アプリを保護します

Astra Control Center

NetApp
November 27, 2023

目次

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接続します。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"。

バックアップとレプリケーションのスケジュールをオフセットして、スケジュールの重複を回避します。たとえば、1時間ごとに1時間の最上部にバックアップを実行し、オフセットを5分、間隔を10分に設定してレプリケーションを開始するようにスケジュールを設定します。

アプリケーションがでサポートされるストレージクラスを使用している場合 `ontap-nas-economy` ドライバ、保護ポリシーは使用できません。バックアップとSnapshotのスケジュールを設定する場合は、Astra Controlでサポートされるストレージクラスに移行します。

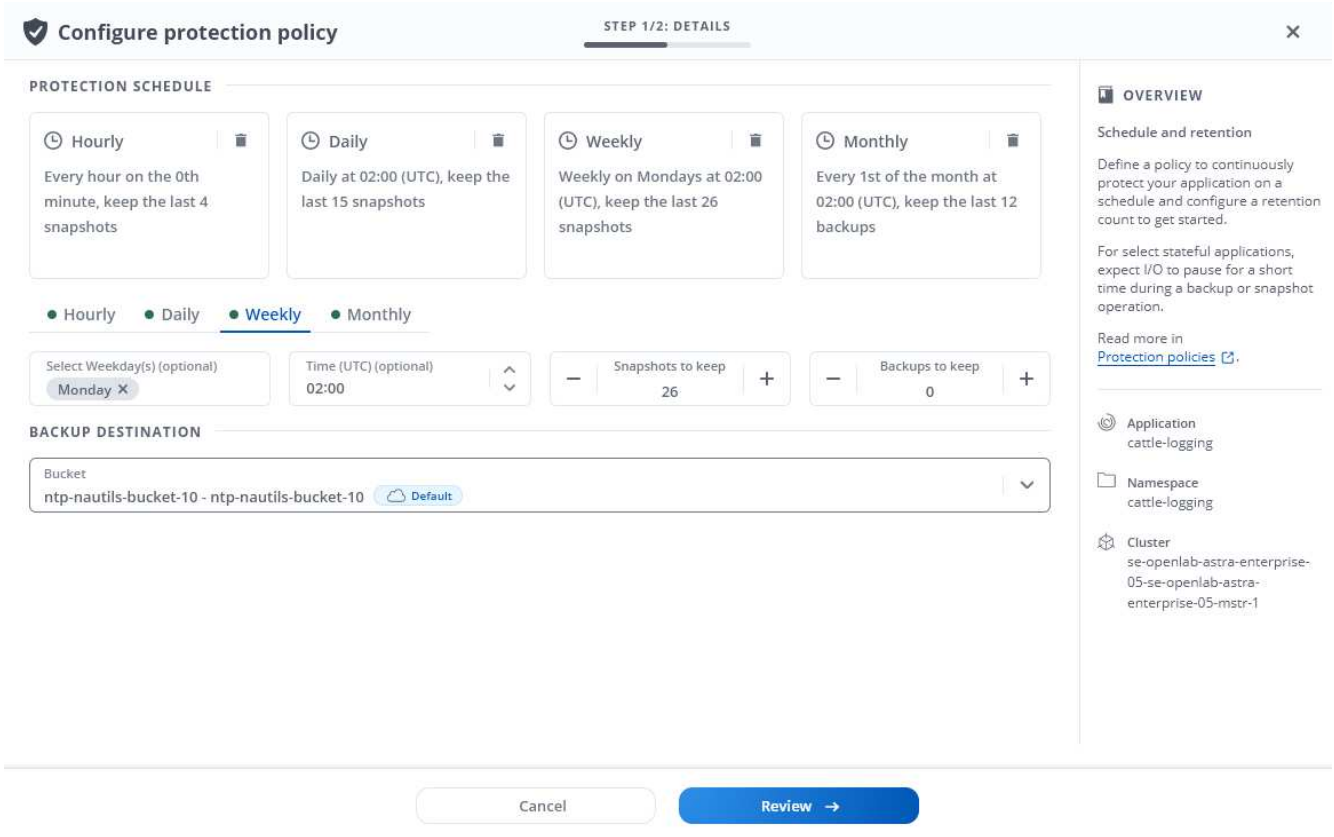
手順

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. ［ 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。

> アプリケーションがでサポートされるストレージクラスを使用している場合 `ontap-nas-economy` ドライバ、スナップショットを作成できません。スナップショットには代替のストレージクラスを使用します。

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、 * 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。

> アプリケーションがでサポートされるストレージクラスを使用している場合 `ontap-nas-economy` ドライバ。を定義していることを確認してください `backendType` のパラメータ "Kubernetesストレージオブジェクト" を使用します `ontap-nas-economy` 保護処理を実行する前にによってバックアップされたアプリケーションのバックアップ `ontap-nas-economy` システムの停止を伴うため、バックアップ処理が完了するまでアプリケーションを使用できなくなります。

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

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 の略" アプリを復元するには、

このタスクについて

- 最初にアプリケーションを保護する:アプリケーションを復元する前に、アプリケーションのスナップショットまたはバックアップを作成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の確認：別のストレージクラスにリストアする場合は、ストレージクラスで同じ永続ボリュームアクセスモード（ReadWriteManyなど）が使用され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。たとえば、ソースの永続ボリュームがRWXアクセスモードを使用している場合は、Azure Managed Disks 、AWS EBS、Google Persistent Disk、など、RWXを提供できないデスティネーションストレージクラスを選択します `ontap-san`を指定すると、リストア処理は失敗します。原因は失敗します。永続ボリュームのアクセスモードの詳細については、を参照してください "Kubernetes" ドキュメント

- 必要なスペースを確保するための計画：NetApp ONTAP ストレージを使用するアプリケーションのインプレース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されるスペースが2倍になることがあります。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。
- （**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デプロイ済みアプリ*：Helm 3でデプロイされたアプリ（またはHelm 2からHelm 3にアップグレードされたアプリ）は完全にサポートされます。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。

> 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。詳細については、を参照してください この例です。

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. [オプション]メニューの[操作]列で、*[リストア]*を選択します。
3. 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。

     > アプリケーションがでサポートされるストレージクラスを使用している場合 `ontap-nas-economy` ドライバ。元のストレージクラスを使用してアプリケーションをリストアする必要があります。アプリケーションを同じネームスペースにリストアする場合、別のストレージクラスを指定することはできません。

