Astra Control Center を使用

Astra Control Center

NetApp
November 21, 2023

目次

Astra Control Center を使用

アプリの管理を開始します

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[アプリケーション]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。

アプリケーション管理の要件

Astra Control には、次のアプリケーション管理要件があります。

- * 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：アプリケーションは、Astra Controlを使用して、単一クラスタ上の1つ以上の指定された名前空間内で定義できます。アプリケーションには、同じクラスタ内の複数のネームスペースにまたがるリソースを含めることができます。Astra Controlでは、複数のクラスタ間でアプリケーションを定義する機能はサポートされていません。
- ストレージクラス：ストレージクラスを明示的に設定し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は、元 々 指定されたストレージクラスが必要です。ストレージクラスを明示的に設定したアプリケーションを、同じストレージクラスを含まないクラスタにクローニングすると、失敗します。
- * Kubernetes リソース * ： Astra Control で収集されていない Kubernetes リソースを使用するアプリケーションには、アプリケーションのデータ管理機能がフル装備されていない可能性があります。Astra Control では、次の Kubernetes リソースが収集されます。

ClusterRole	ClusterRoleBinding	ConfigMap
CronJob	CustomResourceDefinition	CustomResource
DaemonSet	DeploymentConfig	HorizontalPodAutoscaler
Ingress	MutatingWebhook	NetworkPolicy
PersistentVolumeClaim	Pod	PodDisruptionBudget
PodTemplate	ReplicaSet	Role
RoleBinding	Route	Secret
Service	ServiceAccount	StatefulSet
ValidatingWebhook		

サポートされているアプリインストール方法

Astra Control は、次のアプリケーションインストール方法をサポートしています。

- * マニフェストファイル * ： Astra Control は、 kubectl を使用してマニフェストファイルからインストールされたアプリケーションをサポートします。例：

```
kubectl apply -f myapp.yaml
```

- * Helm 3 * ： Helm を使用してアプリケーションをインストールする場合、 Astra Control には Helm バージョン 3 が必要です。Helm 3 （または Helm 2 から Helm 3 にアップグレード）を使用してインストールされたアプリケーションの管理とクローニングが完全にサポートされています。Helm 2 でインストールされたアプリケーションの管理はサポートされていません。
- オペレータが導入したアプリケーション：Astra Controlは、ネームスペースを対象とした演算子を使用してインストールされたアプリケーションをサポートしています。一般的には、「パスバイリファレンス」アーキテクチャではなく「パスバイバリュー」アーキテクチャで設計されています。インストールする演算子とアプリケーションは、同じ名前空間を使用する必要があります。このような名前空間を使用するには、演算子の deployment.yaml ファイルを変更する必要があります。

  これらのパターンに続くいくつかのオペレータアプリを次に示します。

  - "Apache K8ssandra"

    > K8ssandra では、 In Place リストア処理がサポートされます。新しいネームスペースまたはクラスタにリストアするには、アプリケーションの元のインスタンスを停止する必要があります。これは、ピアグループ情報がインスタンス間通信を行わないようにするためです。アプリケーションのクローニングはサポートされていません。

  - "Jenkins CI"
  - "Percona XtraDB クラスタ"

  Astra Controlでは、「パスバイリファレンス」アーキテクチャ（CockroachDBオペレータなど）で設計されたオペレータをクローニングできない場合があります。クローニング処理では、クローニング処理の一環として独自の新しいシークレットが存在する場合でも、クローニングされたオペレータがソースオペレータから Kubernetes シークレットを参照しようとします。Astra Control がソースオペレータの Kubernetes シークレットを認識しないため、クローニング処理が失敗する場合があります。

クラスタにアプリをインストールします

お先にどうぞ "クラスタが追加されました" Astra Controlを使用すると、アプリケーションをインストールしたり、クラスタ上の既存のアプリケーションを管理したりできます。1つ以上の名前空間にスコープされているすべてのアプリケーションを管理できます。

アプリケーションを定義します

Astra Controlがクラスタ上のネームスペースを検出したら、管理するアプリケーションを定義できます。を選択できます 1つ以上のネームスペースにまたがるアプリケーションを管理します または ネームスペース全体を単一のアプリケーションとして管理。データ保護処理に必要な精度のレベルが重要になります。

Astra Controlを使用すると、階層の両方のレベル（ネームスペースとそのネームスペースまたはスパニングネームスペース内のアプリケーション）を別 々 に管理できますが、いずれか一方を選択することを推奨します。Astra Control で実行したアクションは、ネームスペースレベルとアプリケーションレベルの両方で同時に実行される場合、失敗する可能性があります。

たとえば、「Maria」に対して、毎週同じ頻度でバックアップを作成するように設定することもできますが、同じネームスペースにある「MariaDB」をバックアップする頻度を高く設定する必要があるとします。これらのニーズに基づいて、アプリケーションを個別に管理する必要があります。また、シングルネームスペースアプリケーションとして管理する必要はありません。

必要なもの

- KubernetesクラスタをAstra Controlに追加。
- クラスタにインストールされているアプリケーションが1つ以上あります。 サポートされているアプリケーションのインストール方法については、こちらをご覧ください。
- アクティブなポッドが1つ以上あります。
- Astra Controlに追加したKubernetesクラスタ上の既存のネームスペース。
- （オプション）すべてのにKubernetesラベルを付けます "サポートされるKubernetesリソース"。

ラベルは、 Kubernetes オブジェクトに割り当てて識別できるキーと値のペアです。ラベルを使用すると、 Kubernetes オブジェクトのソート、整理、検索が簡単になります。Kubernetes のラベルの詳細については、 "Kubernetes の公式ドキュメントを参照してください"。

このタスクについて

- 開始する前に、を理解しておく必要があります "標準ネームスペースとシステムネームスペースの管理"。
- Astra Controlのアプリケーションで複数の名前空間を使用する場合は、 "ネームスペースの制約を持つユーザロールを変更します" 複数の名前空間をサポートするAstra Control Centerバージョンにアップグレードした後。
- Astra Control API を使用してアプリケーションを管理する方法については、を参照してください "Astra の自動化と API に関する情報"。

アプリケーション管理オプション

- [アプリケーションとして管理するリソースを定義します]
- [アプリケーションとして管理するネームスペースを定義します]

アプリケーションとして管理するリソースを定義します

を指定できます "アプリケーションを構成するKubernetesリソース" Astra Controlで管理したい。アプリケーションを定義すると、Kubernetesクラスタの要素を1つのアプリケーションにグループ化できます。このKubernetesリソースの集まりは、ネームスペースとラベル選択条件によって分類されます。

アプリケーションを定義することで、クローン、スナップショット、バックアップなどのAstra Control操作に含めるものをより細かく制御できます。

アプリケーションを定義するときは、保護ポリシーを使用して複数のアプリケーションにKubernetesリソースを含めないようにしてください。Kubernetesリソースの保護ポリシーが重複していると、原因 のデータが競合する可能性があります 詳細については、例を参照してください。

リソースを共有するアプリケーションでIn Placeリストア処理を実行すると、予期しない結果が生じる可能性があります。アプリケーション間で共有されているリソースは、いずれかのアプリケーションでインプレースリストアが実行されると置き換えられます。たとえば、次のようなシナリオでは、NetApp SnapMirrorレプリケーションを使用する場合に、問題となる状況を想定していません。

1. アプリケーションを定義します `app1` ネームスペースを使用する `ns1`。
2. のレプリケーション関係を設定します `app1`。
3. アプリケーションを定義します `app2` （同じクラスタ上）ネームスペースを使用します `ns1` および `ns2`。
4. のレプリケーション関係を設定します `app2`。
5. のレプリケーションを反転した `app2`。これにより、が起動します `app1` 非アクティブ化するソースクラスタ上のアプリケーション。

**** アプリケーションネームスペースにクラスタ対象リソースを追加する方法については、**** を参照してください。

ネームスペースリソースに関連付けられているクラスタリソースを、自動的に含まれるアストラコントロールに加えてインポートできます。特定のグループ、種類、バージョンのリソースを含むルールを追加し、必要に応じてラベルを付けることができます。この処理は、Astra Controlに自動的に含まれないリソースがある場合などに実行します。

Astra Controlに自動的に含まれる、クラスタを対象としたリソースを除外することはできません。

以下を追加できます `apiVersions` （APIバージョンと組み合わせたグループ）。

リソースの種類	**1**回あたりのバージョン（グループ**+**バージョン）
`ClusterRole`	rbac.authorization.k8s.io/v1
`ClusterRoleBinding`	rbac.authorization.k8s.io/v1
`CustomResource`	apiextensions.k8s.io/v1、apiextensions.k8s.io/v1beta1
`CustomResourceDefinition`	apiextensions.k8s.io/v1、apiextensions.k8s.io/v1beta1
`MutatingWebhookConfiguration`	admissionregistration.k8s.io/v1
`ValidatingWebhookConfiguration`	admissionregistration.k8s.io/v1

手順

1. [アプリケーション（Applications）]ページで、[定義（ Define）]を選択します
2. [アプリケーションの定義（* Define application）]ウィンドウで、アプリケーション名を入力します。
3. **[Cluster]**ドロップダウン・リストから、アプリケーションが実行されているクラスタを選択します。
4. 「名前空間」ドロップダウンリストからアプリケーションの名前空間を選択します。

アプリケーションは、Astra Controlを使用して、単一クラスタ上の1つ以上の指定された名前空間内で定義できます。アプリケーションには、同じクラスタ内の複数のネームスペースにまたがるリソースを含めることができます。Astra Controlでは、複数のクラスタ間でアプリケーションを定義する機能はサポートされていません。

5. （オプション）各ネームスペースにKubernetesリソースのラベルを入力します。ラベルまたはラベルの選択基準（クエリー）を1つ指定できます。

Kubernetes のラベルの詳細については、 "Kubernetes の公式ドキュメントを参照してください"。

6. （オプション）「名前空間の追加」を選択し、ドロップダウンリストから名前空間を選択して、アプリケーションの名前空間を追加します。
7. （オプション）追加するネームスペースのラベルまたはラベルの選択基準を1つ入力します。
8. （オプション）Astra Controlに自動的に含まれるリソースに加えて、クラスタを対象としたリソースを含めるには、*クラスタを対象とした追加のリソースを含める*をチェックし、次の手順を実行します。
   a. 「含めるルールを追加」を選択します。
   b. グループ：ドロップダウンリストから、リソースのAPIグループを選択します。
   c. *kind *:ドロップダウンリストから'オブジェクトスキーマの名前を選択します
   d. バージョン：APIのバージョンを入力します。
   e. ラベルセレクタ：必要に応じて、ルールに追加するラベルを指定します。このラベルは、このラベルに一致するリソースのみを取得するために使用します。ラベルを指定しないと、Astra Controlは、そのクラスタに指定されている種類のリソースのすべてのインスタンスを収集します。
   f. エントリに基づいて作成されたルールを確認します。
   g. 「 * 追加」を選択します。

クラスタを対象としたリソースルールは必要な数だけ作成できます。[アプリケーションの定義の概要]にルールが表示されます。

9. [* 定義（ Define ）] を選択します
10. [定義（Define *）]を選択した後、必要に応じて他のアプリケーションについても同じ手順を繰り返します。

アプリケーションの定義が完了すると、アプリケーションがに表示されます `Healthy` 「アプリケーション」ページのアプリケーションのリストに表示されます。クローンを作成し、バックアップとスナップショットを作成できるようになりました。

追加したアプリケーションの保護列に警告アイコンが表示されている場合は、バックアップされておらず、まだバックアップのスケジュールが設定されていないことを示しています。

特定のアプリケーションの詳細を表示するには、アプリケーション名を選択します。

このアプリに追加されたリソースを表示するには、*リソース*タブを選択します。Resource列でリソース名のあとの番号を選択するか、Searchでリソース名を入力して、追加のクラスタを対象としたリソースを確認します。

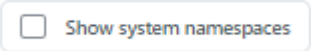
アプリケーションとして管理するネームスペースを定義します

ネームスペースのリソースをアプリケーションとして定義することで、ネームスペース内のすべてのKubernetesリソースをAstra Control管理に追加できます。特定の名前空間内のすべてのリソースを同じような方法で、共通の間隔で管理および保護する場合は、アプリケーションを個別に定義することをお勧めします。

手順

1. クラスタページで、クラスタを選択します。
2. [名前空間]タブを選択します。
3. 管理するアプリケーションリソースを含む名前空間のアクションメニューを選択し、*アプリケーションとして定義*を選択します。

> 複数のアプリケーションを定義する場合は、名前空間リストから選択し、左上隅の*アクション*ボタンを選択して、*アプリケーションとして定義*を選択します。これにより、個 々のネームスペースに複数のアプリケーションが定義されます。マルチネームスペースアプリケーションについては、を参照してください [アプリケーションとして管理するリソースを定義します]。

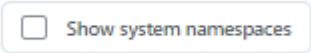
> [システムネームスペースを表示（Show system Namespaces）]チェックボックスを選択して、アプリケーション管理で通常はデフォルトで使用されないシステムネームスペースを表示します。 
>
> "詳細はこちら"。

このプロセスが完了すると、ネームスペースに関連付けられているアプリケーションがに表示されます `Associated applications` 列（Column）：

システムネームスペースについて教えてください。

Astra Controlは、Kubernetesクラスタ上のシステムネームスペースも検出します。これらのシステムネームスペースはデフォルトでは表示されません。システムアプリケーションリソースのバックアップが必要になることがまれです。

選択したクラスタの[ネームスペース]タブからシステムネームスペースを表示するには、[システムネームスペースを表示]チェックボックスをオンにします。



> Astra Control 自体は標準のアプリケーションではなく、「システムアプリケーション」です。Astra Control 自体は管理しないでください。Astra Control 自体は、管理用にデフォルトでは表示されません。

例：リリースごとに保護ポリシーを分ける

この例では、DevOpsチームが「カナリアリリースの導入を管理しています。チームのクラスタにはnginxを実行するポッドが3つあります。そのうちの 2 つのポッドは、安定版リリース専用です。3 番目のポッドはカナリアリリース用です。

DevOpsチームのKubernetes管理者がラベルを追加します `deployment=stable` を使用して、安定版リリースポッドに移動しますチームがラベルを追加します `deployment=canary` カナリアリリースポッドに移動します。

チームの安定版リリースには、 1 時間ごとの Snapshot と日次バックアップの要件が含まれています。カナリアリリースはより一時的なリリースなので、ラベル付きのものは何でも短時間で、よりアグレッシブな保護ポリシーを作成したいと考えています `deployment=canary`。

データの競合を回避するために、管理者は「カナリア」リリース用と「stable」リリース用の2つのアプリケーションを作成します。これにより、 Kubernetes オブジェクトの 2 つのグループに対して、バックアップ、Snapshot 、およびクローニングの処理が分離されます。

詳細については、こちらをご覧ください

- "Astra Control API を使用"
- "アプリの管理を解除します"

アプリを保護します

保護の概要

Astra Control Center を使用して、アプリケーションのバックアップ、クローン、スナップショット、および保護ポリシーを作成できます。アプリケーションをバックアップすることで、サービスや関連データを可能な限り利用できるようになります。災害時にバックアップからリストアすることで、アプリケーションと関連データを最小限の中断で完全にリカバリできます。バックアップ、クローン、 Snapshot を使用すると、ランサムウェアや偶発的なデータ損失、環境障害などの一般的な脅威からデータを保護できます。 "Astra Control Center で使用可能なデータ保護の種類と、それらを使用するタイミングについて説明します"。

また、ディザスタリカバリに備えてアプリケーションをリモートクラスタにレプリケートすることもできます。

アプリケーション保護のワークフロー

次のワークフロー例を使用して、アプリケーションの保護を開始できます。

[1つ] すべてのアプリケーションを保護

アプリケーションをすぐに保護するには、次の手順を実行します。 "すべてのアプリケーションの手動バックアップを作成する"。

[2 つ] 各アプリケーションの保護ポリシーを設定します

将来のバックアップとスナップショットを自動化するには、 "各アプリケーションの保護ポリシーを設定します"。たとえば、週単位のバックアップと日単位の Snapshot をそれぞれ 1 カ月ずつ保持して開始できます。手動バックアップやスナップショットよりも、保護ポリシーを使用してバックアップとスナップショットを自動化することを強く推奨します。

[3つ] 保護ポリシーを調整します

アプリとその使用パターンが変化したら、必要に応じて保護ポリシーを調整して、最適な保護を実現します。

[4.] アプリケーションをリモートクラスタにレプリケートします

"アプリケーションをレプリケートします" NetApp SnapMirrorテクノロジを使用してリモートクラスタにバックアップする場合Astra Controlは、Snapshotをリモートクラスタにレプリケートし、非同期のディザスタリカバリ機能を提供します。

[5 つ**]** 災害が発生した場合は、最新のバックアップまたはレプリケーションを使用してアプリケーションをリモートシステムにリストアします

データ損失が発生した場合は、を使用してリカバリできます "最新のバックアップをリストアしています" まず、各アプリケーションについて説明します。その後、最新の Snapshot をリストアできます（使用可能な場合）。または、リモートシステムへのレプリケーションを使用することもできます。

Snapshot とバックアップでアプリケーションを保護

自動保護ポリシーまたはアドホックベースを使用して、スナップショットやバックアップを作成することで、すべてのアプリケーションを保護します。Astra Control Center UI またはを使用できます "Astra Control API" アプリを保護します。

このタスクについて

- * Helmでアプリケーションを展開*：Helmを使用してアプリケーションを展開する場合、Astra Control CenterにはHelmバージョン3が必要です。Helm 3 （または Helm 2 から Helm 3 にアップグレード）を使用して展開されたアプリケーションの管理とクローニングが完全にサポートされています。Helm 2 で展開されたアプリケーションはサポートされていません。
- （**OpenShift**クラスタのみ）ポリシーの追加：OpenShiftクラスタでアプリをホストするためのプロジェクトを作成すると、プロジェクト（またはKubernetesネームスペース）にSecurityContext UIDが割り当てられます。Astra Control Center でアプリケーションを保護し、 OpenShift でそのアプリケーションを別のクラスタまたはプロジェクトに移動できるようにするには、アプリケーションを任意の UID として実行できるようにポリシーを追加する必要があります。たとえば、次の OpenShift CLI コマンドは、 WordPress アプリケーションに適切なポリシーを付与します。

```
oc new-project wordpress
oc adm policy add-scc-to-group anyuid system:serviceaccounts:wordpress
oc adm policy add-scc-to-user privileged -z default -n wordpress
```

アプリケーションデータの保護に関連する次のタスクを実行できます。

- [保護ポリシーを設定します]
- Snapshot を作成します
- [バックアップを作成します]
- Snapshot とバックアップを表示します
- Snapshot を削除します
- [バックアップをキャンセルします]
- [バックアップを削除します]

保護ポリシーを設定します

保護ポリシーは、定義され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します。Snapshot とバックアップを毎時、日次、週次、および月単位で作成し、保持するコピーの数を指定できます。

1 時間に 1 回以上の頻度でバックアップや Snapshot を実行する必要がある場合は、次の方法があります "Astra Control REST API を使用して、スナップショットとバックアップを作成"。

