Astra Control Service를 사용합니다

Astra Control Service

NetApp
April 18, 2024

This PDF was generated from https://docs.netapp.com/ko-kr/astra-control-service/use/log-in.html on April 18, 2024. Always check docs.netapp.com for the latest.

목차

Astra Control Service를 사용합니다 . . . 1
Astra Control Service에 로그인합니다 . . . 1
애플리케이션 관리 및 보호 . . . 1
앱 및 컴퓨팅 상태 보기 . . . 28
버킷을 관리합니다 . . . 30
실행 중인 작업을 모니터링합니다 . . . 33
계정을 관리합니다 . . . 33
클라우드 인스턴스 관리 . . . 43
Astra Control Provisioner를 활성화합니다 . . . 44
앱 및 클러스터 관리를 취소합니다 . . . 53
Astra Control의 자체 관리형 인스턴스를 구축 . . . 55

Astra Control Service를 사용합니다

Astra Control Service에 로그인합니다

Astra Control Service는 로 이동하여 SaaS 기반 사용자 인터페이스를 통해 액세스할 수 있습니다 https://astra.netapp.io.

> Single Sign-On을 사용하여 회사 디렉터리(통합 ID)의 자격 증명을 사용하여 로그인할 수 있습니다. 자세한 내용은 로 이동하십시오 "Help Center(도움말 센터)" 그런 다음 * Cloud Central 로그인 옵션 * 을 선택합니다.

시작하기 전에

- "BlueXP 사용자 ID입니다".
- "새로운 Astra Control 계정" 또는 "기존 계정에 대한 초대".
- 지원되는 웹 브라우저입니다.

  Astra Control Service는 최신 버전의 Firefox, Safari 및 Chrome을 최소 1280 x 720의 해상도로 지원합니다.

단계

1. 웹 브라우저를 열고 로 이동합니다 https://astra.netapp.io.
2. NetApp BlueXP 자격 증명을 사용하여 로그인합니다.

애플리케이션 관리 및 보호

앱 관리를 시작합니다

먼저 해 "Astra Control에 Kubernetes 클러스터를 추가합니다", 클러스터(Astra Control 외부)에 앱을 설치한 다음 Astra Control의 애플리케이션 페이지로 이동하여 앱을 정의할 수 있습니다.

실행 중인 Pod가 있는 스토리지 리소스가 포함된 앱 또는 실행 중인 Pod 없이 스토리지 리소스가 포함된 앱을 정의하고 관리할 수 있습니다. 실행 중인 Pod가 없는 앱을 데이터 전용 애플리케이션이라고 합니다.

설명합니다

Astra Control에는 다음과 같은 앱 관리 요구 사항이 있습니다.

- * Licensing *: 10개 이상의 네임스페이스를 관리하려면 Astra Control 가입이 필요합니다.
- * 네임스페이스 *: Astra Control을 사용하여 단일 클러스터에서 하나 이상의 지정된 네임스페이스 내에서 응용 프로그램을 정의할 수 있습니다. 앱은 동일한 클러스터 내에서 여러 네임스페이스에 걸쳐 있는 리소스를 포함할 수 있습니다. Astra Control은 여러 클러스터에서 앱을 정의하는 기능을 지원하지 않습니다.
- * 스토리지 클래스 *: 스토리지 클래스가 명시적으로 설정된 앱을 설치하고 앱을 복제해야 하는 경우 클론 작업의 타겟 클러스터에 원래 지정된 스토리지 클래스가 있어야 합니다. 명시적으로 설정된 스토리지 클래스를 가진 애플리케이션을 동일한 스토리지 클래스가 없는 클러스터로 클론 복제하면 실패합니다.

- * Kubernetes 리소스 *: Astra Control에서 수집하지 않은 Kubernetes 리소스를 사용하는 앱에는 전체 앱 데이터 관리 기능이 없을 수 있습니다. Astra Control은 다음과 같은 Kubernetes 리소스를 수집합니다.

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		

지원되는 앱 설치 방법

Astra Control은 다음과 같은 응용 프로그램 설치 방법을 지원합니다.

- * 매니페스트 파일 *: Astra Control은 kubctl을 사용하여 매니페스트 파일에서 설치된 앱을 지원합니다. 예를 들면 다음과 같습니다.

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

- * Helm 3 *: Helm을 사용하여 앱을 설치하는 경우 Astra Control에 Helm 버전 3이 필요합니다. Helm 3(또는 Helm 2에서 Helm 3으로 업그레이드)과 함께 설치된 앱의 관리 및 클론 생성이 완벽하게 지원됩니다. Helm 2가 설치된 앱 관리는 지원되지 않습니다.
- * 운용자 구축 앱 *: Astra Control은 네임스페이스 범위 연산자로 설치된 앱을 지원합니다. 일반적으로 "pass-by-reference" 아키텍처가 아니라 "pass-by-value"로 설계되었습니다. 운영자와 설치하는 앱은 동일한 네임스페이스를 사용해야 합니다. 운영자가 배포 .YAML 파일을 수정해야 할 수도 있습니다.