     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またはバックアップを選択します。これにより、アプリは以前のバージョンに戻ります。
     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。

- 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。

この手順 は、どちらにも使用できます をバックアップされたストレージクラスに追加します `ontap-nas` 同じクラスタ*または*から作成されたストレージクラスを含む別のクラスタにアプリケーションをコピーします `ontap-nas-economy` ドライバ。

i. 復元されたアプリの名前を指定します。
ii. 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。
iii. 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。

iv. 「 * 次へ * 」を選択します。
v. アプリの復元に使用するスナップショットまたはバックアップを選択します。
vi. 「 * 次へ * 」を選択します。
vii. 次のいずれかを選択します。
  - 元のストレージクラスを使用してリストア：ターゲットクラスタに存在しない場合を除き、元々 関連付けられていたストレージクラスがアプリケーションで使用されます。この場合、クラスタのデフォルトのストレージクラスが使用されます。
  - 別のストレージクラスを使用したリストア：ターゲットクラスタに存在するストレージクラスを選択してください。元 々 関連付けられていたストレージクラスに関係なく、すべてのアプリケーションボリュームが、リストアの一環としてこの別のストレージクラスに移動されます。

viii. 「 * 次へ * 」を選択します。

4. フィルタするリソースを選択：
  - すべてのリソースを復元：元のアプリケーションに関連付けられているすべてのリソースを復元します。
  - リソースのフィルタ:元のアプリケーションリソースのサブセットを復元するルールを指定します。
    i. リストアされたアプリケーションにリソースを含めるか除外するかを選択します。
    ii. または[除外ルールを追加]*のいずれかを選択し、アプリケーションのリストア時に正しいリソースをフィルタするようにルールを設定します。設定が正しくなるまで、ルールを編集したり削除したり、ルールを再度作成したりすることができます。

includeルールとexcludeルールの設定については、を参照してください [アプリケーションのリストア中にリソースをフィルタリングします]。

5. 「 * 次へ * 」を選択します。
6. リストア処理の詳細をよく確認し、プロンプトが表示されたら「restore」と入力して*[リストア]*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。

アプリケーションのリストア中にリソースをフィルタリングします

にフィルタルールを追加できます "リストア" リストアされたアプリケーションに含める、またはリストアされたアプリケーションから除外する既存のアプリケーションリソースを指定する処理。指定した名前空間、ラベル、またはGVK（GroupVersionKind）に基づいて、リソースを含めたり除外したりできます。

[Include（含める）]および**[Exclude（除外）]**のシナリオ

- 元のネームスペースを使用する包含ルールを選択した場合（インプレースリストア）：ルールで定義した既存のアプリケーションリソースは削除され、リストアに使用する選択したSnapshotまたはバックアップのリソースで置き換えられます。includeルールで指定しないリソースは変更されません。
- 新しい名前空間を持つ**include**ルールを選択した場合：このルールを使用して、リストアされたアプリケーションで使用する特定のリソースを選択します。対象ルールに指定しないリソースは、リストアされたアプリケーションには含まれません。
- 元のネームスペースを含む除外ルールを選択した場合（インプレースリストア）：除外するように指定したリソースはリストアされず、変更されません。除外するように指定しないリソースは、スナップショットまたはバックアップからリストアされます。対応するStatefulSetがフィルタリングされたリソースに含まれている場合、永続ボリューム上のすべてのデータが削除されて再作成されます。
- 新しい名前空間を持つ除外ルールを選択した場合:このルールを使用して、リストアされたアプリケーションから削除する特定のリソースを選択します。除外するように指定しないリソースは、スナップショットまたはバックアップからリストアされます。

ルールには、includeまたはexcludeタイプがあります。リソースの包含と除外を組み合わせたルールは使用できません。

手順

1. リソースをフィルタするように選択し、[アプリケーションのリストア]ウィザードで[含める]または[除外するルールを追加する]を選択したら、*[除外するルールを追加する]*を選択します。

   Astra Controlで自動的に追加されるクラスタ対象のリソースを除外することはできません。

2. フィルタルールを設定します。

   ネームスペース、ラベル、またはGVKを少なくとも1つ指定する必要があります。フィルタルールを適用したあとに保持するリソースがあれば、リストアしたアプリケーションを正常な状態に保つのに十分であることを確認してください。

   a. ルールの特定のネームスペースを選択します。選択しない場合は、すべての名前空間がフィルタで使用されます。

      アプリケーションに複数のネームスペースが含まれていた場合、新しいネームスペースにリストアすると、リソースが含まれていなくてもすべてのネームスペースが作成されます。

   b. （オプション）リソース名を入力します。

   c. （任意）ラベルセレクタ：を含めます "ラベルセレクタ" をクリックしてルールに追加します。ラベルセレクタは、選択したラベルに一致するリソースのみをフィルタリングするために使用されます。

   d. （オプション）[Use GVK (GroupVersionKind) set]を選択してリソースをフィルタリング*し、追加のフィルタリングオプションを指定します。

      GVKフィルタを使用する場合は、バージョンと種類を指定する必要があります。

      i. （オプション）* Group *：ドロップダウンリストからKubernetes APIグループを選択します。

      ii. 種類：ドロップダウンリストから、フィルタで使用するKubernetesリソースタイプのオブジェクトスキーマを選択します。

      iii. バージョン：Kubernetes APIのバージョンを選択します。

3. エントリに基づいて作成されたルールを確認します。

4. 「 * 追加」を選択します。

   ルールを含むリソースと除外するリソースは必要なだけ作成できます。処理を開始する前に、リストアアプリケーションの概要にルールが表示されます。

経済性に優れたONTAP-NASストレージからONTAP-NASストレージへの移行

Astra Controlを使用できます "アプリケーションのリストア" または "アプリケーションのクローン" に対応するストレージクラスからアプリケーションボリュームを移行する処理 `ontap-nas-economy`では、でサポートされるストレージクラスに制限されたアプリケーション保護オプションが許可されます `ontap-nas` Astra Controlのあらゆる保護オプションを利用できます。クローンまたはリストア処理では、を使用するqtreeベースのボリュームが移行されます `ontap-nas-economy` でサポートされる標準ボリュームへのバックエンド `ontap-nas`。ボリューム（ボリュームが存在するかどうかに関係なく） `ontap-nas-economy` BACKED ONLYまたはMIXEDは、ターゲットストレージクラスに移行されます。移行が完了すると、保護オ

