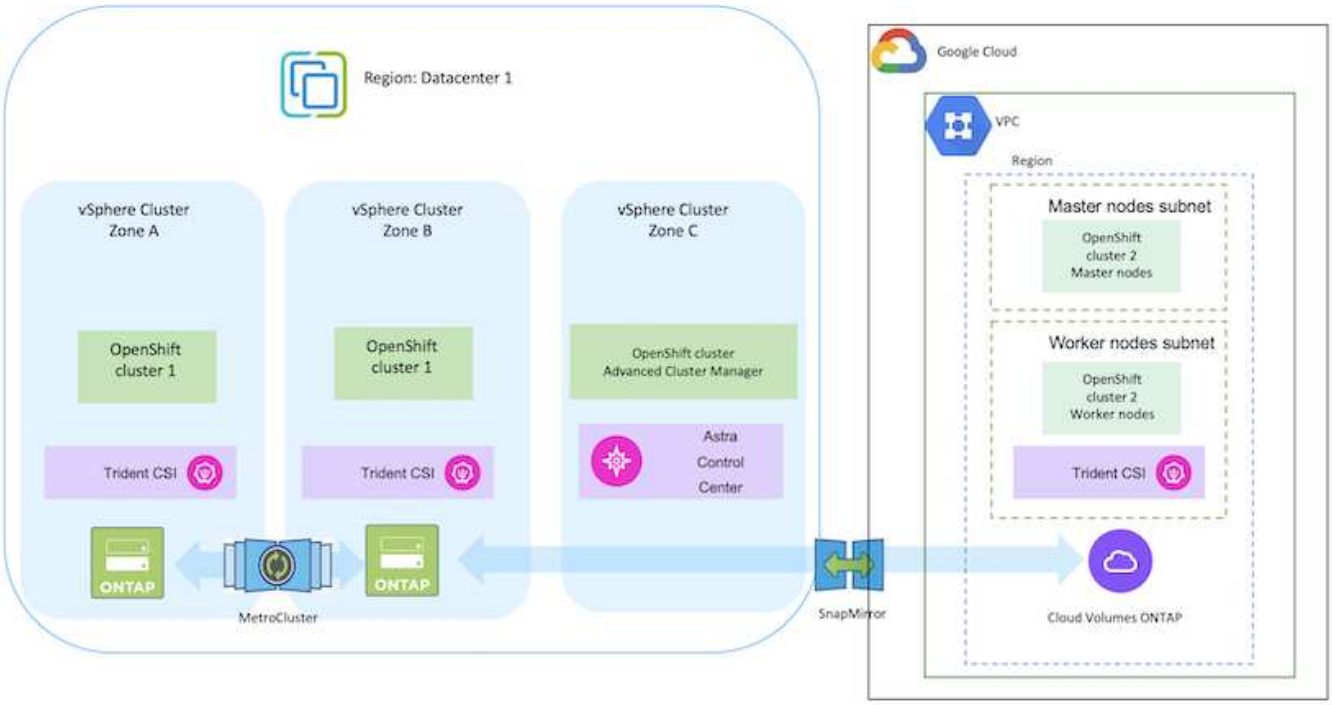
> Kubernetesでは2つのボリュームバインドモードがサポートされます。-***VolumeBindingMode_***が***_Immediate***（デフォルト）に設定されている場合、Astra Tridentはトポロジを認識せずにボリュームを作成します。永続ボリュームは、要求側ポッドのスケジュール要件に依存せずに作成されます。-***VolumeBindingMode_***が***_WaitForFirstConsumer***に設定されている場合、PVCの永続ボリュームの作成とバインドは、そのPVCを使用するポッドがスケジュールされて作成されるまで遅延します。これにより、トポロジの要件に応じたスケジュールの制約を満たすようにボリュームが作成されます。Astra Tridentのストレージバックエンドは、アベイラビリティゾーン（トポロジ対応バックエンド）に基づいて選択的にボリュームをプロビジョニングするように設計できます。ストレージクラスがそのようなバックエンドを使用する場合、ボリュームは、サポートされているリージョン / ゾーンでスケジュールされているアプリケーションから要求された場合にのみ作成されます。（Topology-Aware StorageClass）を参照してください "こちらをご覧ください" を参照してください。

GCPでのRed Hat OpenShift Containerプラットフォームの導入と設定

GCPでのRed Hat OpenShift Containerプラットフォームの導入と設定

このセクションでは、GCPでOpenShiftクラスタをセットアップおよび管理し、それらにステートフルアプリケーションを導入する方法の概要的なワークフローについて説明します。このスライドでは、Astra Tridentを使用してNetApp Cloud Volumes ONTAP ストレージを使用し、永続ボリュームを提供する方法を示しています。ステートフルアプリケーションに対してデータ保護と移行のアクティビティを実行するためのAstra Control Centerの使用方法について詳しく説明します。

次の図は、GCPに導入され、VPNを使用してデータセンターに接続されたクラスタを示しています。



GCPにRed Hat OpenShift Containerプラットフォームクラスタを導入する方法はいくつかあります。このセットアップの概要概要 には、使用した具体的な方法のドキュメントへのリンクが記載されています。その他の方法については、に記載されている関連リンクを参照してください "リソースセクション"。

セットアッププロセスは、次の手順に分けることができます。

CLIから**GCP**に**OCP**クラスタをインストールします。

- 記載されているすべての前提条件を満たしていることを確認します。 "こちらをご覧ください"。
- オンプレミスとGCP間のVPN接続については、pfsense VMを作成して設定しました。手順については、を参照してください "こちらをご覧ください"。
  - pfsenseのリモートゲートウェイアドレスは、Google Cloud PlatformでVPNゲートウェイを作成した後にのみ設定できます。
  - フェーズ2のリモートネットワークIPアドレスは、OpenShiftクラスタインストールプログラムが実行され、クラスタ用のインフラストラクチャコンポーネントが作成された後にのみ設定できます。
  - Google CloudのVPNは、インストールプログラムによってクラスタのインフラストラクチャコンポーネントが作成された後にのみ設定できます。
- 次に、GCPにOpenShiftクラスタをインストールします。
  - インストールプログラムとプルシークレットを入手し、ドキュメントに記載されている手順に従ってクラスタを導入する "こちらをご覧ください"。
  - インストールでGoogle Cloud PlatformにVPCネットワークが作成されます。また、Cloud DNSにプライベートゾーンを作成し、レコードを追加します。
    - VPCネットワークのCIDRブロックアドレスを使用してpfsenseを設定し、VPN接続を確立します。ファイアウォールが正しく設定されていることを確認します。
    - Google Cloud DNSのAレコードのIPアドレスを使用して、オンプレミス環境のDNSにAレコードを追加します。
  - クラスタのインストールが完了し、クラスタのコンソールにログインするためのkubeconfigファイルとユーザ名とパスワードが表示されます。

BlueXPを使用して**GCP**に**Cloud Volumes ONTAP**を導入

- Google Cloudにコネクタをインストールします。手順を参照してください "こちらをご覧ください"。
- コネクタを使用してGoogle CloudにCVOインスタンスを導入します。手順については、こちらを参照してください。 https://docs.netapp.com/us-en/bluexp-cloud-volumes-ontap/task-getting-started-gcp.html

GCPのOCPクラスタにAstra Tridentをインストール

- 図に示すように、Astra Tridentにはさまざまな導入方法がある "こちらをご覧ください"。
- このプロジェクトでは、Astra Tridentのオペレータを手順に従って手動で導入し、Astra Tridentをインストールしました。 "こちらをご覧ください"。
- バックエンドとストレージクラスを作成手順を参照してください "こちらをご覧ください"。

GCPのOCPクラスタをAstra Control Centerに追加します。

- クラスタの管理に必要な最小限の権限を含むクラスタロールを含むKubeConfigファイルを別途作成します。手順は次のとおりです。
  "こちらをご覧ください"。
- 手順に従ってクラスタをAstra Control Centerに追加
  "こちらをご覧ください"

マルチゾーンアーキテクチャに**Tridentの**CSIトポロジ機能を使用

今日のクラウドプロバイダは、Kubernetes / OpenShiftのクラスタ管理者がゾーンベースのクラスタのノードを生成できるようにしています。ノードは、リージョンによって異なるアベイラビリティゾーンに配置することも、リージョンによって配置することもできます。マルチゾーンアーキテクチャでワークロード用のボリュームをプロビジョニングするために、 Astra Trident は CSI トポロジを使用します。CSI トポロジ機能を使用すると、領域およびアベイラビリティゾーンに基づいて、ボリュームへのアクセスをノードのサブセットに制限できます。を参照してください "こちらをご覧ください" を参照してください。

> Kubernetesでは2つのボリュームバインドモードがサポートされます。-***VolumeBindingMode_***が***_Immediate***（デフォルト）に設定されている場合、Astra Tridentはトポロジを認識せずにボリュームを作成します。永続ボリュームは、要求側ポッドのスケジュール要件に依存せずに作成されます。-***VolumeBindingMode_***が***_WaitForFirstConsumer***に設定されている場合、PVCの永続ボリュームの作成とバインドは、そのPVCを使用するポッドがスケジュールされて作成されるまで遅延します。これにより、トポロジの要件に応じたスケジュールの制約を満たすようにボリュームが作成されます。Astra Tridentのストレージバックエンドは、アベイラビリティゾーン（トポロジ対応バックエンド）に基づいて選択的にボリュームをプロビジョニングするように設計できます。ストレージクラスがそのようなバックエンドを使用する場合、ボリュームは、サポートされているリージョン / ゾーンでスケジュールされているアプリケーションから要求された場合にのみ作成されます。（Topology-Aware StorageClass）を参照してください "こちらをご覧ください" を参照してください。

デモビデオ

Google Cloud PlatformへのOpenShiftクラスタのインストール

Astra Control CenterへのOpenShiftクラスタのインポート

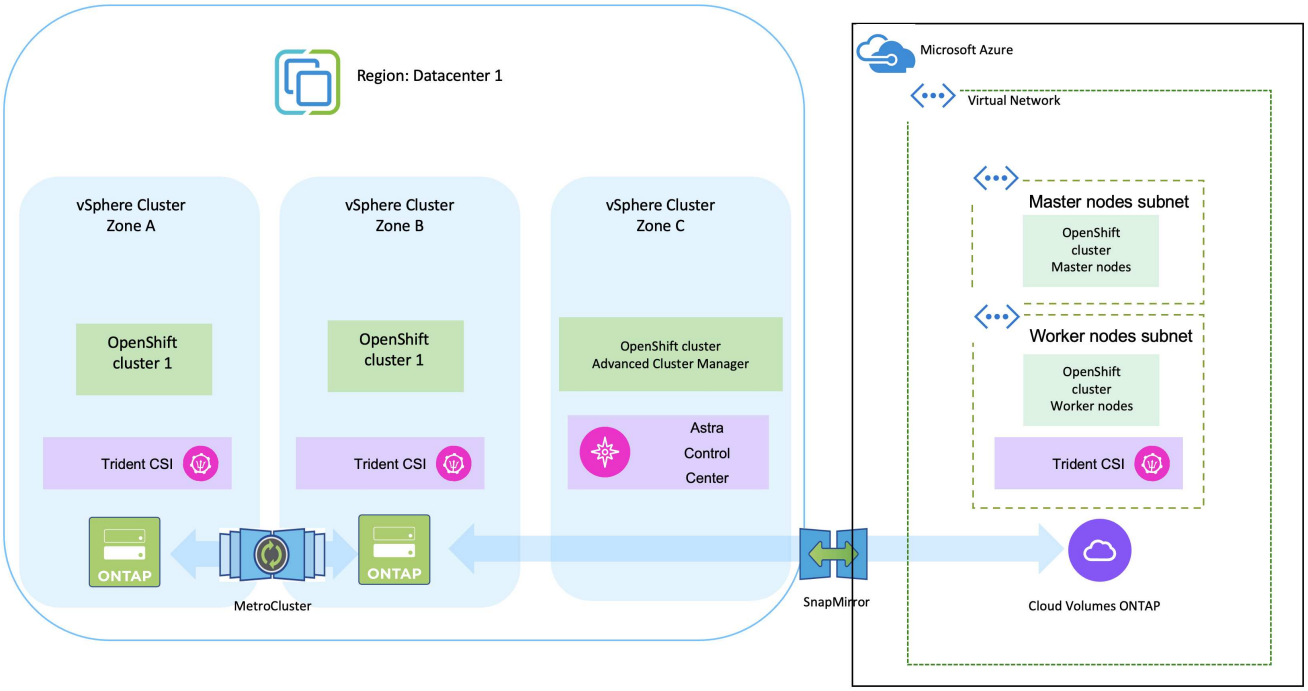
AzureでのRed Hat OpenShift Containerプラットフォームの導入と設定

AzureでのRed Hat OpenShift Containerプラットフォームの導入と設定

このセクションでは、AzureでOpenShiftクラスタをセットアップおよび管理し、それら

にステートフルアプリケーションを導入する方法の概要的なワークフローについて説明します。このスライドでは、Astra Trident / Astra Control Provisionerを使用して永続ボリュームを提供するNetApp Cloud Volumes ONTAPストレージを使用しています。ステートフルアプリケーションに対してデータ保護と移行のアクティビティを実行するためのAstra Control Centerの使用方法について詳しく説明します。

次の図は、Azureに導入され、VPNを使用してデータセンターに接続されたクラスタを示しています。



Red Hat OpenShift Containerプラットフォームクラスタを Azureに導入するには、いくつかの方法があります。このセットアップの概要概要 には、使用した具体的な方法のドキュメントへのリンクが記載されています。その他の方法については、に記載されている関連リンクを参照してください "リソースセクション"。

セットアッププロセスは、次の手順に分けることができます。

CLIを使用して**Azure**に**OCP**クラスタをインストールします。

- 記載されているすべての前提条件を満たしていることを確認します。 "こちらをご覧ください"。
- VPN、サブネット、ネットワークセキュリティグループ、およびプライベートDNSゾーンを作成します。VPNゲートウェイおよびサイト間VPN接続を作成します。
- オンプレミスとAzure間のVPN接続のために、pfsense VMを作成して設定しました。手順については、を参照してください "こちらをご覧ください"。
- インストールプログラムとプルシークレットを入手し、ドキュメントに記載されている手順に従ってクラスタを導入する "こちらをご覧ください"。
- クラスタのインストールが完了し、クラスタのコンソールにログインするためのkubeconfigファイルとユーザ名とパスワードが表示されます。

install-config.yamlファイルの例を以下に示します。

```
apiVersion: v1
baseDomain: sddc.netapp.com
compute:
- architecture: amd64
  hyperthreading: Enabled
  name: worker
  platform:
    azure:
      encryptionAtHost: false
      osDisk:
        diskSizeGB: 512
        diskType: "StandardSSD_LRS"
      type: Standard_D2s_v3
      ultraSSDCapability: Disabled
      #zones:
      #- "1"
      #- "2"
      #- "3"
  replicas: 3
controlPlane:
  architecture: amd64
  hyperthreading: Enabled
  name: master
  platform:
    azure:
      encryptionAtHost: false
      osDisk:
        diskSizeGB: 1024
        diskType: Premium_LRS
      type: Standard_D8s_v3
      ultraSSDCapability: Disabled
```

```
  replicas: 3
metadata:
  creationTimestamp: null
  name: azure-cluster
networking:
  clusterNetwork:
  - cidr: 10.128.0.0/14
    hostPrefix: 23
  machineNetwork:
  - cidr: 10.0.0.0/16
  networkType: OVNKubernetes
  serviceNetwork:
  - 172.30.0.0/16
platform:
  azure:
    baseDomainResourceGroupName: ocp-base-domain-rg
    cloudName: AzurePublicCloud
    computeSubnet: ocp-subnet2
    controlPlaneSubnet: ocp-subnet1
    defaultMachinePlatform:
      osDisk:
        diskSizeGB: 1024
        diskType: "StandardSSD_LRS"
      ultraSSDCapability: Disabled
    networkResourceGroupName: ocp-nc-us-rg
    #outboundType: UserDefinedRouting
    region: northcentralus
    resourceGroupName: ocp-cluster-ncusrg
    virtualNetwork: ocp_vnet_ncus
publish: Internal
pullSecret:
```

BlueXPを使用して**Azure**に**Cloud Volumes ONTAP**を導入

- Azureにコネクタをインストールします。手順を参照してください "こちらをご覧ください"。
- コネクタを使用してAzureにCVOインスタンスを導入します。手順リンク：https://docs.netapp.com/us-en/bluexp-cloud-volumes-ontap/task-getting-started-azure.html [こちら]を参照してください。