  다음은 이러한 패턴을 따르는 일부 운영자 앱에 대한 설명입니다.

  - "아파치 K8ssandra"

    > K8ssandra 의 경우 현재 위치 복원 작업이 지원됩니다. 새 네임스페이스 또는 클러스터에 대한 복원 작업을 수행하려면 응용 프로그램의 원래 인스턴스를 중단해야 합니다. 이는 이월된 피어 그룹 정보가 인스턴스 간 통신으로 이어지지 않도록 하기 위한 것입니다. 앱 복제는 지원되지 않습니다.

  - "젠킨스 CI"
  - "Percona XtraDB 클러스터"

  Astra Control은 "pass-by-reference" 아키텍처(예: CockroachDB 운영자)로 설계된 운영자를 복제하지 못할 수 있습니다. 이러한 유형의 클론 복제 작업 중에 클론 복제 운영자는 클론 복제 프로세스의 일부로 고유한 새로운 암호가 있음에도 불구하고 소스 운영자의 Kubernetes 암호를 참조하려고 합니다. Astra Control이 소스 운영자의

Kubernetes 암호를 모르기 때문에 클론 작업이 실패할 수 있습니다.

클러스터에 앱을 설치합니다

먼저 해 "클러스터가 추가되었습니다" Astra Control은 클러스터에서 앱을 설치하거나 기존 앱을 관리할 수 있습니다. 하나 이상의 네임스페이스로 범위가 지정된 모든 앱을 관리할 수 있습니다.

Astra Control은 스토리지가 Astra Control에서 지원하는 스토리지 클래스에 있는 경우에만 상태 저장 앱을 관리합니다. Astra Control Service는 Astra Control Provisioner 또는 일반 CSI 드라이버에서 지원하는 모든 스토리지 클래스를 지원합니다.

- "GKE 클러스터용 저장소 클래스에 대해 알아봅니다"
- "AKS 클러스터용 스토리지 클래스에 대해 알아보십시오"
- "AWS 클러스터를 위한 스토리지 클래스에 대해 알아보십시오"

앱 정의

Astra Control이 클러스터에서 네임스페이스를 검색한 후 관리할 애플리케이션을 정의할 수 있습니다. 선택할 수 있습니다 하나 이상의 네임스페이스를 포괄하는 응용 프로그램을 관리합니다 또는 전체 네임스페이스를 단일 애플리케이션으로 관리합니다. 데이터 보호 작업에 필요한 세분화 수준으로 세분화됩니다.

Astra Control을 사용하면 계층 구조의 수준(네임스페이스 및 해당 네임스페이스 또는 스패닝 네임스페이스의 응용 프로그램)을 별도로 관리할 수 있지만 가장 좋은 방법은 하나 또는 다른 수준을 선택하는 것입니다. 작업이 네임스페이스 및 앱 수준에서 동시에 발생하면 Astra Control에서 수행하는 작업이 실패할 수 있습니다.

> 예를 들어, "Maria"에 대해 주간 백업 주기를 갖는 백업 정책을 설정할 수 있지만 "MariaDB"(동일한 네임스페이스)를 더 자주 백업해야 할 수 있습니다. 이러한 요구사항에 따라 단일 네임스페이스 앱이 아니라 앱을 별도로 관리해야 합니다.

시작하기 전에

- Astra Control에 Kubernetes 클러스터가 추가되었습니다.
- 클러스터에 설치된 애플리케이션 하나 이상 지원되는 앱 설치 방법에 대해 자세히 알아보십시오.
- Astra Control에 추가한 Kubernetes 클러스터의 기존 네임스페이스
- (선택 사항) Any의 Kubernetes 레이블 "지원되는 Kubernetes 리소스".

> 레이블은 식별을 위해 Kubernetes 객체에 할당할 수 있는 키/값 쌍입니다. 레이블을 사용하면 Kubernetes 오브젝트를 더 쉽게 정렬, 구성 및 찾을 수 있습니다. Kubernetes 레이블에 대해 자세히 알아보려면 "공식 Kubernetes 설명서를 참조하십시오".

이 작업에 대해

- 시작하기 전에, 또한 이해해야 합니다 "표준 및 시스템 네임스페이스 관리".
- Astra Control에서 앱과 여러 네임스페이스를 사용하려는 경우 고려해 보십시오 "네임스페이스 제약 조건을 사용하여 사용자 역할 수정" 앱을 정의하기 전에
- Astra Control API를 사용하여 앱을 관리하는 방법에 대한 지침은 을 참조하십시오 "Astra 자동화 및 API 정보".

애플리케이션 관리 옵션

- 앱으로 관리할 리소스를 정의합니다
- 앱으로 관리할 네임스페이스를 정의합니다

앱으로 관리할 리소스를 정의합니다

를 지정할 수 있습니다 "앱을 구성하는 Kubernetes 리소스" Astra Control을 통해 관리하고자 하는 것입니다. 앱을 정의하면 Kubernetes 클러스터의 요소를 단일 애플리케이션으로 그룹화할 수 있습니다. 이 Kubernetes 리소스 모음은 네임스페이스 및 레이블 선택기 기준에 따라 구성됩니다.