プションの制限がなくなります。

リソースを別のアプリケーションと共有するアプリケーションでは、インプレースリストアが複雑になります

リソースを別のアプリケーションと共有し、意図しない結果を生成するアプリケーションに対して、インプレースリストア処理を実行できます。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。

次に、NetApp SnapMirrorレプリケーションを使用してリストアすると望ましくない状況が発生するシナリオの例を示します。

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。

SnapMirrorテクノロジを使用してストレージバックエンド間でアプリケーションをレプリケート

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、ストレージバックエンド間、同じクラスタ上、または異なるクラスタ間でデータやアプリケーションの変更を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Aへ
- 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。

別のアプリケーションを逆方向に同時に複製できます。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による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。

- * ONTAPクラスタ*：
  - * Astra Trident *：ONTAPをバックエンドとして利用するソースとデスティネーションの両方のKubernetesクラスタに、Astra Tridentバージョン22.10以降が存在している必要があります。
  - ライセンス：Data Protection Bundleを使用するONTAP SnapMirror非同期ライセンスが、ソースとデスティネーションの両方のONTAPクラスタで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" を参照してください。
- ピアリング：
  - *クラスタとSVM *：ONTAPストレージバックエンドにピア関係が設定されている必要があります。を参照してください "クラスタと SVM のピアリングの概要" を参照してください。

    2つのONTAPクラスタ間のレプリケーション関係で使用されるSVM名が一意であることを確認してください。

  - * Astra TridentとSVM *：ピア関係にあるリモートSVMがデスティネーションクラスタのAstra Tridentで使用可能である必要があります。
- * Astra Control Center *：

  "Astra Control Centerを導入" シームレスなディザスタリカバリのための第3の障害ドメインまたはセカンダリサイト。

  - 管理対象クラスタ：次のクラスタをAstra Controlに追加して管理する必要があります（理想的には障害ドメインやサイトが異なる場合）。
    - ソースKubernetesクラスタ
    - デスティネーションKubernetesクラスタ
    - 関連付けられているONTAPクラスタ

- ユーザアカウント：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」ロールの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。

- * Astra Trident / ONTAPの設定*：Astra Control Centerでは、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レプリケーションをサポートするストレージバックエンドを少なくとも1つ設定する必要があります。ソースクラスタとデスティネーションクラスタが同じである場合は、耐障害性を最大限に高めるために、デスティネーションアプリケーションでソースアプリケーションとは別のストレージバックエンドを使用する必要があります。

> Astra Controlレプリケーションでは、単一のストレージクラスを使用するアプリケーションがサポートされます。ネームスペースにアプリケーションを追加するときは、そのアプリケーションのストレージクラスがネームスペース内の他のアプリケーションと同じであることを確認してください。レプリケートされたアプリケーションにPVCを追加するときは、新しいPVCのストレージクラスがネームスペース内の他のPVCと同じであることを確認してください。

レプリケーション関係を設定

レプリケーション関係の設定には、次の作業が含まれます。

- Astra ControlでアプリケーションSnapshotを作成する頻度を選択します（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が含まれます）。
- 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の作成時間の設定

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. >[レプリケーション]*タブを選択します。
3. [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：デスティネーションクラスタを入力します（ソースクラスタと同じでもかまいません）。
   - 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クラスタのピア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。ベストプラクティスとして、デスティネーションストレージクラスでソースストレージクラスとは別のストレージバックエンドを指定することを推奨します。
   - レプリケーションタイプ： `Asynchronous` は、現在使用可能な唯一のレプリケーションタイプです。
   - 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：Astra Controlで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設定します。

◦ オフセット：Astra ControlでSnapshotを作成する時間（分）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。

バックアップとレプリケーションのスケジュールをオフセットして、スケジュールの重複を回避します。たとえば、1時間ごとに1時間の最上部にバックアップを実行し、オフセットを5分、間隔を10分に設定してレプリケーションを開始するようにスケジュールを設定します。

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が、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確認するには、**[**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名の形式は次のとおりです。 `replication-schedule-<string>`。Astra Controlは、レプリケーションに使用された最後のSnapshotを保持します。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初期スナップショットを作成します。
- 最初の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>

例：
正常|確立

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. >[レプリケーション]*タブを選択します。
3. [アクション]メニューから*[フェイルオーバー]*を選択します。
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

フェイルオーバー手順が発生すると、次の処理が実行されます。

- デスティネーションアプリケーションは、最新のレプリケートされたSnapshotに基づいて起動されます。
- 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。
- ソースアプリの保護ポリシーは、フェイルオーバー時にソースアプリ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にコピーされます。
- ソースアプリで1つ以上のリストア後の実行フックが有効になっている場合、それらの実行フックはデスティネーションアプリに対して実行されます。
- 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. >[レプリケーション]*タブを選択します。
3. [操作]メニューから*[再同期]*を選択します。
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

これは、アプリケーションをデスティネーションストレージバックエンドに移動し、元のソースストレージバックエンドに引き続きレプリケートするという計画的な処理です。Astra Controlは、デスティネーションアプリケーションにフェイルオーバーする前に、ソースアプリケーションを停止してデスティネーションにデータをレプリケートします。

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。

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. >[レプリケーション]*タブを選択します。
3. [操作]メニューから*[逆レプリケーション]*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。

- 元のソースアプリのKubernetesリソースのスナップショットが作成されます。
- 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、アプリのボリュームのスナップショットが取得され、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。
- アプリのKubernetesリソースは、元のソースアプリがシャットダウンされた後に複製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スナップショットから復元されます。
- 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、フェイルオーバー処理後に次の一連の処理を使用して「フェイルバック」を実現できます。このワークフローでは、レプリケーションの方向を元に戻すために、Astra Controlがアプリケーションの変更を元のソースアプリケーションにレプリケート（再同期）してからレプリケーションの方向を反転します。

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が完了した関係から開始し、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. >[レプリケーション]*タブを選択します。
3. [操作]メニューから*[再同期]*を選択します。
4. フェイルバック処理の場合は、フェイルオーバーした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されます）。
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. >[レプリケーション]*タブを選択します。
3. [アプリケーションの保護]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：関係が確立中または確立されており、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（成功したSnapshotがレプリケートされ、フェイルオーバーが可能です）、最新のSnapshotはレプリケートに失敗したか失敗しました。

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が使用可能であること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：Astra Controlは、SnapMirror関係を解除し、最後にレプリケートされたアプリケーションのSnapshotからアプリケーションのKubernetesリソースをリストアします。
- フェイルオーバー：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最新の（成功した）レプリケートされ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 の略" アプリケーションのクローン作成と移行を実行します。