AzureのOCPクラスタへのAstra Control Provisionerのインストール

- このプロジェクトでは、すべてのクラスタ（オンプレミスクラスタ、Astra Control Centerが導入されているオンプレミスクラスタ、およびAzureのクラスタ）にAstra Control Provisioner（ACP）をインストールしました。Astra Control Provisionerの詳細 "こちらをご覧ください"。
- バックエンドとストレージクラスを作成手順を参照してください "こちらをご覧ください"。

AzureのOCPクラスタをAstra Control Centerに追加します。

- クラスタの管理に必要な最小限の権限を含むクラスタロールを含むKubeConfigファイルを別途作成します。手順は次のとおりです。
  "こちらをご覧ください"。
- 手順に従ってクラスタをAstra Control Centerに追加
  "こちらをご覧ください"

マルチゾーンアーキテクチャに**Tridentの**CSIトポロジ機能を使用

今日のクラウドプロバイダは、Kubernetes / OpenShiftのクラスタ管理者がゾーンベースのクラスタのノードを生成できるようにしています。ノードは、リージョンによって異なるアベイラビリティゾーンに配置することも、リージョンによって配置することもできます。マルチゾーンアーキテクチャでワークロード用のボリュームをプロビジョニングするために、 Astra Trident は CSI トポロジを使用します。CSI トポロジ機能を使用すると、領域およびアベイラビリティゾーンに基づいて、ボリュームへのアクセスをノードのサブセットに制限できます。を参照してください "こちらをご覧ください" を参照してください。

> Kubernetesでは2つのボリュームバインドモードがサポートされます。-***VolumeBindingMode_***が***_Immediate***（デフォルト）に設定されている場合、Astra Tridentはトポロジを認識せずにボリュームを作成します。永続ボリュームは、要求側ポッドのスケジュール要件に依存せずに作成されます。-***VolumeBindingMode_***が***_WaitForFirstConsumer***に設定されている場合、PVCの永続ボリュームの作成とバインドは、そのPVCを使用するポッドがスケジュールされて作成されるまで遅延します。これにより、トポロジの要件に応じたスケジュールの制約を満たすようにボリュームが作成されます。Astra Tridentのストレージバックエンドは、アベイラビリティゾーン（トポロジ対応バックエンド）に基づいて選択的にボリュームをプロビジョニングするように設計できます。ストレージクラスがそのようなバックエンドを使用する場合、ボリュームは、サポートされているリージョン / ゾーンでスケジュールされているアプリケーションから要求された場合にのみ作成されます。（Topology-Aware StorageClass）を参照してください "こちらをご覧ください" を参照してください。

デモビデオ

Astra Controlを使用したアプリケーションのフェイルオーバーとフェイルバック

Astra Control Centerを使用したデータ保護

このページには、VMware vSphereまたはAstra Control Center（ACC）を使用してクラウドで実行されるRed Hat OpenShift Containerベースのアプリケーションのデータ保護オプションが表示されます。

ユーザがRed Hat OpenShiftを使用してアプリケーションを最新化する過程で、偶発的な削除やその他の人的エラーからユーザを保護するためのデータ保護戦略を策定する必要があります。多くの場合、データを管理か

ら保護するために、規制やコンプライアンスの目的で保護戦略が必要になります。

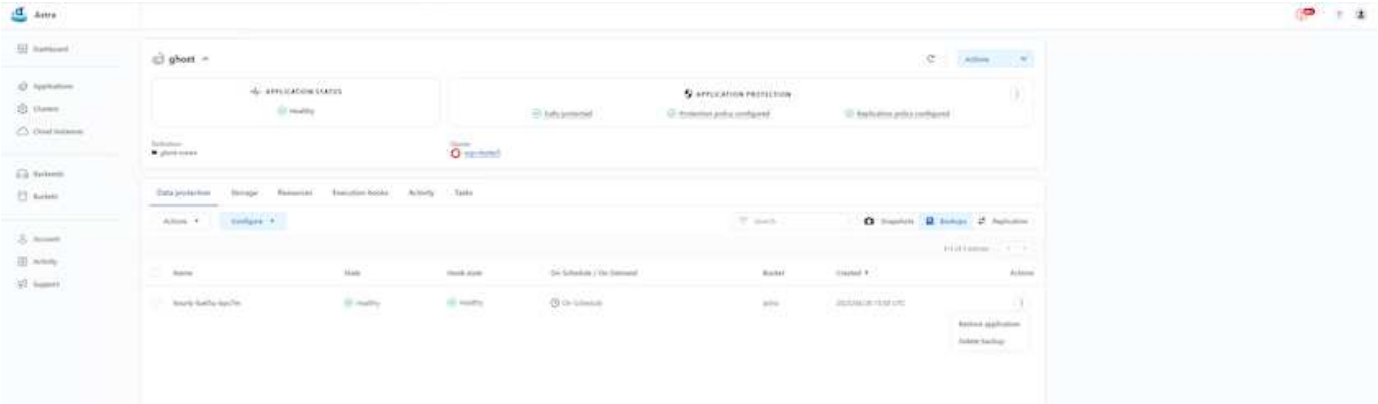
データ保護の要件は、ポイントインタイムコピーへのリバートから別の障害ドメインへの自動フェイルオーバーまで、人手を介さずにさまざまです。多くのお客様がONTAP をKubernetesアプリケーションに最適なストレージプラットフォームとして選択しています。その理由は、マルチテナンシー、マルチプロトコル、ハイパフォーマンスと容量のサービス、マルチサイト環境のレプリケーションとキャッシュ、セキュリティと柔軟性などの豊富な機能があるからです。

お客様は、データセンターの拡張機能としてクラウド環境を設定している場合があります。これにより、クラウドのメリットを活用できるだけでなく、将来的にワークロードを移行するための適切な位置付けを得ることができます。このようなお客様にとって、OpenShiftアプリケーションとデータをクラウド環境にバックアップすることは避けられません。その後、アプリケーションと関連データをクラウドまたはデータセンターのOpenShiftクラスタにリストアできます。

ACCを使用したバックアップと復元

アプリケーション所有者は、ACCによって検出されたアプリケーションを確認および更新できます。ACCはCSIを使用してSnapshotコピーを作成し、ポイントインタイムSnapshotコピーを使用してバックアップを実行できます。バックアップ先は、クラウド環境内のオブジェクトストアにすることができます。スケジュールされたバックアップの保護ポリシーと保持するバックアップバージョンの数を設定できます。最小RPOは1時間です。

ACCを使用したバックアップからのアプリケーションのリストア

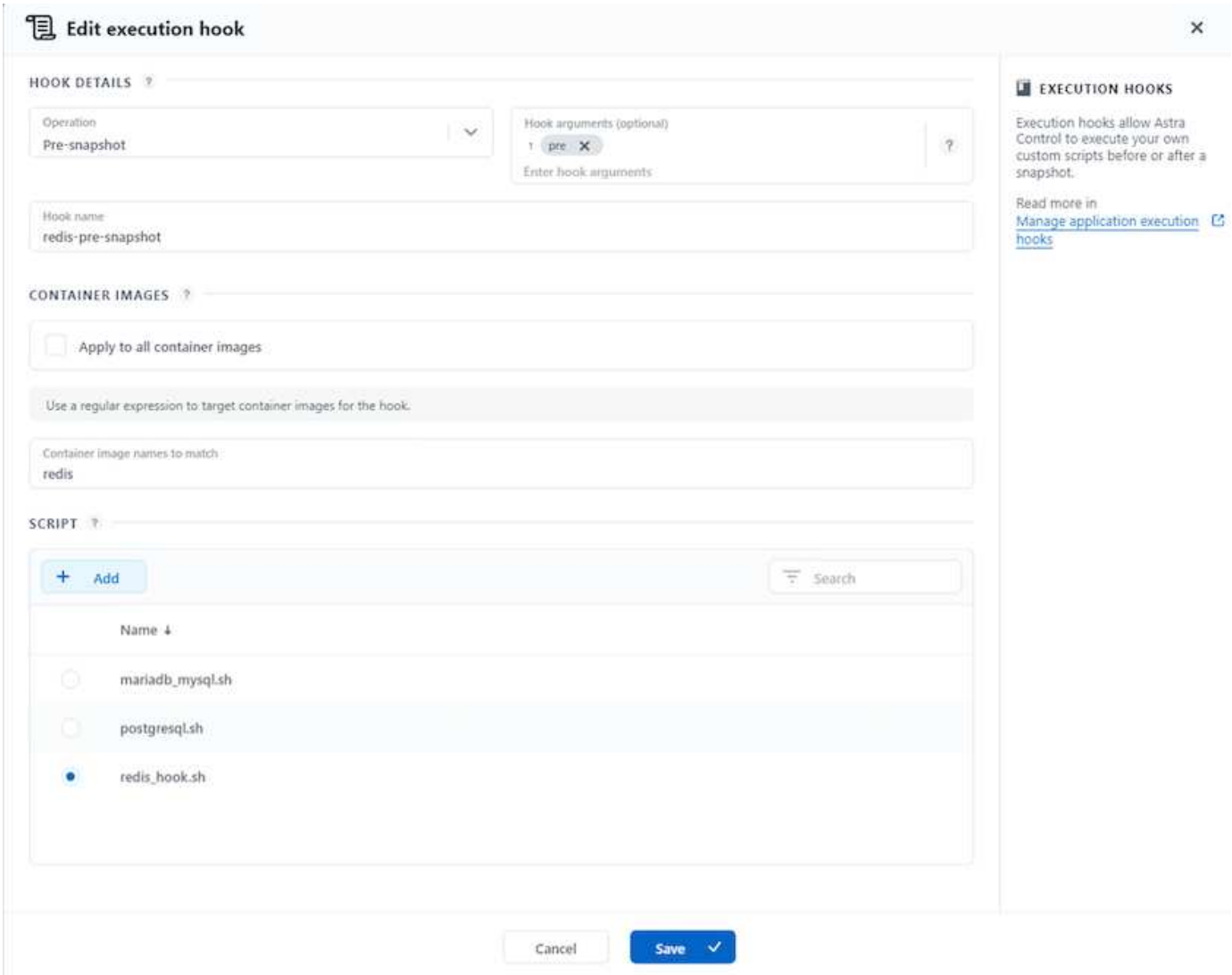


アプリケーション固有の実行フック

ストレージアレイレベルのデータ保護機能を使用できますが、アプリケーションのバックアップとリストアの整合性を確保するために追加の手順が必要になることがよくあります。アプリケーション固有の追加手順は次のとおりです。- Snapshotコピーの作成前または作成後。-バックアップの作成前または作成後。- Snapshotコピーまたはバックアップからリストアしたあと。Astra Controlでは、実行フックと呼ばれるカスタムスクリプトとしてコード化されたアプリケーション固有の手順を実行できます。

ネットアップの "オープンソースプロジェクトVerda" 一般的なクラウドネイティブアプリケーションの実行フックを提供し、アプリケーションを簡単に保護し、堅牢で、オーケストレーションを容易にします。リポジトリにないアプリケーションに十分な情報がある場合は、そのプロジェクトに貢献してください。

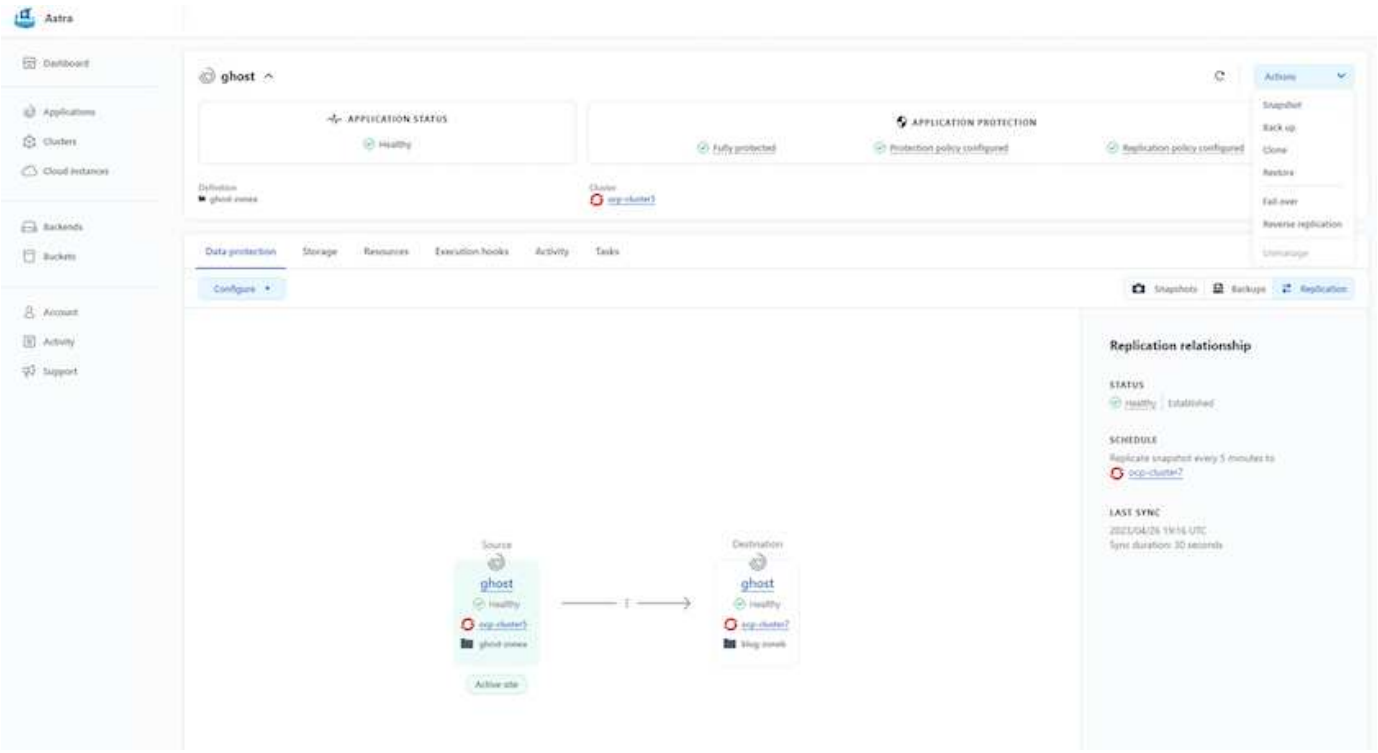
redisアプリケーションの**pre-Snapshot**用のサンプル実行フック。



ACCを使用したレプリケーション

リージョンを保護する場合や、RPOとRTOの低い解決策 を実現する場合は、別のサイト（できれば別のリージョン）で実行されている別のKubernetesインスタンスにアプリケーションをレプリケートできます。ACCは、最短5分でRPOを実現するONTAP 非同期SnapMirrorを利用します。を参照してください "こちらをご覧ください" SnapMirrorのセットアップ手順を参照してください。

ACCを使用した**SnapMirror**



SANエコノミーおよびNASエコノミーのストレージドライバは、レプリケーション機能をサポートしていません。を参照してください "こちらをご覧ください" を参照してください。

デモビデオ：

"Astra Control Centerを使用したディザスタリカバリのデモビデオ"

Astra Control Centerによるデータ保護

Astra Control Centerのデータ保護機能の詳細を確認できます "こちらをご覧ください"

ACCを使用したディザスタリカバリ（レプリケーションを使用したフェイルオーバーとフェイルバック）

Astra Controlを使用したアプリケーションのフェイルオーバーとフェイルバック

Astra Control Centerを使用したデータ移行

このページには、Astra Control Center（ACC）を使用したRed Hat OpenShiftクラスタ上のコンテナワークロードのデータ移行オプションが表示されます。具体的には、ACCを使用して、一部のワークロードまたはすべてのワークロードをオンプレミスのデータセンターからクラウドに移動したり、テスト目的でアプリケーションをクラウドにクローニングしたり、データセンターからクラウドに移行したりできます