앱을 정의하면 클론, 스냅샷, 백업을 비롯한 Astra Control 작업에 포함할 항목을 보다 세부적으로 제어할 수 있습니다.

> 앱을 정의할 때 보호 정책이 있는 여러 앱에 Kubernetes 리소스를 포함하지 않아야 합니다. Kubernetes 리소스의 보호 정책이 중복되어 데이터 충돌이 발생할 수 있습니다.

앱 네임스페이스에 클러스터 범위 리소스를 추가하는 방법에 대해 자세히 알아보십시오.

Namespace 리소스와 연결된 클러스터 리소스 및 자동으로 포함된 Astra Control을 가져올 수 있습니다. 특정 그룹, 종류, 버전 및 레이블(선택 사항)의 리소스를 포함할 규칙을 추가할 수 있습니다. Astra Control에 자동으로 포함되지 않는 리소스가 있는 경우 이 작업을 수행할 수 있습니다.

Astra Control에 의해 자동으로 포함되는 클러스터 범위 리소스는 제외할 수 없습니다.

다음을 추가할 수 있습니다 `apiVersions` (API 버전과 결합된 그룹):

자원 종류	**apiVersions(그룹 + 버전)**
`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`	Admissions registration.k8s.io/v1
`ValidatingWebhookConfiguration`	Admissions registration.k8s.io/v1

단계

1. 응용 프로그램 페이지에서 * 정의 * 를 선택합니다.
2. 응용 프로그램 정의 * 창에서 응용 프로그램 이름을 입력합니다.
3. 응용 프로그램이 실행되는 클러스터를 * 클러스터 * 드롭다운 목록에서 선택합니다.
4. Namespace* 드롭다운 목록에서 응용 프로그램의 네임스페이스를 선택합니다.

> Astra Control을 사용하여 단일 클러스터에서 하나 이상의 지정된 네임스페이스 내에서 앱을 정의할 수 있습니다. 앱은 동일한 클러스터 내에서 여러 네임스페이스에 걸쳐 있는 리소스를 포함할 수 있습니다. Astra Control은 여러 클러스터에서 앱을 정의하는 기능을 지원하지 않습니다.

5. (선택 사항) 각 네임스페이스에서 Kubernetes 리소스에 대한 레이블을 입력합니다. 단일 레이블 또는 레이블 선택 조건(쿼리)을 지정할 수 있습니다.

> Kubernetes 레이블에 대해 자세히 알아보려면 "공식 Kubernetes 설명서를 참조하십시오".

6. (선택 사항) * 네임스페이스 추가 * 를 선택하고 드롭다운 목록에서 네임스페이스를 선택하여 앱에 대한 네임스페이스를 추가합니다.
7. (선택 사항) 추가하는 모든 추가 네임스페이스에 대한 단일 레이블 또는 레이블 선택기 조건을 입력합니다.
8. (선택 사항) Astra Control에 자동으로 포함되는 리소스 외에 클러스터 범위 리소스를 포함하려면 * 추가 클러스터 범위 리소스 포함 * 을 선택하여 다음을 완료합니다.
   a. 포함 규칙 추가 * 를 선택합니다.
   b. * Group *: 드롭다운 목록에서 리소스의 API 그룹을 선택합니다.
   c. * Kind *: 드롭다운 목록에서 개체 스키마의 이름을 선택합니다.
   d. * 버전 *: API 버전을 입력합니다.
   e. * 라벨 선택기 *: 규칙에 추가할 라벨을 선택적으로 포함합니다. 이 레이블은 이 레이블과 일치하는 리소스만 검색하는 데 사용됩니다. 레이블을 제공하지 않으면 Astra Control은 해당 클러스터에 대해 지정된 리소스 유형의 모든 인스턴스를 수집합니다.
   f. 항목에 따라 만들어진 규칙을 검토합니다.
   g. 추가 * 를 선택합니다.

> 클러스터 범위의 리소스 규칙을 원하는 만큼 만들 수 있습니다. 규칙은 애플리케이션 요약 정의에 나타납니다.

9. 정의 * 를 선택합니다.
10. 정의 * 를 선택한 후 필요에 따라 다른 앱에 대해 프로세스를 반복합니다.

앱 정의를 마치면 앱이 에 나타납니다 `Healthy` 응용 프로그램 페이지의 응용 프로그램 목록에서 상태를 지정합니다. 이제 클론을 생성하고 백업과 스냅샷을 생성할 수 있습니다.

> 방금 추가한 앱에는 Protected(보호) 열 아래에 백업이 없고 아직 백업이 예약되지 않았음을 나타내는 경고 아이콘이 있을 수 있습니다.