作業を開始する前に

- デスティネーションボリュームを確認：別のストレージクラスにクローニングする場合は、ストレージクラスで同じ永続ボリュームアクセスモード（ReadWriteManyなど）が使用され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的ボリュームのアクセスモードが異なると、クローニング処理は失敗します。たとえば、ソースの永続ボリュームがRWXアクセスモードを使用している場合は、Azure Managed Disks、AWS EBS、Google Persistent Disk、など、RWXを提供できないデスティネーションストレージクラスを選択します `ontap-san`を指定すると、クローン処理は失敗します。原因は失敗します。永続ボリュームのアクセスモードの詳細については、を参照してください "Kubernetes" ドキュメント
- 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。
- * ontap-nas-economy-backed storage class *：アプリケーションがに基づくストレージクラスを使用している場合 `ontap-nas-economy` ドライバ：クローン処理のバックアップ部分はシステムの停止を伴います。バックアップが完了するまで、ソースアプリケーションは使用できません。クローン処理のリストア部分は無停止で実行されます。
- 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。
- *特定のバージョンのPostgreSQL *：同じクラスタ内のアプリケーションクローンは、Bitnami PostgreSQL 11.5.0チャートで一貫して失敗します。正常にクローニングするには、以前のバージョンのグラフを使用してください。

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。
- アプリケーションに関連付けられている元のストレージクラスを保持するか、別のストレージクラスを選択します。

  アプリケーションのストレージクラスをネイティブクラウドプロバイダのストレージクラスやサポートされているその他のストレージクラスに移行できます。 をバックアップされたストレージクラスに追加します `ontap-nas` を使用するか、から作成されたストレージクラスを含む別のクラスタにアプリケーションをコピーします `ontap-nas-economy` ドライバ。

別のストレージクラスを選択し、このストレージクラスがリストア時に存在しない場合は、エラーが返されます。

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
- フェイルオーバー後

実行フックフィルタ

アプリケーションに実行フックを追加または編集するときに、実行フックにフィルタを追加して、フックが一致するコンテナを管理できます。フィルタは、すべてのコンテナで同じコンテナイメージを使用し、各イメージを別の目的（Elasticsearchなど）に使用するアプリケーションに便利です。フィルタを使用すると、一部の同一コンテナで実行フックが実行されるシナリオを作成できます。1つの実行フックに対して複数のフィルタを作成すると、それらは論理AND演算子と結合されます。実行フックごとに最大10個のアクティブフィルタを使用できます。

実行フックに追加する各フィルタは、正規表現を使用してクラスタ内のコンテナを照合します。フックがコンテナと一致すると、そのコンテナに関連付けられたスクリプトがフックによって実行されます。フィルタの正規表現では、正規表現2（RE2）構文を使用します。この構文では、一致リストからコンテナを除外するフィルタの作成はサポートされていません。実行フックフィルタの正規表現でAstra Controlがサポートする構文については、を参照してください "正規表現2（RE2）構文のサポート"。

> リストアまたはクローン処理のあとに実行される実行フックにネームスペースフィルタを追加し、リストアまたはクローンのソースとデスティネーションが異なるネームスペースにある場合、ネームスペースフィルタはデスティネーションネームスペースにのみ適用されます。

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、実行中のアプリケーションの機能を低下させたり、完全に無効にしたり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使用されるロジックは、バックアップが完了したとは見なされません。

- 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、*アクティビティ*ページ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）。
- レプリケートされたソースアプリケーションをAstra Control Centerがデスティネーションアプリケーションにフェイルオーバーすると、フェイルオーバーの完了後にソースアプリケーションに対して有効になっているフェイルオーバー後の実行フックがデスティネーションアプリケーションに対して実行されます。

> Astra Control Center 23.04でリストア後のフックを実行していて、Astra Control Centerを23.07にアップグレードした場合、フェイルオーバーレプリケーション後にリストア後の実行フックが実行されなくなります。アプリケーションのフェイルオーバー後の実行フックを新しく作成する必要があります。また、フェイルオーバー用の既存のリストア後フックの処理タイプを「リストア後」から「フェイルオーバー後」に変更することもできます。

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、すべての異なるタイプのフックを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	フェールオーバーフックの実行
1.	クローン	N	N	新規	同じ	Y	N	Y	N
2.	クローン	N	N	新規	違う	Y	Y	Y	N
3.	クローンまたはリストア	Y	N	新規	同じ	N	N	Y	N
4.	クローンまたはリストア	N	Y	新規	同じ	N	N	Y	N
5.	クローンまたはリストア	Y	N	新規	違う	N	N	Y	N

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	フェールオーバーフックの実行
6.	クローンまたはリストア	N	Y	新規	違う	N	N	Y	N
7.	リストア	Y	N	既存	同じ	N	N	Y	N
8	リストア	N	Y	既存	同じ	N	N	Y	N
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	N
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	N
11	バックアップ	Y	該当なし	該当なし	該当なし	N	N	該当なし	N
12	フェイルオーバー	Y	該当なし	レプリケーションで作成	違う	N	N	N	Y
13	フェイルオーバー	Y	該当なし	レプリケーションで作成	同じ	N	N	N	Y

実行フックの例

にアクセスします "NetApp Verda GitHubプロジェクト" Apache CassandraやElasticsearchなどの一般的なアプリケーションの実行フックをダウンロードします。また、独自のカスタム実行フックを構築するための例やアイデアを得ることもできます。

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、一致するコンテナの数、作成時間、および実行時間（プリ/ポストオペレーション）を確認できます。を選択できます + アイコンをクリックして、実行するコンテナのリストを展開します。このアプリケーションの実行フックに関連するイベントログを表示するには、*アクティビティ*タブに移動します。

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

各実行フックは、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。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つのスクリプトを変更するだけで多く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してAstra Controlに追加できます。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、次の
   a. *操作*ドロップダウンメニューから操作タイプを選択して、フックをいつ実行するかを決定します。
   b. フックの一意の名前を入力します。
   c. （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。
5. （オプション）フックフィルタの詳細（* Hook Filter Details *）領域で、実行フックが実行されるコンテナを制御するフィルタを追加できます。
   a. [フィルタの追加]を選択します。
   b. [フックフィルタータイプ*]列で、フィルターを適用する属性をドロップダウンメニューから選択します。
   c. [**Regex**]列に、フィルタとして使用する正規表現を入力します。Astra Controlでは、を使用します "正規表現2（RE2）正規表現の正規表現構文"。