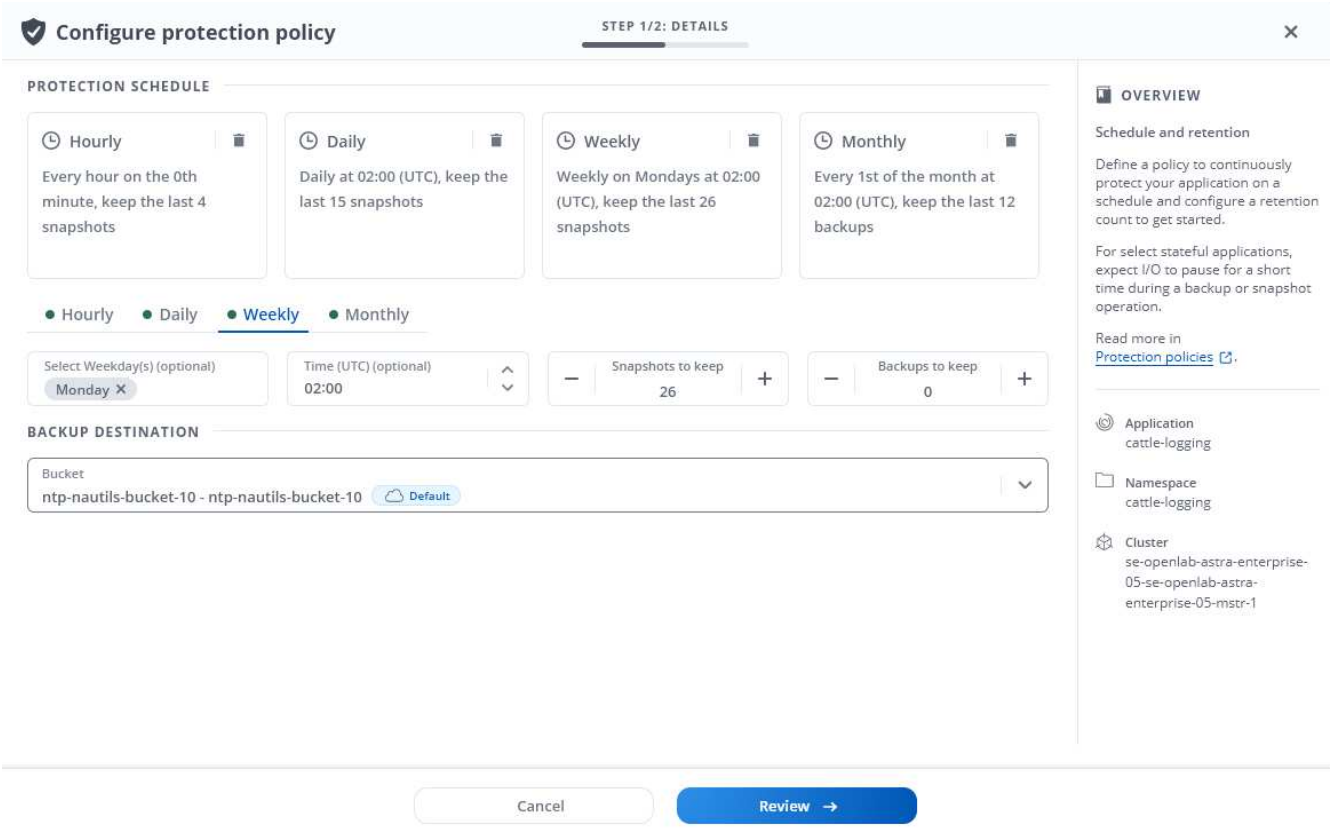
手順

1. 「 * アプリケーション」を選択し、アプリケーションの名前を選択します。
2. [* データ保護 *] を選択します。
3. ［ 保護ポリシーの設定 ］ を選択します。
4. 毎時、日次、週次、および月単位で保持する Snapshot とバックアップの数を選択して、保護スケジュールを定義します。

   スケジュールは、毎時、毎日、毎週、および毎月の各スケジュールで同時に定義できます。保持レベルを設定するまで、スケジュールはアクティブになりません。

   バックアップの保持レベルを設定する際に、バックアップを格納するバケットを選択できます。

   次の例では、 Snapshot とバックアップの保護スケジュールとして、毎時、毎日、毎週、毎月の 4 つを設定します。



5. [* Review （レビュー）] を選択します
6. ［ * 保護ポリシーの設定 * ］ を選択します

結果

Astra Control は、定義したスケジュールと保持ポリシーを使用して、スナップショットとバックアップを作成し、保持することによって、データ保護ポリシーを実装します。

Snapshot を作成します

オンデマンド Snapshot はいつでも作成できます。

手順

1. 「 * アプリケーション * 」を選択します。
2. 目的のアプリケーションの * アクション * 列のオプションメニューから、 * スナップショット * を選択します。
3. スナップショットの名前をカスタマイズし、*次へ*を選択します。
4. Snapshot の概要を確認し、「 * Snapshot * 」を選択します。

結果

スナップショットプロセスが開始されます。スナップショットは'ステータスが* Healthy である場合に成功します（ Data protection **>** Snapshots ページの State *列）

バックアップを作成します

アプリケーションはいつでもバックアップできます。

> Astra Control Center の S3 バケットは、使用可能容量を報告しません。Astra Control Center で管理されているアプリケーションのバックアップまたはクローニングを行う前に、 ONTAP または StorageGRID 管理システムでバケット情報を確認します。

手順

1. 「 * アプリケーション * 」を選択します。
2. 目的のアプリケーションの*アクション*列のオプションメニューから、*バックアップ*を選択します。
3. バックアップ名をカスタマイズする。
4. 既存のスナップショットからアプリケーションをバックアップするかどうかを選択します。このオプションを選択すると、既存の Snapshot のリストから選択できます。
5. ストレージバケットのリストから、バックアップのデスティネーションバケットを選択します。
6. 「 * 次へ * 」を選択します。
7. バックアップの概要を確認し、「バックアップ」を選択します。

結果

Astra Control ：アプリケーションのバックアップを作成

> ネットワークに障害が発生している場合や、処理速度が異常に遅い場合は、バックアップ処理がタイムアウトする可能性があります。その結果、バックアップは失敗します。

実行中のバックアップをキャンセルする必要がある場合は、の手順に従ってください [バックアップをキャンセルします]。バックアップを削除するには、完了するまで待ってから、の手順を実行します [バックアップを削除します]。

データ保護処理（クローン、バックアップ、リストア）が完了して永続ボリュームのサイズを変更したあと、新しいボリュームのサイズが UI に表示されるまでに最大 20 分かかります。データ保護処理にかかる時間は数分です。また、ストレージバックエンドの管理ソフトウェアを使用してボリュームサイズの変更を確認できます。

Snapshot とバックアップを表示します

アプリケーションのスナップショットとバックアップは、 [データ保護（ Data Protection ）] タブで表示できます。

手順

1. 「 * アプリケーション」を選択し、アプリケーションの名前を選択します。
2. [* データ保護 *] を選択します。

   デフォルトでは、 Snapshot が表示されます。

3. バックアップのリストを表示するには、「 * Backups * 」を選択します。

Snapshot を削除します

不要になったスケジュール済みまたはオンデマンドの Snapshot を削除します。

現在レプリケート中のSnapshotは削除できません。

手順

1. 「 * アプリケーション」を選択し、管理アプリの名前を選択します。
2. [* データ保護 *] を選択します。
3. 目的のスナップショットの * アクション * 列のオプションメニューから、 * スナップショットの削除 * を選択します。
4. 削除を確認するために「 delete 」と入力し、「 * はい、 Snapshot を削除します * 」を選択します。

結果

Astra Control がスナップショットを削除します。

バックアップをキャンセルします

実行中のバックアップをキャンセルすることができます。

バックアップをキャンセルするには、バックアップが実行されている必要があります `Running` 状態。にあるバックアップはキャンセルできません `Pending` 状態。

手順

1. 「 * アプリケーション」を選択し、アプリケーションの名前を選択します。
2. [* データ保護 *] を選択します。
3. 「 * Backups * 」を選択します。
4. 目的のバックアップの[アクション（* Actions ）]列の[オプション（**Options**）]メニューから、[*キャンセル（* Cancel *）]を選択します。
5. 処理を確認するために「CANCEL」と入力し、「* Yes、cancel backup *」を選択します。

バックアップを削除します

不要になったスケジュール済みまたはオンデマンドのバックアップを削除します。

> 実行中のバックアップをキャンセルする必要がある場合は、の手順に従ってください [バックアップをキャンセルします]。バックアップを削除するには、完了するまで待ってから、次の手順を実行します。

手順

1. 「 * アプリケーション」を選択し、アプリケーションの名前を選択します。
2. [* データ保護 *] を選択します。
3. 「 * Backups * 」を選択します。
4. 目的のバックアップの [* アクション *] 列の [オプション] メニューから、 [* バックアップの削除 *] を選択します。
5. 削除を確認するために「 delete 」と入力し、「 * はい、バックアップを削除 * 」を選択します。

結果

Astra Control がバックアップを削除する。

アプリケーションのリストア

Astra Control を使用すると、スナップショットまたはバックアップからアプリケーションをリストアできます。同じクラスタにアプリケーションをリストアする場合、既存の Snapshot からのリストアは高速です。Astra Control UI またはを使用できます "Astra Control API" アプリを復元するには、

> NetApp ONTAP ストレージを使用するアプリケーションのIn Placeリストアを実行すると、リストアしたアプリケーションで使用するスペースは2倍になります。In Placeリストアを実行したあとに、リストアしたアプリケーションから不要なSnapshotを削除して、ストレージスペースを解放します。

このタスクについて

- 最初にアプリを保護する:復元する前に、アプリケーションのスナップショットを取るか、バックアップすることを強くお勧めします。リストアに失敗した場合に、Snapshotまたはバックアップからクローニングできます。
- デスティネーションボリュームを確認：別のクラスタにリストアする場合は、同じ永続的ボリュームアクセスモード（ReadWriteManyなど）をクラスタが使用していることを確認してください。デスティネーションの永続ボリュームアクセスモードが異なると、リストア処理は失敗します。

- （**OpenShift**クラスタのみ）ポリシーの追加：OpenShiftクラスタでアプリをホストするためのプロジェクトを作成すると、プロジェクト（またはKubernetesネームスペース）にSecurityContext UIDが割り当てられます。Astra Control Center でアプリケーションを保護し、 OpenShift でそのアプリケーションを別のクラスタまたはプロジェクトに移動できるようにするには、アプリケーションを任意の UID として実行できるようにポリシーを追加する必要があります。たとえば、次の OpenShift CLI コマンドは、 WordPress アプリケーションに適切なポリシーを付与します。

```
oc new-project wordpress
oc adm policy add-scc-to-group anyuid system:serviceaccounts:wordpress
oc adm policy add-scc-to-user privileged -z default -n wordpress
```

- * Helmはアプリケーションを展開しました*：Helm 3を使用して展開されたアプリケーションのクローン作成（またはHelm 2からHelm 3にアップグレード）は完全にサポートされています。Helm 2 で展開されたアプリケーションはサポートされていません。

手順

1. 「 * アプリケーション」を選択し、アプリケーションの名前を選択します。
2. 「 * データ保護 * 」を選択します。
3. Snapshot からリストアする場合は、 * Snapshots * アイコンを選択したままにします。それ以外の場合は、「 * Backups * 」アイコンを選択してバックアップからリストアします。
4. リストア元のスナップショットまたはバックアップの [* アクション *] 列の [オプション] メニューから、[* アプリケーションのリストア *] を選択します。
5. リストアタイプを選択します。
   - 元のネームスペースにリストア：この手順 を使用して、アプリケーションを元のクラスタにインプレースでリストアします。

     > リソースを共有するアプリケーションでIn Placeリストア処理を実行すると、予期しない結果が生じる可能性があります。アプリケーション間で共有されているリソースは、いずれかのアプリケーションでインプレースリストアが実行されると置き換えられます。たとえば、次のようなシナリオでは、NetApp SnapMirrorレプリケーションを使用する場合に、問題となる状況を想定していません。
     >
     > i. アプリケーションを定義します `app1` ネームスペースを使用する `ns1`。
     > ii. のレプリケーション関係を設定します `app1`。
     > iii. アプリケーションを定義します `app2` （同じクラスタ上）ネームスペースを使用します `ns1` および `ns2`。
     > iv. のレプリケーション関係を設定します `app2`。
     > v. のレプリケーションを反転した `app2`。これにより、が起動します `app1` 非アクティブ化するソースクラスタ上のアプリケーション。

     i. アプリをインプレースで復元するために使用するスナップショットを選択します。これにより、アプリは以前のバージョンのに戻ります。
     ii. 「 * 次へ * 」を選択します。

> 以前に削除したネームスペースにリストアすると、同じ名前の新しいネームスペースがリストアプロセスで作成されます。以前に削除したネームスペースでアプリケーションを管理する権限を持つユーザは、新しく作成したネームスペースに手動で権限を復元する必要があります。

iii. リストア操作の詳細を確認し、「restore」と入力して、「* Restore *」を選択します。

- 新しい名前空間に復元：この手順 を使用して、アプリを別のクラスタまたはソースとは異なる名前空間で別のクラスタに復元します。

  i. リストアするアプリケーションのデスティネーションクラスタを選択します。

  ii. アプリケーションに関連付けられている各ソースネームスペースのデスティネーションネームスペースを入力します。

  > Astra Controlは、このリストアオプションの一部として新しいデスティネーションネームスペースを作成します。指定するデスティネーションネームスペースがデスティネーションクラスタに存在していないことを確認してください。

  iii. 「 * 次へ * 」を選択します。

  iv. アプリケーションのリストアに使用するスナップショットを選択します。

  v. 「 * 次へ * 」を選択します。

  vi. リストア処理の詳細を確認し、* Restore *を選択します。

結果

Astra Control は、指定した情報に基づいてアプリケーションを復元します。アプリケーションをインプレースでリストアした場合、既存の永続ボリュームのコンテンツが、リストアしたアプリケーションの永続ボリュームのコンテンツに置き換えられます。

> データ保護処理（クローン、バックアップ、またはリストア）が完了して永続ボリュームのサイズを変更したあと、Web UIに新しいボリュームサイズが表示されるまでに最大20分かかります。データ保護処理にかかる時間は数分です。また、ストレージバックエンドの管理ソフトウェアを使用してボリュームサイズの変更を確認できます。

> ネームスペースの名前/ IDまたはネームスペースのラベルでネームスペースの制約を受けているメンバーユーザは、同じクラスタの新しいネームスペース、または組織のアカウントに含まれる他のクラスタにアプリケーションをクローニングまたはリストアできます。ただし、同じユーザが、クローニングまたはリストアされたアプリケーションに新しいネームスペースからアクセスすることはできません。クローンまたはリストア処理によって新しいネームスペースが作成されると、アカウントの管理者 / 所有者はメンバーユーザアカウントを編集し、該当するユーザに新しいネームスペースへのアクセスを許可するロールの制限を更新できます。

SnapMirrorテクノロジを使用してアプリケーションをリモートシステムにレプリケート

Astra Controlを使用すると、NetApp SnapMirrorテクノロジの非同期レプリケーション機能を使用して、RPO（目標復旧時点）とRTO（目標復旧時間）の低いアプリケーションのビジネス継続性を構築できます。設定が完了すると、アプリケーションはデータやアプリケーションの変更をクラスタ間でレプリケートできるようになります。

バックアップ/リストアとレプリケーションの比較については、を参照してください "データ保護の概念"。

アプリケーションは、オンプレミスのみ、ハイブリッド、マルチクラウドなど、さまざまなシナリオでレプリケートできます。

- オンプレミスサイトAからオンプレミスサイトBへ
- Cloud Volumes ONTAP を使用してオンプレミスからクラウドに移行できます
- Cloud Volumes ONTAP を使用したクラウドをオンプレミスに移行
- Cloud Volumes ONTAP を使用したクラウドからクラウドへ（同じクラウドプロバイダ内の異なるリージョン間または異なるクラウドプロバイダ間）

Astra Controlを使用すれば、オンプレミスのクラスタからクラウドへ（Cloud Volumes ONTAP を使用）、またはクラウド間（Cloud Volumes ONTAP からCloud Volumes ONTAP へ）にアプリケーションをレプリケートできます。

> （別のクラスタまたはサイトで実行されている）別のアプリケーションを逆方向に同時にレプリケートできます。たとえば、アプリケーションA、B、Cはデータセンター1からデータセンター2にレプリケートでき、アプリケーションX、Y、Zはデータセンター2からデータセンター1にレプリケートできます。

Astra Controlを使用すると、アプリケーションのレプリケーションに関連する次のタスクを実行できます。

- [レプリケーション関係を設定]
- [デスティネーションクラスタでレプリケートされたアプリケーションをオンラインにする（フェイルオーバー）]
- [フェイルオーバーしたレプリケーションを再同期します]
- [アプリケーションのレプリケーションを反転する]
- [アプリケーションを元のソースクラスタにフェイルバックします]
- [アプリケーションレプリケーション関係を削除します]

レプリケーションの前提条件

Astra Controlアプリケーションのレプリケーションを開始するには、次の前提条件を満たしている必要があります。

- シームレスな災害復旧を実現するために、第3の障害ドメインまたはセカンダリサイトにAstra Control Centerを導入することをお勧めします。
- アプリケーションのホストKubernetesクラスタとデスティネーションKubernetesクラスタは、理想的には異なる障害ドメインまたはサイトで、ONTAP クラスタと一緒に管理する必要があります。
- ONTAP クラスタとホストSVMをペアリングする必要があります。を参照してください "クラスタと SVM のピアリングの概要"。
- ペアリングしたリモートSVMがデスティネーションクラスタからAstra Tridentに接続されている必要があります。
- ソースとデスティネーションの両方のONTAP クラスタにAstra Tridentバージョン22.07以降が存在する必要があります。

- ソースとデスティネーションの両方のONTAP クラスタで、データ保護バンドルを使用したONTAP SnapMirror非同期ライセンスが有効になっている必要があります。を参照してください "ONTAP のSnapMirrorライセンスの概要"。
- ONTAP ストレージバックエンドをAstra Control Centerに追加する場合は、「admin」ロールでユーザクレデンシャルを適用します。このロールにはアクセス方法があります `http` および `ontapi` ONTAP ソースとデスティネーションの両方のクラスタで有効にします。を参照してください "ONTAP ドキュメントの「ユーザーアカウントの管理」を参照してください" を参照してください。
- ソースとデスティネーションの両方のKubernetesクラスタとONTAP クラスタをAstra Controlで管理する必要があります。

> （別のクラスタまたはサイトで実行されている）別のアプリケーションを逆方向に同時にレプリケートできます。たとえば、アプリケーションA、B、Cはデータセンター1からデータセンター2にレプリケートでき、アプリケーションX、Y、Zはデータセンター2からデータセンター1にレプリケートできます。

- * Astra Trident / ONTAP 構成 * ： Astra Control Center では、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がレプリケーション用に提供する次のONTAP ドライバをサポートしています。
  - ONTAP - NAS
  - ONTAP-NAS-flexgroup
  - ONTAP - SAN

方法をご確認ください "SnapMirrorテクノロジを使用してアプリケーションをリモートシステムにレプリケート"。

レプリケーション関係を設定

レプリケーション関係を設定するには、レプリケーションポリシーを構成する次の作業を行います。

- Astra ControlでのアプリケーションSnapshotの作成頻度の選択（アプリケーションのKubernetesリソースと、アプリケーションの各ボリュームのボリュームSnapshotを含む）
- レプリケーションスケジュールの選択（Kubernetesリソースと永続ボリュームデータを含む）
- Snapshotを作成する時刻を設定します

手順

1. Astra Controlの左ナビゲーションから、「アプリケーション」を選択します。
2. ［アプリケーション］ページで、［データ保護］>［レプリケーション］タブを選択します。
3. ［データ保護］＞［レプリケーション］タブで、［レプリケーションポリシーの設定］を選択します。または、[アプリケーション保護]ボックスから[アクション]オプションを選択し、[レプリケーションポリシーの構成]を選択します。
4. 次の情報を入力または選択します。
   - デスティネーションクラスタ：ソースとは異なるデスティネーションクラスタを入力してください。
   - デスティネーションストレージクラス：デスティネーションONTAP クラスタでペアリングされているSVMを使用するストレージクラスを選択または入力します。
   - レプリケーションタイプ：現在使用できるレプリケーションタイプは「非同期」のみです。

- デスティネーションネームスペース：デスティネーションクラスタの新規または既存のデスティネーションネームスペースを入力します。
- （任意）[Add namespace]を選択し、ドロップダウンリストからネームスペースを選択して、ネームスペースを追加します。
- レプリケーション頻度：Snapshotを作成してデスティネーションにレプリケートする頻度を指定します。
- オフセット：Astra Controlでスナップショットを作成する時間の上部から分数を設定します。オフセットを使用すると、他のスケジュールされた処理と競合しないようにすることができます。たとえば、10：02から5分ごとにSnapshotを作成する場合は、オフセットの分として「02」を入力します。結果は、10：02、10：07、10：12などになります