> 특정 앱의 세부 정보를 보려면 앱 이름을 선택합니다.

이 앱에 추가된 리소스를 보려면 * 리소스 * 탭을 선택하십시오. 리소스 열에서 리소스 이름 뒤의 숫자를 선택하거나 검색에 리소스 이름을 입력하여 추가 클러스터 범위 리소스가 포함되도록 합니다.

앱으로 관리할 네임스페이스를 정의합니다

네임스페이스의 리소스를 애플리케이션으로 정의하여 Astra Control 관리에 네임스페이스의 모든 Kubernetes 리소스를 추가할 수 있습니다. 이 방법은 앱을 개별적으로 정의하는 것이 좋습니다 "특정 네임스페이스의 모든 리소스를 관리하고 보호하려고 합니다" 비슷한 방식으로, 일정한 간격으로.

단계

1. 클러스터 페이지에서 클러스터를 선택합니다.

2. Namespaces* 탭을 선택합니다.
3. 관리하려는 앱 리소스가 포함된 네임스페이스의 작업 메뉴를 선택하고 * 응용 프로그램으로 정의 * 를 선택합니다.

> 여러 응용 프로그램을 정의하려면 네임스페이스 목록에서 선택하고 왼쪽 위 모서리에 있는 * 작업 * 버튼을 선택한 다음 * 응용 프로그램으로 정의 * 를 선택합니다. 이렇게 하면 개별 네임스페이스에 여러 개의 개별 응용 프로그램이 정의됩니다. 다중 네임스페이스 응용 프로그램의 경우 을 참조하십시오 앱으로 관리할 리소스를 정의합니다.

> 기본적으로 앱 관리에 사용되지 않는 시스템 네임스페이스를 표시하려면 * Show system namespaces * 확인란을 선택합니다. Show system namespaces "자세히 보기".

프로세스가 완료되면 해당 네임스페이스와 연결된 응용 프로그램이 '연결된 응용 프로그램' 열에 나타납니다.

시스템 네임스페이스는 어떻습니까?

Astra Control은 Kubernetes 클러스터에서 시스템 네임스페이스를 검색합니다. 기본적으로 이러한 시스템 네임스페이스는 표시되지 않습니다. 시스템 앱 리소스를 백업해야 하는 경우는 드뭅니다.

선택한 클러스터의 Namespaces 탭에서 * Show system namespaces * 확인란을 선택하여 시스템 네임스페이스를 표시할 수 있습니다.

Show system namespaces

> Astra Control 자체는 표준 앱이 아니며 "시스템 앱"입니다. Astra Control 자체를 관리하려고 해서는 안 됩니다. 관리 시 Astra Control 자체는 기본적으로 표시되지 않습니다.

스냅샷 및 백업으로 애플리케이션 보호

자동화된 보호 정책을 사용하거나 필요에 따라 스냅샷과 백업을 생성하여 앱을 보호합니다. Astra UI 또는 를 사용할 수 있습니다 "Astra Control API" 앱을 보호합니다.

에 대해 자세히 알아보십시오 "Astra Control의 데이터 보호".

앱 데이터 보호와 관련된 다음 작업을 수행할 수 있습니다.

- 보호 정책을 구성합니다
- 스냅샷을 생성합니다
- 백업을 생성합니다
- ONTAP - NAS - 경제성 작업을 위한 백업 및 복원 지원
- 변경 불가능한 백업을 생성합니다
- 스냅샷 및 백업을 봅니다
- 스냅샷을 삭제합니다
- 백업을 취소합니다

- 백업을 삭제합니다

보호 정책을 구성합니다

보호 정책은 정의된 일정에 따라 스냅샷, 백업 또는 둘 다를 생성하여 앱을 보호합니다. 시간별, 일별, 주별 및 월별 스냅샷과 백업을 생성하도록 선택할 수 있으며, 보존할 복제본 수를 지정할 수 있습니다.

시간당 한 번 이상 백업 또는 스냅샷을 자주 실행해야 하는 경우 를 수행할 수 있습니다 "Astra Control REST API를 사용하여 스냅샷과 백업을 생성합니다".

> WORM(Write Once Read Many) 버킷에 대한 변경 불가능한 백업을 생성하는 보호 정책을 정의하는 경우 백업의 보존 시간이 버킷에 대해 구성된 보존 기간보다 짧지 않은지 확인합니다.