データ移行

アプリケーションをある環境から別の環境に移行するには、ACCの次の機能のいずれかを使用できます。

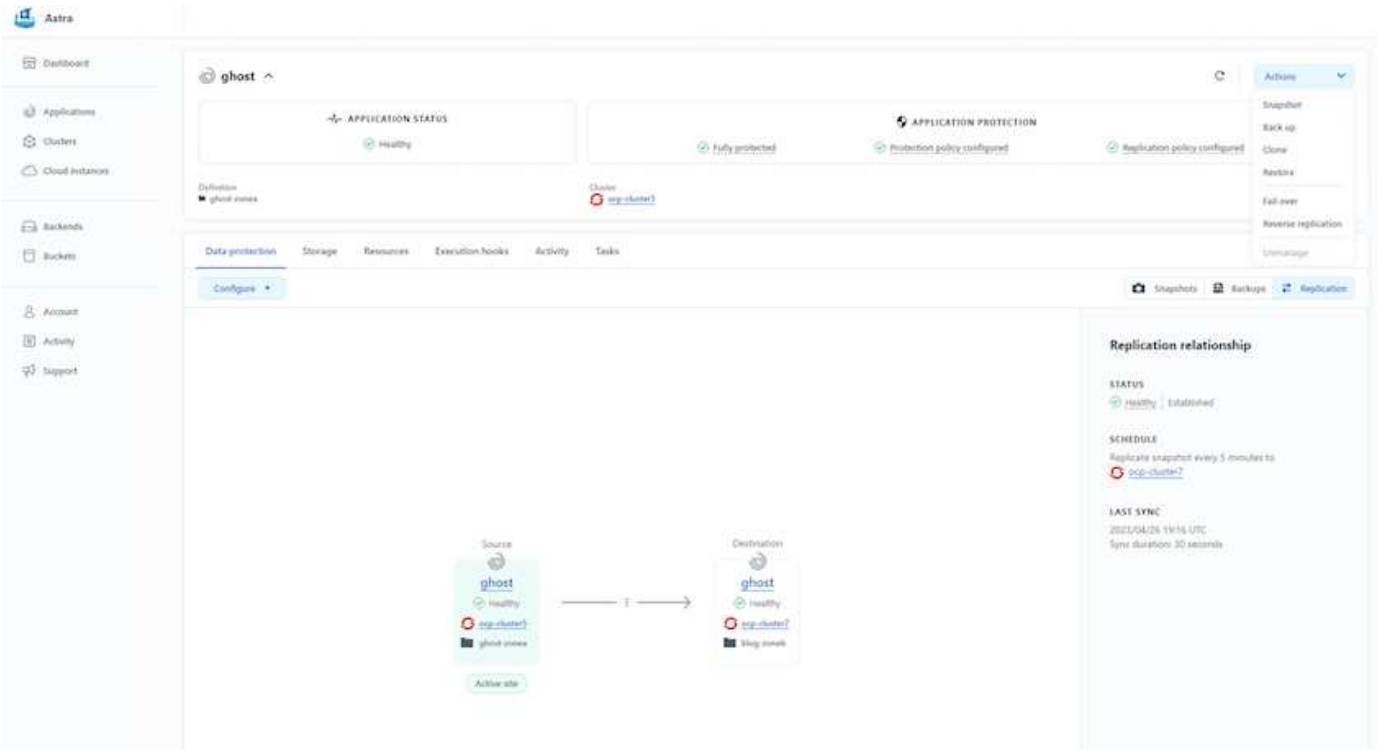
- レプリケーション

- バックアップとリストア
- クローン

を参照してください "データ保護セクション" レプリケーションおよびバックアップとリストアオプションの場合。を参照してください "こちらをご覧ください" クローン作成の詳細については、を参照してください。

> Astraレプリケーション機能は、Trident Container Storage Interface（CSI）でのみサポートされます。ただし、NASエコノミードライバとSANエコノミードライバでは、レプリケーションはサポートされていません。

ACCを使用したデータ複製の実行



Red Hat OpenShift Containerワークロード向けのネットアップハイブリッドマルチクラウドソリューション

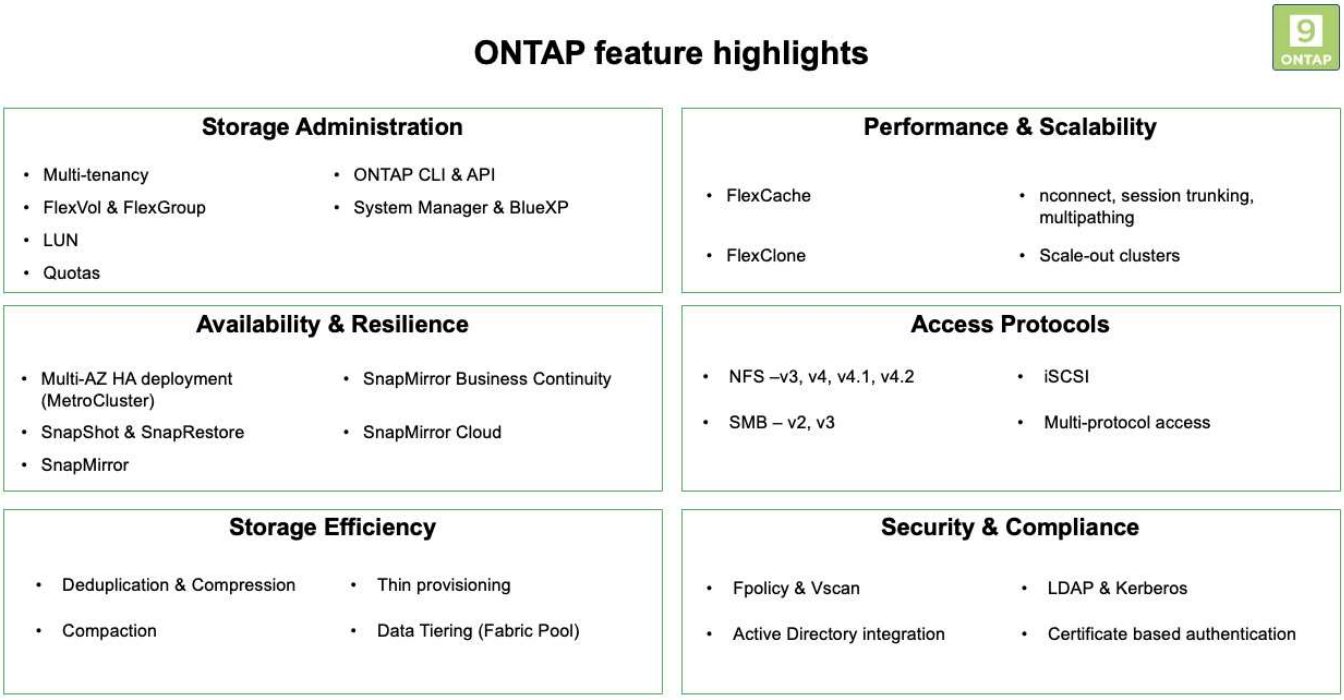
概要

ネットアップでは、従来型エンタープライズアプリケーションを最新化し、Kubernetesを中心に構築されたコンテナとオーケストレーションプラットフォームを使用して新しいアプリケーションを構築するお客様が大幅に増えています。Red Hat OpenShift Container Platformは、多くのお客様に採用されている例の1つです。

企業内でコンテナを採用するお客様がますます増えています。ネットアップは、ステートフルアプリケーションの永続的ストレージのニーズに加え、データ保護、データセキュリティ、データ移行などの従来のデータ管理のニーズにも応えることができます。しかし、これらのニーズは、さまざまな戦略、ツール、方法を使用して満たしています。

- NetApp ONTAP **ベースのストレージオプションを次に示します。コンテナとKubernetes環境にセキュリティ、データ保護、信頼性、柔軟性を提供します。

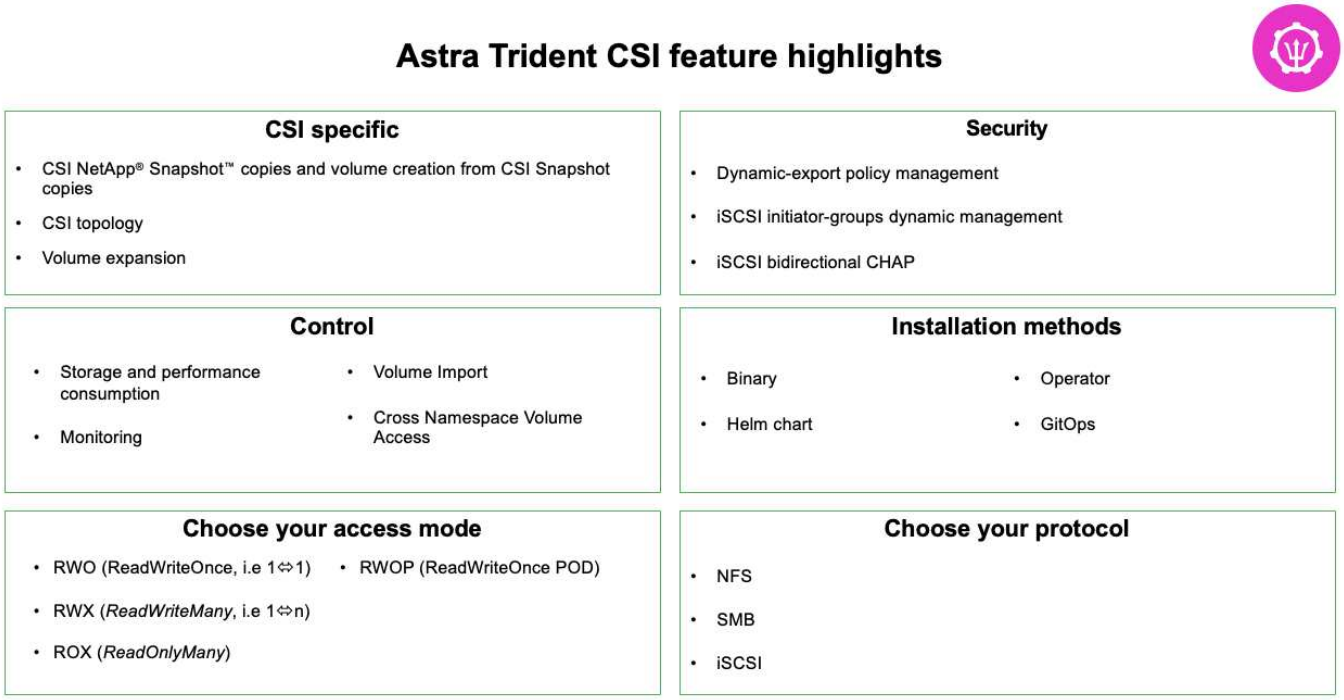
- オンプレミスの自己管理型ストレージ：
- ネットアップファブリック接続ストレージ（FAS ）、ネットアップオールフラッシュFAS アレイ（AFF ）、ネットアップオールSANアレイ（ASA ）、ONTAP Select
  - オンプレミスのプロバイダ管理ストレージ：
- NetApp Keystone が提供するストレージサービス（STaaS）
  - クラウド内の自己管理型ストレージ：
- NetApp Cloud Volumes ONTAP（CVO）は、ハイパースケーラに自己管理型ストレージを提供します
  - クラウド上のプロバイダが管理するストレージ：
- Cloud Volumes Service for Google Cloud（CVS）、Azure NetApp Files （ANF）、Amazon FSx for NetApp ONTAP は、ハイパースケーラにフルマネージドストレージを提供します



- NetApp BlueXP **を使用すると、すべてのストレージ資産とデータ資産を単一のコントロールプレーン/インターフェイスから管理できます。

BlueXPを使用して、クラウドストレージ（Cloud Volumes ONTAP やAzure NetApp Files など）の作成と管理、データの移動、保護、分析、オンプレミスやエッジの多くのストレージデバイスの管理を行うことができます。

- NetApp Astra Trident **はCSI準拠のストレージオーケストレーションツールです。上記のさまざまなネットアップストレージオプションを利用して、永続的ストレージをすばやく簡単に利用できます。ネットアップが保守、サポートしているオープンソースのソフトウェアです。



ビジネスクリティカルなコンテナワークロードに必要なのは、永続的ボリュームだけではありません。同社のデータ管理要件では、アプリケーションのKubernetesオブジェクトの保護と移行も必要です。

> アプリケーションデータには、ユーザデータに加えてKubernetesオブジェクトが含まれます。例を次に示します。 -ポッド仕様、PVC、デプロイ、サービスなどのKubernetesオブジェクト- 設定マップやシークレットなどのカスタム設定オブジェクト- Snapshotコピー、バックアップ、クローンなどの永続的データ- CRSやCRDなどのカスタムリソース

- NetApp Astra Control **は、フルマネージドと自己管理型の両方のソフトウェアとして提供され、堅牢なアプリケーションデータ管理のためのオーケストレーションを提供します。を参照してください "Astra のドキュメント" を参照してください。

このリファレンスドキュメントでは、NetApp Astra Control Centerを使用して、Red Hat OpenShiftコンテナプラットフォームに導入されたコンテナベースアプリケーションの移行と保護について検証します。また、解決策 では、コンテナプラットフォームを管理するためのRed Hat Advanced Cluster Management（ACM）の導入と使用に関する詳細についても説明しています。また、Astra Trident CSIプロビジョニングツールを使用して、ネットアップストレージとRed Hat OpenShiftコンテナプラットフォームを統合する方法についても詳しく説明します。Astra Control Centerはハブクラスタに導入され、コンテナアプリケーションとその永続的ストレージライフサイクルの管理に使用されます。最後に、Amazon FSx for NetApp ONTAP （FSxN）を永続的ストレージとして使用し、AWS（ROSA）のマネージドRed Hat OpenShiftクラスタ上のコンテナワークロードのレプリケーション、フェイルオーバー、フェイルバックのための解決策 を提供します。

NetApp解決策 とAWS上のマネージドRed Hat OpenShift Containerプラットフォームのワークロード

NetApp解決策 とAWS上のマネージドRed Hat OpenShift Containerプラットフォームのワークロード

お客様は、「クラウド生まれ」の場合もあれば、一部のワークロードやすべてのワークロードをデータセンターからクラウドに移行する準備ができた時点で、最新化に向けた取り組みを進めている場合もあります。ワークロードの実行に、プロバイダが管理す

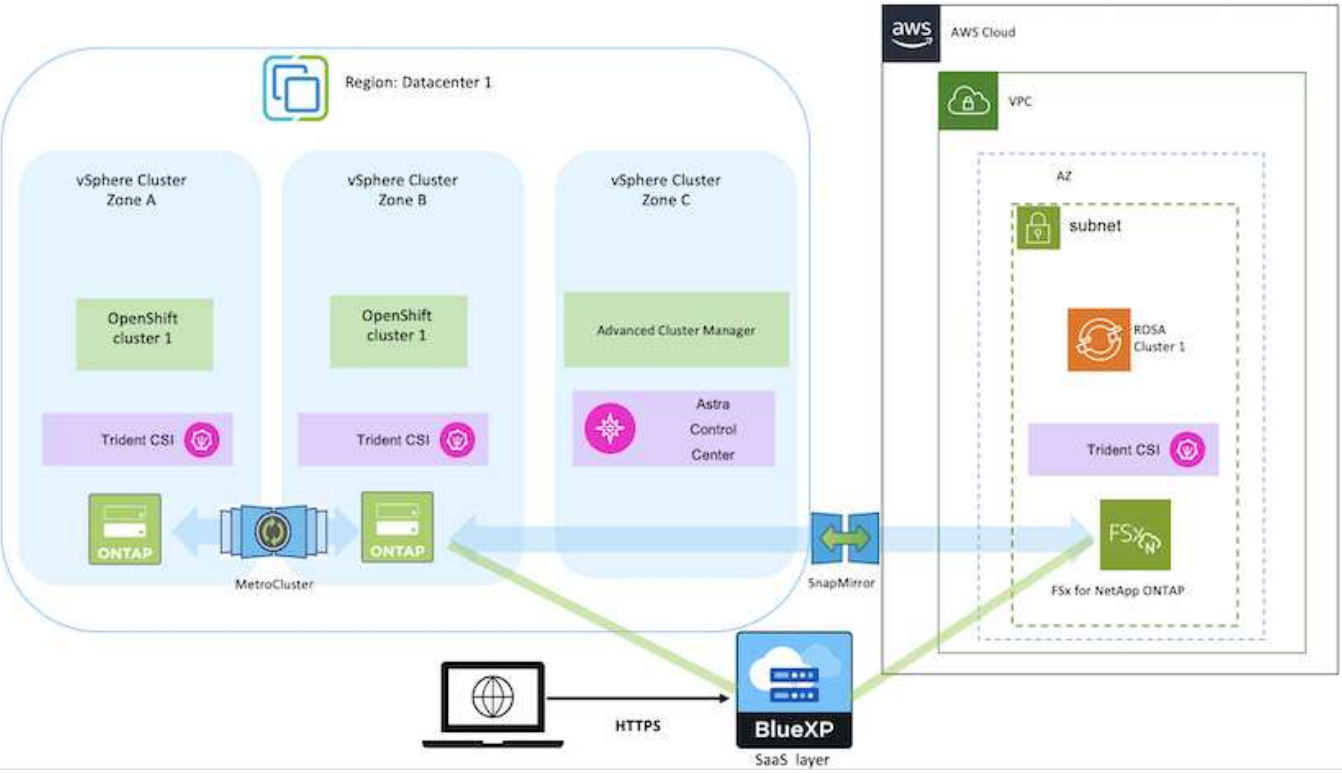
るOpenShiftコンテナとプロバイダが管理するネットアップストレージをクラウドで使用することもできます。コンテナワークロードに対応した本番環境を成功させるためには、マネージドRed Hat OpenShiftコンテナクラスタ（ROSA）をクラウドに計画して導入する必要があります。AWSクラウドにいる場合は、ストレージのニーズに合わせてFSx for NetApp ONTAP を導入することもできます。