5. 「次へ」を選択し、概要を確認して、「保存」を選択します。

> 最初に、最初のスケジュールが実行される前にステータスに「app_mirror」と表示されます。

Astra Control：レプリケーションに使用するアプリケーションSnapshotを作成

6. アプリケーションのスナップショットステータスを表示するには、アプリケーション>*スナップショット*タブを選択します。

   Snapshot名には「replication-schedule -<string>」の形式を使用します。Astra Controlは、レプリケーションに使用された最後のSnapshotを保持古いレプリケーションSnapshotは、レプリケーションが正常に完了すると削除されます。

結果

これにより、レプリケーション関係が作成されます。

Astra Controlは、関係を確立した結果として次のアクションを実行します。

- デスティネーションにネームスペースを作成します（存在しない場合）。
- 送信元アプリケーションのPVCに対応する宛先ネームスペースにPVCを作成します。
- アプリケーションと整合性のある最初のSnapshotを作成します。
- 初期Snapshotを使用して、永続ボリュームのSnapMirror関係を確立します。

データ保護ページには、レプリケーション関係の状態とステータスが表示されます。<Health status>|<Relationship life cycle state>

たとえば、Normal | Establishedです

レプリケーションの状態とステータスの詳細については、このトピックの最後を参照してください。

デスティネーションクラスタでレプリケートされたアプリケーションをオンラインにする（フェイルオーバー）

Astra Controlを使用すると、レプリケートされたアプリケーションをデスティネーションクラスタに「フェイルオーバー」できます。この手順 はレプリケーション関係を停止し、デスティネーションクラスタでアプリケーションをオンラインにします。ソースクラスタのアプリケーションが稼働していた場合、この手順 はそのアプリケーションを停止しません。

手順

1. Astra Controlの左ナビゲーションから、「アプリケーション」を選択します。
2. ［アプリケーション］ページで、［データ保護］>［レプリケーション］タブを選択します。
3. [データ保護（Data Protection）]>[複製（Replication）]タブの[アクション（Actions）]メニューから、[フェールオーバー*（フェールオーバー*）]を選択し
4. フェイルオーバーページで、情報を確認し、*フェイルオーバー*を選択します。

結果

手順 のフェイルオーバーでは、次の処理が実行されます。

- デスティネーションクラスタでは、レプリケートされた最新のSnapshotに基づいてアプリケーションが開始されます。
- ソースクラスタとアプリケーション（動作している場合）は停止されず、引き続き実行されます。
- レプリケーションの状態は「フェイルオーバー」に変わり、完了すると「フェイルオーバー」に変わります。
- ソースアプリケーションの保護ポリシーは、フェイルオーバー時にソースアプリケーションに存在するスケジュールに基づいて、デスティネーションアプリケーションにコピーされます。
- Astra Controlには、ソースクラスタとデスティネーションクラスタの両方のアプリケーションと、それぞれの健全性が表示されます。

フェイルオーバーしたレプリケーションを再同期します

再同期処理によってレプリケーション関係が再確立されます。関係のソースを選択して、ソースクラスタまたはデスティネーションクラスタにデータを保持することができます。この処理は、SnapMirror関係を再確立し、ボリュームのレプリケーションを任意の方向に開始します。

レプリケーションを再確立する前に、新しいデスティネーションクラスタ上のアプリケーションが停止されます。

> 再同期プロセスの間、ライフサイクルの状態は「Establishing」と表示されます。

手順

1. Astra Controlの左ナビゲーションから、「アプリケーション」を選択します。
2. ［アプリケーション］ページで、［データ保護］>［レプリケーション］タブを選択します。
3. [データ保護（Data Protection）]>[レプリケーション（Replication）]タブの[アクション（Actions）]メニューから、[*再同期（Resync *）]を
4. 再同期（Resync）ページで、保持するデータを含むソースまたはデスティネーションのアプリケーションインスタンスを選択します。

   > デスティネーションのデータが上書きされるため、再同期元は慎重に選択してください。

5. 続行するには、* Resync *を選択します。
6. 「resync」と入力して確定します。
7. 「* Yes、resync *」を選択して終了します。

結果

- Replication（レプリケーション）ページに、レプリケーションステータスとしてEstablishing（確立）が表示されます。
- Astra Controlは、新しいデスティネーションクラスタのアプリケーションを停止します。
- SnapMirror resyncを使用して、指定した方向に永続的ボリュームのレプリケーションを再確立します。
- [レプリケーション]ページに、更新された関係が表示されます。

アプリケーションのレプリケーションを反転する

元のソースクラスタへのレプリケートを続行したまま、アプリケーションをデスティネーションクラスタに移動する計画的処理です。Astra Controlは、ソースクラスタ上のアプリケーションを停止し、デスティネーションにデータをレプリケートしてから、デスティネーションクラスタにアプリケーションをフェイルオーバーします。

この状況では、ソースとデスティネーションを交換しようとしています。元のソースクラスタが新しいデスティネーションクラスタになり、元のデスティネーションクラスタが新しいソースクラスタになります。

手順

1. Astra Controlの左ナビゲーションから、「アプリケーション」を選択します。
2. ［アプリケーション］ページで、［データ保護］>［レプリケーション］タブを選択します。
3. [データ保護（Data Protection）]>[レプリケーション（Replication）]タブの[アクション（Actions）]メニューから、[レプリケーションを反転（Reverse replication）]を選択します
4. リバース・レプリケーションのページで情報を確認し、「リバース・レプリケーション」を選択して続行します。

結果

リバースレプリケーションの結果、次の処理が実行されます。

- Snapshotは、元のソースアプリケーションのKubernetesリソースから作成されます。
- 元のソースアプリケーションのポッドは、アプリケーションのKubernetesリソースを削除することで正常に停止されます（PVCとPVはそのまま維持されます）。
- ポッドがシャットダウンされると、アプリケーションのボリュームのSnapshotが作成されてレプリケートされます。
- SnapMirror関係が解除され、デスティネーションボリュームが読み取り/書き込み可能な状態になります。
- アプリケーションのKubernetesリソースは、元のソースアプリケーションのシャットダウン後にレプリケートされたボリュームデータを使用して、シャットダウン前のSnapshotからリストアされます。
- 逆方向にレプリケーションが再確立されます。

アプリケーションを元のソースクラスタにフェイルバックします

Astra Controlを使用すると、次の一連の操作を使用して、「フェイルオーバー」操作後に「フェイルバック」を実行できます。このワークフローでは、元のレプリケーション方向を復元するために、レプリケーションの方向を反転する前に、Astra Controlによってアプリケーションの変更が元のソースクラスタにレプリケート（再同期）されます。

このプロセスは、デスティネーションへのフェイルオーバーを完了した関係から開始され、次の手順を実行し

ます。

- フェイルオーバー状態から開始します。
- 関係を再同期します。
- レプリケーションを反転する。

手順

1. Astra Controlの左ナビゲーションから、「アプリケーション」を選択します。
2. ［アプリケーション］ページで、［データ保護］>［レプリケーション］タブを選択します。
3. [データ保護（Data Protection）]>[レプリケーション（Replication）]タブの[アクション（Actions）]メニューから、[*再同期（Resync *）]を
4. フェイルバック処理の場合は、フェイルオーバーしたオーバーアプリケーションを再同期処理のソースとして選択します（フェイルオーバー後に書き込まれたデータは保持します）。
5. 「resync」と入力して確定します。
6. 「* Yes、resync *」を選択して終了します。
7. 再同期が完了したら、[データ保護（Data Protection）]>[レプリケーション（Replication）]タブの[アクション（Actions）]メニューから[*レプリケーションを反転（Reverse replication）]を選択します。
8. リバース・レプリケーションのページで、情報を確認し、*リバース・レプリケーション*を選択します。

結果

このコマンドは、「resync」処理と「reverse relationship」処理の結果を組み合わせて、レプリケーションが再開された元のソースクラスタ上のアプリケーションを元のデスティネーションクラスタにオンラインにします。

アプリケーションレプリケーション関係を削除します

関係を削除すると、2つの異なるアプリケーション間に関係がなくなります。

手順

1. Astra Controlの左ナビゲーションから、「アプリケーション」を選択します。
2. ［アプリケーション］ページで、［データ保護］>［レプリケーション］タブを選択します。
3. [データ保護]>[レプリケーション]タブの[アプリケーション保護]ボックスまたは関係図で、[レプリケーション関係の削除*]を選択します。

結果

レプリケーション関係を削除すると、次の処理が実行されます。

- 関係が確立されていても、アプリケーションがデスティネーションクラスタでオンラインになっていない（フェイルオーバーした）場合、Astra Controlは、初期化中に作成されたPVCを保持し、「空」の管理対象アプリケーションをデスティネーションクラスタに残します。また、作成されたバックアップを保持するためにデスティネーションアプリケーションを保持します。
- アプリケーションがデスティネーションクラスタでオンラインになった（フェイルオーバーした）場合、Astra ControlはPVCと宛先アプリケーションを保持します。ソースとデスティネーションのアプリケーションは、独立したアプリケーションとして扱われるようになりました。バックアップスケジュールは、両方のアプリケーションで維持されますが、相互に関連付けられていません。

レプリケーション関係のヘルスステータスと関係のライフサイクル状態

Astra Controlには、関係の健全性と、レプリケーション関係のライフサイクルの状態が表示されます。

レプリケーション関係のヘルスステータス

レプリケーション関係の健常性は、次のステータスで示されます。

- 正常：関係が確立されているか確立されており、最新のSnapshotが転送されました。
- 警告：関係がフェイルオーバーされているかフェイルオーバーされています（そのためソースアプリは保護されなくなりました）。
- * 重要 *
  - 関係が確立されているか、フェイルオーバーされていて、前回の調整が失敗しました。
  - 関係が確立され、新しいPVCの追加を最後に調整しようとしても失敗しています。
  - 関係は確立されていますが（Snapshotが正常にレプリケートされ、フェイルオーバーが可能になります）、最新のSnapshotはレプリケートに失敗したか、レプリケートに失敗しています。

レプリケーションのライフサイクル状態

次の状態は、レプリケーションのライフサイクルの各段階を表しています。

- * Establishing *：新しいレプリケーション関係を作成中です。Astra Controlは、必要に応じてネームスペースを作成し、デスティネーションクラスタの新しいボリュームにPersistent Volumeクレーム（PVC；永続ボリューム要求）を作成し、SnapMirror関係を作成します。このステータスは、レプリケーションが再同期中であること、またはレプリケーションを反転中であることを示している可能性もあり
- * established *：レプリケーション関係が存在します。Astra Controlは、PVCが使用可能かどうかを定期的にチェックし、レプリケーション関係をチェックし、アプリケーションのSnapshotを定期的に作成し、アプリケーション内の新しいソースPVCを特定します。その場合は、レプリケーションに含めるリソースがAstra Controlによって作成されます。
- フェイルオーバー：SnapMirror関係が解除され、アプリケーションのKubernetesリソースが最後にレプリケートされたアプリケーションのSnapshotからリストアされます。
- *フェイルオーバーした場合：Astra Controlは、ソースクラスタからのレプリケーションを停止し、デスティネーションでレプリケートされた最新の（成功した）アプリケーションSnapshotを使用して、Kubernetesリソースをリストアします。
- * resyncing *：Astra Controlは、SnapMirror resyncを使用して、再同期元の新しいデータを再同期先に再同期します。この処理では、同期の方向に基づいて、デスティネーション上の一部のデータが上書きされる可能性があります。Astra Controlは、デスティネーションネームスペースで実行されているアプリケーションを停止し、Kubernetesアプリケーションを削除します。再同期処理の実行中、ステータスは「Establishing」と表示されます。
- リバース：は、元のソースクラスタへのレプリケーションを続行しながらアプリケーションをデスティネーションクラスタに移動する予定の処理です。Astra Controlは、ソースクラスタ上のアプリケーションを停止し、デスティネーションにデータをレプリケートしてから、デスティネーションクラスタにアプリケーションをフェイルオーバーします。リバースレプリケーションの間、ステータスは「Establishing」と表示されます。
- 削除中：
  - レプリケーション関係が確立されたものの、まだフェイルオーバーされていない場合は、レプリケーション中に作成されたPVCがAstra Controlによって削除され、デスティネーションの管理対象アプリ

ケーションが削除されます。

- レプリケーションがすでにフェイルオーバーされている場合、Astra ControlはPVCと宛先アプリケーションを保持します。

アプリケーションのクローン作成と移行

既存のアプリケーションをクローニングして、同じKubernetesクラスタまたは別のクラスタに重複するアプリケーションを作成できます。Astra Control でアプリケーションをクローニングすると、アプリケーション構成と永続的ストレージのクローンが作成されます。

Kubernetes クラスタ間でアプリケーションとストレージを移動する必要がある場合は、クローニングが役立ちます。たとえば、 CI / CD パイプラインや Kubernetes ネームスペース間でワークロードを移動できます。Astra Control Center UIまたはを使用できます "Astra Control API" アプリケーションのクローン作成と移行を実行します。

必要なもの

- アプリケーションを別のクラスタにクローニングするには、ソースクラスタとデスティネーションクラスタを含むクラウドインスタンス（同じでない場合）にデフォルトのバケットを用意する必要があります。クラウドインスタンスごとにデフォルトのバケットを割り当てる必要があります。
- クローン処理中に、IngressClassリソースまたはwebhookを必要とするアプリケーションが正常に機能するためには、これらのリソースがデスティネーションクラスタですでに定義されていない必要があります。

> OpenShift 環境でのアプリケーションのクローニングでは、 Astra Control Center が 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。そのため、これらの処理を許可するには、 ONTAP ボリュームのエクスポートポリシーを設定する必要があります。次のコマンドを使用して実行できます。
>
> 1. `export-policy rule modify -vserver <storage virtual machine name> -policyname <policy name> -ruleindex 1 -superuser sys`
> 2. `export-policy rule modify -vserver <storage virtual machine name> -policyname <policy name> -ruleindex 1 -anon 65534`

クローンの制限事項

- 明示的なストレージクラス：ストレージクラスを明示的に設定したアプリケーションを導入し、その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、ターゲットクラスタには元 々 指定されたストレージクラスが必要です。ストレージクラスを明示的に設定したアプリケーションを、同じストレージクラスを含まないクラスタにクローニングすると、失敗します。
- クローンとユーザーの制約：名前空間の名前/ IDまたは名前空間のラベルによって名前空間の制約を持つメンバーユーザーは、同じクラスタ上の新しい名前空間、または組織のアカウント内の他の任意のクラスタに対して、アプリケーションのクローンまたはリストアを実行できます。ただし、同じユーザが、クローニングまたはリストアされたアプリケーションに新しいネームスペースからアクセスすることはできません。クローンまたはリストア処理によって新しいネームスペースが作成されると、アカウントの管理者 / 所有者はメンバーユーザアカウントを編集し、該当するユーザに新しいネームスペースへのアクセスを許可するロールの制限を更新できます。
- クローンはデフォルトバケットを使用：アプリケーションのバックアップまたはアプリケーションのリストア時に、オプションでバケットIDを指定できます。ただし、アプリケーションのクローニング処理で

は、定義済みのデフォルトバケットが常に使用されます。クローンのバケットを変更するオプションはありません。どのバケットを使用するかを制御する必要がある場合は、どちらかを選択できます "バケットのデフォルト設定を変更する" または、を実行します "バックアップ" その後にを押します "リストア" 個別。

- * Jenkins CI*を使用：オペレータがデプロイしたJenkins CIのインスタンスをクローニングする場合は、永続データを手動で復元する必要があります。これは、アプリケーションの展開モデルの制限事項です。
- * S3バケットを使用している場合*：Astra Control CenterのS3バケットは使用可能容量を報告しません。Astra Control Center で管理されているアプリケーションのバックアップまたはクローニングを行う前に、 ONTAP または StorageGRID 管理システムでバケット情報を確認します。

OpenShift に関する考慮事項

- クラスタおよび**OpenShift**バージョン：クラスタ間でアプリケーションをクローニングする場合、ソースクラスタとデスティネーションクラスタはOpenShiftの同じディストリビューションである必要があります。たとえば、 OpenShift 4.7 クラスタからアプリケーションをクローニングする場合は、 OpenShift 4.7 でもあるデスティネーションクラスタを使用します。
- *プロジェクトおよびUID *：OpenShiftクラスタでアプリをホストするプロジェクトを作成すると、プロジェクト（またはKubernetes名前空間）にSecurityContext UIDが割り当てられます。Astra Control Center でアプリケーションを保護し、 OpenShift でそのアプリケーションを別のクラスタまたはプロジェクトに移動できるようにするには、アプリケーションを任意の UID として実行できるようにポリシーを追加する必要があります。たとえば、次の OpenShift CLI コマンドは、 WordPress アプリケーションに適切なポリシーを付与します。

```
oc new-project wordpress
oc adm policy add-scc-to-group anyuid system:serviceaccounts:wordpress
oc adm policy add-scc-to-user privileged -z default -n wordpress
```

手順

1. 「 * アプリケーション * 」を選択します。
2. 次のいずれかを実行します。
   - 目的のアプリケーションの [* アクション * （ * Actions * ）] 列で [オプション（ Options ）] メニューを選択します。
   - 目的のアプリケーションの名前を選択し、ページの右上にあるステータスドロップダウンリストを選択します。
3. 「 * Clone * 」を選択します。
4. クローンの詳細を指定します。
   - 名前を入力します。
   - クローンのデスティネーションクラスタを選択してください。
   - クローンのデスティネーションネームスペースを入力してください。アプリケーションに関連付けられた各ソースネームスペースは、定義した宛先ネームスペースにマッピングされます。

     Astra Controlでは、クローニング処理の一環として新しいデスティネーションネームスペースが作成されます。指定するデスティネーションネームスペースがデスティネーションクラスタに存在していないことを確認してください。

   - 「 * 次へ * 」を選択します。

- 既存の Snapshot からクローンを作成するかバックアップを作成するかを選択します。このオプションを選択しない場合、 Astra Control Center はアプリケーションの現在の状態からクローンを作成します。
  - 既存のSnapshotまたはバックアップからクローニングする場合は、使用するSnapshotまたはバックアップを選択します。

5. 「 * 次へ * 」を選択します。
6. クローンに関する情報を確認し、* Clone *を選択します。

結果

Astra Controlは、入力した情報に基づいてアプリケーションをクローニングします。新しいアプリケーションクローンがに含まれている場合、クローニング処理は成功します `Healthy` 「アプリケーション」ページで説明します。

クローンまたはリストア処理によって新しいネームスペースが作成されると、アカウントの管理者 / 所有者はメンバーユーザアカウントを編集し、該当するユーザに新しいネームスペースへのアクセスを許可するロールの制限を更新できます。

> データ保護処理（クローン、バックアップ、またはリストア）が完了して永続ボリュームのサイズを変更したあと、新しいボリュームのサイズがUIに表示されるまでに最大20分かかります。データ保護処理にかかる時間は数分です。また、ストレージバックエンドの管理ソフトウェアを使用してボリュームサイズの変更を確認できます。

アプリケーション実行フックを管理します

実行フックは、管理対象アプリケーションのデータ保護操作と組み合わせて実行するように構成できるカスタムアクションです。たとえば、データベースアプリケーションがある場合、実行フックを使用して、スナップショットの前にすべてのデータベーストランザクションを一時停止し、スナップショットの完了後にトランザクションを再開できます。これにより、アプリケーションと整合性のある Snapshot を作成できます。

実行フックのタイプ

Astra Controlは、実行可能なタイミングに基づいて、次の種類の実行フックをサポートします。

- Snapshot前
- Snapshot後
- バックアップ前
- バックアップ後
- リストア後のPOSTコマンドです

カスタム実行フックに関する重要な注意事項

アプリケーションの実行フックを計画するときは、次の点を考慮してください。

- 実行フックは、スクリプトを使用してアクションを実行する必要があります。多くの実行フックは、同じスクリプトを参照できます。

- Astra Controlでは、実行フックが実行可能なシェルスクリプトの形式で記述されるようにするスクリプトが必要です。
- スクリプトのサイズは96KBに制限されています。
- Astra Controlは、実行フックの設定と一致条件を使用して、スナップショット、バックアップ、または復元操作に適用できるフックを決定します。
- 実行フックの障害はすべて'ソフトな障害ですフックが失敗しても'他のフックとデータ保護操作は試行されますただし、フックが失敗すると、 * アクティビティ * ページイベントログに警告イベントが記録されます。
- 実行フックを作成、編集、または削除するには、 Owner 、 Admin 、または Member 権限を持つユーザーである必要があります。
- 実行フックの実行に 25 分以上かかる場合 ' フックは失敗し ' 戻りコードが N/A のイベント・ログ・エントリが作成されます該当する Snapshot はタイムアウトして失敗とマークされ、タイムアウトを通知するイベントログエントリが生成されます。
- アドホックデータ保護操作の場合、すべてのフックイベントが生成され、[**Activity**]ページのイベントログに保存されます。ただし、スケジュールされたデータ保護処理については、フック障害イベントだけがイベントログに記録されます（スケジュールされたデータ保護処理自体によって生成されたイベントは記録されたままです）。