正規表現フィールドに他のテキストを含まない属性（ポッド名など）の正確な名前でフィルタリングすると、部分文字列の照合が実行されます。正確な名前とその名前だけを照合するには、完全に一致する文字列の一致構文を使用します（例： `^exact_podname$`）。

   d. フィルタをさらに追加するには、*フィルタを追加*を選択します。

実行フックの複数のフィルタは、論理AND演算子と結合されます。実行フックごとに最大10個のアクティブフィルタを使用できます。

6. 完了したら、「次へ」を選択します。
7. [* 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。
8. 「 * 次へ * 」を選択します。
9. 実行フックの設定を確認します。
10. 「 * 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。

実行フックを編集します

実行フックを編集して、その属性、フィルタ、または使用するスクリプトを変更できます。実行フックを編集するには、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. 編集するフックの*アクション*列のオプションメニューを選択します。
4. 「 * 編集 * 」を選択します。
5. 各セクションを完了したら、「次へ」を選択して、必要な変更を行います。
6. [保存（ Save ）] を選択します。

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。
5. 表示されたダイアログで、「delete」と入力して確定します。
6. [はい]を選択し、実行フックを削除します。*

を参照してください。

- "NetApp Verda GitHubプロジェクト"

Astra Control Centerを使用したAstra Control Centerの保護

Astra Control Centerが実行されているKubernetesクラスタで致命的なエラーに対する耐障害性を高めるには、Astra Control Centerアプリケーション自体を保護します。セカンダリのAstra Control Centerインスタンスを使用してAstra Control Centerをバックアップおよびリストアするか、基盤となるストレージでONTAPを使用している場合はAstraレプリケーションを使用できます。

これらのシナリオでは、Astra Control Centerの2つ目のインスタンスを別のフォールトドメインに導入して設定し、プライマリAstra Control Centerインスタンスとは別の2つ目のKubernetesクラスタで実行します。2つ目のAstra Controlインスタンスは、プライマリのAstra Control Centerインスタンスのバックアップに使用され、場合によってはリストアにも使用されます。リストアまたはレプリケートされたAstra Control Centerインスタンスは、引き続きアプリケーションクラスタアプリケーションのアプリケーションデータ管理機能を提供し、それらのアプリケーションのバックアップやSnapshotへのアクセスをリストアします。

作業を開始する前に

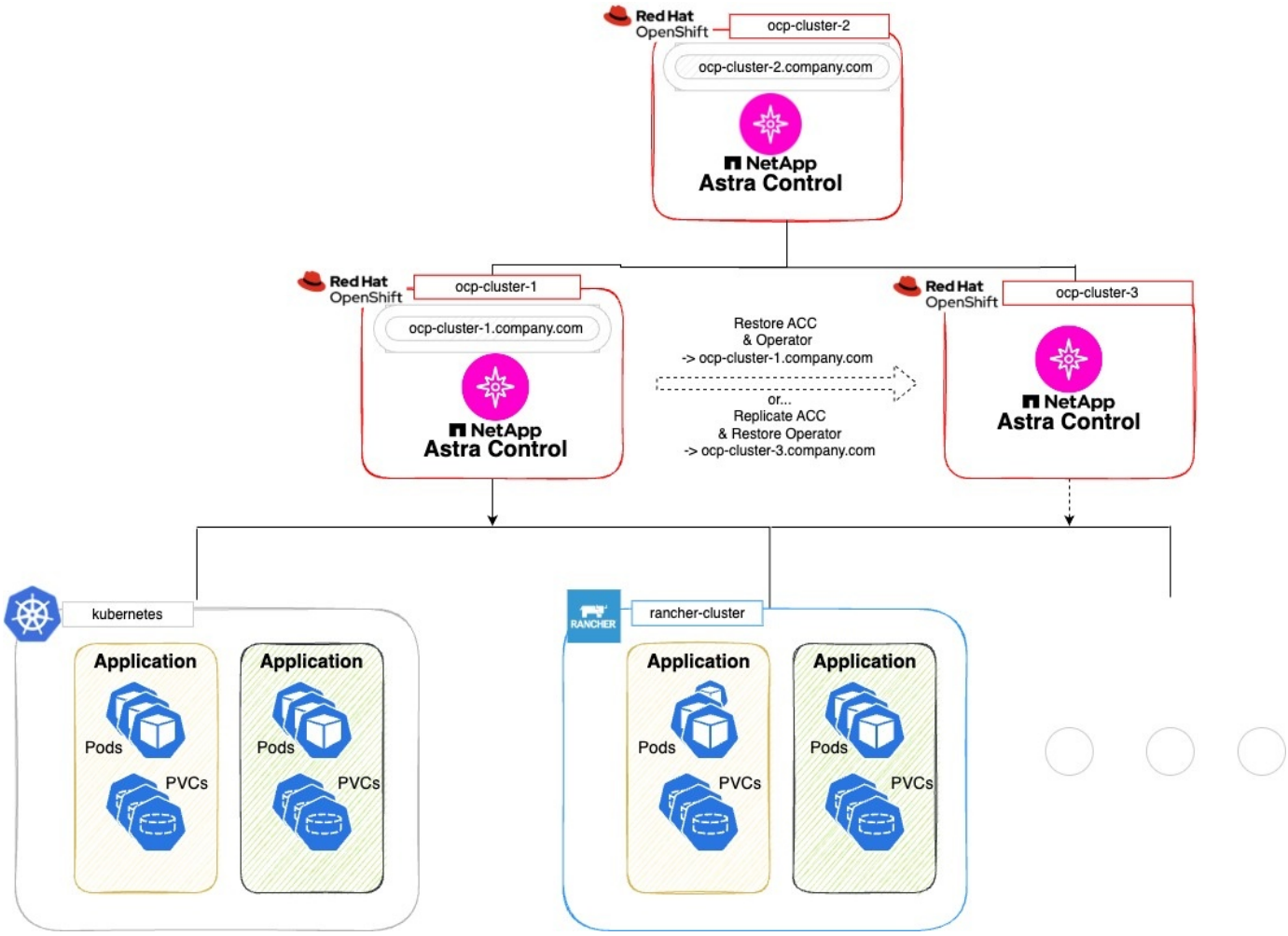
Astra Control Centerの保護シナリオを設定する前に、次の情報を確認してください。