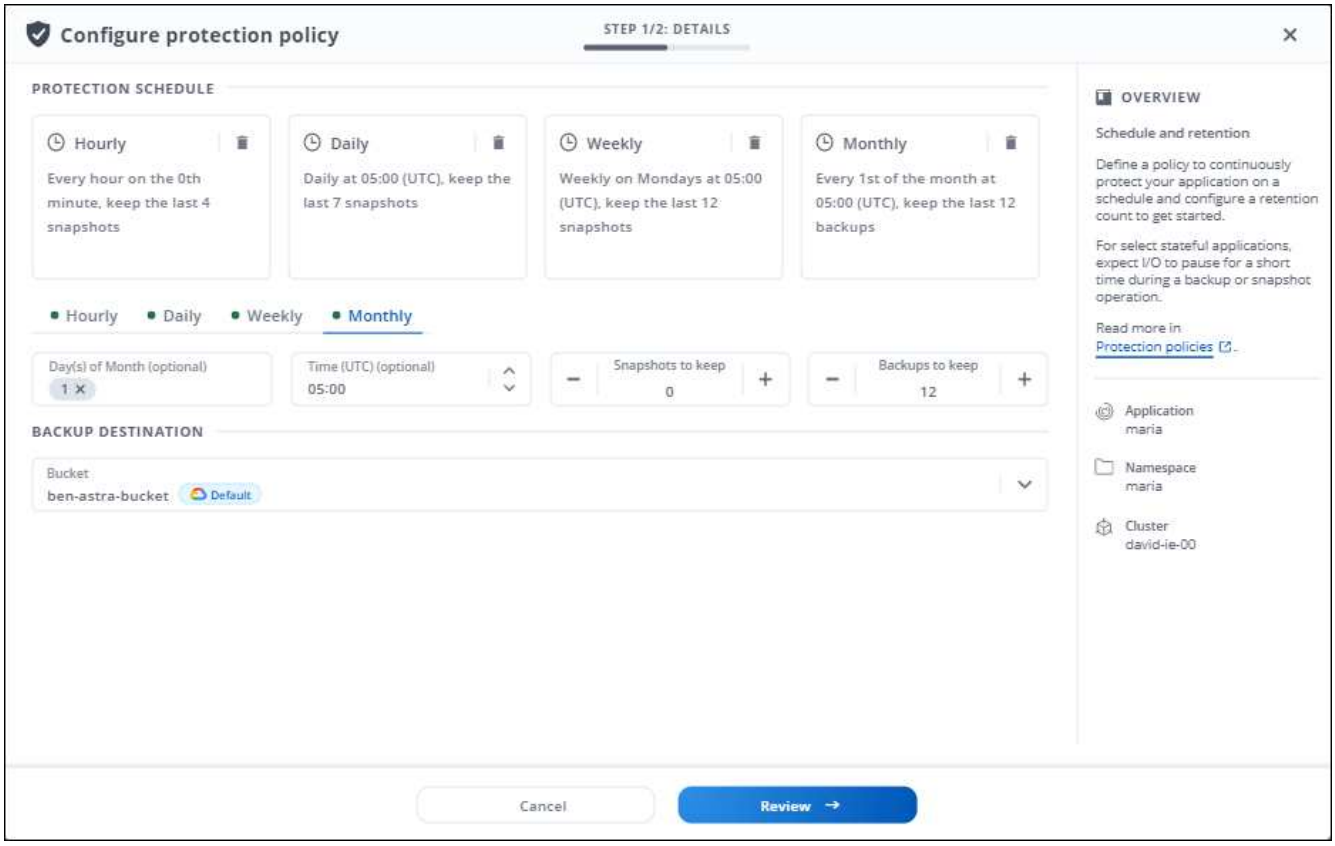
단계

1. 응용 프로그램 * 을 선택한 다음 관리되는 응용 프로그램의 이름을 선택합니다.
2. 데이터 보호 * 를 선택합니다.
3. 보호 정책 구성 * 을 선택합니다.
4. 시간별, 일별, 주별 및 월별 스케줄에 대해 유지할 스냅샷 및 백업 수를 선택하여 보호 스케줄을 정의합니다.

   시간별, 일별, 주별 및 월별 스케줄을 동시에 정의할 수 있습니다. 스냅샷 및 백업에 대한 보존 레벨을 설정하기 전에는 스케줄이 활성화되지 않습니다.

   백업의 보존 레벨을 설정할 때 백업을 저장할 버킷을 선택할 수 있습니다.

   다음 예에서는 스냅샷 및 백업의 경우 매시간, 일별, 주별 및 월별로 4개의 보호 스케줄을 설정합니다.



5. Review * 를 선택합니다.
6. 보호 정책 설정 * 을 선택합니다.

결과

Astra Control은 정의한 스케줄 및 보존 정책을 사용하여 스냅샷 및 백업을 생성하고 유지함으로써 데이터 보호 정책을 구현합니다.

스냅샷을 생성합니다

언제든지 주문형 스냅샷을 생성할 수 있습니다.

이 작업에 대해

Astra Control은 다음 드라이버를 통해 지원되는 스토리지 클래스를 사용하여 스냅샷 생성을 지원합니다.

- `ontap-nas`
- `ontap-san`
- `ontap-san-economy`

> 앱이 에서 지원하는 저장소 클래스를 사용하는 경우 `ontap-nas-economy` 드라이버, 스냅샷을 생성할 수 없습니다. 스냅샷에 대체 스토리지 클래스를 사용합니다.