> - Istioサービスメッシュに参加するアプリケーションの実行フックを作成する場合は、フックがサービスメッシュコンテナではなく、元のアプリケーションコンテナに対して実行されていることを確認します。Istioサービスメッシュを使用するアプリケーションに対して実行されるすべての実行フックにフィルターregexを適用することで、Istioサービスメッシュコンテナを除外できます。
> - 実行フックは、実行中のアプリケーションの機能を低下させるか、完全に無効にすることが多いため、カスタム実行フックの実行時間を最小限に抑えるようにしてください。
> - 実行フックが関連付けられている状態でバックアップまたはスナップショット操作を開始した後'キャンセルした場合でも'バックアップまたはスナップショット操作がすでに開始されていればフックは実行できますつまり、バックアップ後の実行フックでは、バックアップが完了したとは判断できません。

実行順序

データ保護操作を実行すると、実行フックイベントが次の順序で実行されます。

1. 適用可能なカスタムプリオペレーション実行フックは、適切なコンテナで実行されます。カスタムのプリオペレーションフックは必要なだけ作成して実行できますが、操作前のこれらのフックの実行順序は保証も構成もされていません。
2. データ保護処理が実行されます。
3. 適用可能なカスタムポストオペレーション実行フックは、適切なコンテナで実行されます。必要な数のカスタムポストオペレーションフックを作成して実行できますが、操作後のこれらのフックの実行順序は保証されず、設定もできません。

同じ種類の実行フック（スナップショット前など）を複数作成する場合、これらのフックの実行順序は保証されません。ただし、異なるタイプのフックの実行順序は保証されています。たとえば、5つの異なるタイプのフックをすべて持つ構成の実行順序は、次のようになります。

1. 予備フックが実行されます

2. スナップショット前フックが実行されます
3. スナップショット後フックが実行されます
4. バックアップ後のフックが実行されます
5. 復元後のフックが実行されます

シナリオ番号2のこの設定の例は、の表を参照してください [フックが実行されるかどうかを確認します]。

> 本番環境で実行スクリプトを有効にする前に、必ず実行フックスクリプトをテストしてください。'kubectl exec' コマンドを使用すると、スクリプトを簡単にテストできます。本番環境で実行フックを有効にしたら、作成されたSnapshotとバックアップをテストして整合性があることを確認します。これを行うには、アプリケーションを一時的なネームスペースにクローニングし、スナップショットまたはバックアップをリストアしてから、アプリケーションをテストします。

フックが実行されるかどうかを確認します

次の表を使用して、アプリケーションでカスタム実行フックが実行されるかどうかを判断します。

アプリケーションの高レベルの処理は、すべてスナップショット、バックアップ、またはリストアの基本的な処理のいずれかを実行することで構成されることに注意してください。シナリオによっては、クローニング処理はこれらの処理のさまざまな組み合わせで構成されるため、クローン処理を実行する実行フックはさまざまです。

In Placeリストア処理では既存のSnapshotまたはバックアップが必要になるため、これらの処理ではSnapshotまたはバックアップフックは実行されません。

> 開始してスナップショットを含むバックアップをキャンセルし'実行フックが関連付けられている場合は'一部のフックが実行され'ほかのフックが実行されないことがありますつまり、バックアップ後の実行フックでは、バックアップが完了したとは判断できません。キャンセルしたバックアップに関連する実行フックがある場合は、次の点に注意してください。
>
> - バックアップ前およびバックアップ後のフックは常に実行されます。
> - バックアップに新しいスナップショットが含まれており'スナップショットが開始されている場合は'スナップショット前フックとスナップショット後フックが実行されます
> - スナップショットの開始前にバックアップがキャンセルされた場合は'スナップショット前フックとスナップショット後フックは実行されません

シナリオ（**Scenario**）	操作	既存の**Snapshot**	既存のバックアップ	ネームスペース	クラスタ	スナップショットフックが実行されます	バックアップフックが実行されます	フックを元に戻します
1.	クローン	N	N	新規	同じ	Y	N	Y
2.	クローン	N	N	新規	違う	Y	Y	Y
3.	クローンまたはリストア	Y	N	新規	同じ	N	N	Y

シナリオ（Scenario）	操作	既存のSnapshot	既存のバックアップ	ネームスペース	クラスタ	スナップショットフックが実行されます	バックアップフックが実行されます	フックを元に戻します
4.	クローンまたはリストア	N	Y	新規	同じ	N	N	Y
5.	クローンまたはリストア	Y	N	新規	違う	N	Y	Y
6.	クローンまたはリストア	N	Y	新規	違う	N	N	Y
7.	リストア	Y	N	既存	同じ	N	N	Y
8.	リストア	N	Y	既存	同じ	N	N	Y
9.	スナップショット	該当なし	該当なし	該当なし	該当なし	Y	該当なし	該当なし
10.	バックアップ	N	該当なし	該当なし	該当なし	Y	Y	該当なし
11.	バックアップ	Y	該当なし	該当なし	該当なし	N	Y	該当なし

実行フックの例

にアクセスします "NetApp Verda GitHubプロジェクト" 例を見て、実行フックをどのように構成するかを考えてみましょう。これらの例は、テンプレートまたはテストスクリプトとして使用できます。

既存の実行フックを表示します

アプリケーションの既存のカスタム実行フックを表示できます。

手順

1. 「 * アプリケーション」に移動し、管理アプリの名前を選択します。
2. [実行フック *] タブを選択します。

   有効または無効になっているすべての実行フックを結果リストに表示できます。フックのステータス、ソース、および実行時刻（プリ/ポストオペレーション）を表示できます。実行フックに関連するイベントログを表示するには、左側のナビゲーション領域の * アクティビティ * ページに移動します。

既存のスクリプトを表示します

アップロードされた既存のスクリプトを表示できます。このページでは、使用中のスクリプトと、使用中のフックを確認することもできます。

手順

1. 「アカウント」に移動します。

2. [スクリプト]タブを選択します。

   このページには、アップロードされた既存のスクリプトのリストが表示されます。[使用者*]列には、各スクリプトを使用している実行フックが表示されます。

スクリプトを追加します

実行フックが参照できるスクリプトを1つ以上追加できます。多くの実行フックは、同じスクリプトを参照できます。これにより、1つのスクリプトのみを変更することで、多数の実行フックを更新できます。

手順

1. 「アカウント」に移動します。
2. [スクリプト]タブを選択します。
3. 「 * 追加」を選択します。
4. 次のいずれかを実行します。
   - カスタムスクリプトをアップロードする。
     i. [ファイルのアップロード（ Upload file ）] オプションを選択します。
     ii. ファイルを参照してアップロードします。
     iii. スクリプトに一意の名前を付けます。
     iv. （オプション）他の管理者がスクリプトについて知っておく必要があるメモを入力します。
     v. 「スクリプトを保存」を選択します。
   - クリップボードからカスタムスクリプトを貼り付けます。
     i. [貼り付け（Paste）]または[タイプ（* type）]オプションを選択する
     ii. テキストフィールドを選択し、スクリプトテキストをフィールドに貼り付けます。
     iii. スクリプトに一意の名前を付けます。
     iv. （オプション）他の管理者がスクリプトについて知っておく必要があるメモを入力します。
5. 「スクリプトを保存」を選択します。

結果

新しいスクリプトが、[スクリプト]タブのリストに表示されます。

スクリプトを削除します

不要になって実行フックで使用されなくなったスクリプトは、システムから削除できます。

手順

1. 「アカウント」に移動します。
2. [スクリプト]タブを選択します。
3. 削除するスクリプトを選択し、「アクション」列のメニューを選択します。
4. 「 * 削除」を選択します。

> スクリプトが1つまたは複数の実行フックに関連付けられている場合、*Delete*アクションは使用できません。スクリプトを削除するには、まず関連する実行フックを編集し、別のスクリプトに関連付けます。

カスタム実行フックを作成します

アプリケーションのカスタム実行フックを作成できます。を参照してください [実行フックの例] フックの例を参照してください。実行フックを作成するには、 Owner 、 Admin 、または Member のいずれかの権限が必要です。

> 実行フックとして使用するカスタムシェルスクリプトを作成する場合は、特定のコマンドを実行するか、実行可能ファイルへの完全パスを指定する場合を除き、ファイルの先頭に適切なシェルを指定するようにしてください。

手順

1. 「 * アプリケーション」を選択し、管理アプリの名前を選択します。
2. [実行フック *] タブを選択します。
3. 「 * 追加」を選択します。
4. フックの詳細*（* Hook Details ）領域で、*操作（* Operation *）ドロップダウンメニューから操作タイプを選択して、フックを実行するタイミングを決定します。
5. フックの一意の名前を入力します。
6. （オプション）実行中にフックに渡す引数を入力し、各引数を入力した後で Enter キーを押して、それぞれを記録します。
7. [* Container Images * （コンテナイメージ * ）] 領域で、アプリケーションに含まれるすべてのコンテナイメージに対してフックを実行する必要がある場合は、 [* Apply to all container images * （すべてのコンテナイメージに適用 * ）] チェックボックスを有効にします。代わりに、フックが 1 つ以上の指定されたコンテナイメージに対してのみ機能する場合は、 * Container image names to match * フィールドにコンテナイメージ名を入力します。
8. [* スクリプト * （ * Script * ）] 領域で、次のいずれかを実行します。
   - 新しいスクリプトを追加します。
     i. 「 * 追加」を選択します。
     ii. 次のいずれかを実行します。
       - カスタムスクリプトをアップロードする。
         I. [ファイルのアップロード（ Upload file ）] オプションを選択します。
         II. ファイルを参照してアップロードします。
         III. スクリプトに一意の名前を付けます。
         IV. （オプション）他の管理者がスクリプトについて知っておく必要があるメモを入力します。
         V. 「スクリプトを保存」を選択します。
       - クリップボードからカスタムスクリプトを貼り付けます。
         I. [貼り付け（Paste）]または[タイプ（* type）]オプションを選択する

        II. テキストフィールドを選択し、スクリプトテキストをフィールドに貼り付けます。
        III. スクリプトに一意の名前を付けます。
        IV. （オプション）他の管理者がスクリプトについて知っておく必要があるメモを入力します。
    - リストから既存のスクリプトを選択します。

      このスクリプトを使用するように実行フックに指示します。

9. [* フックを追加 *] を選択します。

実行フックの状態を確認します

スナップショット、バックアップ、または復元操作の実行が終了したら、操作の一部として実行された実行フックの状態を確認できます。このステータス情報を使用して、実行フックを保持するか、変更するか、削除するかを決定できます。

手順

1. 「 * アプリケーション」を選択し、管理アプリの名前を選択します。
2. [データ保護]タブを選択します。
3. 実行中のSnapshotを表示するには「* Snapshots」を選択し、実行中のバックアップを表示するには「* Backups」を選択します。

   フック状態*は、操作完了後の実行フックランのステータスを示します。状態にカーソルを合わせると、詳細を確認できます。たとえば、スナップショット中に実行フック障害が発生した場合、そのスナップショットのフック状態にカーソルを合わせると、失敗した実行フックのリストが表示されます。各失敗の理由を確認するには、左側のナビゲーション領域の*アクティビティ*ページを確認します。

スクリプトの使用状況を表示します

どの実行フックがAstra Control Web UIの特定のスクリプトを使用しているかを確認できます。

手順

1. 「 * アカウント * 」を選択します。
2. [スクリプト]タブを選択します。

   スクリプトのリストにある* Used by *列には、リスト内の各スクリプトを使用しているフックの詳細が表示されます。

3. 目的のスクリプトの[使用者*]列の情報を選択します。

   より詳細なリストが表示され、スクリプトを使用しているフックの名前と、それらが実行されるように構成されている操作のタイプが示されます。

実行フックを無効にします

アプリケーションのスナップショットの前または後に実行を一時的に禁止する場合は、実行フックを無効にできます。実行フックを無効にするには、 Owner 、 Admin 、または Member のいずれかの権限が必要です。

手順

1. 「 * アプリケーション」を選択し、管理アプリの名前を選択します。
2. [実行フック *] タブを選択します。
3. 無効にするフックの * アクション * 列のオプションメニューを選択します。
4. [**Disable**] を選択します。

実行フックを削除します

不要になった実行フックは完全に削除できます。実行フックを削除するには、 Owner 、 Admin 、または Member のいずれかの権限が必要です。

手順

1. 「 * アプリケーション」を選択し、管理アプリの名前を選択します。
2. [実行フック *] タブを選択します。
3. 削除するフックの * アクション * 列のオプションメニューを選択します。
4. 「 * 削除」を選択します。

を参照してください。

- "NetApp Verda GitHubプロジェクト"

アプリケーションとクラスタの健常性を監視

アプリケーションとクラスタの健常性の概要を表示します

ダッシュボード * を選択すると、アプリ、クラスター、ストレージバックエンド、それらのヘルスの概要が表示されます。

これらは静的な数値やステータスだけでなく、それぞれからドリルダウンすることもできます。たとえば、アプリが完全に保護されていない場合は、アイコンの上にカーソルを置くと、完全に保護されていないアプリを特定できます。その理由が含まれます。

アプリケーションタイル

「 * アプリケーション * 」タイルは、次の項目を識別するのに役立ちます。

- Astra で現在管理しているアプリケーションの数。
- それらの管理アプリが正常であるかどうか。
- アプリケーションが完全に保護されているかどうか（最新のバックアップがある場合は保護されます）。
- 検出されたものの、まだ管理されていないアプリケーションの数。

  アプリケーションが検出された後で管理または無視するため、この数はゼロになるのが理想的です。さらに、ダッシュボードで検出されたアプリケーションの数を監視して、開発者がクラスタに新しいアプリケーションを追加するタイミングを特定します。

クラスタタイル

クラスタタイルには、 Astra Control Center を使用して管理しているクラスタの健常性に関する同様の詳細が表示され、ドリルダウンしてアプリと同様に詳細を確認できます。

ストレージバックエンドはタイル張りです

「ストレージバックエンド * 」タイルは、ストレージバックエンドの健全性を特定するための情報を提供します。これには次のものが含まれます。

- 管理対象のストレージバックエンドの数
- これらの管理バックエンドが正常であるかどうか
- バックエンドが完全に保護されているかどうか
- 検出されたがまだ管理されていないバックエンドの数。

クラスタの健全性を表示してストレージクラスを管理します

Astra Control Center で管理するクラスタを追加すると、その場所、ワーカーノード、永続ボリューム、ストレージクラスなど、クラスタに関する詳細を表示できます。管理対象クラスタのデフォルトのストレージクラスを変更することもできます。

クラスタの健常性と詳細を表示します

クラスタの場所、ワーカーノード、永続ボリューム、ストレージクラスなどの詳細を表示できます。

手順

1. Astra Control Center UI で、 [* Clusters] を選択します。
2. [* Clusters] ページで、詳細を表示するクラスタを選択します。

> クラスタの構成 `removed` クラスタとネットワークの接続が正常であると表示される（Kubernetes APIを使用してクラスタに外部からアクセスしようとすると成功する）場合は、Astra Controlに指定したkubeconfigが無効になる可能性があります。クラスタでの証明書のローテーションまたは有効期限が原因の可能性があります。この問題を修正するには、を使用して、 Astra Control のクラスタに関連付けられたクレデンシャルを更新します "Astra Control API の略"。

3. [Overview （概要）] 、 [* Storage （ * ストレージ）] 、 [* Activity * （アクティビティ * ）] タブの情報を表示して、必要な情報を検索します。
   - * 概要 * ：状態を含むワーカーノードの詳細。
   - * ストレージ * ：ストレージクラスと状態を含む、コンピューティングに関連付けられた永続的ボリューム。
   - * アクティビティ * ：クラスタに関連するアクティビティを表示します。

> Astra Control Center * Dashboard * から始まるクラスタ情報を表示することもできます。[* クラスタ *] タブの [* リソースサマリ *] で、管理対象クラスタを選択して [* クラスタ *] ページに移動できます。[* Clusters] ページが表示されたら、上記の手順を実行します。

デフォルトのストレージクラスを変更する

クラスタのデフォルトのストレージクラスは変更できます。Astra Controlは、クラスタを管理する際に、クラスタのデフォルトストレージクラスを追跡します。

> kubectlコマンドを使用してストレージクラスを変更しないでください。代わりに、この手順 を使用してください。kubectlを使用して変更を行った場合、Astra Controlはその変更を元に戻します。

手順

1. Astra Control Center Web UIで、[* Clusters]を選択します。
2. [* Clusters]ページで、変更するクラスタを選択します。
3. [* ストレージ *] タブを選択します。
4. 「ストレージクラス」カテゴリを選択します。
5. デフォルトとして設定するストレージクラスの* Actions *メニューを選択します。
6. 「デフォルトに設定」を選択します。

アプリの状態と詳細を表示します

アプリケーションの管理を開始すると、アプリケーションのステータス（正常かどうか）、保護ステータス（障害発生時に完全に保護されているかどうか）、ポッド、永続的ストレージなどを識別できる詳細が Astra Control に表示されます。

手順

1. Astra Control Center UI で、 * アプリケーション * を選択し、アプリの名前を選択します。
2. 情報を確認します。
   - アプリステータス：Kubernetesでのアプリの状態を反映するステータスを提供します。たとえば、ポッドと永続ボリュームはオンラインか？アプリケーションが正常な状態でない場合は、 Kubernetes のログでクラスタの問題を調べてトラブルシューティングする必要があります。Astra は、壊れたアプリケーションの修正に役立つ情報を提供していません。
   - アプリ保護ステータス：アプリの保護状態を表示します。
     - * 完全に保護されている * ：アプリにはアクティブなバックアップスケジュールがあり、 1 週間も経過していない正常なバックアップがあります
     - * 部分的に保護 * ：アプリケーションには、アクティブなバックアップスケジュール、アクティブなスナップショットスケジュール、または正常なバックアップまたはスナップショットがあります
     - * 保護されていない * ：完全に保護されていない、または部分的に保護されていないアプリ

       _ 最新のバックアップがあるまで、完全に保護することはできません _ 。これは、永続ボリュームから離れたオブジェクトストアにバックアップが格納されるために重要です。障害や事故によってクラスタと永続的ストレージが消去された場合は、バックアップをリカバリする必要があります。スナップショットを使用してリカバリすることはできません。
   - 概要：アプリケーションに関連付けられているポッドの状態に関する情報。

- データ保護：データ保護ポリシーを設定し、既存のスナップショットとバックアップを表示できます。
- ストレージ：アプリケーションレベルの永続的ボリュームを表示します。永続ボリュームの状態は、Kubernetes クラスタから見たものです。
- リソース：バックアップおよび管理されているリソースを確認できます。
- アクティビティ：アプリケーションに関連するアクティビティを表示します。

> Astra Control Center * Dashboard * から始まるアプリ情報を表示することもできます。[* アプリケーション *] タブの [リソースの概要 *] で、管理アプリを選択して [* アプリケーション *] ページに移動できます。[**Applications**] ページが表示されたら、上記の手順に従います。

アカウントを管理します

ローカルユーザとロールを管理します

Astra Control UIを使用して、Astra Control Centerインストールのユーザーを追加、削除、および編集できます。Astra Control UI またはを使用できます "Astra Control API" ユーザを管理するには、を実行

LDAPを使用して、選択したユーザの認証を実行することもできます。

LDAP を使用する

LDAPは、分散ディレクトリ情報にアクセスするための業界標準プロトコルであり、エンタープライズ認証に広く使用されています。Astra Control CenterをLDAPサーバーに接続して、選択したAstra Controlユーザーの認証を実行できます。大まかには、AstraとLDAPを統合し、Astra ControlユーザおよびLDAP定義に対応するグループを定義することです。Astra Control APIまたはWeb UIを使用して、LDAP認証とLDAPユーザおよびグループを設定できます。詳細については、次のドキュメントを参照してください。