- プライマリ**Astra Control Center**インスタンスを実行する**Kubernetes**クラスタ：このクラスタは、アプリケーションクラスタを管理するプライマリAstra Control Centerインスタンスをホストします。
- セカンダリ**Astra Control Center**インスタンスを実行しているプライマリと同じ**Kubernetes**ディストリビューションタイプの**2**つ目の**Kubernetes**クラスタ：このクラスタは、プライマリAstra Control Centerインスタンスを管理するAstra Control Centerインスタンスをホストします。
- プライマリと同じ**Kubernetes**ディストリビューションタイプの**3**つ目の**Kubernetes**クラスタ：このクラスタは、Astra Control Centerのリストアまたはレプリケートされたインスタンスをホストします。現在プライマリに導入されているものと同じAstra Control Centerネームスペースを使用する必要があります。たとえば、Astra Control Centerがネームスペースに導入されている場合 `netapp-acc` ソースクラスタで、ネームスペース `netapp-acc` デスティネーションKubernetesクラスタ上のどのアプリケーションでも使用できない状態である必要があります。
- * S3互換バケット*：各Astra Control Centerインスタンスには、アクセス可能なS3互換オブジェクトストレージバケットがあります。
- 設定されたロードバランサ：ロードバランサはAstraのIPアドレスを提供し、アプリケーションクラスタと両方のS3バケットへのネットワーク接続を確立する必要があります。

- クラスタは**Astra Control Center**の要件に準拠：Astra Control Center保護で使用される各クラスタは、"Astra Control Centerの一般的な要件"。

このタスクについて

以下の手順では、以下のコマンドを使用してAstra Control Centerを新しいクラスタにリストアするために必要な手順について説明します。 バックアップとリストア または レプリケーション。手順は、ここに示す構成例に基づいています。



この設定例では、次の情報が表示されています。

- プライマリ**Astra Control Center**インスタンスを実行する**Kubernetes**クラスタ：
  - OpenShiftクラスタ： `ocp-cluster-1`
  - Astra Control Centerプライマリインスタンス： `ocp-cluster-1.company.com`
  - このクラスタは、アプリケーションクラスタを管理します。
- セカンダリ**Astra Control Center**インスタンスを実行しているプライマリと同じ**Kubernetes**ディストリビューションタイプの**2**つ目の**Kubernetes**クラスタ：
  - OpenShiftクラスタ： `ocp-cluster-2`
  - Astra Control Centerのセカンダリインスタンス： `ocp-cluster-2.company.com`
  - このクラスタを使用して、プライマリのAstra Control Centerインスタンスをバックアップするか、別

のクラスタへのレプリケーションを設定します（この例では、 `ocp-cluster-3` クラスタ）。

- リストア処理に使用されるプライマリと同じ**Kubernetes**ディストリビューションタイプの**3**つ目の**Kubernetes**クラスタ：
  - OpenShiftクラスタ： `ocp-cluster-3`
  - Astra Control Center 3つ目のインスタンス： `ocp-cluster-3.company.com`
  - このクラスタは、Astra Control Centerのリストアまたはレプリケーションのフェイルオーバーに使用されます。

> アプリケーションクラスタは、上図のKubernetesクラスタとRancherクラスタからわかるように、3つのAstra Control Centerクラスタの外部に配置するのが理想的です。

図には示されていません。

- すべてのクラスタに、TridentがインストールされたONTAPバックエンドがあります。
- この構成では、OpenShiftクラスタがMetalLBをロードバランサとして使用しています。
- SnapshotコントローラとVolumeSnapshotClassもすべてのクラスタにインストールされています（を参照）。 "前提条件"。

ステップ1オプション：Astra Control Centerのバックアップとリストア

この手順では、バックアップとリストアを使用して新しいクラスタにAstra Control Centerをリストアするために必要な手順について説明します。

この例では、Astra Control Centerは常に `netapp-acc` 名前空間と演算子は、 `netapp-acc-operator` ネームスペース：

> ここでは説明しませんが、Astra Control Centerのオペレータは、Astra CRと同じネームスペースに導入することもできます。

作業を開始する前に

- プライマリのAstra Control Centerをクラスタにインストールしておきます。
- セカンダリのAstra Control Centerを別のクラスタにインストールしておきます。

手順

1. プライマリAstra Control Centerアプリケーションとデスティネーションクラスタを、（実行中の）セカンダリAstra Control Centerインスタンスから管理 `ocp-cluster-2` クラスタ）：
   a. セカンダリAstra Control Centerインスタンスにログインします。
   b. "プライマリAstra Control Centerクラスタを追加" (`ocp-cluster-1`）。
   c. "デスティネーションの3つ目のクラスタを追加" (`ocp-cluster-3`）をクリックします。
2. セカンダリのAstra Control CenterでAstra Control CenterとAstra Control Centerオペレータを管理します。
   a. [アプリケーション（Applications）]ページで、[定義（ Define）]を選択します
   b. [アプリケーションの定義]ウィンドウで、新しいアプリケーション名を入力します。 (`netapp-acc`）。

c. プライマリAstra Control Centerを実行しているクラスタを選択 (`ocp-cluster-1`）をクリックします。

d. を選択します `netapp-acc` Astra Control Centerのネームスペース（*[ネームスペース]*ドロップダウンリスト）。

e. [クラスタリソース]ページで、*[クラスタを対象とした追加のリソースを含める]*をオンにします。

f. 「含めるルールを追加」を選択します。

g. 次のエントリを選択し、*[追加]*を選択します。

- ラベルセレクタ:acc-crd
- グループ：apiextensions.k8s.io
- バージョン：v1
- 種類: CustomResourceDefinition

h. アプリケーション情報を確認します。

i. [* 定義（ Define ）] を選択します

「* define *」を選択した後、演算子に対して「アプリケーションの定義」プロセスを繰り返します。`netapp-acc-operator`）をクリックし、 `netapp-acc-operator` [アプリケーションの定義]ウィザードの名前空間。

3. Astra Control Centerとオペレータのバックアップ：

a. セカンダリAstra Control Centerで、[Applications]タブを選択して[Applications]ページに移動します。

b. "バックアップ" Astra Control Centerアプリケーション (`netapp-acc`）。

c. "バックアップ" 演算子 (`netapp-acc-operator`）。

4. Astra Control Centerとオペレータをバックアップしたら、次のツールでディザスタリカバリ（DR）シナリオをシミュレートします。 "Astra Control Centerのアンインストール" プライマリクラスタから削除します。