단계

1. 응용 프로그램 * 을 선택합니다.
2. 원하는 앱의 * Actions * 열에 있는 옵션 메뉴에서 * Snapshot * 을 선택합니다.
3. 스냅샷 이름을 사용자 지정하고 * 다음 * 을 선택합니다.
4. 스냅샷 요약을 검토하고 * Snapshot * 을 선택합니다.

결과

스냅샷 프로세스가 시작됩니다. 데이터 보호 * > * 스냅샷 * 페이지의 * 상태 * 열에서 상태가 * 정상 * 인 경우 스냅샷이 성공합니다.

백업을 생성합니다

언제든지 앱을 백업할 수도 있습니다.

> Azure NetApp Files 스토리지에서 호스트되는 애플리케이션을 백업할 때 스토리지 공간이 처리되는 방식에 유의하십시오. 을 참조하십시오 "애플리케이션 백업" 를 참조하십시오.

> Astra Control은 다음 드라이버를 통해 지원되는 스토리지 클래스를 사용하여 백업 생성을 지원합니다.
>
> - `ontap-nas`
> - `ontap-nas-economy`
> - `ontap-san`
> - `ontap-san-economy`

이 작업에 대해

Astra Control의 버킷은 사용 가능한 용량을 보고하지 않습니다. Astra Control에서 관리되는 앱을 백업 또는 클론 복제하기 전에 적절한 스토리지 관리 시스템에서 버킷 정보를 확인하십시오.

앱이 에서 지원하는 저장소 클래스를 사용하는 경우 `ontap-nas-economy` 드라이버, 당신은 필요합니다 백업 및 복원을 활성화합니다 기능. 을(를) 정의했는지 확인합니다 `backendType` 매개 변수 을 선택합니다 "Kubernetes 스토리지 오브젝트입니다" 을 값으로 사용합니다 `ontap-nas-economy` 보호 작업을 수행하기 전에

단계

1. 응용 프로그램 * 을 선택합니다.
2. 원하는 앱의 * Actions * 열에 있는 옵션 메뉴에서 * Back Up * 을 선택합니다.
3. 백업 이름을 사용자 지정합니다.
4. 기존 스냅샷에서 앱을 백업할지 여부를 선택합니다. 이 옵션을 선택하면 기존 스냅샷 목록에서 선택할 수 있습니다.
5. 스토리지 버킷 목록에서 백업할 대상 버킷을 선택합니다.
6. 다음 * 을 선택합니다.
7. 백업 요약을 검토하고 * 백업 * 을 선택합니다.

결과

Astra Control은 앱 백업을 생성합니다.

- 네트워크에 정전이 발생했거나 비정상적으로 느린 경우 백업 작업이 시간 초과될 수 있습니다. 이로 인해 백업이 실패합니다.
- 실행 중인 백업을 취소해야 하는 경우 의 지침을 따릅니다 백업을 취소합니다. 백업을 삭제하려면 백업이 완료될 때까지 기다린 다음 의 지침을 따르십시오 백업을 삭제합니다.
- 데이터 보호 작업(클론, 백업, 복원)과 후속 영구 볼륨 크기 조정 후 UI에 새 볼륨 크기가 표시되기까지 최대 20분이 지연됩니다. 데이터 보호 작업이 몇 분 내에 성공적으로 완료되며 스토리지 백엔드에 관리 소프트웨어를 사용하여 볼륨 크기 변경을 확인할 수 있습니다.

ONTAP - NAS - 경제성 작업을 위한 백업 및 복원 지원

Astra Control Provisioner는 를 사용하는 스토리지 백엔드에 대해 설정할 수 있는 백업 및 복원 기능을 제공합니다 `ontap-nas-economy` 스토리지 클래스.

시작하기 전에

- Astra Control Provisioner 또는 Astra Trident를 활성화했습니다.
- Astra Control에서 애플리케이션을 정의했습니다. 이 응용 프로그램은 이 절차를 완료할 때까지 제한된 보호 기능을 제공합니다.
- 있습니다 `ontap-nas-economy` 스토리지 백엔드의 기본 스토리지 클래스로 선택됩니다.

구성 단계를 위해 확장합니다

1. ONTAP 스토리지 백엔드에서 다음을 수행합니다.

   a. 를 호스팅하는 SVM을 찾습니다 `ontap-nas-economy`응용 프로그램의 볼륨을 기반으로 합니다.

   b. 볼륨이 생성된 ONTAP에 연결된 터미널에 로그인합니다.

   c. SVM에 대한 스냅샷 디렉토리 숨기기:

   > 이러한 변경은 전체 SVM에 영향을 줍니다. 숨겨진 디렉토리에 계속 액세스할 수 있습니다.

```
nfs modify -vserver <svm name> -v3-hide-snapshot enabled
```

+

> ONTAP 스토리지 백엔드의 스냅샷 디렉토리가 숨겨져 있는지 확인합니다. 이 디렉토리를 숨기지 않으면 특히 NFSv3을 사용하는 경우에는 애플리케이션에 대한 액세스가 손실될 수 있습니다.