- "リモート認証とユーザーの管理には、Astra Control APIを使用します"
- "リモートユーザとリモートグループの管理には、Astra Control UIを使用します"
- "リモート認証を管理するには、Astra Control UIを使用します"

ユーザを追加します

アカウント所有者と管理者は、 Astra Control Center のインストールにさらにユーザーを追加できます。

手順

1. 「アカウントの管理」ナビゲーション領域で、「 * アカウント * 」を選択します。
2. [**Users**] タブを選択します。
3. [ユーザーの追加] を選択します。
4. ユーザ名、 E メールアドレス、および一時パスワードを入力します。

   ユーザは初回ログイン時にパスワードを変更する必要があります。

5. 適切なシステム権限を持つユーザロールを選択します。

   各ロールには次の権限があります。

   - * Viewer * はリソースを表示できます。
   - メンバー * には、ビューア・ロールの権限があり、アプリとクラスタの管理、アプリの管理解除、スナップショットとバックアップの削除ができます。
   - **Admin** にはメンバーの役割権限があり、 Owner 以外の他のユーザーを追加および削除できます。
   - * Owner * には Admin ロールの権限があり、任意のユーザーアカウントを追加および削除できます。
6. メンバーロールまたはビューアロールを持つユーザーに制約を追加するには、 * 制約へのロールの制限 * チェックボックスをオンにします。

   制約の追加の詳細については、を参照してください "ローカルユーザとロールを管理します"。

7. 「 * 追加」を選択します。

パスワードを管理します

Astra Control Center では、ユーザーアカウントのパスワードを管理できます。

パスワードを変更します

ユーザアカウントのパスワードはいつでも変更できます。

手順

1. 画面の右上にあるユーザアイコンを選択します。
2. * プロファイル * を選択します。
3. [* アクション * （ * Actions * ）] 列の [オプション（ Options ）] メニューから、 [* パスワードの変更 * （ * Change Password ）] を選択します
4. パスワードの要件に準拠するパスワードを入力します。
5. 確認のためパスワードをもう一度入力します。
6. 「 * パスワードの変更 * 」を選択します。

別のユーザのパスワードをリセットします

アカウントに Admin ロールまたは Owner ロールの権限がある場合は、自分だけでなく他のユーザアカウントのパスワードもリセットできます。パスワードをリセットする場合は、ログイン時にユーザが変更しなければならない一時パスワードを割り当てます。

手順

1. 「アカウントの管理」ナビゲーション領域で、「 * アカウント * 」を選択します。
2. [* アクション * （ * Actions * ）] ドロップダウンリストを選択します。
3. 「 * パスワードのリセット * 」を選択します。
4. パスワードの要件に適合する一時パスワードを入力します。
5. 確認のためパスワードをもう一度入力します。

次回ユーザがログインするときに、パスワードの変更を求めるプロンプトが表示されます。

6. 「 * パスワードのリセット * 」を選択します。

ユーザを削除します

所有者ロールまたは管理者ロールを持つユーザは、いつでもそのアカウントから他のユーザを削除できます。

手順

1. 「アカウントの管理」ナビゲーション領域で、「 * アカウント * 」を選択します。
2. [* ユーザー *] タブで、削除する各ユーザーの行にあるチェックボックスをオンにします。
3. [* アクション * （ * Actions * ）] 列の [オプション（ Options ）] メニューから、 [* ユーザー / 秒を削除（ * Remove user/s * ）] を選択する
4. プロンプトが表示されたら、「 remove 」という単語を入力して削除を確認し、「 * Yes 、 Remove User * 」を選択します。

結果

Astra Control Center は、アカウントからユーザーを削除します。

ロールの管理

ロールを管理するには、ネームスペースの制約を追加し、ユーザロールをその制約に制限します。これにより、組織内のリソースへのアクセスを制御できます。Astra Control UI またはを使用できます "Astra Control API" をクリックしてください。

ロールに名前空間制約を追加します

管理者または所有者ユーザーは、メンバーまたはビューアーの役割に名前空間の制約を追加できます。

手順

1. 「アカウントの管理」ナビゲーション領域で、「 * アカウント * 」を選択します。
2. [**Users**] タブを選択します。
3. [* アクション * （ * Actions * ）] 列で、メンバーまたはビューアーの役割を持つユーザーのメニューボタンを選択します。
4. [役割の編集] を選択します。
5. [ロールを制約に制限する *] チェックボックスをオンにします。

   このチェックボックスは、メンバーロールまたはビューアロールでのみ使用できます。[*Role] ドロップダウン・リストから別のロールを選択できます

6. [* 制約の追加 *] を選択します。

   使用可能な制約の一覧は、ネームスペースまたはネームスペースラベルで確認できます。

7. [制約タイプ * （ Constraint type * ）] ドロップダウンリストで、ネームスペースの構成方法に応じて、 [* Kubernetes namespace] * または [* Kubernetes namespace label*] を選択します。

8. リストから 1 つ以上の名前空間またはラベルを選択して、それらの名前空間にロールを制限する制約を構成します。
9. [* 確認 *] を選択します。

   [役割の編集 *] ページには、この役割に選択した拘束のリストが表示されます。

10. [* 確認 *] を選択します。

    [**Account**] ページでは、 [*Role] 列のメンバまたはビューアの役割の制約を表示できます。

> 制約を追加せずに役割の制約を有効にし、 * 確認 * を選択すると、役割には完全な制限があると見なされます（役割は、名前空間に割り当てられているリソースへのアクセスを拒否されます）。

ロールから名前空間制約を削除します

管理者または所有者ユーザーは、役割から名前空間の制約を削除できます。

手順

1. 「アカウントの管理」ナビゲーション領域で、「 * アカウント * 」を選択します。
2. [**Users**] タブを選択します。
3. [* アクション * （ * Actions * ）] 列で、アクティブな拘束を持つメンバーまたはビューアーの役割を持つユーザーのメニューボタンを選択する。
4. [役割の編集] を選択します。
   - 役割の編集 * （ Edit role * ）ダイアログには、役割のアクティブな拘束が表示されます。
5. 削除する拘束の右側にある * X * を選択します。
6. [* 確認 *] を選択します。

を参照してください。

- "ユーザロールとネームスペース"

リモート認証を管理する

LDAPは、分散ディレクトリ情報にアクセスするための業界標準プロトコルであり、エンタープライズ認証に広く使用されています。Astra Control CenterをLDAPサーバーに接続して、選択したAstra Controlユーザーの認証を実行できます。

大まかには、AstraとLDAPを統合し、Astra ControlユーザおよびLDAP定義に対応するグループを定義することです。Astra Control APIまたはWeb UIを使用して、LDAP認証とLDAPユーザおよびグループを設定できます。

> Astra Control Centerは、LDAPの「メール」属性でメールアドレスを使用して、リモートユーザーを検索し、追跡します。この属性は、ディレクトリ内のオプションのフィールドまたは空のフィールドです。Astra Control Centerに表示するリモートユーザの場合は、このフィールドにEメールアドレスが存在している必要があります。この電子メールアドレスは、Astra Control Centerのユーザー名として認証に使用されます。

LDAPS認証用の証明書を追加します

LDAPサーバのプライベートTLS証明書を追加して、LDAPS接続を使用する際にAstra Control CenterがLDAPサーバで認証できるようにします。この処理は、1回だけ、またはインストールした証明書の有効期限が切れたときにのみ実行してください。

手順

1. 「アカウント」に移動します。
2. ［証明書］タブを選択します。
3. 「 * 追加」を選択します。
4. をアップロードします `.pem` クリップボードからファイルの内容をファイルまたは貼り付けます。
5. [**Trusted**]チェックボックスをオンにします。
6. [証明書の追加]を選択します。

リモート認証を有効にします

LDAP認証を有効にして、Astra ControlとリモートLDAPサーバ間の接続を設定できます。

必要なもの

LDAPSを使用する場合は、Astra Control CenterがLDAPサーバに対して認証できるように、Astra Control CenterにLDAPサーバのプライベートTLS証明書がインストールされていることを確認してください。を参照してください LDAPS認証用の証明書を追加します 手順については、を参照し

手順

1. 「*アカウント」>「接続」に移動します。
2. [* Remote Authentication （リモート認証）]ペインで、設定メニューを選択します。
3. 「 * 接続」を選択します。
4. サーバのIPアドレス、ポート、および優先接続プロトコル（LDAPまたはLDAPS）を入力します。

> ベストプラクティスとして、LDAPサーバに接続するときはLDAPSを使用してください。LDAPSに接続する前に、LDAPサーバのプライベートTLS証明書をAstra Control Centerにインストールする必要があります。

5. サービスアカウントのクレデンシャルをEメール形式で入力します（administrator@example.com ）。Astra Controlは、LDAPサーバとの接続時にこれらのクレデンシャルを使用します。
6. [ユーザー一致]セクションで、LDAPサーバーからユーザー情報を取得するときに使用するベースDNと適切なユーザー検索フィルタを入力します。
7. ［グループ一致］セクションで、グループ検索ベースDNと適切なカスタムグループ検索フィルタを入力します。

> 正しいベース識別名（DN）と、* User Match および Group Match *の適切な検索フィルタを使用してください。ベースDNは、検索を開始するディレクトリツリーのレベルをAstra Controlに指示し、検索フィルタは、Astra Controlが検索するディレクトリツリーの部分を制限します。

8. [送信] を選択します。

結果

[リモート認証]ペインのステータスは、LDAPサーバーへの接続が確立されると、[保留中]になり、次に[接続済み]になります。

リモート認証を無効にします

LDAPサーバへのアクティブな接続を一時的に無効にすることができます。

> LDAPサーバへの接続を無効にすると、すべての設定が保存され、Astra Controlに追加されたすべてのリモートユーザとリモートグループがそのLDAPサーバから保持されます。このLDAPサーバにいつでも再接続できます。

手順

1. 「*アカウント」>「接続」に移動します。
2. [* Remote Authentication （リモート認証）]ペインで、設定メニューを選択します。
3. [**Disable**] を選択します。

結果

[* Remote Authentication （リモート認証）]ペインのステータスが[* Disabled（無効）]に変わります。すべてのリモート認証設定、リモートユーザ、およびリモートグループが維持され、いつでも接続を再度有効にすることができます。

リモート認証の設定を編集します

LDAPサーバーへの接続を無効にした場合、または*リモート認証*ペインが「接続エラー」状態にある場合は、設定を編集できます。

> 「リモート認証」ペインが「無効」状態の場合、LDAPサーバのURLまたはIPアドレスを編集することはできません。必要です [リモート認証を切断します] 最初に。

手順

1. 「*アカウント」>「接続」に移動します。
2. [* Remote Authentication （リモート認証）]ペインで、設定メニューを選択します。
3. 「 * 編集 * 」を選択します。
4. 必要な変更を行い、* Edit *を選択します。

リモート認証を切断します

LDAPサーバから切断して、Astra Controlから構成設定を削除できます。

LDAPサーバから切断すると、そのLDAPサーバのすべての構成設定がAstra Controlから削除されるだけでなく、そのLDAPサーバから追加されたすべてのリモートユーザとリモートグループも削除されます。

手順

1. 「*アカウント」>「接続」に移動します。
2. [* Remote Authentication （リモート認証）]ペインで、設定メニューを選択します。
3. 「切断」を選択します。

結果

「リモート認証」パネルのステータスが「切断済み」に変わります。リモート認証設定、リモートユーザ、およびリモートグループがAstra Controlから削除される。

リモートユーザとリモートグループを管理します

Astra ControlシステムでLDAP認証を有効にしている場合は、LDAPユーザおよびグループを検索して、承認されたシステムのユーザに含めることができます。

リモートユーザを追加します

アカウント所有者と管理者は、リモートユーザをAstra Controlに追加できます。

同じEメールアドレスのローカルユーザがシステムにすでに存在する場合は、リモートユーザを追加できません。ユーザをリモートユーザとして追加するには、最初にローカルユーザをシステムから削除してください。

Astra Control Centerは、LDAPの「メール」属性でメールアドレスを使用して、リモートユーザーを検索し、追跡します。この属性は、ディレクトリ内のオプションのフィールドまたは空のフィールドです。Astra Control Centerに表示するリモートユーザの場合は、このフィールドにEメールアドレスが存在している必要があります。この電子メールアドレスは、Astra Control Centerのユーザー名として認証に使用されます。

手順

1. [**Account**（アカウント*）]領域に移動します。
2. [*Users & groups]タブを選択します。
3. ページの右端で、*リモートユーザー*を選択します。
4. 「 * 追加」を選択します。
5. 必要に応じて、ユーザのEメールアドレスを* Filter by email *フィールドに入力して、LDAPユーザを検索します。
6. リストから1人以上のユーザを選択します。
7. ユーザにロールを割り当てます。

ユーザとユーザのグループに異なるロールを割り当てると、より権限の高いロールが優先されます。

8. 必要に応じて、このユーザに1つ以上のネームスペースの制約を割り当て、*ロールを制約に制限*を選択して適用します。新しい名前空間制約を追加するには、*制約の追加*を選択します。

> ユーザにLDAPグループメンバーシップを使用して複数のロールを割り当てると、最も権限の高いロールの制約だけが有効になります。たとえば'ローカルビューアロールを持つユーザーがメンバーロールにバインドされた3つのグループを結合すると'メンバーロールからの制約の合計が有効になり'ビューアロールからの制約はすべて無視されます

9. 「* 追加」を選択します。

結果

新しいユーザがリモートユーザのリストに表示されます。このリストでは、ユーザーに対するアクティブな拘束を表示したり、*アクション*メニューからユーザーを管理したりできます。

リモートグループを追加します

複数のリモートユーザを一度に追加するには、アカウント所有者と管理者がリモートグループをAstra Controlに追加します。リモートグループを追加すると、そのグループ内のすべてのリモートユーザがAstra Controlに追加され、同じロールを継承します。

手順

1. [**Account**（アカウント*）]領域に移動します。
2. [*Users & groups]タブを選択します。
3. ページの右端で、*リモートグループ*を選択します。
4. 「* 追加」を選択します。

   このウィンドウには、Astra Controlがディレクトリから取得したLDAPグループの共通名と識別名のリストが表示されます。

5. 必要に応じて、「共通名でフィルタ」フィールドにグループの共通名を入力してLDAPグループを検索します。
6. リストから1つ以上のグループを選択します。
7. グループにロールを割り当てます。

> 選択したロールは、このグループのすべてのユーザに割り当てられます。ユーザとユーザのグループに異なるロールを割り当てると、より権限の高いロールが優先されます。

8. 必要に応じて、このグループに1つ以上の名前空間制約を割り当て、*制約にロールを制限*を選択して適用します。新しい名前空間制約を追加するには、*制約の追加*を選択します。

> ユーザにLDAPグループメンバーシップを使用して複数のロールを割り当てると、最も権限の高いロールの制約だけが有効になります。たとえば'ローカルビューアロールを持つユーザーがメンバーロールにバインドされた3つのグループを結合すると'メンバーロールからの制約の合計が有効になり'ビューアロールからの制約はすべて無視されます

9. 「* 追加」を選択します。

結果

新しいグループがリモートグループのリストに表示され、このグループ内のすべてのリモートユーザがリモートユーザのリストに表示されます。このリストでは、*アクション*メニューからグループの詳細を表示したり、グループを管理したりできます。

通知を表示および管理します

アクションが完了または失敗すると、 Astra から通知が表示されます。たとえば、アプリケーションのバックアップが正常に完了した場合に通知が表示されます。

これらの通知は、インターフェイスの右上から管理できます。

手順

1. 右上の未読通知の数を選択します。
2. 通知を確認し、 [* 既読としてマークする *] または [すべての通知を表示する *] を選択します。

   [すべての通知を表示する *] を選択した場合は、 [通知] ページがロードされます。

3. [* 通知 *] ページで、通知を表示し、既読としてマークする通知を選択し、 [* アクション *] を選択して、 [* 既読としてマークする *] を選択します。

クレデンシャルを追加および削除します

ONTAP S3 、 OpenShift で管理される Kubernetes クラスタ、未管理の Kubernetes クラスタなどのローカルプライベートクラウドプロバイダのクレデンシャルを、お客様のアカウントにいつでも追加、削除できます。Astra Control Center は、これらのクレデンシャルを使用して、クラスタ上の Kubernetes クラスタとアプリケーションを検出し、ユーザに代わってリソースをプロビジョニングします。

Astra Control Center のすべてのユーザーが同じ資格情報セットを共有することに注意してください。

クレデンシャルを追加する

クラスターの管理時に、 Astra Control Center に資格情報を追加できます。新しいクラスタを追加してクレデンシャルを追加する手順については、を参照してください "Kubernetes クラスタを追加"。

> 自分で作成する場合は `kubeconfig` ファイルには、* 1つの*コンテキストエレメントのみを定義する必要があります。を参照してください "Kubernetes のドキュメント" を参照してください `kubeconfig` ファイル。

クレデンシャルを削除する

アカウントからのクレデンシャルの削除はいつでも実行できます。クレデンシャルは、のあとに削除してください "関連するすべてのクラスタの管理を解除します"。

Astra Control Center は、 Astra Control Center の認証情報を使用してバックアップバケットに認証するため、 Astra Control Center に追加する最初の資格情報セットは常に使用されています。これらのクレデンシャルは削除しないことを推奨します。

手順

1. 「 * アカウント * 」を選択します。
2. [*Credentials] タブを選択します。
3. 削除するクレデンシャルの [状態 *] 列で [オプション] メニューを選択します。
4. 「 * 削除」を選択します。
5. 削除を確認するために「削除」と入力し、「はい」、「認証情報を削除」を選択します。

結果

Astra Control Center は、アカウントから資格情報を削除します。

アカウントのアクティビティを監視

Astra Control アカウントのアクティビティの詳細を表示できます。たとえば、新しいユーザを招待したとき、クラスタが追加されたとき、 Snapshot が作成されたときなどです。アカウントアクティビティを CSV ファイルにエクスポートすることもできます。

KubernetesクラスタをAstra Controlから管理し、Astra ControlをCloud Insights に接続した場合、Astra ControlはCloud Insights にイベントログを送信する。ポッドの導入やPVCの添付ファイルに関する情報などのログ情報が、Astra Control Activityログに記録されます。この情報を使用して、管理しているKubernetesクラスタの問題を特定します。

Astra Control のアカウントアクティビティをすべて表示

1. 「 * Activity * 」を選択します。
2. フィルタを使用してアクティビティのリストを絞り込むか、検索ボックスを使用して探しているものを正確に検索します。
3. アカウントアクティビティを CSV ファイルにダウンロードするには、「 * CSV にエクスポート」を選択します。

特定のアプリケーションのアカウントアクティビティを表示します

1. 「 * アプリケーション」を選択し、アプリケーションの名前を選択します。
2. 「 * Activity * 」を選択します。

クラスタのアカウントアクティビティを表示します

1. 「 * クラスタ」を選択し、クラスタの名前を選択します。
2. 「 * Activity * 」を選択します。

対応が必要なイベントを解決するための操作を実行します

1. 「 * Activity * 」を選択します。
2. 注意が必要なイベントを選択してください。

3. [**Take action**] ドロップダウンオプションを選択します。

   このリストから、実行できる対処方法のほか、問題 に関するドキュメントを参照したり、問題 の解決に役立つサポートを受けたりできます。

既存のライセンスを更新する

評価用ライセンスをフルライセンスに変換したり、既存の評価用ライセンスまたはフルライセンスを新しいライセンスで更新したりできます。フルライセンスがない場合は、ネットアップの営業担当者に連絡して、ライセンスとシリアル番号の全文を入手してください。Astra Control Center UIまたはを使用できます "Astra Control API" 既存のライセンスを更新します。

手順

1. にログインします "NetApp Support Site"。
2. Astra Control Center のダウンロードページにアクセスし、シリアル番号を入力して、ネットアップライセンスファイル（ NLF ）をダウンロードする。
3. Astra Control Center UI にログインします。
4. 左側のナビゲーションから、 * アカウント * > * ライセンス * を選択します。
5. [**Account**>*License*] ページで、既存のライセンスのステータスドロップダウンメニューを選択し、[**Replace**] を選択します。
6. ダウンロードしたライセンスファイルを参照します。
7. 「 * 追加」を選択します。