新しいクラスタ（この手順で説明する3つ目のKubernetesクラスタ）にAstra Control Centerをリストアし、新しくインストールしたAstra Control Centerのプライマリクラスタと同じDNSを使用します。

5. セカンダリAstra Control Centerを使用 "リストア" バックアップから作成したAstra Control Centerアプリケーションのプライマリインスタンス：

a. [Applications]*を選択し、Astra Control Centerアプリケーションの名前を選択します。

b. [オプション]メニューの[操作]列で、*[リストア]*を選択します。

c. リストアタイプとして*[新しいネームスペースにリストアする]*を選択します。

d. リストア名を入力 (`netapp-acc`）。

e. デスティネーションの3番目のクラスタを選択 (`ocp-cluster-3`）。

f. 元のネームスペースと同じネームスペースになるようにデスティネーションネームスペースを更新します。

g. [Restore Source]ページで、リストア・ソースとして使用するアプリケーション・バックアップを選択します。

h. [元のストレージクラスを使用してリストアする]*を選択します。

i. [すべてのリソースをリストア]*を選択します。

j. リストア情報を確認し、*[Restore]*を選択して、Astra Control Centerをデスティネーションクラスタにリストアするリストアプロセスを開始します。（ocp-cluster-3）。アプリケーションが起動すると、リストアが完了します。 available 状態。

6. デスティネーションクラスタでAstra Control Centerを設定します。

a. ターミナルを開き、kubeconfigを使用してデスティネーションクラスタに接続します。（ocp-cluster-3）をクリックします。

b. を確認します ADDRESS Astra Control Center構成の列で参照されるプライマリシステムのDNS名は次のとおりです。

```
kubectl get acc -n netapp-acc
```

対応：

```
NAME  UUID                                   VERSION    ADDRESS
READY
astra 89f4fd47-0cf0-4c7a-a44e-43353dc96ba8 23.07.0-24 ocp-cluster-
1.company.com           True
```

a. 状況に応じて ADDRESS 上記の応答のフィールドにプライマリAstra Control Centerインスタンスのfqdnがない場合は、Astra Control CenterのDNSを参照するように設定を更新します。

```
kubectl edit acc -n netapp-acc
```

i. を変更します astraAddress の下 spec: FQDNへ (ocp-cluster-1.company.com （この例では）のプライマリAstra Control Centerインスタンス。

ii. 設定を保存します。

iii. アドレスが更新されたことを確認します。

```
kubectl get acc -n netapp-acc
```

b. にアクセスします Astra Control Centerのオペレータのリストア セクションを参照して、リストアプロセスを完了してください。

ステップ1オプション：レプリケーションを使用してAstra Control Centerを保護

この手順では、設定に必要な手順について説明します。 "Astra Control Centerのレプリケーション" を使用して、プライマリAstra Control Centerインスタンスを保護します。

この例では、Astra Control Centerは常に netapp-acc 名前空間と演算子は、 netapp-acc-operator ネー

ムスペース：

作業を開始する前に

- プライマリのAstra Control Centerをクラスタにインストールしておきます。
- セカンダリのAstra Control Centerを別のクラスタにインストールしておきます。

手順

1. セカンダリAstra Control Centerインスタンスからプライマリ Astra Control Centerアプリケーションとデスティネーションクラスタを管理します。
   a. セカンダリAstra Control Centerインスタンスにログインします。
   b. "プライマリAstra Control Centerクラスタを追加" (`ocp-cluster-1`）。
   c. "デスティネーションの3つ目のクラスタを追加" (`ocp-cluster-3`）をクリックします。
2. セカンダリのAstra Control CenterでAstra Control CenterとAstra Control Centerオペレータを管理します。
   a. [Clusters]*を選択し、プライマリAstra Control Centerが含まれるクラスタを選択します。（`ocp-cluster-1`）。
   b. [名前空間]タブを選択します。
   c. 選択するオプション `netapp-acc` および `netapp-acc-operator` 名前空間。
   d. [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。
   e. 定義されたアプリケーションを表示するには、*[アプリケーションで表示]*を選択します。
3. レプリケーションのバックエンドを構成します。

   > レプリケーションには、プライマリのAstra Control Centerクラスタとデスティネーションクラスタが必要 (`ocp-cluster-3`）別のピアONTAPストレージバックエンドを使用します。
   > 各バックエンドがピアリングされてAstra Controlに追加されると、[Backends]ページの*[Discovered]*タブにバックエンドが表示されます。

   a. "ピアバックエンドの追加" をプライマリクラスタのAstra Control Centerに接続します。
   b. "ピアバックエンドの追加" デスティネーションクラスタのAstra Control Centerに接続します。
4. レプリケーションを設定します。
   a. [Applications]画面で、 `netapp-acc` アプリケーション：
   b. [レプリケーションポリシーの設定]*を選択します。
   c. 選択するオプション `ocp-cluster-3` をデスティネーションクラスタとして指定します。
   d. ストレージクラスを選択します。
   e. 入力するコマンド `netapp-acc` をデスティネーションネームスペースとして指定します。
   f. 必要に応じてレプリケーション頻度を変更します。
   g. 「 * 次へ * 」を選択します。
   h. 設定が正しいことを確認し、*[保存]*を選択します。

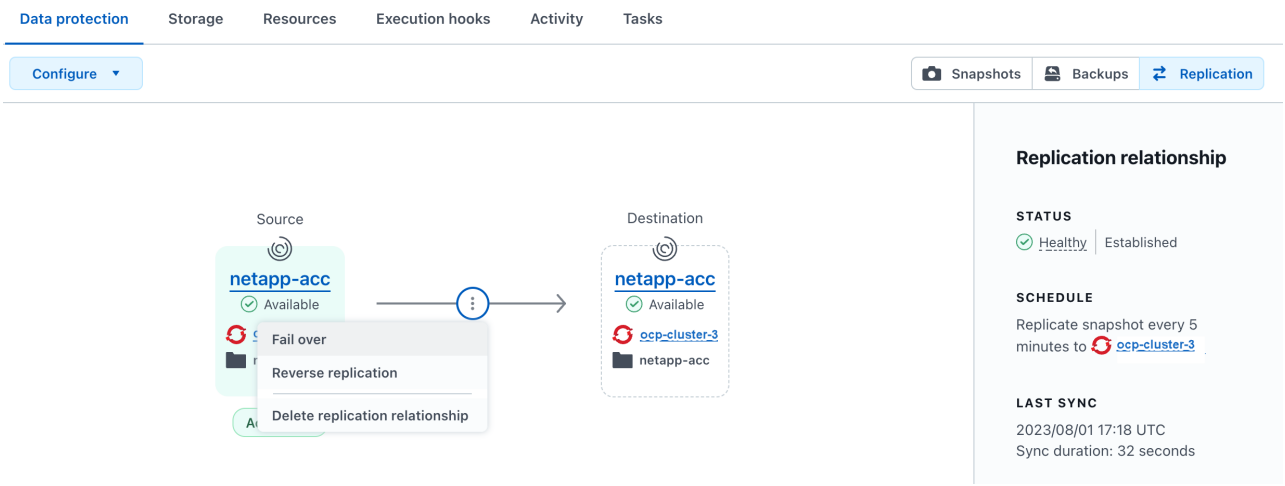
   レプリケーション関係の移行元 `Establishing` 終了： `Established`。アクティブな場合、このレプ