2. Astra Control Provisioner 또는 Astra Trident에서 다음을 수행합니다.

   a. ONTAP-NAS 경제적이며 애플리케이션과 연결된 각 PV에 대해 스냅샷 디렉토리를 사용할 수 있습니다.

```
tridentctl update volume <pv name> --snapshot-dir=true --pool
-level=true -n trident
```

   b. 연결된 각 PV에 대해 스냅샷 디렉토리가 활성화되었는지 확인합니다.

```
tridentctl get volume <pv name> -n trident -o yaml | grep
snapshotDir
```

   응답:

```
snapshotDirectory: "true"
```

3. Astra Control에서 연결된 모든 스냅샷 디렉토리를 활성화한 후 애플리케이션을 업데이트하여 Astra Control이 변경된 값을 인식하도록 합니다.

결과

Astra Control을 사용하여 애플리케이션을 백업 및 복원할 준비가 되었습니다. 각 PVC는 백업 및 복원을 위해 다른 응용 프로그램에서 사용할 수도 있습니다.

변경 불가능한 백업을 생성합니다

백업을 저장하는 버킷의 보존 정책에서 금지하는 한 변경 불가능한 백업은 수정, 삭제 또는 덮어쓸 수 없습니다. 보존 정책이 구성된 버킷에 애플리케이션을 백업하여 변경 불가능한 백업을 만들 수 있습니다. 을 참조하십시오 "데이터 보호" 변경 불가능한 백업 작업에 대한 중요한 정보를 참조하십시오.

시작하기 전에

보존 정책을 사용하여 대상 버킷을 구성해야 합니다. 사용하는 스토리지 공급자에 따라 이 방법이 달라집니다. 자세한 내용은 다음 스토리지 제공업체 설명서를 참조하십시오.

- * Amazon Web Services *: "버킷을 생성할 때 S3 오브젝트 잠금을 설정하고 기본 보존 기간으로 기본 보존 모드를 "거버넌스"로 설정합니다".
- * Google Cloud *: "보존 정책을 사용하여 버킷을 구성하고 보존 기간을 지정합니다".
- * Microsoft Azure *: "컨테이너 수준 범위에서 시간 기반 보존 정책을 사용하여 BLOB 스토리지 버킷을 구성합니다".
- * NetApp StorageGRID *: "버킷을 생성할 때 S3 오브젝트 잠금을 설정하고 기본 보존 기간을 사용하여 기본 보존 모드를 "규정 준수"로 설정합니다".

> Astra Control의 버킷은 사용 가능한 용량을 보고하지 않습니다. Astra Control에서 관리되는 앱을 백업 또는 클론 복제하기 전에 적절한 스토리지 관리 시스템에서 버킷 정보를 확인하십시오.

> 앱이 에서 지원하는 저장소 클래스를 사용하는 경우 `ontap-nas-economy` 드라이버, 을(를) 정의했는지 확인하십시오 `backendType` 매개 변수 을 선택합니다 "Kubernetes 스토리지 오브젝트입니다" 을 값으로 사용합니다 `ontap-nas-economy` 보호 작업을 수행하기 전에

단계

1. 응용 프로그램 * 을 선택합니다.
2. 원하는 앱의 * Actions * 열에 있는 옵션 메뉴에서 * Back Up * 을 선택합니다.
3. 백업 이름을 사용자 지정합니다.
4. 기존 스냅샷에서 앱을 백업할지 여부를 선택합니다. 이 옵션을 선택하면 기존 스냅샷 목록에서 선택할 수 있습니다.
5. 스토리지 버킷 목록에서 백업할 대상 버킷을 선택합니다. WORM(Write Once Read Many) 버킷은 버킷 이름 옆에 "잠김" 상태로 표시됩니다.

   > 버켓이 지원되지 않는 유형인 경우 버킷을 가리키거나 선택할 때 표시됩니다.

6. 다음 * 을 선택합니다.
7. 백업 요약을 검토하고 * 백업 * 을 선택합니다.

결과

Astra Control은 앱의 변경 불가능한 백업을 생성한다.

- 네트워크에 정전이 발생했거나 비정상적으로 느린 경우 백업 작업이 시간 초과될 수 있습니다. 이로 인해 백업이 실패합니다.
- 동일한 앱의 변경 불가능한 백업을 두 번 동일한 버킷에 동시에 생성하려는 경우 Astra Control이 두 번째 백업을 시작하지 못합니다. 첫 번째 백업이 완료될 때까지 기다린 후 다른 백업을 시작하십시오.
- 실행 중인 변경 불가능한 백업은 취소할 수 없습니다.
- 데이터 보호 작업(클론, 백업, 복원)과 후속 영구 볼륨 크기 조정 후 UI에 새 볼륨 크기가 표시되기까지 최대 20분이 지연됩니다. 데이터 보호 작업이 몇 분 내에 성공적으로 완료되며 스토리지 백엔드에 관리 소프트웨어를 사용하여 볼륨 크기 변경을 확인할 수 있습니다.