[**Account**>*Licenses*] ページには、ライセンス情報、有効期限、ライセンスシリアル番号、アカウント ID 、および使用されている CPU ユニットが表示されます。

を参照してください。

- "Astra Control Center のライセンス"

バケットを管理する

アプリケーションや永続的ストレージをバックアップする場合や、クラスタ間でアプリケーションをクローニングする場合は、オブジェクトストアバケットプロバイダが不可欠です。Astra Control Center を使用して、オフクラスタのバックアップ先として、アプリケーションのオブジェクトストアプロバイダを追加します。

アプリケーション構成と永続的ストレージを同じクラスタにクローニングする場合、バケットは必要ありません。

次の Amazon Simple Storage Service （ S3 ）バケットプロバイダのいずれかを使用します。

- NetApp ONTAP S3
- NetApp StorageGRID S3 の略

- Microsoft Azure
- 汎用 S3

> Amazon Web Services（AWS）とGoogle Cloud Platform（GCP）では、汎用のS3バケットタイプを使用します。

> Astra Control CenterはAmazon S3を汎用のS3バケットプロバイダとしてサポートしていますが、Astra Control Centerは、AmazonのS3をサポートしていると主張するすべてのオブジェクトストアベンダーをサポートしているわけではありません。

バケットの状態は次のいずれかになります。

- Pending ：バケットの検出がスケジュールされています。
- Available ：バケットは使用可能です。
- Removed ：バケットには現在アクセスできません。

Astra Control API を使用してバケットを管理する方法については、を参照してください "Astra の自動化と API に関する情報"。

バケットの管理に関連して次のタスクを実行できます。

- "バケットを追加します"
- [バケットを編集する]
- [デフォルトバケットを設定する]
- [バケットのクレデンシャルをローテーションするか、削除する]
- [バケットを削除する]

> Astra Control Center の S3 バケットは、使用可能容量を報告しません。Astra Control Center で管理されているアプリケーションのバックアップまたはクローニングを行う前に、 ONTAP または StorageGRID 管理システムでバケット情報を確認します。

バケットを編集する

バケットのアクセスクレデンシャル情報を変更したり、選択したバケットがデフォルトバケットかどうかを変更したりできます。

> バケットを追加するときは、正しいバケットプロバイダを選択し、そのプロバイダに適したクレデンシャルを指定します。たとえば、タイプとして NetApp ONTAP S3 が許可され、StorageGRID クレデンシャルが受け入れられますが、このバケットを使用して原因の以降のアプリケーションのバックアップとリストアはすべて失敗します。を参照してください "リリースノート"。

手順

1. 左側のナビゲーションから、*バケット*を選択します。
2. [アクション（* Actions ）]列のメニューから、[*編集（ Edit）]を選択します。

3. バケットタイプ以外の情報を変更します。

> バケットタイプは変更できません。

4. 「 * Update * 」を選択します。

デフォルトバケットを設定する

クラスタ間でクローニングを実行する場合、Astra Controlにはデフォルトバケットが必要です。すべてのクラスタにデフォルトバケットを設定するには、次の手順を実行します。

手順

1. 「* Cloud Instances *」に移動します。
2. リスト内のクラウドインスタンスの*アクション*列でメニューを選択します。
3. 「 * 編集 * 」を選択します。
4. [* Bucket*]リストで、デフォルトにするバケットを選択します。
5. [保存（ Save ）] を選択します。

バケットのクレデンシャルをローテーションするか、削除する

Astra Controlは、バケットのクレデンシャルを使用してS3バケットにアクセスし、シークレットキーを提供することで、Astra Control Centerがバケットと通信できるようにします。

バケットのクレデンシャルをローテーションする

クレデンシャルのローテーションを行う場合は、バックアップが進行中でないとき（スケジュール設定またはオンデマンド）に、ローテーションを継続して実行してください。

クレデンシャルの編集やローテーションを行う手順

1. 左側のナビゲーションから、*バケット*を選択します。
2. [* アクション * （ * Actions * ）] 列の [オプション（ Options ）] メニューから、 [* 編集（ * Edit ）] を選択する。
3. 新しいクレデンシャルを作成します。
4. 「 * Update * 」を選択します。

バケットのクレデンシャルを削除する

バケットのクレデンシャルを削除するのは、新しいクレデンシャルがバケットに適用されている場合やバケットがアクティブに使用されなくなった場合だけにしてください。

> Astra Control に追加する最初のクレデンシャルセットは、 Astra Control がバックアップバケットの認証にクレデンシャルを使用するため、常に使用されています。バケットがアクティブな状態で使用されている場合は、これらのクレデンシャルを削除しないでください。削除すると、バックアップが失敗してバックアップが使用できなくなります。

アクティブなバケットクレデンシャルを削除する場合は、を参照してください "バケットのクレデンシャル削除のトラブルシューティング"。

Astra Control APIを使用してS3クレデンシャルを削除する方法については、を参照してください "Astra の自動化と API に関する情報"。

バケットを削除する

使用されなくなったバケットや正常でないバケットを削除することができます。これは、オブジェクトストアの設定をシンプルかつ最新の状態に保つために役立ちます。

デフォルトバケットを削除することはできません。そのバケットを削除する場合は、最初に別のバケットをデフォルトとして選択します。

必要なもの

- 開始する前に、このバケットの実行中または完了済みのバックアップがないことを確認してください。
- アクティブな保護ポリシーでバケットが使用されていないことを確認する必要があります。

ある場合は、続行できません。

手順

1. 左ナビゲーションから、 * バケット * を選択します。
2. [アクション * （ Actions * ）] メニューから、 [* 削除（ Remove ）] を選択します。

   Astra Control を使用すると、最初にバケットを使用してバックアップを実行するスケジュールポリシーが存在せず、削除しようとしているバケットにアクティブなバックアップが存在しないようにすることができます。

3. 「 remove 」と入力して操作を確認します。
4. 「 * Yes 、 remove bucket * 」を選択します。

詳細については、こちらをご覧ください

- "Astra Control API を使用"

ストレージバックエンドを管理します

ストレージバックエンドとして Astra Control のストレージクラスタを管理することで、永続ボリューム（ PVS ）とストレージバックエンドの間のリンケージを取得できるだけでなく、追加のストレージ指標も取得できます。ストレージ容量と健全性の詳細を監視できます。 Astra Control Center が Cloud Insights に接続されている場合のパフォーマンスも監視できます。

Astra Control API を使用してストレージバックエンドを管理する方法については、を参照してください "Astra の自動化と API に関する情報"。

ストレージバックエンドの管理に関連して、次のタスクを実行できます。

- "ストレージバックエンドを追加します"
- [ストレージバックエンドの詳細を表示します]
- [ストレージバックエンドの管理を解除します]
- [ストレージバックエンドを削除します]

ストレージバックエンドの詳細を表示します

ストレージバックエンドの情報は、ダッシュボードまたはバックエンドオプションで確認できます。

ダッシュボードでストレージバックエンドの詳細を確認します

手順

1. 左側のナビゲーションから、 * ダッシュボード * を選択します。
2. ダッシュボードのストレージバックエンドパネルで状態を確認します。
    - * 正常でない * ：ストレージが最適な状態ではありません。これは、レイテンシの問題やコンテナの問題が原因でアプリケーションがデグレードした場合などに発生します。
    - * すべて正常 * ：ストレージは管理されており、最適な状態です。
    - * 検出 * ：ストレージは検出されましたが、 Astra Control では管理されていません。

バックエンドからストレージバックエンドの詳細を表示するオプションを選択します

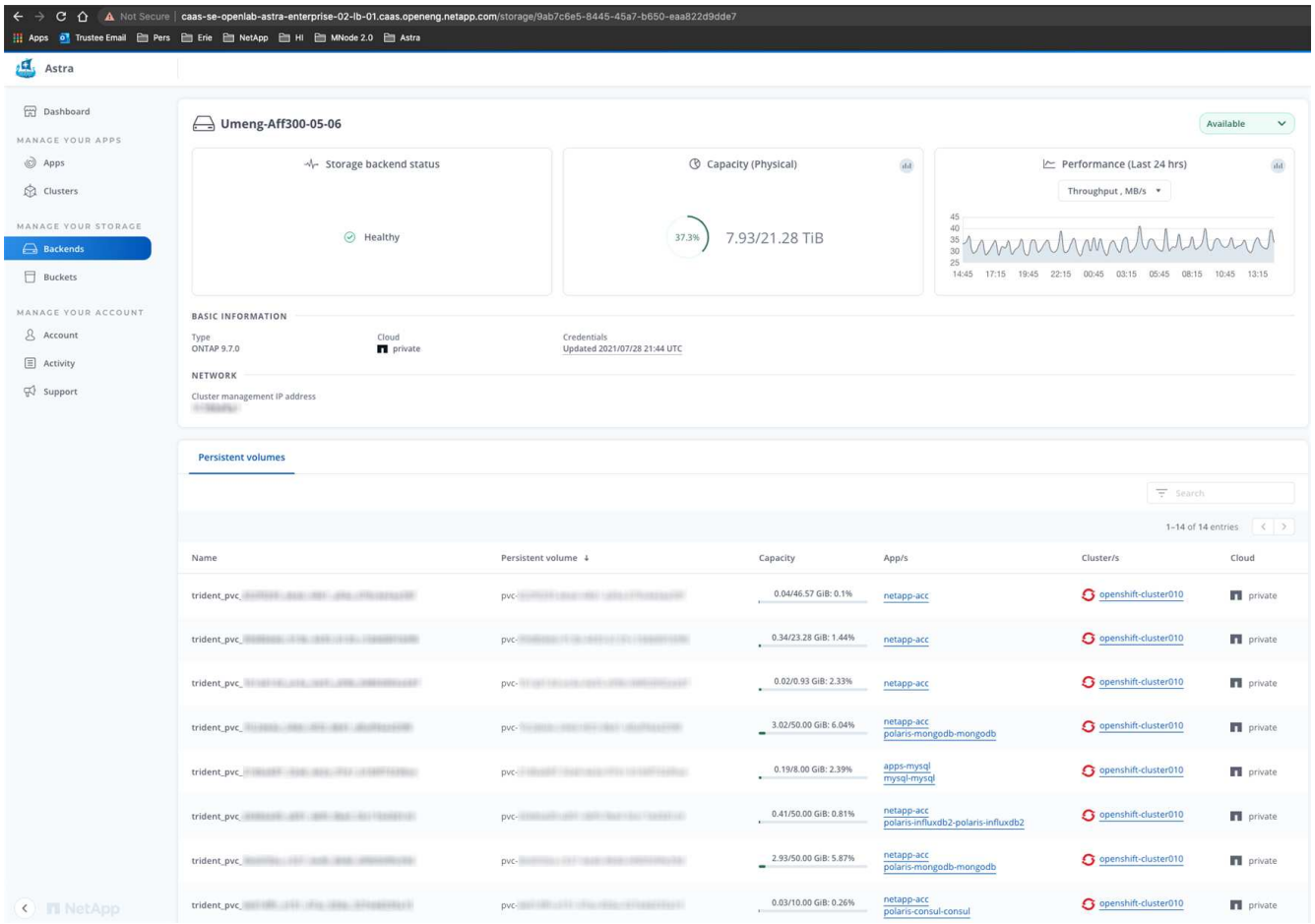
バックエンドの健常性、容量、パフォーマンス（ IOPS スループット、レイテンシ）に関する情報を表示します。

Kubernetesアプリケーションが使用しているボリュームが表示され、選択したストレージバックエンドに格納されます。Cloud Insights を使用すると、追加情報 が表示されます。を参照してください "Cloud Insights のドキュメント"。

手順

1. 左側のナビゲーション領域で、 * Backends * を選択します。
2. ストレージバックエンドを選択します。

NetApp Cloud Insights に接続した場合、 Cloud Insights からの抜粋がバックエンドのページに表示されます。



3. Cloud Insights に直接移動するには、指標画像の横にある * Cloud Insights * アイコンを選択します。

ストレージバックエンドの管理を解除します

バックエンドの管理を解除できます。

手順

1. 左のナビゲーションから、 * Backends * を選択します。
2. ストレージバックエンドを選択します。
3. * アクション * 列のオプションメニューから、 * 管理解除 * を選択します。
4. 「 unmanage 」と入力して操作を確定します。
5. 「 * Yes 、 unmanage storage backend * 」を選択します。

ストレージバックエンドを削除します

使用されなくなったストレージバックエンドを削除できます。これは、設定をシンプルかつ最新の状態に保つために役立ちます。

必要なもの

- ストレージバックエンドが管理対象外であることを確認します。
- ストレージバックエンドにクラスタに関連付けられたボリュームがないことを確認します。

手順

1. 左ナビゲーションから、 * Backends * を選択します。
2. バックエンドが管理されている場合は、管理を解除します。
   a. [*Managed] を選択します。
   b. ストレージバックエンドを選択します。
   c. [* アクション * （ * Actions * ）] オプションから、 [* アンマネージ * （ * Unmanage * ）] を
   d. 「 unmanage 」と入力して操作を確定します。
   e. 「 * Yes 、 unmanage storage backend * 」を選択します。
3. [* Discovered （検出済み）] を選択
   a. ストレージバックエンドを選択します。
   b. [* アクション * （ * Actions * ）] オプションから、 [* 削除（ * Remove ）] を選択する。
   c. 「 remove 」と入力して操作を確認します。
   d. 「 * Yes 、 remove storage backend * 」を選択します。

詳細については、こちらをご覧ください

- "Astra Control API を使用"

実行中のタスクを監視します

Astra Controlの過去24時間に完了、失敗、またはキャンセルされた実行中のタスクとタスクの詳細を表示できます。たとえば、実行中のバックアップ、リストア、またはクローン処理のステータスを表示して、完了した割合や推定残り時間などの詳細を確認できます。実行済みのスケジュール済み処理または手動で開始した処理のステータスを表示できます。

実行中または完了済みのタスクを表示している間に、タスクの詳細を展開して、各サブタスクのステータスを確認できます。進行中のタスクまたは完了したタスクの場合はタスクの進捗状況バーが緑色で表示され、キャンセルされたタスクの場合は青色で表示され、エラーのために失敗したタスクの場合は赤色で表示されます。

> クローン処理の場合、タスクサブタスクはSnapshotとSnapshotのリストア処理で構成されます。

失敗したタスクの詳細については、を参照してください "アカウントのアクティビティを監視"。

手順

1. タスクの実行中に、「アプリケーション」に移動します。
2. リストからアプリケーションの名前を選択します。
3. アプリケーションの詳細で、[タスク]タブを選択します。

   現在または過去のタスクの詳細を表示したり、タスクの状態でフィルタリングしたりできます。

タスクは最大24時間、*タスク*リストに保存されます。を使用して、この制限とその他のタスクモニタの設定を行うことができます "Astra Control API の略"。

Cloud Insights 、Prometheus、Fluentd接続でインフラを監視します

複数のオプション設定を構成して、 Astra Control Center の操作性を高めることができます。インフラ全体を監視して詳細を把握するには、NetApp Cloud Insights への接続を作成し、Prometheusを設定するか、Fluentd接続を追加します。

Astra Control Centerを実行しているネットワークで、インターネットに接続するためのプロキシが必要な場合（サポートバンドルをNetApp Support Site にアップロードする場合、またはCloud Insights への接続を確立する場合）は、Astra Control Centerでプロキシサーバを設定する必要があります。

- Cloud Insights に接続します
- Prometheusに接続
- Fluentd に接続します

Cloud Insights またはNetApp Support Site への接続に使用するプロキシサーバを追加します

Astra Control Centerを実行しているネットワークで、インターネットに接続するためのプロキシが必要な場合（サポートバンドルをNetApp Support Site にアップロードする場合、またはCloud Insights への接続を確立する場合）は、Astra Control Centerでプロキシサーバを設定する必要があります。

Astra Control Center は、プロキシサーバー用に入力した詳細を検証しません。正しい値を入力してください。

手順

1. * admin * / * owner * 権限を持つアカウントを使用して Astra Control Center にログインします。
2. [**Account**>*Connections*] を選択します。
3. ドロップダウンリストから [**Connect**] を選択して、プロキシサーバを追加します。

4. プロキシサーバの名前または IP アドレスとプロキシポート番号を入力します。
5. プロキシサーバで認証が必要な場合は、このチェックボックスをオンにしてユーザ名とパスワードを入力します。
6. 「 * 接続」を選択します。

結果

入力したプロキシ情報が保存されている場合は、 **Account**>*Connections* ページの **HTTP Proxy** セクションに、接続されていることが示され、サーバー名が表示されます。

プロキシサーバの設定を編集します

プロキシサーバの設定を編集できます。

手順

1. * admin * / * owner * 権限を持つアカウントを使用して Astra Control Center にログインします。
2. [**Account**>*Connections*] を選択します。
3. ドロップダウンリストから * Edit * を選択して、接続を編集します。
4. サーバの詳細と認証情報を編集します。
5. [保存（ Save ）] を選択します。

プロキシサーバ接続を無効にします

プロキシサーバ接続を無効にすることができます。他の接続が中断される可能性があることを無効にする前に警告が表示されます。

手順

1. * admin * / * owner * 権限を持つアカウントを使用して Astra Control Center にログインします。
2. [**Account**>*Connections*] を選択します。
3. 接続を無効にするには、ドロップダウンリストから * 切断 * を選択します。
4. 表示されたダイアログボックスで、処理を確認します。

Cloud Insights に接続します

NetApp Cloud Insights を Astra Control Center インスタンスに接続すると、インフラ全体を監視して詳細に把握できます。Cloud Insights は、 Astra Control Center ライセンスに含まれています。

Cloud Insights には、 Astra Control Center が使用するネットワークから、またはプロキシサーバー経由で間接的にアクセスできる必要があります。

Cloud Insights にアストラコントロールセンターを接続すると、 Acquisition Unit ポッドが作成されます。このポッドは、 Astra Control Center で管理されているストレージバックエンドからデータを収集し、 Cloud Insights にプッシュします。このポッドには、 8GB の RAM と 2 つの CPU コアが必要です。

Cloud Insights 接続を有効にすると、スループット情報をバックエンド * ページで確認できるほか、ストレージバックエンドを選択したあとにここから Cloud Insights に接続できます。ダッシュボード * の情報はクラスタセクションでも確認できます。また、そこから Cloud Insights に接続できます。

必要なもの

- admin * / * owner * 権限を持つ Astra Control Center アカウント。
- 有効な Astra Control Center ライセンス。
- Astra Control Center を実行しているネットワークで、インターネットに接続するためにプロキシが必要な場合は、プロキシサーバーです。

Cloud Insights を初めて使用する場合は、の機能について理解しておいてください。を参照してください "Cloud Insights のドキュメント"。

手順

1. * admin * / * owner * 権限を持つアカウントを使用して Astra Control Center にログインします。
2. [**Account**>*Connections*] を選択します。
3. 接続を追加するには、ドロップダウンリストで * 切断されている * と表示されている * 接続 * を選択します。

オプションを表示して、Cloud Insights 接続を有効にします。"]