FSx for NetApp ONTAP は、AWSのコンテナ導入にデータの保護、信頼性、柔軟性を提供します。Astra Tridentは、お客様のステートフルアプリケーション向けに永続的FSxNストレージを利用するための動的ストレージプロビジョニングツールとして機能します。

ROSAは、コントロールプレーンノードが複数のアベイラビリティゾーンに分散した状態でHAモードで導入できるため、FSx ONTAP は、高可用性を提供し、AZの障害から保護するマルチAZオプションを使用してプロビジョニングすることもできます。

> ファイルシステムの優先アベイラビリティゾーン（AZ）からAmazon FSxファイルシステムにアクセスする場合、データ転送料金は発生しません。価格設定の詳細については、を参照してください "こちらをご覧ください"。

OpenShift Containerワークロード向けのデータ保護と移行用解決策

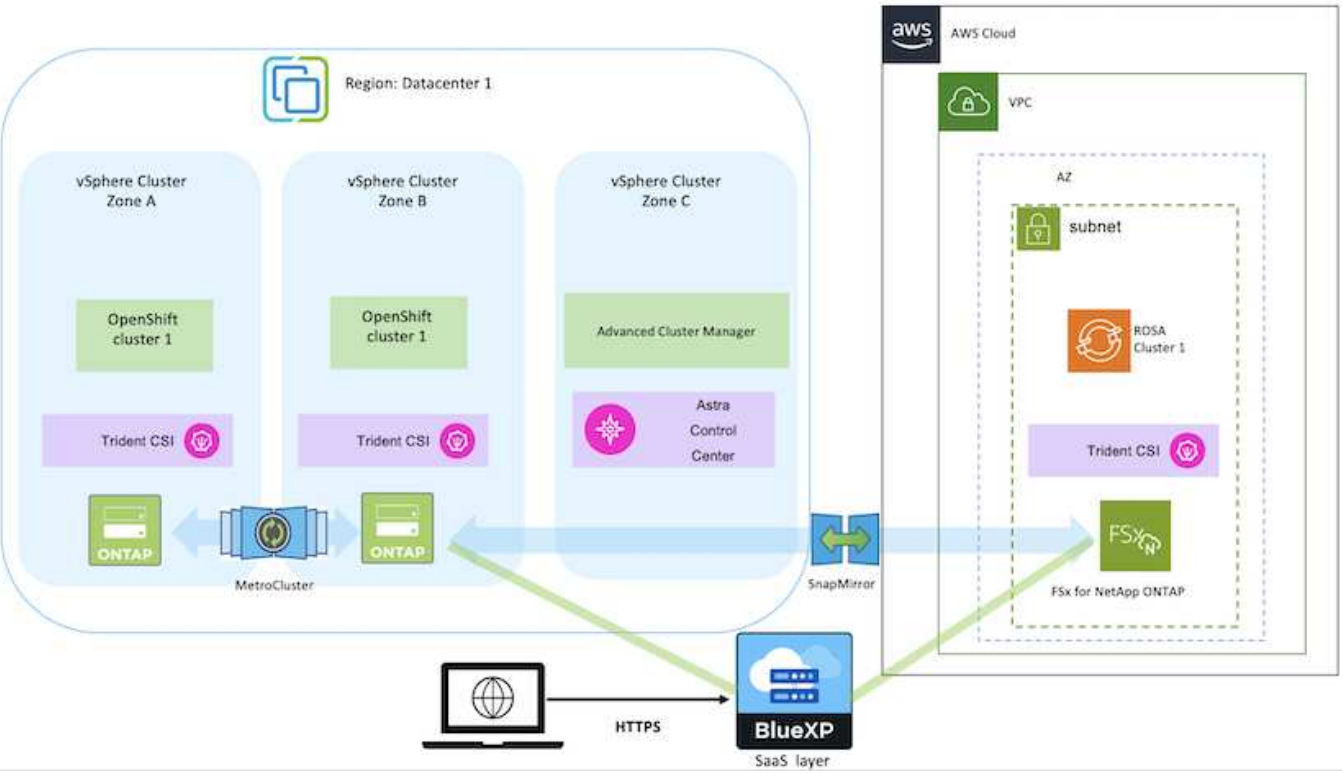


AWSにマネージド**Red Hat OpenShift Container**プラットフォームを導入して設定します

このセクションでは、AWS（ROSA）でマネージドRed Hat OpenShiftクラスタをセットアップする大まかなワークフローについて説明します。このスライドでは、Astra TridentによるストレージバックエンドとしてManaged FSx for NetApp ONTAP （FSxN）を使用して永続ボリュームを提供しています。BlueXPを使用したAWSへのFSxNの導

入について詳しく説明します。また、ROSAクラスタ上のステートフルアプリケーションに対して、BlueXPとOpenShiftのGitOps（Argo CD）を使用してデータ保護と移行のアクティビティを実行する方法についても詳しく説明します。

次の図は、AWSに導入され、FSxNをバックエンドストレージとして使用するROSAクラスタを示しています。



この解決策 は、AWSの2つのVPCで2つのROSAクラスタを使用して検証されました。各ROSAクラスタは、Astra Tridentを使用してFSxNに統合されています。ROSAクラスタとFSxNをAWSに導入するには、いくつかの方法があります。このセットアップの概要概要 には、使用した具体的な方法のドキュメントへのリンクが記載されています。その他の方法については、に記載されている関連リンクを参照してください "リソースセクション"。

セットアッププロセスは、次の手順に分けることができます。

ROSAクラスタをインストールします

- 2つのVPCを作成し、VPC間にVPCピアリング接続を設定します。
- を参照してください "こちらをご覧ください" ROSAクラスタのインストール手順については、を参照してください。

FSxNをインストールします

- BlueXPからVPCにFSxNをインストールします。を参照してください "こちらをご覧ください"（BlueXPアカウントの作成と使用を開始するため）を参照してください "こちらをご覧ください" FSxNのインストールに使用します。を参照してください "こちらをご覧ください" FSxNを管理するためにAWSでコネクタを作成します。
- AWSを使用してFSxNを導入する。を参照してください "こちらをご覧ください" AWSコンソールを使用した導入用。

ROSAクラスタへの**Trident**のインストール（**Helm**チャートを使用）

- Helmチャートを使用して、ROSAクラスタにTridentをインストールします。HelmチャートのURL：https://netapp.github.io/trident-helm-chart

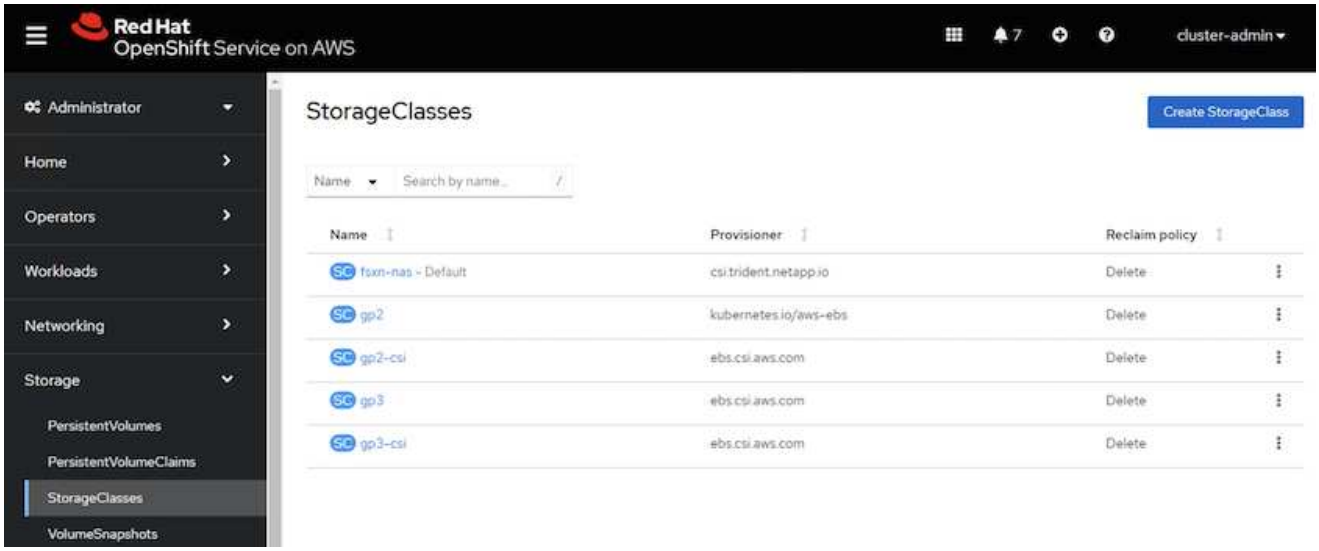
ROSAクラスタ向けのFSxNとAstra Tridentの統合

OpenShift GitOpsを使用すると、ApplicationSetを使用してArgoCDに登録されたすべての管理対象クラスタにAstra Trident CSIを導入できます。

```yaml
apiVersion: argoproj.io/v1alpha1
kind: ApplicationSet
metadata:
  name: trident-operator
spec:
  generators:
  - clusters: {}
    # selector:
    #   matchLabels:
    #     tridentversion: '23.04.0'
  template:
    metadata:
      name: '{{nameNormalized}}-trident'
    spec:
      destination:
        namespace: trident
        server: '{{server}}'
      source:
        repoURL: 'https://netapp.github.io/trident-helm-chart'
        targetRevision: 23.04.0
        chart: trident-operator
      project: default
      syncPolicy:
        syncOptions:
          - CreateNamespace=true
```

Tridentを使用したバックエンドとストレージクラスの作成（**FSxN**向け）

- を参照してください "こちらをご覧ください" バックエンドとストレージクラスの作成の詳細については、を参照してください。
- OpenShiftコンソールから、Trident CSIを使用してFsxN用に作成したストレージクラスをデフォルトで作成します。下のスクリーンショットを参照：



OpenShift GitOpsを使用したアプリケーションの導入（**Argo CD**）

- クラスタにOpenShift GitOpsオペレータをインストールします。手順を参照してください "こちらをご覧ください"。
- クラスタ用の新しいArgo CDインスタンスをセットアップします。手順を参照してください "こちらをご覧ください"。

Argo CDのコンソールを開き、アプリをデプロイします。たとえば、Argo CDとHelm Chartを使用してJenkins Appをデプロイできます。アプリケーションを作成するときに、次の詳細が提供されました。プロジェクト:デフォルトクラスタ: https://kubernetes.default.svc名前空間: Jenkins Helm ChartのURL : https://charts.bitnami.com/bitnami

Helmパラメータ：global.storageClass：fsxn -nas

データ保護

このページには、Astra Control Serviceを使用したAWS（ROSA）クラスタでのマネージドRed Hat OpenShiftのデータ保護オプションが表示されます。Astra Control Service（ACS）では、使いやすいグラフィカルユーザインターフェイスを使用して、クラスタの追加、クラスタ上で実行されるアプリケーションの定義、アプリケーション対応のデータ管理アクティビティの実行を行うことができます。ACS関数には、ワークフローの自動化を可能にするAPIを使用してアクセスすることもできます。

Astra Control（ACSまたはACC）に搭載されるのは、NetApp Astra Tridentです。Astra Tridentは、Red Hat OpenShift、EKS、AKS、SUSE Rancher、Anthosなど、いくつかのタイプのKubernetesクラスタを統合しま

す。 FAS / AFF、ONTAP Select、CVO、Google Cloud Volumes Service、Azure NetApp Files、Amazon FSx for NetApp ONTAPなど、さまざまな種類のNetApp ONTAPストレージを使用できます。

ここでは、ACSを使用した次のデータ保護オプションの詳細について説明します。

- ある地域で実行されているROSAアプリケーションのバックアップと復元と、別の地域への復元を示すビデオ。
- ROSAアプリケーションのスナップショットと復元を示すビデオ。
- ROSAクラスタ（Amazon FSx for NetApp ONTAP）のインストール、NetApp Astra Tridentを使用したストレージバックエンドとの統合、ROSAクラスタへのPostgreSQLアプリケーションのインストール、ACSを使用したアプリケーションのスナップショットの作成とそこからのアプリケーションのリストアの詳細を順を追って説明します。
- ACSを使用してFSx for ONTAPを使用してROSAクラスタ上のMySQLアプリケーションのスナップショットを作成し、そのスナップショットからリストアする手順の詳細を示すブログ。

バックアップ/バックアップからのリストア

次のビデオは、あるリージョンで実行されているROSAアプリケーションのバックアップと、別のリージョンへのリストアを示しています。

AWSでのRed Hat OpenShift向けFSx NetApp ONTAPサービス

Snapshot / Snapshotからのリストア

次のビデオでは、ROSAアプリケーションのスナップショットを作成してからスナップショットから復元する方法を示します。

Amazon FSx for NetApp ONTAPストレージを使用したRed Hat OpenShift Service on AWS（ROSA）クラスタでのアプリケーションのスナップショット/リストア

ブログ

- "Amazon FSxストレージを使用したROSAクラスタ上のアプリケーションのデータ管理にAstra Control Serviceを使用"

スナップショットを作成してそこからリストアするためのステップバイステップの詳細

セットアップの前提条件

- "AWS アカウント"
- "Red Hat OpenShiftアカウント"
- IAMユーザ "適切な権限" ROSAクラスタを作成してアクセスするには
- "AWS CLI"
- "ローザCLI"
- "OpenShift CLI"（OC）
- サブネットと適切なゲートウェイおよびルートを備えたvPC
- "Rosaクラスタインストール済み" VPCに挿入

- "NetApp ONTAP 対応の Amazon FSX" 同じVPCに作成
- ROSAクラスタへのアクセス "OpenShiftハイブリッドクラウドコンソール"

次のステップ

1. 管理者ユーザを作成し、クラスタにログインします。
2. クラスタ用のkubeconfigファイルを作成します。
3. クラスタにAstra Tridentをインストール
4. Trident CSIプロビジョニングツールを使用して、バックエンド、ストレージクラス、Snapshotクラスの構成を作成
5. クラスタにPostgreSQLアプリケーションを導入します。
6. データベースを作成し、レコードを追加します。
7. クラスタをACSに追加します。
8. ACSでアプリケーションを定義します。
9. ACSを使用してスナップショットを作成します。
10. PostgreSQLアプリケーションでデータベースを削除します。
11. ACSを使用してスナップショットから復元します。
12. アプリがスナップショットから復元されたことを確認します。

1：管理者ユーザを作成してクラスタにログイン

次のコマンドを使用してadminユーザを作成し、ROSAクラスタにアクセスします（adminユーザを作成する必要があるのは、インストール時にadminユーザを作成しなかった場合だけです）。

```
rosa create admin --cluster=<cluster-name>
```

次のような出力が表示されます。を使用してクラスタにログインします。 `oc login` コマンドは出力に表示されます。

```
W: It is recommended to add an identity provider to login to this cluster.
See 'rosa create idp --help' for more information.
I: Admin account has been added to cluster 'my-rosa-cluster'. It may take up
to a minute for the account to become active.
I: To login, run the following command:
oc login https://api.my-rosa-cluster.abcd.p1.openshiftapps.com:6443 \
--username cluster-admin \
--password FWGYL-2mkJI-00000-00000
```