リケーションは、レプリケーション設定が削除されるまで5分おきに実行されます。

5. プライマリシステムが破損しているかアクセスできなくなった場合は、レプリケーションをもう一方のクラスタにフェイルオーバーします。

フェイルオーバーが正常に実行されるように、デスティネーションクラスタにAstra Control Centerがインストールされていないことを確認してください。

a. 縦の楕円アイコンを選択し、*フェイルオーバー*を選択します。



b. 詳細を確認し、*[フェイルオーバー]*を選択してフェイルオーバープロセスを開始します。

レプリケーション関係のステータスがに変わります。 `Failing over` 次に `Failed over` 完了したら、

6. フェイルオーバーの設定を完了します。

a. ターミナルを開き、3番目のクラスタのkubeconfigを使用して接続します。（`ocp-cluster-3`）。このクラスタにはAstra Control Centerがインストールされています。

b. 3つ目のクラスタのAstra Control Center FQDNを確認（`ocp-cluster-3`）。

c. Astra Control CenterのDNSを参照するように設定を更新します。

```
kubectl edit acc -n netapp-acc
```

i. を変更します `astraAddress` の下 `spec:` FQDNを使用（`ocp-cluster-3.company.com`）をクリックします。

ii. 設定を保存します。

iii. アドレスが更新されたことを確認します。

```
kubectl get acc -n netapp-acc
```

d. 必要なすべてのtraefik CRDが存在することを確認します。

```
kubectl get crds | grep traefik
```

必要なtraefik CRD：

```
ingressroutes.traefik.containo.us
ingressroutes.traefik.io
ingressroutetcps.traefik.containo.us
ingressroutetcps.traefik.io
ingressrouteudps.traefik.containo.us
ingressrouteudps.traefik.io
middlewares.traefik.containo.us
middlewares.traefik.io
middlewaretcps.traefik.containo.us
middlewaretcps.traefik.io
serverstransports.traefik.containo.us
serverstransports.traefik.io
tlsoptions.traefik.containo.us
tlsoptions.traefik.io
tIsstores.traefik.containo.us
tIsstores.traefik.io
traefikservices.traefik.containo.us
traefikservices.traefik.io
```

a. 上記のCRDの一部がない場合は、次の手順を実行します。
   i. に進みます "traefikドキュメント"。
   ii. 「定義」領域をファイルにコピーします。
   iii. 変更を適用：

   ```
   kubectl apply -f <file name>
   ```

   iv. traefikを再起動します。

   ```
   kubectl get pods -n netapp-acc | grep -e "traefik" | awk '{print
   $1}' | xargs kubectl delete pod -n netapp-acc"
   ```

b. にアクセスします Astra Control Centerのオペレータのリストア セクションを参照して、リストアプロセスを完了してください。

ステップ2：Astra Control Centerのオペレータをリストア

セカンダリのAstra Control Centerを使用して、プライマリのAstra Control Centerオペレータをバックアップ

からリストアデスティネーションネームスペースは、ソースネームスペースと同じである必要があります。Astra Control Centerをプライマリソースクラスタから削除しても、同じリストア手順を実行するためのバックアップは引き続き存在します。

手順

1. *アプリケーション*を選択し、オペレータアプリの名前を選択します。（`netapp-acc-operator`）。
2. [操作]列の[オプション]メニューから*[リストア]*を選択します。
3. リストアタイプとして*[新しいネームスペースにリストアする]*を選択します。
4. デスティネーションの3番目のクラスタを選択 (`ocp-cluster-3`）。
5. ネームスペースをプライマリソースクラスタに関連付けられているネームスペースと同じに変更する (`netapp-acc-operator`）。
6. リストア・ソースとして以前に作成されたバックアップを選択します。
7. [元のストレージクラスを使用してリストアする]*を選択します。
8. [すべてのリソースをリストア]*を選択します。
9. 詳細を確認し、*[リストア]*をクリックしてリストアプロセスを開始します。

   [Applications]ページには、Astra Control Centerオペレータがデスティネーションの第3のクラスタにリストアされていることが表示される (`ocp-cluster-3`）。プロセスが完了すると、状態はとして表示されます。 `Available`。10分以内に、ページでDNSアドレスが解決されます。

結果

Astra Control Centerとその登録済みクラスタ、Snapshotとバックアップを使用した管理対象アプリケーションを、デスティネーションの第3のクラスタで利用できるようになりました。（`ocp-cluster-3`）。元のインスタンスに対して使用していた保護ポリシーは、新しいインスタンスにも適用されます。スケジュールされたバックアップやオンデマンドのバックアップやスナップショットを引き続き作成できます。

トラブルシューティング

システムの健全性と保護プロセスが成功したかどうかを確認します。

- ポッドが実行されていません：すべてのポッドが実行中であることを確認します。

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

  一部のポッドが `CrashLookBackOff` 状態、再起動し、次の状態に移行する必要があります。 `Running` 状態。

- システムステータスの確認：Astra Control Centerシステムが `ready` 都道府県：

  ```
  kubectl get acc -n netapp-acc
  ```

  対応：

```
NAME  UUID                                   VERSION    ADDRESS
READY
astra 89f4fd47-0cf0-4c7a-a44e-43353dc96ba8 23.07.0-24 ocp-cluster-
1.company.com           True
```

- 導入ステータスの確認：Astra Control Centerの導入情報を表示して `Deployment State` はです `Deployed`。

```
kubectl describe acc astra -n netapp-acc
```

- リストアした**Astra Control Center UI**で**404**エラーが返される： `AccTraefik` 入力オプションとして、traefik CRD すべてインストールされていることを確認します。

著作権に関する情報

商標に関する情報