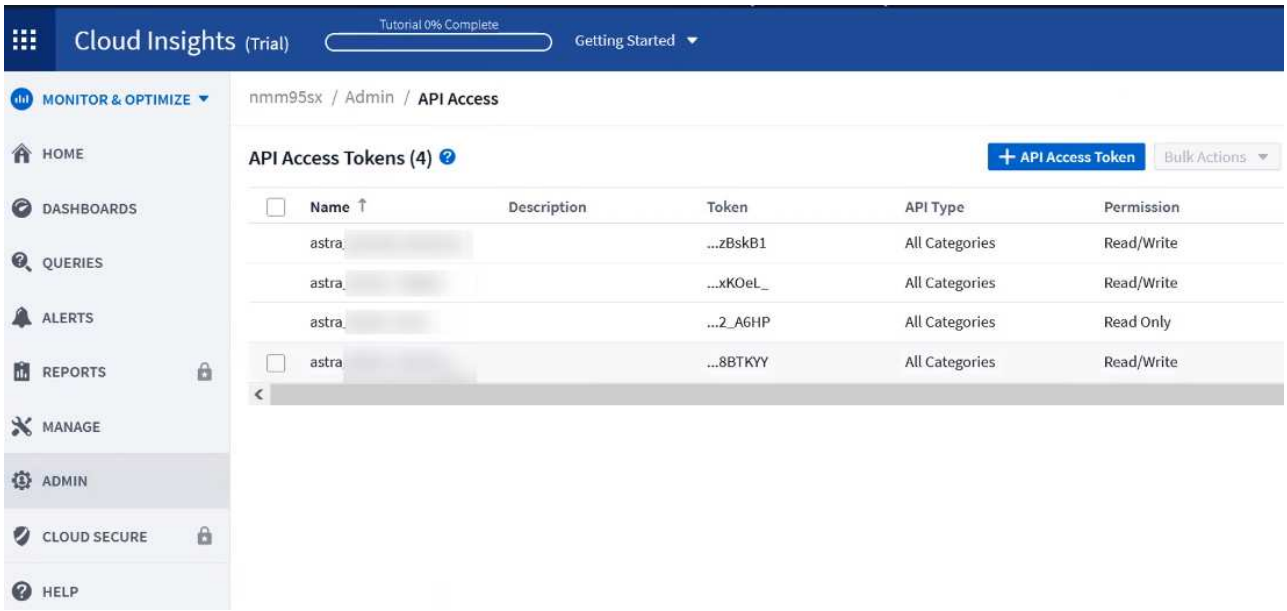
4. Cloud Insights API トークンとテナント URL を入力します。テナント URL の形式は次のようになります。

```
https://<environment-name>.c01.cloudinsights.netapp.com/
```

テナント URL は、 Cloud Insights ライセンスを取得すると取得されます。テナント URL がない場合は、を参照してください "Cloud Insights のドキュメント"。

a. をダウンロードしてください "API トークン"をクリックし、 Cloud Insights テナントの URL にログインします。

b. Cloud Insights で、 * Admin*>* API Access* をクリックして、 * Read/Write * と * Read Only* API Access トークンの両方を生成します。



c. 「 * Read Only * 」キーをコピーします。Cloud Insights 接続を有効にするには、 [Astra Control Center] ウィンドウに貼り付ける必要があります。Read API Access Token Key 権限で、 Assets 、 Alerts 、 Acquisition Unit 、 and Data Collection を選択します。

d. 「 * Read/Write 」キーをコピーします。Astra Control Center * Connect Cloud Insights * ウィンドウに貼り付ける必要があります。Read/Write API Access Tokenキーの権限で、Data Ingestion、Log Ingestion、Acquisition Unit、およびData Collectionを選択します。

> * 読み取り専用 * キーと * 読み取り / 書き込み * キーを生成することを推奨します。両方の目的で同じキーを使用することは推奨しません。デフォルトでは、トークンの有効期限は 1 年に設定されています。トークンが期限切れになるまでの最大期間を指定するために、デフォルトの選択を維持することをお勧めします。トークンの有効期限が切れると、テレメトリが停止します。

e. Cloud Insights からコピーしたキーを Astra コントロールセンターに貼り付けます。

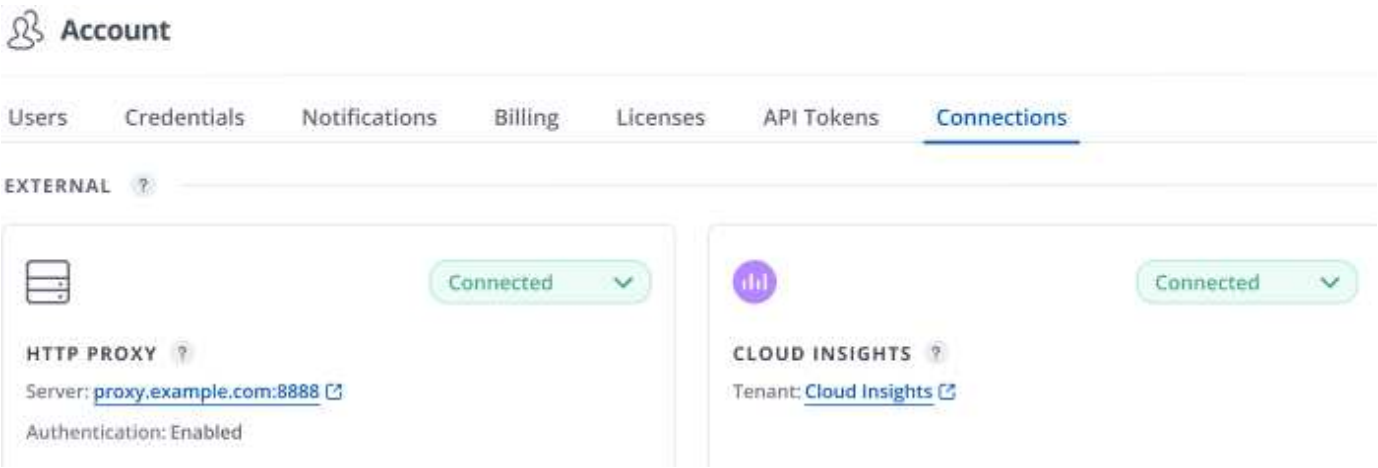
5. 「 * 接続」を選択します。

> [* 接続] を選択すると、 [* アカウント * > * 接続 *] ページの [* Cloud Insights *] セクションで、接続の状態が [* 保留中] に変わります。接続が有効になり、ステータスが * 接続済み * に変わるまで数分かかることがあります。

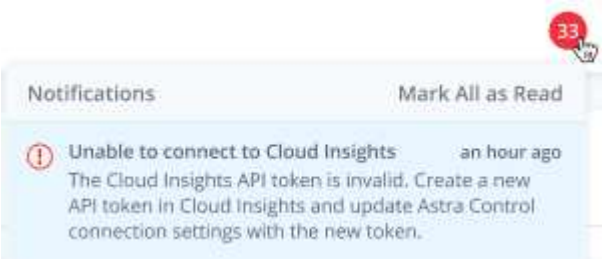
> Astra Control Center と Cloud Insights UI の間を簡単に行き来するには、両方にログインしていることを確認します。

Cloud Insights でデータを表示します

接続に成功した場合は、「 * アカウント * > * 接続 * 」ページの「 * Cloud Insights * 」セクションに接続されていることが示され、テナントの URL が表示されます。Cloud Insights にアクセスして、データが正常に受信されて表示されることを確認できます。

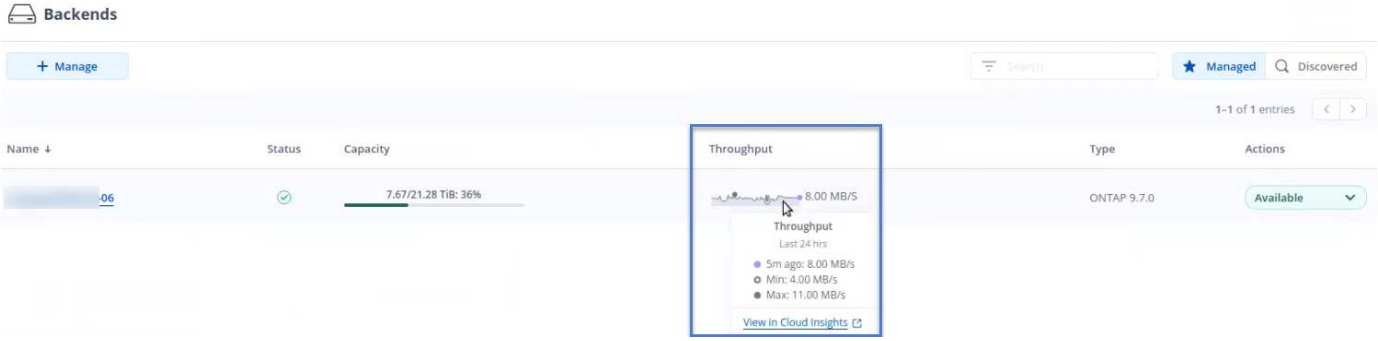


何らかの理由で接続に失敗した場合、ステータスは「 * 失敗 * 」と表示されます。失敗の理由は、 UI の右上にある * Notifications * で確認できます。



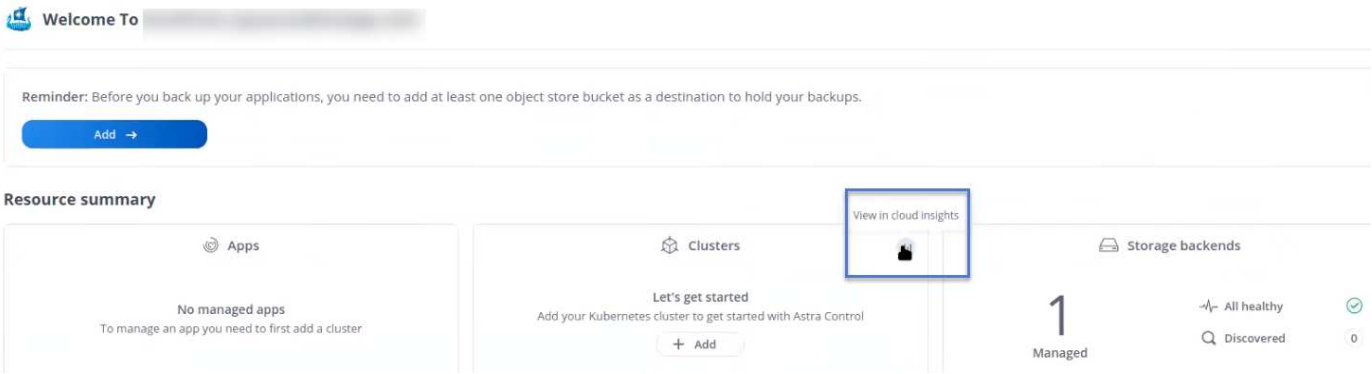
同じ情報は、「 * アカウント * > * 通知 * 」にも記載されています。

Astra Control Center では、スループット情報をバックエンド * ページで表示したり、ストレージバックエンドを選択した後にここから Cloud Insights に接続したりできます。



Cloud Insights に直接移動するには、指標画像の横にある * Cloud Insights * アイコンを選択します。

また、情報は * ダッシュボード * でも確認できます。



Cloud Insights 接続を有効にした後、 Astra Control Center に追加したバックエンドを削除すると、バックエンドは Cloud Insights へのレポートを停止します。

Cloud Insights 接続を編集します

Cloud Insights 接続を編集できます。

編集できるのは API キーのみです。Cloud Insights テナント URL を変更するには、 Cloud Insights 接続を切断して新しい URL に接続することを推奨します。

手順

1. * admin * / * owner * 権限を持つアカウントを使用して Astra Control Center にログインします。
2. [**Account**>*Connections*] を選択します。
3. ドロップダウンリストから * Edit * を選択して、接続を編集します。
4. Cloud Insights 接続設定を編集します。
5. [保存（ Save ）] を選択します。

Cloud Insights 接続を無効にします

Cloud Insights 接続は、 Astra Control Center で管理されている Kubernetes クラスタに対して無効にすることができます。Cloud Insights 接続を無効にしても、すでに Cloud Insights にアップロードされている計測データは削除されません。

手順

1. * admin * / * owner * 権限を持つアカウントを使用して Astra Control Center にログインします。
2. [**Account**>*Connections*] を選択します。
3. 接続を無効にするには、ドロップダウンリストから * 切断 * を選択します。
4. 表示されたダイアログボックスで、処理を確認します。操作を確定すると、 [**Account**>*Connections*] ページで、 Cloud Insights のステータスが [*Pending （保留中）] に変わります。ステータスが * 切断された * に変わるまで数分かかります。

Prometheusに接続

Prometheusを使用して、Astra Control Centerのデータを監視できます。Kubernetesクラスタの指標エンドポイントから指標を収集するようにPrometheusを設定したり、Prometheusを使用して指標データを表示したり

することもできます。

Prometheusの使用の詳細については、でそれぞれのドキュメントを参照してください "Prometheusでの作業の開始"。

必要なもの

PrometheusパッケージがAstra Control Centerクラスタ、またはAstra Control Centerクラスタと通信可能な別のクラスタにダウンロードしてインストールされていることを確認します。

の公式ドキュメントに記載されている手順に従ってください "Prometheus をインストールする"。

Prometheusは、Astra Control Center Kubernetesクラスタと通信できる必要があります。PrometheusがAstra Control Centerクラスタにインストールされていない場合は、Astra Control Centerクラスタで実行されている指標サービスと通信できることを確認する必要があります。

Prometheus を設定する

Astra Control Centerは、KubernetesクラスタのTCPポート9090で指標サービスを公開します。このサービスから指標を収集するには、 Prometheus を設定する必要があります。

手順

1. Prometheusサーバにログインします。
2. にクラスタエントリを追加します `prometheus.yml` ファイル。を参照してください `yml` ファイルで、クラスタに関する次のようなエントリをに追加します `scrape_configs section`：

```
job_name: '<Add your cluster name here. You can abbreviate. It just
needs to be a unique name>'
  metrics_path: /accounts/<replace with your account ID>/metrics
  authorization:
     credentials: <replace with your API token>
  tls_config:
     insecure_skip_verify: true
  static_configs:
    - targets: ['<replace with your astraAddress. If using FQDN, the
prometheus server has to be able to resolve it>']
```

> を設定した場合は `tls_config insecure_skip_verify` 終了： `true`では、TLS暗号化プロトコルは必要ありません。

3. Prometheusサービスを再起動します。

```
sudo systemctl restart prometheus
```

Prometheusにアクセスする

PrometheusのURLにアクセスします。

手順

1. ブラウザで、Prometheus URLをポート9090と入力します。
2. * Status **>** Targets *を選択して、接続を確認します。

Prometheusでデータを表示する

Prometheusを使用してAstra Control Centerのデータを表示できます。

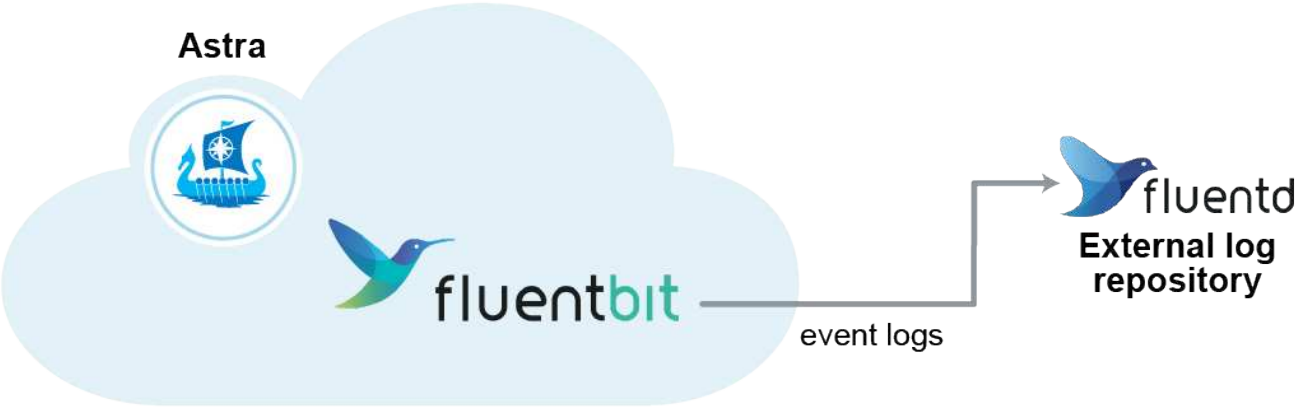
手順

1. ブラウザで、PrometheusのURLを入力します。
2. Prometheusメニューで* Graph *を選択します。
3. メトリクスエクスプローラを使用するには、[**Execute**]の横にあるアイコンを選択します。
4. 選択するオプション `scrape_samples_scraped` をクリックし、* Execute *を選択します。
5. 時間の経過に伴うサンプルのスクレイピングを確認するには、* Graph *を選択します。

> 複数のクラスタデータが収集された場合、各クラスタの指標は異なる色で表示されます。

Fluentd に接続します

Astra Control Centerによって監視されているシステムからFluentdエンドポイントにログ（Kubernetesイベント）を送信できます。Fluentd 接続はデフォルトで無効になっています。



> 管理対象クラスタのイベントログのみが Fluentd に転送されます。

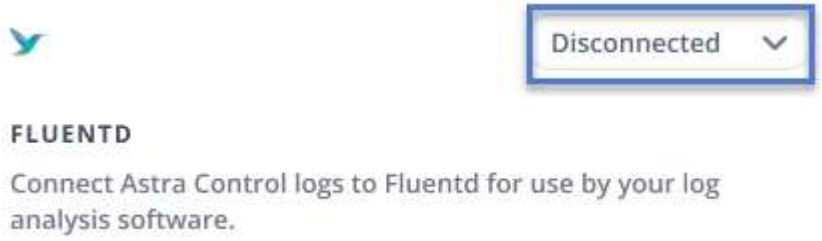
必要なもの

- admin * / * owner * 権限を持つ Astra Control Center アカウント。
- Kubernetes クラスタに Astra Control Center をインストールして実行

> Astra Control Center では、 Fluentd サーバーに入力した詳細は検証されません。必ず正しい値を入力してください。

手順

1. * admin * / * owner * 権限を持つアカウントを使用して Astra Control Center にログインします。
2. [**Account**>*Connections*] を選択します。
3. 接続を追加するには、ドロップダウンリストから [* 接続（ * Connect * ）] を選択します。



4. Fluentd サーバーのホスト IP アドレス、ポート番号、および共有キーを入力します。
5. 「 * 接続」を選択します。

結果

Fluentd サーバーに入力した詳細が保存されている場合は、 * アカウント * > * 接続 * ページの * Fluentd * セクションに接続されていることが示されます。これで、接続した Fluentd サーバーにアクセスし、イベントログを表示できます。

何らかの理由で接続に失敗した場合、ステータスは「 * 失敗 * 」と表示されます。失敗の理由は、 UI の右上にある * Notifications * で確認できます。

同じ情報は、「 * アカウント * > * 通知 * 」にも記載されています。

> ログ収集に問題がある場合は、ワーカーノードにログインして、ログがにあることを確認する必要があります `/var/log/containers/`。

Fluentd 接続を編集します

Fluentd 接続を Astra Control Center インスタンスに編集できます。

手順

1. * admin * / * owner * 権限を持つアカウントを使用して Astra Control Center にログインします。
2. [**Account**>*Connections*] を選択します。
3. ドロップダウンリストから * Edit * を選択して、接続を編集します。
4. Fluentd エンドポイントの設定を変更します。
5. [保存（ Save ）] を選択します。

Fluentd 接続を無効にします

Astra Control Center インスタンスへの Fluentd 接続を無効にできます。

手順

1. * admin * / * owner * 権限を持つアカウントを使用して Astra Control Center にログインします。

2. [**Account**>*Connections*] を選択します。
3. 接続を無効にするには、ドロップダウンリストから * 切断 * を選択します。
4. 表示されたダイアログボックスで、処理を確認します。

アプリケーションとクラスタの管理を解除します

管理する必要がなくなったアプリケーションやクラスタを Astra Control Center から削除します。

アプリの管理を解除します

バックアップ、スナップショット、またはクローンを作成する必要がなくなったアプリケーションの管理を Astra Control Center から停止します。

アプリケーションの管理を解除すると、次のようになります。

- 既存のバックアップと Snapshot がすべて削除されます。
- アプリケーションとデータは引き続き使用できます。

手順

1. 左側のナビゲーションバーから、「 * アプリケーション * 」を選択します。
2. アプリケーションを選択します。
3. [アクション]列の[オプション]メニューから、*Unmanage*を選択します。
4. 情報を確認します。
5. 「 unmanage 」と入力して確定します。
6. 「はい、アプリケーションの管理を解除」を選択します。

結果

Astra Control Center がアプリケーションの管理を停止。

クラスタの管理を解除します

管理する必要がなくなったクラスタをAstra Control Centerから管理しないようにします。

> クラスタの管理を解除する前に、クラスタに関連付けられているアプリケーションの管理を解除する必要があります。

クラスタの管理を解除する場合：

- この処理を実行すると、 Astra Control Center でクラスタが管理されなくなります。クラスタの構成は変更されず、クラスタも削除されません。
- Trident はクラスタからアンインストールされません。 "Trident のアンインストール方法をご確認ください"。