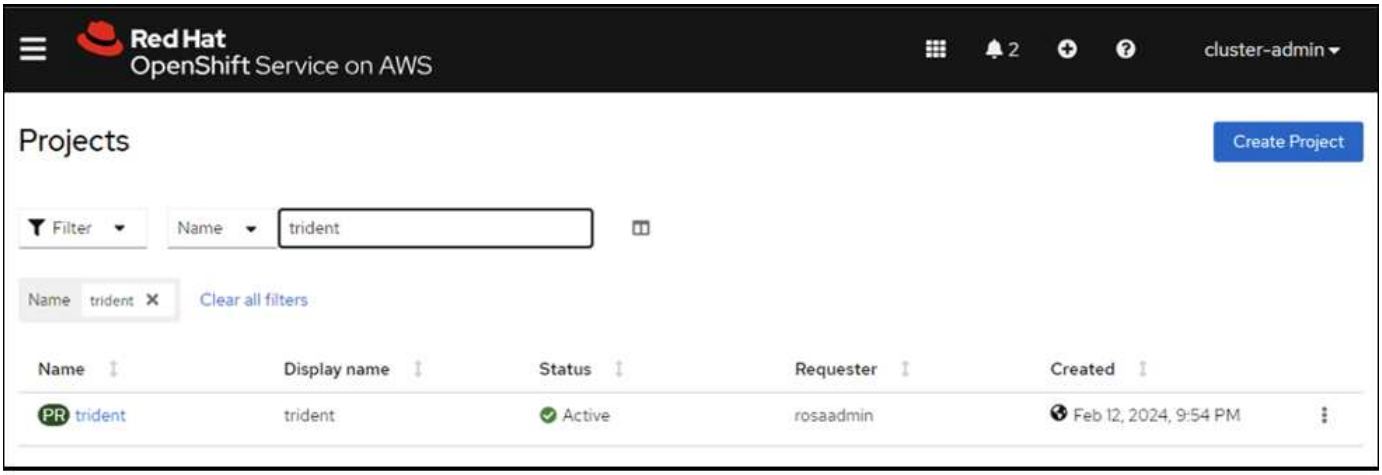
トークンを使用してクラスタにログインすることもできます。クラスタの作成時にすでにadminユーザを作成している場合は、Red Hat OpenShift Hybrid Cloudコンソールからadminユーザのクレデンシャルを使用してクラスタにログインできます。右上隅にログインしているユーザの名前が表示されていることをクリックすると、 `oc login` コマンドラインのコマンド（トークンログイン）。

2.クラスタの**kubeconfig**ファイルを作成

手順に従います "こちらをご覧ください" ROSAクラスタ用のkubeconfigファイルを作成します。このkubeconfigファイルは、あとでクラスタをACSに追加するときに使用されます。

3.クラスタへの**Astra Trident**のインストール

ROSAクラスタにAstra Trident（最新バージョン）をインストールこれを行うには、以下の手順のいずれかに従うことができます。 "こちらをご覧ください"。クラスタのコンソールからhelmを使用してTridentをインストールするには、まずTridentというプロジェクトを作成します。



次に、[開発者]ビューからHelmチャートリポジトリを作成します。URLフィールドの使用 `'https://netapp.github.io/trident-helm-chart'`。次に、Tridentオペレータ用のHelmリリースを作成します。

Project: trident

Create Helm Chart Repository

Add helm chart repository.

Configure via: (•) Form view ( ) YAML view

Scope type

( ) Namespaced scoped (ProjectHelmChartRepository)

Add Helm Chart Repository in the selected namespace.

(•) Cluster scoped (HelmChartRepository)

Add Helm Chart Repository at the cluster level and in all namespaces.

Name *

trident

A unique name for the Helm Chart repository.

Display name

Astra Trident

A display name for the Helm Chart repository.

Description

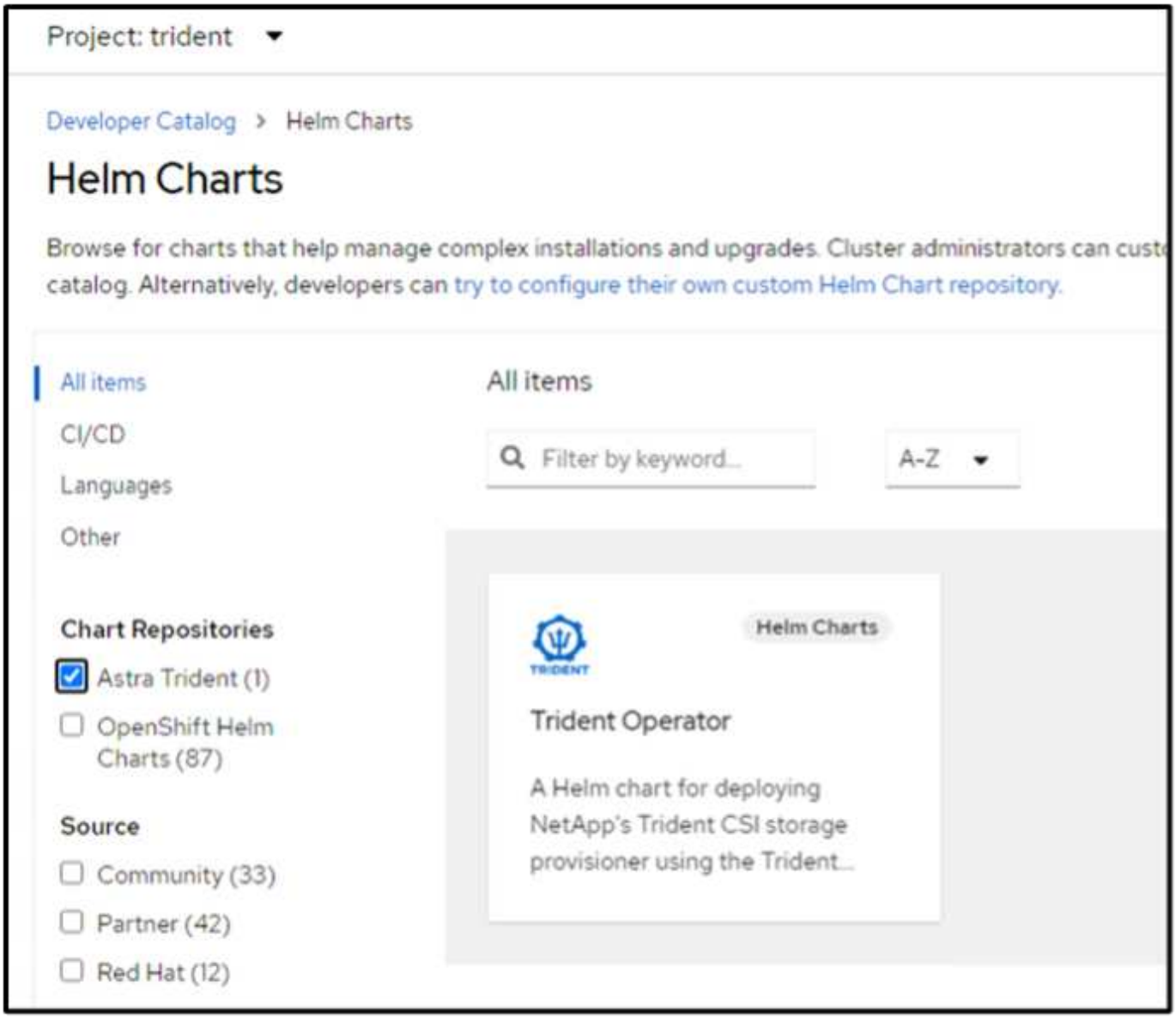
NetApp Astra Trident

A description for the Helm Chart repository.

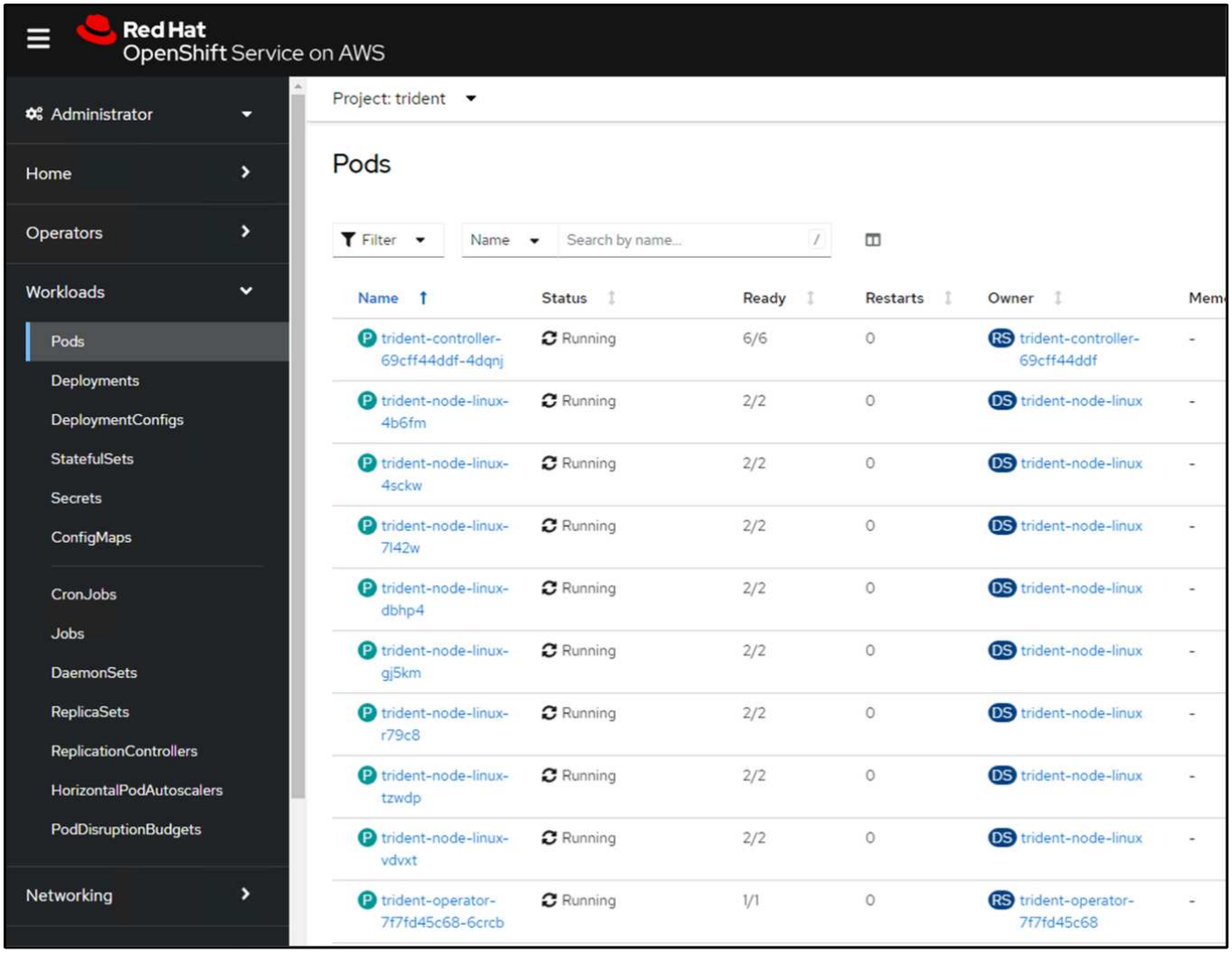
[ ] Disable usage of the repo in the developer catalog.

URL *

https://netapp.github.io/trident-helm-chart



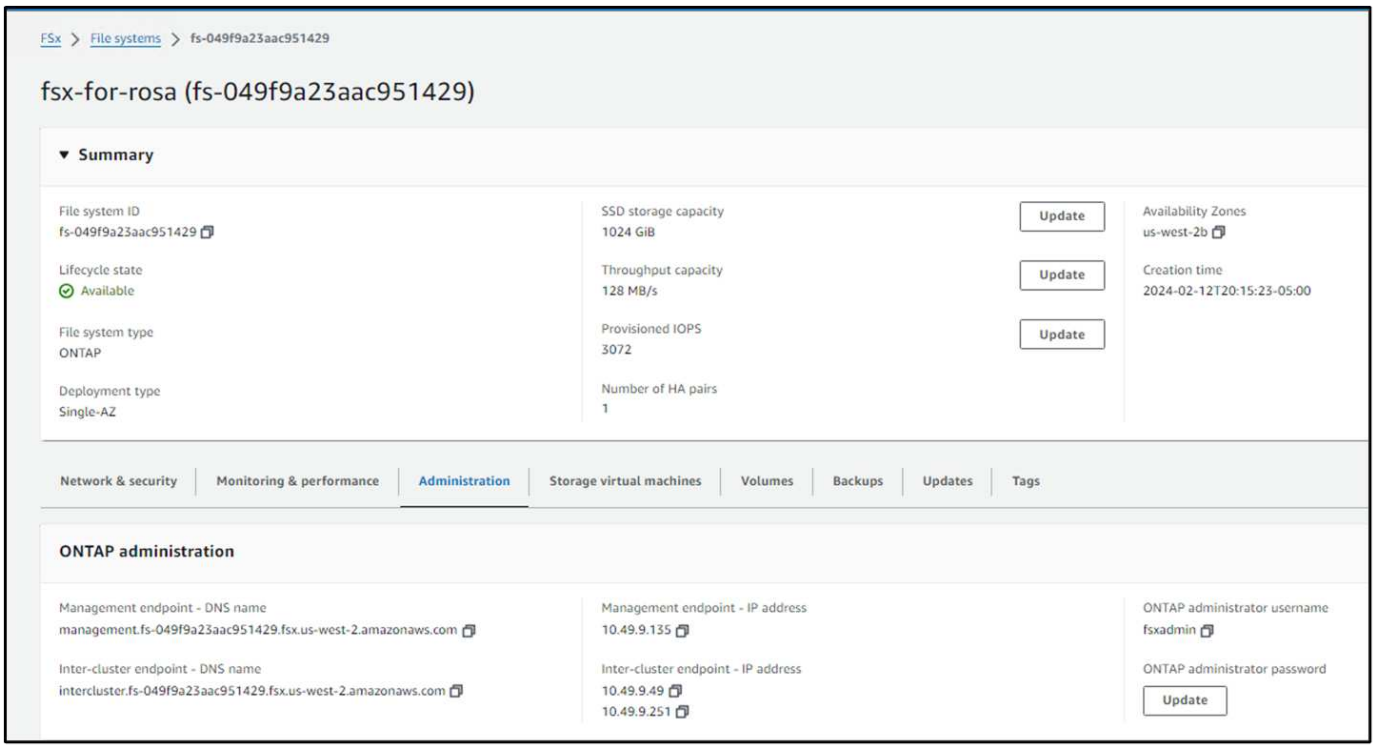
コンソールの管理者ビューに戻り、Tridentプロジェクトでポッドを選択して、すべてのTridentポッドが実行されていることを確認します。



4.Trident CSIプロビジョニングツールを使用して、バックエンド、ストレージクラス、スナップショットクラスの構成を作成

以下のYAMLファイルを使用して、Tridentバックエンドオブジェクト、ストレージクラスオブジェクト、およびVolumesnapshotオブジェクトを作成します。作成したAmazon FSx for NetApp ONTAPファイルシステム、管理LIF、およびファイルシステムのSVM名のクレデンシャルを、バックエンドの構成YAMLで指定してください。これらの詳細を確認するには、Amazon FSxのAWSコンソールに移動し、ファイルシステムを選択して、[管理]タブに移動します。また、[UPDATE]をクリックして、 `fsxadmin` ユーザ：

> コマンドラインを使用して、ハイブリッドクラウドコンソールからオブジェクトを作成したり、YAMLファイルを使用してオブジェクトを作成したりできます。



- Tridentバックエンド構成**

```
apiVersion: v1
kind: Secret
metadata:
  name: backend-tbc-ontap-nas-secret
type: Opaque
stringData:
  username: fsxadmin
  password: <password>
---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: ontap-nas
spec:
  version: 1
  storageDriverName: ontap-nas
  managementLIF: <management lif>
  backendName: ontap-nas
  svm: fsx
  credentials:
    name: backend-tbc-ontap-nas-secret
```

ストレージクラス

```yaml
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: ontap-nas
provisioner: csi.trident.netapp.io
parameters:
  backendType: "ontap-nas"
  media: "ssd"
  provisioningType: "thin"
  snapshots: "true"
allowVolumeExpansion: true
```

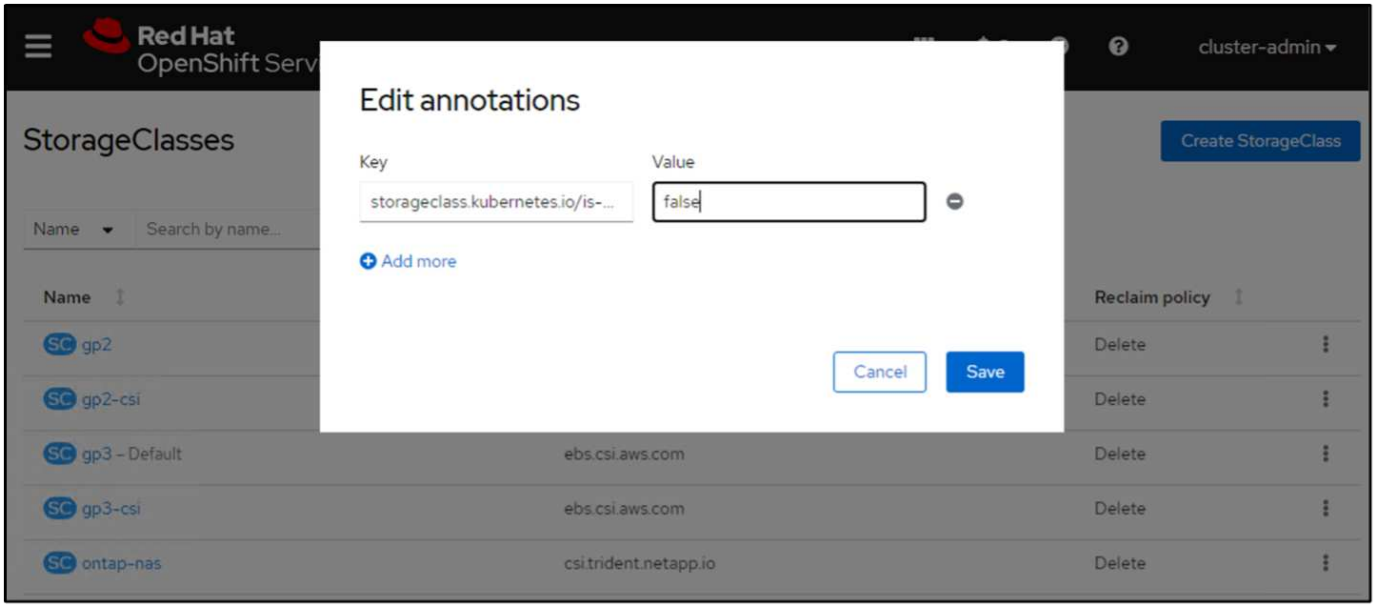
スナップショットクラス

```yaml
apiVersion: snapshot.storage.k8s.io/v1
kind: VolumeSnapshotClass
metadata:
  name: trident-snapshotclass
driver: csi.trident.netapp.io
deletionPolicy: Delete
```

以下のコマンドを実行して、バックエンド、ストレージクラス、およびtrident-snapshotclassオブジェクトが作成されたことを確認します。

```
[ec2-user@ip-10-49-11-132 storage]$ kubectl get tbc -n trident
NAME        BACKEND NAME   BACKEND UUID                           PHASE   STATUS
ontap-nas   ontap-nas      8a5e4583-2dac-46bb-b01e-fa7c3816f121   Bound   Success
[ec2-user@ip-10-49-11-132 storage]$ kubectl get sc
NAME            PROVISIONER             RECLAIMPOLICY   VOLUMEBINDINGMODE      ALLOWVOLUMEEXPANSION   AGE
gp2             kubernetes.io/aws-ebs   Delete          WaitForFirstConsumer   true                   3h23m
gp2-csi         ebs.csi.aws.com         Delete          WaitForFirstConsumer   true                   3h19m
gp3 (default)   ebs.csi.aws.com         Delete          WaitForFirstConsumer   true                   3h23m
gp3-csi         ebs.csi.aws.com         Delete          WaitForFirstConsumer   true                   3h19m
ontap-nas       csi.trident.netapp.io   Delete          Immediate              true                   141m
[ec2-user@ip-10-49-11-132 storage]$ kubectl get Volumesnapshotclass
NAME                    DRIVER                  DELETIONPOLICY   AGE
csi-aws-vsc             ebs.csi.aws.com         Delete           3h19m
trident-snapshotclass   csi.trident.netapp.io   Delete           6m56s
[ec2-user@ip-10-49-11-132 storage]$
```

この時点で重要な変更点は、あとで導入するPostgreSQLアプリケーションでデフォルトのストレージクラスを使用できるように、ONTAP-NASをgp3ではなくデフォルトのストレージクラスに設定することです。クラスタのOpenShiftコンソールで、[Storage]で[StorageClasses]を選択します。現在のデフォルトクラスのアノテーションをfalseに編集し、ontap-nasストレージクラスに対してstorageclass.kubernetes.io/is-default-classをtrueに設定して追加します。



5.クラスタに**PostgreSQL**アプリケーションを導入する

次のように、コマンドラインからアプリケーションをデプロイできます。

```
helm install postgresql bitnami/postgresql -n postgresql --create-namespace
```

```
[ec2-user@ip-10-49-11-132 astra]$ helm install postgresql bitnami/postgresql -n postgresql --create-namespace
NAME: postgresql
LAST DEPLOYED: Tue Feb 13 14:46:16 2024
NAMESPACE: postgresql
STATUS: deployed
REVISION: 1
TEST SUITE: None
NOTES:
CHART NAME: postgresql
CHART VERSION: 14.0.4
APP VERSION: 16.2.0

** Please be patient while the chart is being deployed **

PostgreSQL can be accessed via port 5432 on the following DNS names from within your cluster:

    postgresql.postgresql.svc.cluster.local - Read/Write connection

To get the password for "postgres" run:

    export POSTGRES_PASSWORD=$(kubectl get secret --namespace postgresql postgresql -o jsonpath="{.data.postgres-password}" | base64 -d)

To connect to your database run the following command:

    kubectl run postgresql-client --rm --tty -i --restart='Never' --namespace postgresql --image docker.io/bitnami/postgresql:16.2.0-debian-11-r1 --env="PG
PASSWORD=$POSTGRES_PASSWORD" \
      --command -- psql --host postgresql -U postgres -d postgres -p 5432

    > NOTE: If you access the container using bash, make sure that you execute "/opt/bitnami/scripts/postgresql/entrypoint.sh /bin/bash" in order to avoid
the error "psql: local user with ID 1001} does not exist"

To connect to your database from outside the cluster execute the following commands:

    kubectl port-forward --namespace postgresql svc/postgresql 5432:5432 &
    PGPASSWORD="$POSTGRES_PASSWORD" psql --host 127.0.0.1 -U postgres -d postgres -p 5432

WARNING: The configured password will be ignored on new installation in case when previous PostgreSQL release was deleted through the helm command. In that
 case, old PVC will have an old password, and setting it through helm won't take effect. Deleting persistent volumes (PVs) will solve the issue.
[ec2-user@ip-10-49-11-132 astra]$
```

アプリケーションポッドが実行されていない場合は、セキュリティコンテキストの制約が原因でエラーが発生している可能性があります。

```
[ec2-user@ip-10-49-11-132 astra]$ kubectl get all -n postgresql
NAME                    TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)    AGE
service/postgresql      ClusterIP   172.30.245.50   <none>        5432/TCP   12m
service/postgresql-hl   ClusterIP   None            <none>        5432/TCP   12m

NAME                          READY   AGE
statefulset.apps/postgresql   0/1     12m
[ec2-user@ip-10-49-11-132 astra]$ kubectl get events -n postgresql
LAST SEEN   TYPE      REASON                 OBJECT                                  MESSAGE
2m39s       Normal    WaitForFirstConsumer   persistentvolumeclaim/data-postgresql-0   waiting for first consumer to be created before binding
12m         Normal    SuccessfulCreate       statefulset/postgresql                  create Claim data-postgresql-0 Pod postgresql-0 in StatefulSet postg
resql success
107s        Warning   FailedCreate           statefulset/postgresql                  create Pod postgresql-0 in StatefulSet postgresql failed error: pods
 "postgresql-0" is forbidden: unable to validate against any security context constraint: [provider "trident-controller": Forbidden: not usable by user or
serviceaccount, provider "anyuid": Forbidden: not usable by user or serviceaccount, provider restricted-v2: .spec.securityContext.fsGroup: Invalid value: [
]int64{1001}: 1001 is not an allowed group, provider restricted-v2: .containers[0].runAsUser: Invalid value: 1001: must be in the ranges: [1001010000, 1001
019999], provider "restricted": Forbidden: not usable by user or serviceaccount, provider "nonroot-v2": Forbidden: not usable by user or serviceaccount, pr
ovider "nonroot": Forbidden: not usable by user or serviceaccount, provider "pcap-dedicated-admins": Forbidden: not usable by user or serviceaccount, provi
der "hostmount-anyuid": Forbidden: not usable by user or serviceaccount, provider "machine-api-termination-handler": Forbidden: not usable by user or servi
ceaccount, provider "hostnetwork-v2": Forbidden: not usable by user or serviceaccount, provider "hostnetwork": Forbidden: not usable by user or serviceacco
unt, provider "hostaccess": Forbidden: not usable by user or serviceaccount, provider "splunkforwarder": Forbidden: not usable by user or serviceaccount, p
rovider "trident-node-linux": Forbidden: not usable by user or serviceaccount, provider "node-exporter": Forbidden: not usable by user or serviceaccount, p
rovider "privileged": Forbidden: not usable by user or serviceaccount]
[ec2-user@ip-10-49-11-132 astra]$
```

を編集してエラーを修正します。 `runAsUser` および `fsGroup` フィールド `statefuleset.apps/postgresql` の出力にあるuidを持つオブジェクト `oc get project` 次のようにコマンドを実行します。

```
[ec2-user@ip-10-49-11-132 astra]$ oc get project postgresql -o yaml | grep uid-range
    openshift.io/sa.scc.uid-range: 1001010000/10000
[ec2-user@ip-10-49-11-132 astra]$ oc edit -n postgresql statefulset.apps/postgresql
statefulset.apps/postgresql edited
[ec2-user@ip-10-49-11-132 astra]$
```

PostgreSQLアプリケーションを実行し、Amazon FSx for NetApp ONTAPストレージを基盤とする永続ボリュームを使用する必要があります。

```
[ec2-user@ip-10-49-11-132 astra]$ oc get pods -n postgresql
NAME            READY   STATUS    RESTARTS   AGE
postgresql-0    1/1     Running   0          2m46s
[ec2-user@ip-10-49-11-132 astra]$
```

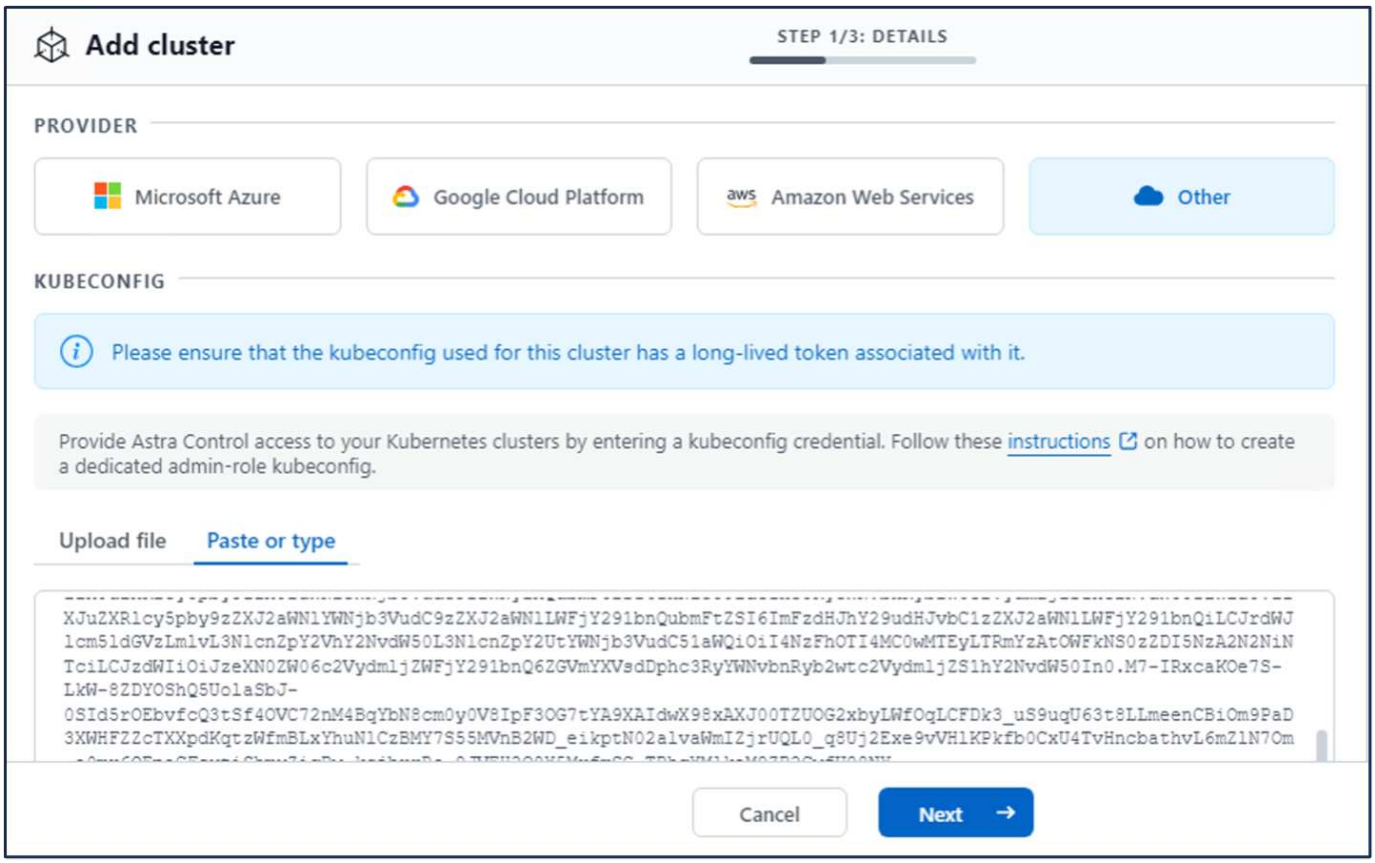
```
[ec2-user@ip-10-49-11-132 storage]$ kubectl get pvc -n postgresql
NAME                STATUS   VOLUME                                     CAPACITY   ACCESS MODES   STORAGECLASS   AGE
data-postgresql-0   Bound    pvc-dd09524a-de75-4825-9424-03a9b91195ca   8Gi        RWO            ontap-nas      4m2s
[ec2-user@ip-10-49-11-132 storage]$
```