手順

1. 左側のナビゲーションバーから、 * クラスタ * を選択します。
2. 管理する必要がなくなったクラスタのチェックボックスを選択します。
3. * アクション * 列のオプションメニューから、 * 管理解除 * を選択します。
4. クラスタの管理を解除することを確認し、「 * Yes 、 unmanage cluster * 」を選択します。

結果

クラスタのステータスが「Removing *」に変わります。その後、クラスタが「*クラスタ」ページから削除され、Astra Control Centerによって管理されなくなります。

> * Astra Control Center と Cloud Insights が接続されていない場合 * 、クラスタの管理を解除すると、テレメトリ・データの送信用にインストールされたすべてのリソースが削除されます。* Astra Control CenterとCloud Insights が接続されている場合*、クラスタの管理を解除すると、のみが削除されます `fluentbit` および `event-exporter` ポッド

Astra Control Center をアップグレードします

Astra Control Centerをアップグレードするには、NetApp Support Site からインストールバンドルをダウンロードし、以下の手順を実行します。この手順を使用して、インターネット接続環境またはエアギャップ環境の Astra コントロールセンターをアップグレードできます。

必要なもの

- アップグレードする前に、を参照してください "運用環境の要件" 環境がAstra Control Center導入の最小要件を満たしていることを確認する。環境に次の要素が必要です。
  - サポートされているAstra Tridentバージョンで、実行しているバージョンを確認するには、既存のAstra Control Centerに対して次のコマンドを実行します。

    ```
    kubectl get tridentversion -n trident
    ```

    を参照してください "Astra Trident のドキュメント" 古いバージョンからアップグレードするには、次の手順に従います。

    > Kubernetes 1.25にアップグレードするには、Astra Trident 22.10 *にアップグレードする必要があります。

  - サポートされているKubernetesディストリビューションで実行しているバージョンを確認するには、既存のAstra Control Centerに対して次のコマンドを実行します。 `kubectl get nodes -o wide`
  - 十分なクラスタリソースを使用してクラスタリソースを確認し、既存のAstra Control Centerクラスタで次のコマンドを実行します。 `kubectl describe node <node name>`
  - Astra Control Centerイメージのプッシュおよびアップロードに使用できるレジストリ
  - デフォルトのストレージクラスデフォルトのストレージクラスを決定するには、既存のアストラコントロールセンターに対して次のコマンドを実行します。 `kubectl get storageclass`

- （OpenShiftのみ）すべてのクラスタオペレータが正常な状態であり、使用可能であることを確認します。

```
kubectl get clusteroperators
```

- すべての API サービスが正常な状態であり、使用可能であることを確認します。

```
kubectl get apiservices
```

- アップグレードを開始する前に、Astra Control Center UIからログアウトします。

このタスクについて

Astra Control Center のアップグレードプロセスでは、次の手順を実行できます。

- Astra Control Centerをダウンロードして展開します
- NetApp Astra kubectlプラグインを削除して、再度インストールします
- [イメージをローカルレジストリに追加します]
- 更新された Astra Control Center オペレータをインストールします
- Astra Control Center をアップグレードします
- [システムステータスを確認します]

> Astra Control Centerオペレータ（たとえば、 `kubectl delete -f astra_control_center_operator_deploy.yaml`）Astra Control Centerのアップグレードまたは操作中はいつでもポッドを削除しないようにします。

> スケジュール、バックアップ、 Snapshot が実行されていないときは、メンテナンス時間内にアップグレードを実行します。

Astra Control Centerをダウンロードして展開します

1. にアクセスします "Astra Control Centerの製品ダウンロードページ" をクリックしますNetApp Support Site 。ドロップダウンメニューから最新バージョンまたは別のバージョンを選択できます。
2. Astra Control Centerを含むバンドルをダウンロードします (`astra-control-center-[version].tar.gz`）。
3. （推奨ですがオプション）Astra Control Centerの証明書と署名のバンドルをダウンロードします (`astra-control-center-certs-[version].tar.gz`）バンドルの署名を確認するには、次の手順を実行します。

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

```
openssl dgst -sha256 -verify certs/AstraControlCenter-public.pub
-signature certs/astra-control-center-[version].tar.gz.sig astra-
control-center-[version].tar.gz
```

出力にはと表示されます `Verified OK` 検証が成功したあとに、

4. Astra Control Centerバンドルからイメージを抽出します。

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

NetApp Astra kubectlプラグインを削除して、再度インストールします

NetApp Astra kubectlコマンドラインプラグインは、Astra Control Centerの導入とアップグレードに関連する一般的なタスクを実行する際に時間を節約します。

1. プラグインがインストールされているかどうかを確認します。

```
kubectl astra
```

2. 次のいずれかを実行します。
   - プラグインがインストールされている場合、コマンドはkubectlプラグインのヘルプを返す必要があります。既存のバージョンのkubectl-mstraを削除するには、次のコマンドを実行します。 `delete /usr/local/bin/kubectl-astra`。
   - コマンドからエラーが返された場合は、プラグインがインストールされていません。次の手順に進んでインストールしてください。
3. プラグインをインストールします。
   a. 使用可能なNetApp Astra kubectlプラグインのバイナリを表示し、オペレーティングシステムとCPUアーキテクチャに必要なファイルの名前をメモします。

> kubectlプラグインライブラリはtarバンドルの一部であり、フォルダに解凍されます `kubectl-astra`。

```
ls kubectl-astra/
```

   a. 正しいバイナリを現在のパスに移動し、名前をに変更します `kubectl-astra`：

```
cp kubectl-astra/<binary-name> /usr/local/bin/kubectl-astra
```

イメージをローカルレジストリに追加します

1. コンテナエンジンに応じた手順を実行します。

Docker です

1. tarballのルートディレクトリに移動します。次のファイルとディレクトリが表示されます。

   ```
   acc.manifest.bundle.yaml
   acc/
   ```

2. Astra Control Centerのイメージディレクトリにあるパッケージイメージをローカルレジストリにプッシュします。を実行する前に、次の置換を行ってください `push-images` コマンドを実行します
   - <BUNDLE_FILE> をAstra Controlバンドルファイルの名前に置き換えます（`acc.manifest.bundle.yaml`）。
   - <MY_FULL_REGISTRY_PATH> をDockerリポジトリのURLに置き換えます。次に例を示します。 "<a href="https://<docker-registry>"" class="bare">https://<docker-registry>"。
   - <MY_REGISTRY_USER> をユーザ名に置き換えます。
   - <MY_REGISTRY_TOKEN> をレジストリの認証済みトークンに置き換えます。

     ```
     kubectl astra packages push-images -m <BUNDLE_FILE> -r
     <MY_FULL_REGISTRY_PATH> -u <MY_REGISTRY_USER> -p
     <MY_REGISTRY_TOKEN>
     ```

ポドマン

1. tarballのルートディレクトリに移動します。次のファイルとディレクトリが表示されます。

   ```
   acc.manifest.bundle.yaml
   acc/
   ```

2. レジストリにログインします。

   ```
   podman login <YOUR_REGISTRY>
   ```

3. 使用するPodmanのバージョンに合わせてカスタマイズされた次のいずれかのスクリプトを準備して実行します。<MY_FULL_REGISTRY_PATH> を’サブディレクトリを含むリポジトリのURLに置き換えます

   ```
   <strong>Podman 4</strong>
   ```

```
export REGISTRY=<MY_FULL_REGISTRY_PATH>
export PACKAGENAME=acc
export PACKAGEVERSION=22.11.0-82
export DIRECTORYNAME=acc
for astraImageFile in $(ls ${DIRECTORYNAME}/images/*.tar) ; do
astraImage=$(podman load --input ${astraImageFile} | sed 's/Loaded
image: //')
astraImageNoPath=$(echo ${astraImage} | sed 's:.*/::')
podman tag ${astraImageNoPath} ${REGISTRY}/netapp/astra/
${PACKAGENAME}/${PACKAGEVERSION}/${astraImageNoPath}
podman push ${REGISTRY}/netapp/astra/${PACKAGENAME}/${
PACKAGEVERSION}/${astraImageNoPath}
done
```

```
<strong>Podman 3</strong>
```

```
export REGISTRY=<MY_FULL_REGISTRY_PATH>
export PACKAGENAME=acc
export PACKAGEVERSION=22.11.0-82
export DIRECTORYNAME=acc
for astraImageFile in $(ls ${DIRECTORYNAME}/images/*.tar) ; do
astraImage=$(podman load --input ${astraImageFile} | sed 's/Loaded
image: //')
astraImageNoPath=$(echo ${astraImage} | sed 's:.*/::')
podman tag ${astraImageNoPath} ${REGISTRY}/netapp/astra/
${PACKAGENAME}/${PACKAGEVERSION}/${astraImageNoPath}
podman push ${REGISTRY}/netapp/astra/${PACKAGENAME}/${
PACKAGEVERSION}/${astraImageNoPath}
done
```

> レジストリ設定に応じて、スクリプトが作成するイメージパスは次のようになります。 https://netappdownloads.jfrog.io/docker-astra-control-prod/netapp/astra/acc/22.11.0-82/image:version

更新された **Astra Control Center** オペレータをインストールします

1. ディレクトリを変更します。

```
cd manifests
```

2. Astra Control Centerオペレータ配置YAMLを編集します (`astra_control_center_operator_deploy.yaml`)を参照して、ローカルレジストリとシークレットを参照してください。

```
vim astra_control_center_operator_deploy.yaml
```

a. 認証が必要なレジストリを使用する場合は、のデフォルト行を置換または編集します `imagePullSecrets: []` 次の条件を満たす場合：

```
imagePullSecrets:
- name: <astra-registry-cred_or_custom_name_of_secret>
```

b. 変更 `[your_registry_path]` をクリックします `kube-rbac-proxy` でイメージをプッシュしたレジストリパスへのイメージ 前の手順。

c. 変更 `[your_registry_path]` をクリックします `acc-operator` でイメージをプッシュしたレジストリパスへのイメージ 前の手順。

d. に次の値を追加します `env` セクション。

```
- name: ACCOP_HELM_UPGRADETIMEOUT
  value: 300m
```

```
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    control-plane: controller-manager
  name: acc-operator-controller-manager
  namespace: netapp-acc-operator
spec:
  replicas: 1
  selector:
    matchLabels:
      control-plane: controller-manager
  strategy:
    type: Recreate
  template:
    metadata:
      labels:
        control-plane: controller-manager
    spec:
      containers:
      - args:
```

```
        - --secure-listen-address=0.0.0.0:8443
        - --upstream=http://127.0.0.1:8080/
        - --logtostderr=true
        - --v=10
        image: [your_registry_path]/kube-rbac-proxy:v4.8.0
        name: kube-rbac-proxy
        ports:
        - containerPort: 8443
          name: https
      - args:
        - --health-probe-bind-address=:8081
        - --metrics-bind-address=127.0.0.1:8080
        - --leader-elect
        env:
        - name: ACCOP_LOG_LEVEL
          value: "2"
        - name: ACCOP_HELM_UPGRADETIMEOUT
          value: 300m
        image: [your_registry_path]/acc-operator:[version x.y.z]
        imagePullPolicy: IfNotPresent
        livenessProbe:
          httpGet:
            path: /healthz
            port: 8081
          initialDelaySeconds: 15
          periodSeconds: 20
        name: manager
        readinessProbe:
          httpGet:
            path: /readyz
            port: 8081
          initialDelaySeconds: 5
          periodSeconds: 10
        resources:
          limits:
            cpu: 300m
            memory: 750Mi
          requests:
            cpu: 100m
            memory: 75Mi
        securityContext:
          allowPrivilegeEscalation: false
      imagePullSecrets: []
      securityContext:
        runAsUser: 65532
      terminationGracePeriodSeconds: 10
```

3. 更新された Astra Control Center オペレータをインストールします。

```
kubectl apply -f astra_control_center_operator_deploy.yaml
```

回答例：

```
namespace/netapp-acc-operator unchanged
customresourcedefinition.apiextensions.k8s.io/astracontrolcenters.astra.
netapp.io configured
role.rbac.authorization.k8s.io/acc-operator-leader-election-role
unchanged
clusterrole.rbac.authorization.k8s.io/acc-operator-manager-role
configured
clusterrole.rbac.authorization.k8s.io/acc-operator-metrics-reader
unchanged
clusterrole.rbac.authorization.k8s.io/acc-operator-proxy-role unchanged
rolebinding.rbac.authorization.k8s.io/acc-operator-leader-election-
rolebinding unchanged
clusterrolebinding.rbac.authorization.k8s.io/acc-operator-manager-
rolebinding configured
clusterrolebinding.rbac.authorization.k8s.io/acc-operator-proxy-
rolebinding unchanged
configmap/acc-operator-manager-config unchanged
service/acc-operator-controller-manager-metrics-service unchanged
deployment.apps/acc-operator-controller-manager configured
```

4. ポッドが実行中であることを確認します

```
kubectl get pods -n netapp-acc-operator
```

Astra Control Center をアップグレードします

1. Astra Control Centerカスタムリソース（CR）を編集します。

```
kubectl edit AstraControlCenter -n [netapp-acc or custom namespace]
```

2. Astraのバージョン番号を変更します (`astraVersion` の内部 `Spec`）をアップグレードするバージョンにアップグレードします。

```
spec:
  accountName: "Example"
  astraVersion: "[Version number]"
```

3. イメージレジストリパスが、イメージをでプッシュしたレジストリパスと一致することを確認します 前の手順。更新 `imageRegistry` の内部 `Spec` 前回のインストール以降にレジストリが変更されている場合。

```
  imageRegistry:
    name: "[your_registry_path]"
```

4. に次の項目を追加します `CRDs` の内部の設定 `Spec`：

```
crds:
  shouldUpgrade: true
```

5. 内に次の行を追加します `additionalValues` の内部 `Spec` Astra Control Center CRで、次の手順を実行します。

```
additionalValues:
    nautilus:
      startupProbe:
        periodSeconds: 30
        failureThreshold: 600
```

ファイルエディタを保存して終了すると、変更が適用され、アップグレードが開始されます。

6. （オプション）ポッドが終了し、再び使用可能になったことを確認します。

```
watch kubectl get pods -n [netapp-acc or custom namespace]
```

7. アップグレードが完了して準備ができたことを示すため、Astra Controlのステータス状態が表示されるまで待ちます (`True`）：

```
kubectl get AstraControlCenter -n [netapp-acc or custom namespace]
```

対応：

```
NAME    UUID                                      VERSION     ADDRESS
READY
astra   9aa5fdae-4214-4cb7-9976-5d8b4c0ce27f      22.11.0-82
10.111.111.111  True
```

処理中のアップグレードステータスを監視するには、次のコマンドを実行します。`kubectl get AstraControlCenter -o yaml -n [netapp-acc or custom namespace]`

Astra Control Centerのオペレータログを調べるには、次のコマンドを実行します。`kubectl logs deploy/acc-operator-controller-manager -n netapp-acc-operator -c manager -f`

システムステータスを確認します

1. Astra Control Center にログインします。
2. バージョンがアップグレードされたことを確認します。UIの* Support *ページを参照してください。
3. すべての管理対象クラスタとアプリケーションが引き続き存在し、保護されていることを確認します。

Astra Control Center をアンインストールします

試用版からフルバージョンの製品にアップグレードする場合は、 Astra Control Center コンポーネントの削除が必要になることがあります。Astra Control Center と Astra Control Center Operator を削除するには、この手順で説明されているコマンドを順に実行します。

アンインストールに問題がある場合は、を参照してください [アンインストールに関する問題のトラブルシューティング]。

必要なもの

- Astra Control Center UI を使用して、すべての管理を解除します "クラスタ"。

手順

1. Astra Control Center を削除します。次のコマンド例は、デフォルトのインストールに基づいています。カスタム構成を作成した場合は、コマンドを変更します。

```
kubectl delete -f astra_control_center.yaml -n netapp-acc
```

結果

```
astracontrolcenter.astra.netapp.io "astra" deleted
```

2. を削除するには、次のコマンドを使用します `netapp-acc` ネームスペース：

```
kubectl delete ns netapp-acc
```

結果

```
namespace "netapp-acc" deleted
```

3. Astra Control Center オペレータシステムコンポーネントを削除するには、次のコマンドを使用します。

```
kubectl delete -f astra_control_center_operator_deploy.yaml
```

結果

```
namespace/netapp-acc-operator deleted
customresourcedefinition.apiextensions.k8s.io/astracontrolcenters.astra.
netapp.io deleted
role.rbac.authorization.k8s.io/acc-operator-leader-election-role deleted
clusterrole.rbac.authorization.k8s.io/acc-operator-manager-role deleted
clusterrole.rbac.authorization.k8s.io/acc-operator-metrics-reader
deleted
clusterrole.rbac.authorization.k8s.io/acc-operator-proxy-role deleted
rolebinding.rbac.authorization.k8s.io/acc-operator-leader-election-
rolebinding deleted
clusterrolebinding.rbac.authorization.k8s.io/acc-operator-manager-
rolebinding deleted
clusterrolebinding.rbac.authorization.k8s.io/acc-operator-proxy-
rolebinding deleted
configmap/acc-operator-manager-config deleted
service/acc-operator-controller-manager-metrics-service deleted
deployment.apps/acc-operator-controller-manager deleted
```

アンインストールに関する問題のトラブルシューティング

Astra Control Center のアンインストールで発生した問題に対処するには、次の回避策を実行します。

Astra Control Center をアンインストールしても、管理対象クラスタで監視オペレータポッドがクリーンアップされない

Astra Control Center をアンインストールする前にクラスタの管理を解除していない場合は、次のコマンドを使用して、ネットアップ監視ネームスペースとネームスペース内のポッドを手動で削除できます。

手順

1. 削除 `acc-monitoring` エージェント：

```
kubectl delete agents acc-monitoring -n netapp-monitoring
```

結果

```
agent.monitoring.netapp.com "acc-monitoring" deleted
```

2. ネームスペースを削除します。

```
kubectl delete ns netapp-monitoring
```

結果

```
namespace "netapp-monitoring" deleted
```

3. リソースの削除を確認します。

```
kubectl get pods -n netapp-monitoring
```

結果

```
No resources found in netapp-monitoring namespace.
```

4. 監視エージェントが削除されたことを確認：

```
kubectl get crd|grep agent
```

サンプル結果：

```
agents.monitoring.netapp.com                      2021-07-21T06:08:13Z
```

5. カスタムリソース定義（ CRD ）情報の削除：

```
kubectl delete crds agents.monitoring.netapp.com
```

結果

```
customresourcedefinition.apiextensions.k8s.io
"agents.monitoring.netapp.com" deleted
```

Astra Control Center をアンインストールしても、 **Traefik CRD** をクリーンアップできない

Traefik CRD を手動で削除できます。CRD はグローバルリソースであり、削除するとクラスタ上の他のアプリケーションに影響を与える可能性があります。

手順

1. クラスタにインストールされている Traefik CRD を表示します。

```
kubectl get crds |grep -E 'traefik'
```

応答

```
ingressroutes.traefik.containo.us             2021-06-23T23:29:11Z
ingressroutetcps.traefik.containo.us          2021-06-23T23:29:11Z
ingressrouteudps.traefik.containo.us          2021-06-23T23:29:12Z
middlewares.traefik.containo.us               2021-06-23T23:29:12Z
middlewaretcps.traefik.containo.us            2021-06-23T23:29:12Z
serverstransports.traefik.containo.us         2021-06-23T23:29:13Z
tlsoptions.traefik.containo.us                2021-06-23T23:29:13Z
tlsstores.traefik.containo.us                 2021-06-23T23:29:14Z
traefikservices.traefik.containo.us           2021-06-23T23:29:15Z
```

2. CRD を削除します。

```
kubectl delete crd ingressroutes.traefik.containo.us
ingressroutetcps.traefik.containo.us
ingressrouteudps.traefik.containo.us middlewares.traefik.containo.us
serverstransports.traefik.containo.us tlsoptions.traefik.containo.us
tlsstores.traefik.containo.us traefikservices.traefik.containo.us
middlewaretcps.traefik.containo.us
```

詳細については、こちらをご覧ください

- "アンインストールに関する既知の問題"