6.データベースの作成とレコードの追加

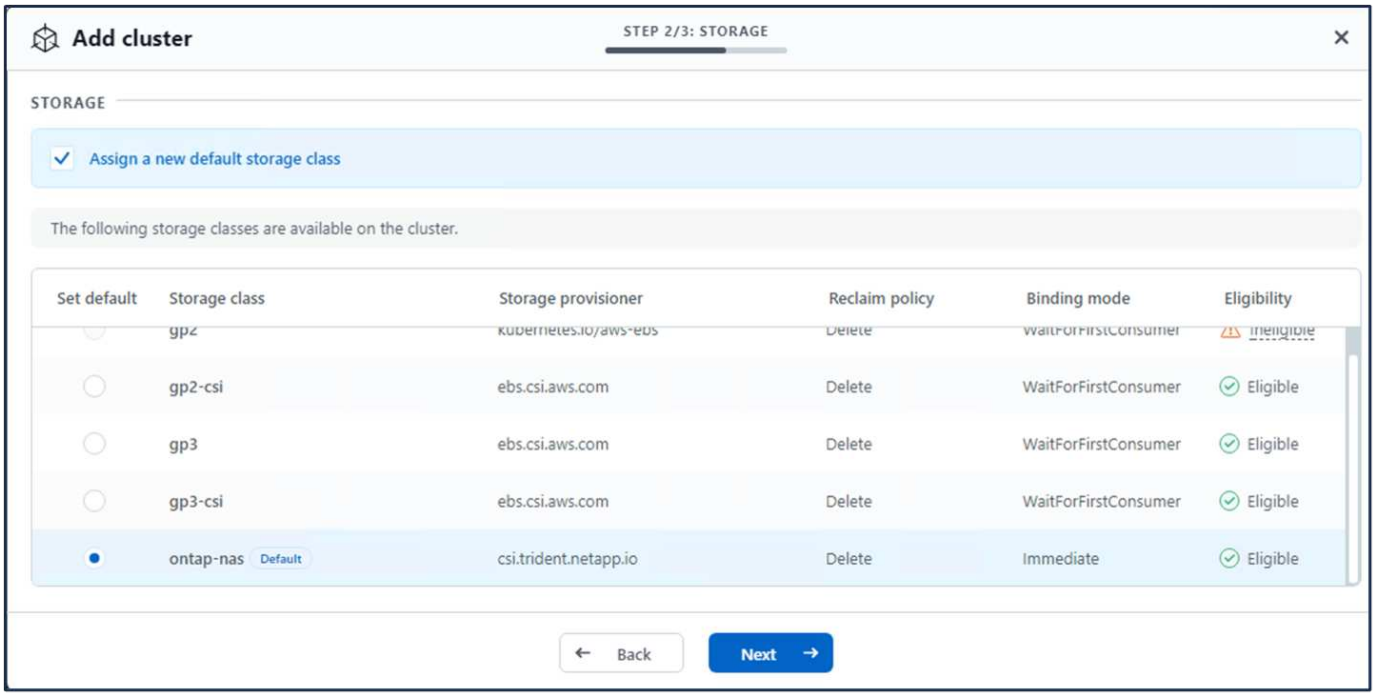
```
[ec2-user@ip-10-49-11-132 astra]$ export POSTGRES_PASSWORD=$(kubectl get secret --namespace postgresql postgresql -o jsonpath="
{.data.postgres-password}" | base64 -d)
[ec2-user@ip-10-49-11-132 astra]$ kubectl run postgresql-client --rm --tty -i --restart='Never' --namespace postgresql --image
docker.io/bitnami/postgresql:16.2.0-debian-11-r1 --env="PGPASSWORD=$POSTGRES_PASSWORD" \
> --command -- psql --host postgresql -U postgres -d postgres -p 5432
Warning: would violate PodSecurity "restricted:v1.24": allowPrivilegeEscalation != false (container "postgresql-client" must se
t securityContext.allowPrivilegeEscalation=false), unrestricted capabilities (container "postgresql-client" must set securityCo
ntext.capabilities.drop=["ALL"]), runAsNonRoot != true (pod or container "postgresql-client" must set securityContext.runAsNonR
oot=true), seccompProfile (pod or container "postgresql-client" must set securityContext.seccompProfile.type to "RuntimeDefault
" or "Localhost")
If you don't see a command prompt, try pressing enter.

postgres=# CREATE DATABASE erp;
CREATE DATABASE
postgres=# \c erp
You are now connected to database "erp" as user "postgres".
erp=# CREATE TABLE PERSONS(ID INT PRIMARY KEY NOT NULL, FIRSTNAME TEXT NOT NULL, LASTNAME TEXT NOT NULL);
CREATE TABLE
erp=# INSERT INTO PERSONS VALUES(1,'John','Doe');
INSERT 0 1
erp=# \dt
          List of relations
 Schema |  Name   | Type  |  Owner
--------+---------+-------+----------
 public | persons | table | postgres
(1 row)

erp=# SELECT * FROM persons;
 id | firstname | lastname
----+-----------+----------
  1 | John      | Doe
(1 row)
```

7.ACSへのクラスタの追加

ACSにログインします。クラスタを選択し、[Add]をクリックします。[Other]を選択し、kubeconfigファイルをアップロードまたは貼り付けます。



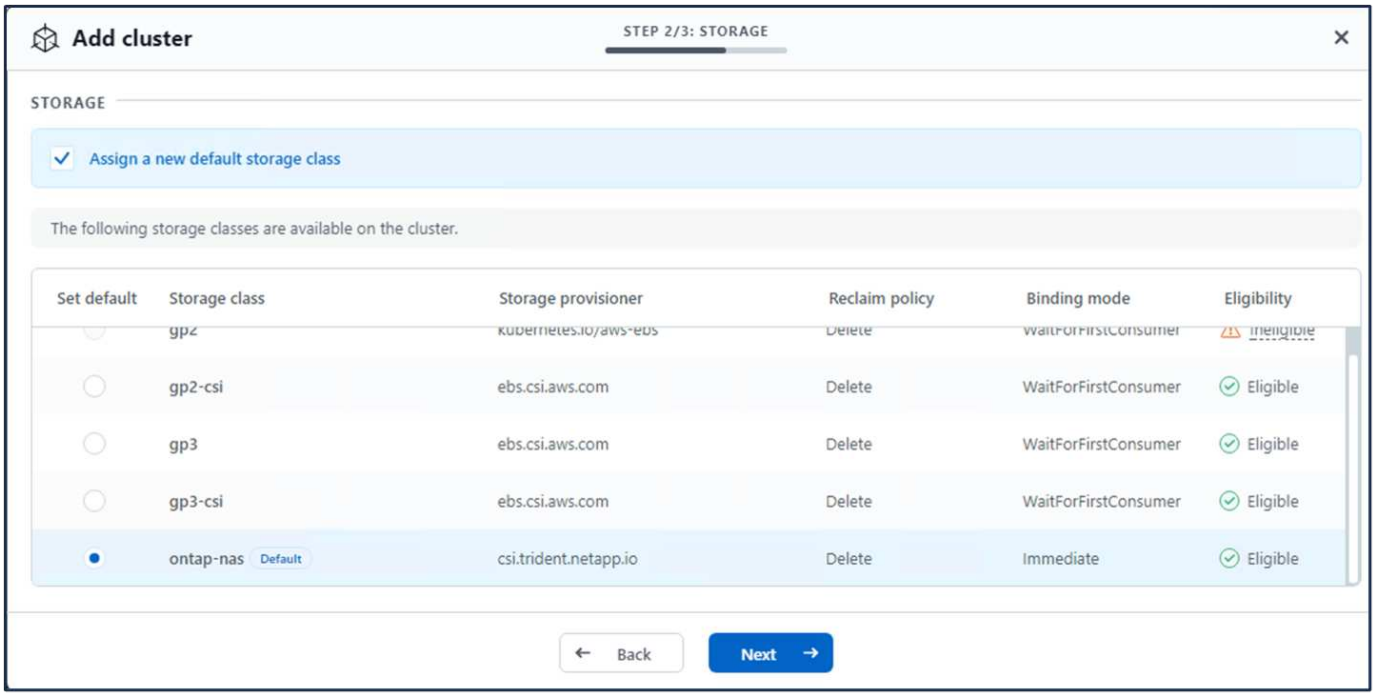
をクリックし、**ACS**のデフォルトのストレージクラスとして**[ONTAP-NAS]**を選択します。**[**次へ**]***をクリックし、詳細を確認して[クラスタを追加]*をクリックします。



8.ACSでのアプリケーションの定義

ACSでPostgreSQLアプリケーションを定義します。ランディングページで*、[定義]を選択し、適切な詳細を入力します。**[**次へ**]***を数回クリックし、詳細を確認して[定義]*をクリックします。アプリケーションがACSに

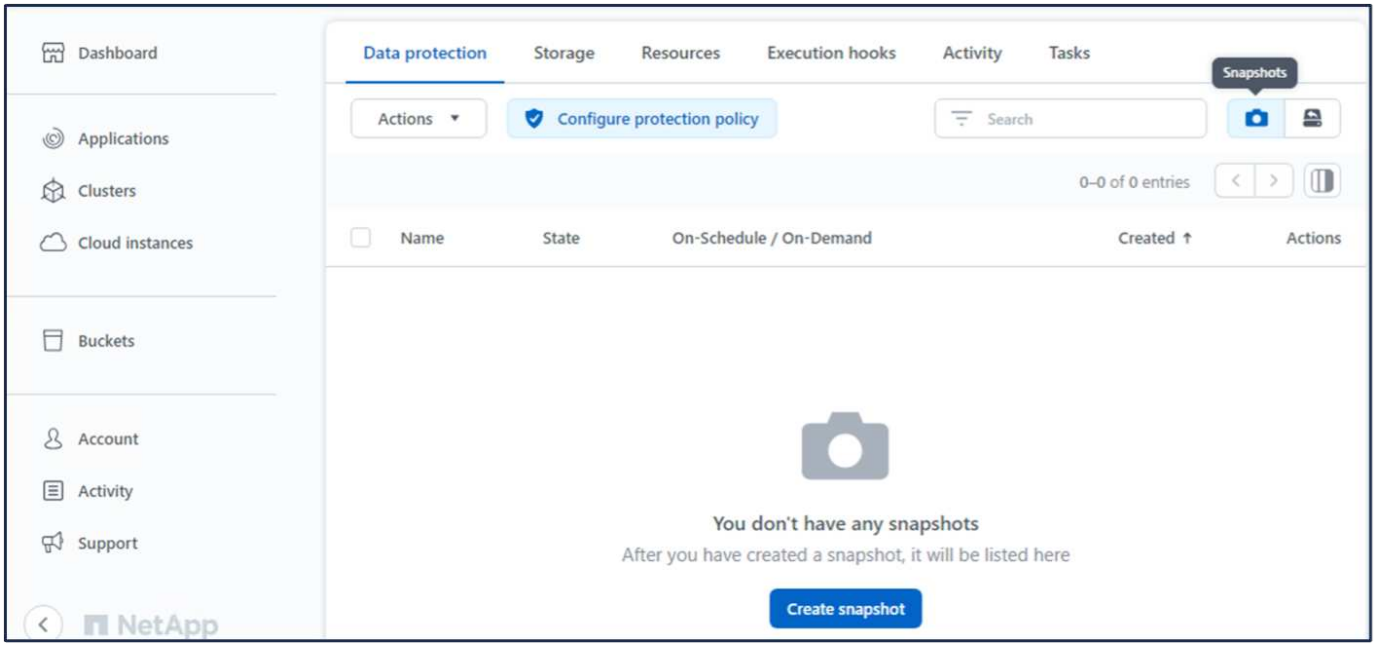
追加されます。

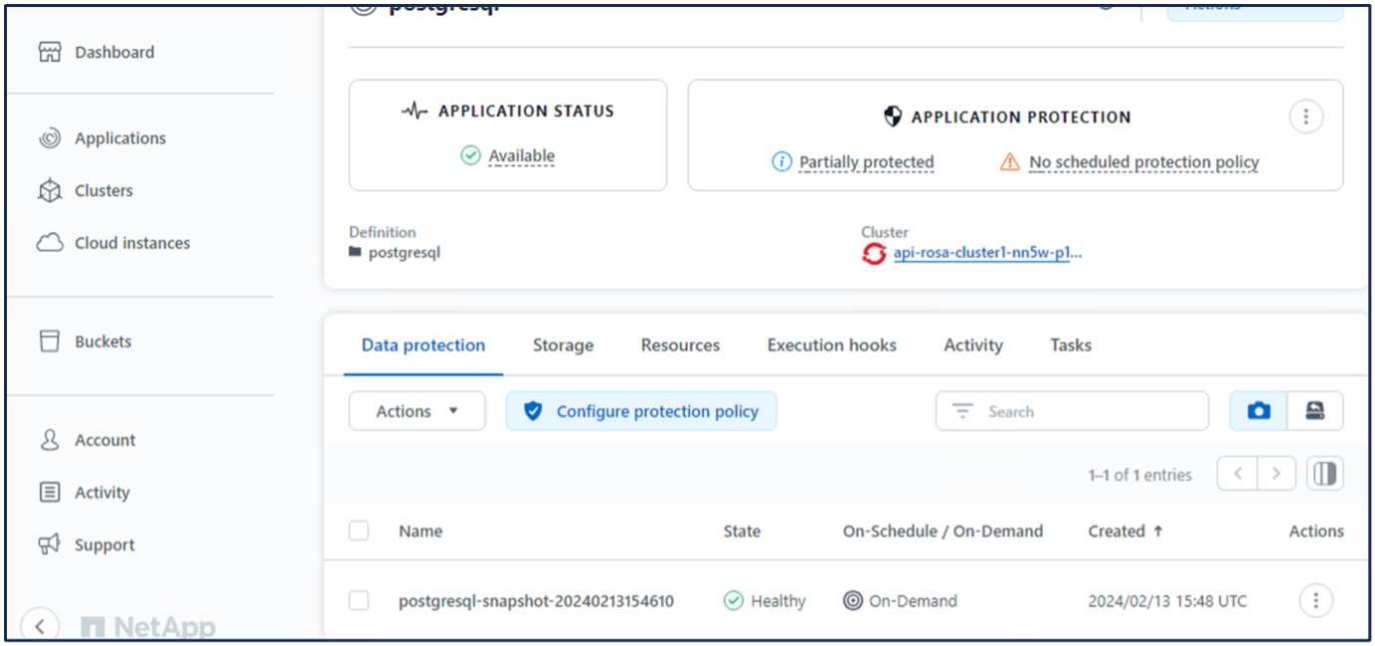


9.ACSを使用したスナップショットの作成

ACSでスナップショットを作成するには、さまざまな方法があります。アプリケーションを選択し、アプリケーションの詳細が表示されたページからスナップショットを作成できます。[Create snapshot]をクリックすると、オンデマンドSnapshotを作成したり、保護ポリシーを設定したりできます。

をクリックして名前を指定し、詳細を確認して[Snapshot]*をクリックするだけで、オンデマンドSnapshotを作成できます。処理が完了すると、Snapshotの状態が「Healthy」に変わります。





10。PostgreSQLアプリケーション内のデータベースの削除

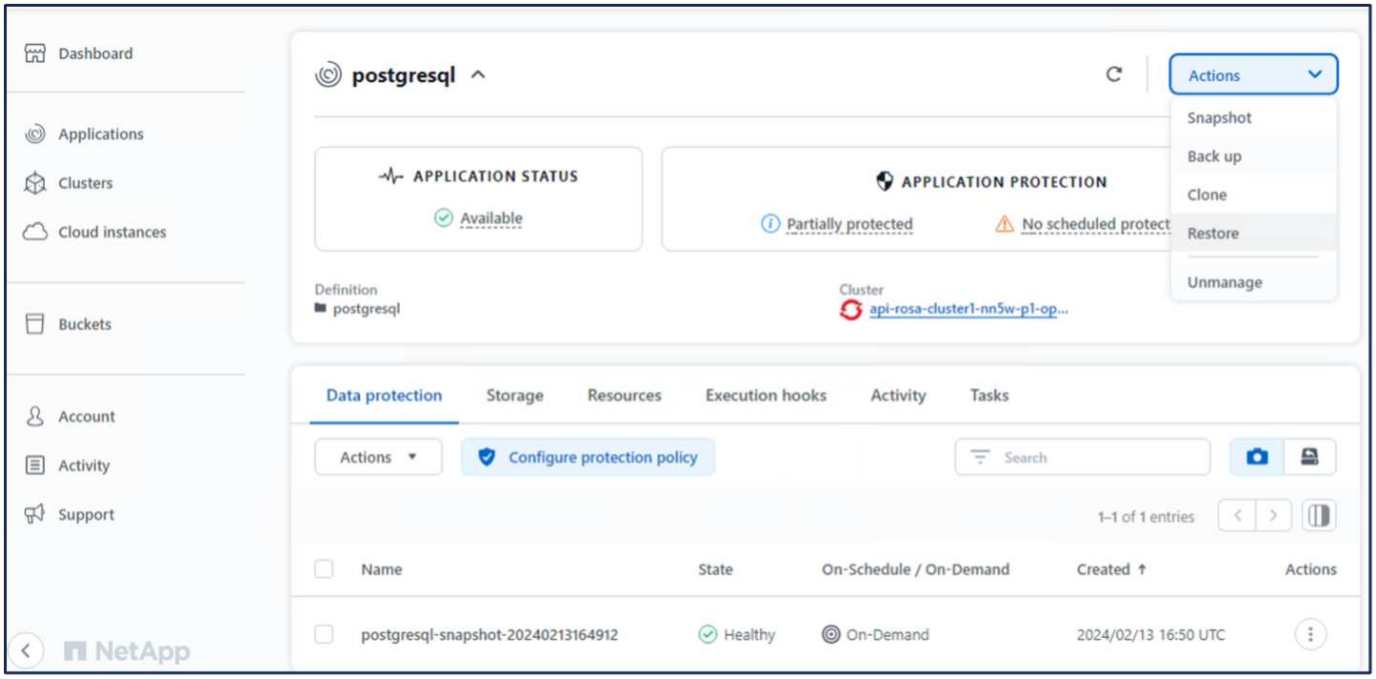
PostgreSQLに再度ログインし、利用可能なデータベースを一覧表示し、以前に作成したデータベースを削除して、データベースが削除されたことを確認します。

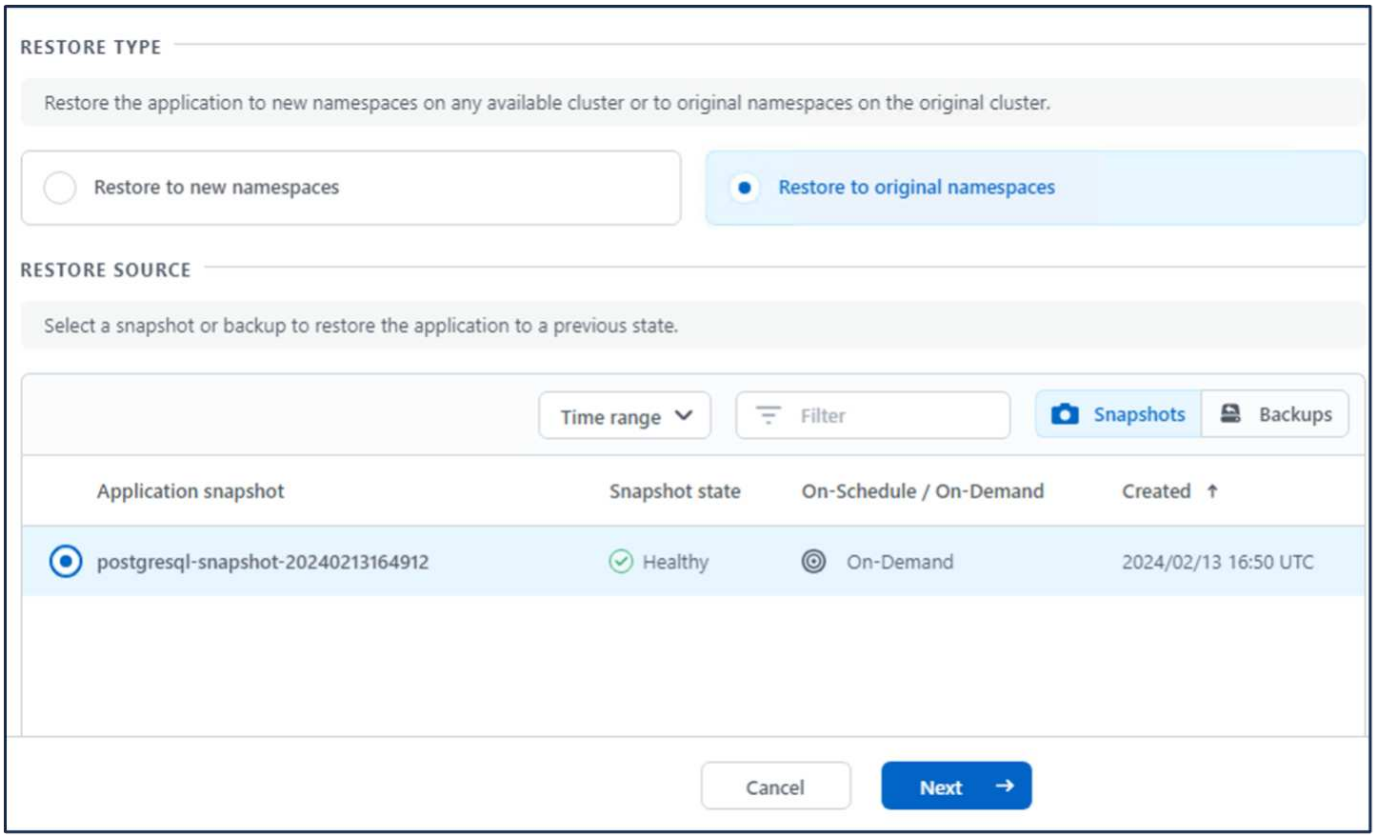
```
postgres=# \l
                                                   List of databases
   Name    |  Owner   | Encoding | Locale Provider |   Collate   |    Ctype    | ICU Locale | ICU Rules |   Access priv
-----------+----------+----------+-----------------+-------------+-------------+------------+-----------+---------------
 erp       | postgres | UTF8     | libc            | en_US.UTF-8 | en_US.UTF-8 |            |           |
 postgres  | postgres | UTF8     | libc            | en_US.UTF-8 | en_US.UTF-8 |            |           |
 template0 | postgres | UTF8     | libc            | en_US.UTF-8 | en_US.UTF-8 |            |           | =c/postgres
+
           |          |          |                 |             |             |            |           | postgres=CTc/
 template1 | postgres | UTF8     | libc            | en_US.UTF-8 | en_US.UTF-8 |            |           | =c/postgres
+
           |          |          |                 |             |             |            |           | postgres=CTc/
(4 rows)

postgres=# DROP DATABASE erp;
DROP DATABASE
postgres=# \l
                                                   List of databases
   Name    |  Owner   | Encoding | Locale Provider |   Collate   |    Ctype    | ICU Locale | ICU Rules |   Access priv
-----------+----------+----------+-----------------+-------------+-------------+------------+-----------+---------------
 postgres  | postgres | UTF8     | libc            | en_US.UTF-8 | en_US.UTF-8 |            |           |
 template0 | postgres | UTF8     | libc            | en_US.UTF-8 | en_US.UTF-8 |            |           | =c/postgres
+
           |          |          |                 |             |             |            |           | postgres=CTc/
 template1 | postgres | UTF8     | libc            | en_US.UTF-8 | en_US.UTF-8 |            |           | =c/postgres
+
           |          |          |                 |             |             |            |           | postgres=CTc/
(3 rows)
```

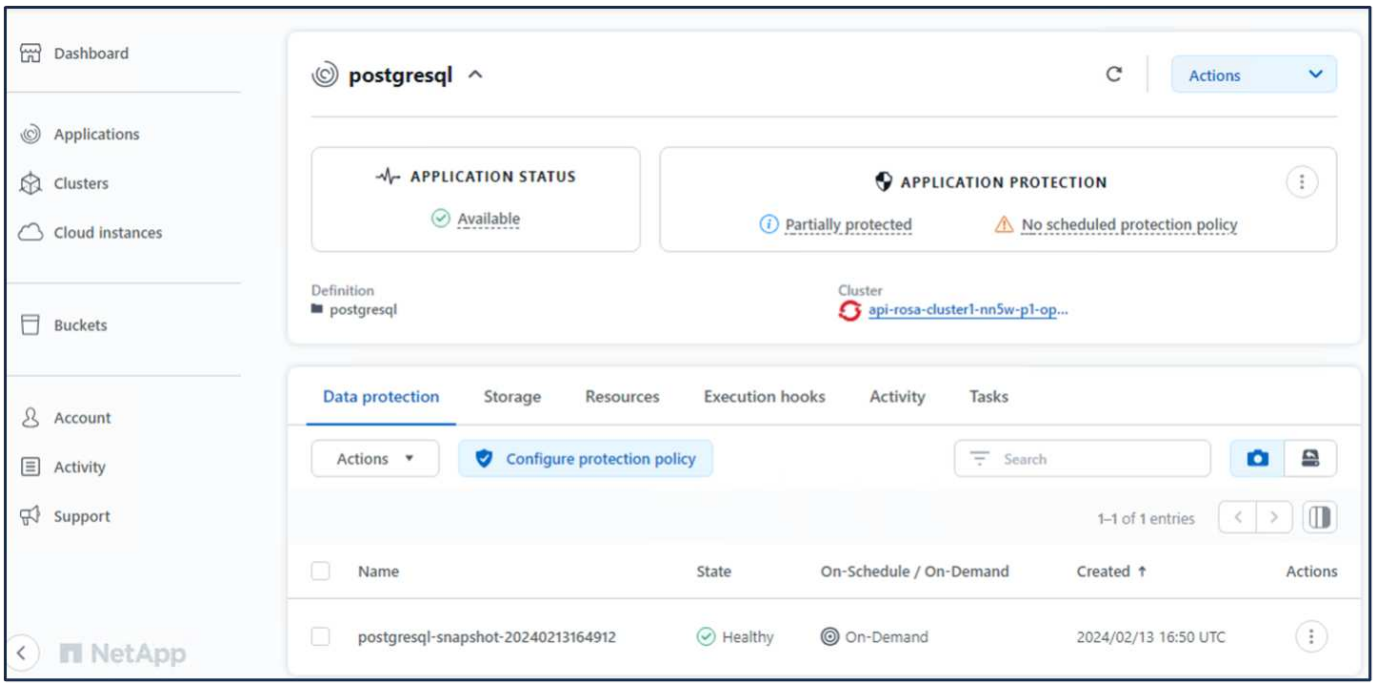
11.ACSを使用したスナップショットからのリストア

スナップショットからアプリケーションを復元するには、ACS UIランディングページに移動し、アプリケー

ションを選択して[Restore]を選択します。リストア元のスナップショットまたはバックアップを選択する必要があります。（通常は、設定したポリシーに基づいて複数のが作成されます）。次の2つの画面で適切な選択を行い、*[復元]*をクリックします。スナップショットからリストアされると、アプリケーションのステータスがRestoring（復元中）からAvailable（使用可能）に変わります。







12.アプリケーションがスナップショットから復元されたことを確認します

PostgreSQLクライアントにログインすると、以前に使用していたテーブルとレコードが表示されます。 これで終わりです。ボタンをクリックするだけで、アプリケーションは以前の状態に復元されます。Astra Controlを使用することで、お客様はそれを簡単に実現できます。

```
[ec2-user@ip-10-49-11-132 ~]$ kubectl run postgresql-client --rm --tty -i --restart='Never' --namespace postgresql --image docker.io/bitnami/postgresql:16.
2.0-debian-11-r1 --env="PGPASSWORD=$POSTGRES_PASSWORD" --command -- psql --host postgresql -U postgres -d postgres -p 5432
Warning: would violate PodSecurity "restricted:v1.24": allowPrivilegeEscalation != false (container "postgresql-client" must set securityContext.allowPrivi
legeEscalation=false), unrestricted capabilities (container "postgresql-client" must set securityContext.capabilities.drop=["ALL"]), runAsNonRoot != true (
pod or container "postgresql-client" must set securityContext.runAsNonRoot=true), seccompProfile (pod or container "postgresql-client" must set securityCon
text.seccompProfile.type to "RuntimeDefault" or "Localhost")
If you don't see a command prompt, try pressing enter.

postgres=# \l
                                                  List of databases
   Name    |  Owner   | Encoding | Locale Provider |   Collate   |    Ctype    | ICU Locale | ICU Rules |   Access privileges
-----------+----------+----------+-----------------+-------------+-------------+------------+-----------+-----------------------
 erp       | postgres | UTF8     | libc            | en_US.UTF-8 | en_US.UTF-8 |            |           |
 postgres  | postgres | UTF8     | libc            | en_US.UTF-8 | en_US.UTF-8 |            |           |
 template0 | postgres | UTF8     | libc            | en_US.UTF-8 | en_US.UTF-8 |            |           | =c/postgres          +
           |          |          |                 |             |             |            |           | postgres=CTc/postgres
 template1 | postgres | UTF8     | libc            | en_US.UTF-8 | en_US.UTF-8 |            |           | =c/postgres          +
           |          |          |                 |             |             |            |           | postgres=CTc/postgres
(4 rows)

postgres=# \c erp
You are now connected to database "erp" as user "postgres".
erp=# \dt
          List of relations
 Schema |  Name   | Type  |  Owner
--------+---------+-------+----------
 public | persons | table | postgres
(1 row)

erp=# SELECT * from PERSONS;
 id | firstname | lastname
----+-----------+----------
  1 | John      | Doe
(1 row)
```

データ移行

このページでは、永続的ストレージにFSx for NetApp ONTAP を使用したマネージドRed Hat OpenShiftクラスタでのコンテナワークロードのデータ移行オプションを示します。

データ移行

AWS上のRed Hat OpenShiftサービスとFSx for NetApp ONTAP （FSxN）は、AWSによるサービスポートフォリオに含まれています。FSxNは、単一のAZまたは複数のAZオプションで使用できます。複数のAZオプションを使用すると、アベイラビリティゾーンの障害からデータを保護できます。FSxNをAstra Tridentと統合することで、ROSAクラスタ上のアプリケーションに永続的ストレージを提供できます。

Helmチャートを使用した**FSxN**と**Trident**の統合

RosaクラスタとAmazon FSx for ONTAPの統合

コンテナアプリケーションの移行には、次の作業が含まれます。

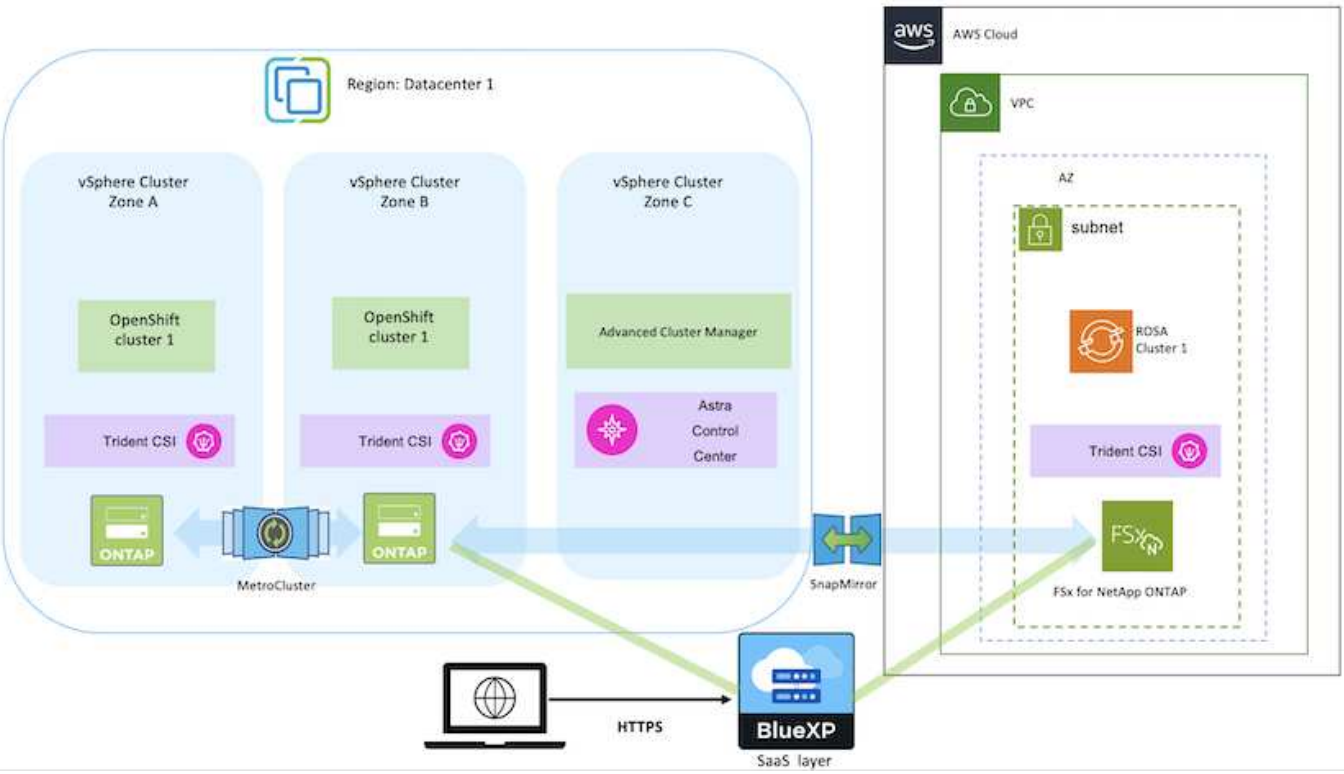
- 永続ボリューム：これはBlueXPを使用して実行できます。もう1つの方法は、Astra Control Centerを使用して、オンプレミスからクラウド環境へのコンテナアプリケーションの移行を処理する方法です。自動化も同じ目的で使用できます。
- アプリケーションメタデータ:これはOpenShift GitOps (Argo CD)を使用して実行できます。

永続的ストレージに**FSxN**を使用した**ROSA**クラスタ上のアプリケーションのフェイルオーバーとフェイルバック

次のビデオは、BlueXPとArgo CDを使用したアプリケーションのフェイルオーバーとフェイルバックのシナリオのデモです。

ROSAクラスタ上のアプリケーションのフェールオーバーとフェールバック

OpenShift Containerワークロード向けのデータ保護と移行用解決策



著作権に関する情報

Copyright © 2024 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権を有し、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権については、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。