스냅샷 및 백업을 봅니다

Data Protection 탭에서 앱의 스냅샷 및 백업을 볼 수 있습니다.

변경 불가능한 백업은 사용 중인 버킷 옆에 "잠김" 상태로 표시됩니다.

단계

1. 응용 프로그램 * 을 선택한 다음 관리되는 응용 프로그램의 이름을 선택합니다.
2. 데이터 보호 * 를 선택합니다.

   스냅샷은 기본적으로 표시됩니다.

3. 백업 목록을 참조하려면 * backups * 를 선택합니다.

스냅샷을 삭제합니다

더 이상 필요하지 않은 예약된 스냅샷 또는 주문형 스냅샷을 삭제합니다.

단계

1. 응용 프로그램 * 을 선택한 다음 관리되는 응용 프로그램의 이름을 선택합니다.
2. 데이터 보호 * 를 선택합니다.
3. 원하는 스냅샷에 대한 * Actions * 열의 Options 메뉴에서 * Delete snapshot * 을 선택합니다.
4. 삭제를 확인하려면 "delete"라는 단어를 입력하고 * Yes, Delete snapshot * 을 선택합니다.

결과

Astra Control이 스냅샷을 삭제합니다.

백업을 취소합니다

진행 중인 백업을 취소할 수 있습니다.

백업을 취소하려면 백업이 에 있어야 합니다 `Running` 상태. 에 있는 백업은 취소할 수 없습니다 `Pending` 상태.

실행 중인 변경 불가능한 백업은 취소할 수 없습니다.

단계

1. 응용 프로그램 * 을 선택한 다음 앱 이름을 선택합니다.
2. 데이터 보호 * 를 선택합니다.
3. Backups * 를 선택합니다.
4. 원하는 백업에 대한 * Actions * 열의 Options 메뉴에서 * Cancel * 을 선택합니다.
5. 작업을 확인하려면 "취소"라는 단어를 입력하고 * 예, 백업 취소 * 를 선택합니다.

백업을 삭제합니다

더 이상 필요하지 않은 예약된 백업 또는 필요 시 백업을 삭제합니다.

> 실행 중인 백업을 취소해야 하는 경우 의 지침을 따릅니다 백업을 취소합니다. 백업을 삭제하려면 백업이 완료될 때까지 기다린 다음 이 지침을 따르십시오.

> 보존 기간이 만료되기 전에는 변경 불가능한 백업을 삭제할 수 없습니다.

단계

1. 응용 프로그램 * 을 선택한 다음 앱 이름을 선택합니다.
2. 데이터 보호 * 를 선택합니다.
3. Backups * 를 선택합니다.
4. 원하는 백업에 대한 * Actions * 열의 Options 메뉴에서 * Delete backup * 을 선택합니다.
5. 삭제를 확인하려면 "delete"라는 단어를 입력하고 * Yes, Delete backup * 을 선택합니다.

결과

Astra Control이 백업을 삭제합니다.

[기술 미리 보기] 전체 클러스터를 보호합니다

클러스터에서 관리되지 않는 네임스페이스의 예약 및 자동 백업을 생성할 수 있습니다. 이러한 워크플로우는 NetApp에서 Kubernetes 서비스 계정, 역할 바인딩 및 cron 작업으로 제공되며 Python 스크립트로 조정됩니다.

작동 방식

전체 클러스터 백업 워크플로우를 구성하고 설치하면 cron 작업이 주기적으로 실행되고 아직 관리되지 않은 모든 네임스페이스를 보호하므로 설치 중에 선택한 일정에 따라 보호 정책이 자동으로 생성됩니다.

전체 클러스터 백업 워크플로우로 클러스터의 모든 관리되지 않는 네임스페이스를 보호하지 않으려면 레이블 기반 백업 워크플로우를 활용할 수 있습니다. 레이블 기반 백업 워크플로는 또한 cron 작업을 사용하지만 관리되지 않는 모든 네임스페이스를 보호하는 대신 브론즈, 실버 또는 골드 백업 정책을 기반으로 선택적으로 네임스페이스를 보호하기 위해 제공하는 레이블을 통해 네임스페이스를 식별합니다.

선택한 워크플로의 범위에 속하는 새 네임스페이스가 만들어지면 관리자 작업 없이 자동으로 보호됩니다. 이러한 워크플로는 클러스터 단위로 구현되므로 클러스터의 중요도에 따라 각 클러스터에서 고유한 보호 수준을 가진 워크플로우를 사용할 수 있습니다.

예: 전체 클러스터 보호

예를 들어, 전체 클러스터 백업 워크플로우를 구성하고 설치하면 네임스페이스의 모든 앱이 관리자의 추가 작업 없이 주기적으로 관리 및 보호됩니다. 워크플로를 설치할 때 네임스페이스가 존재할 필요가 없습니다. 나중에 네임스페이스가 추가되면 네임스페이스가 보호됩니다.

예: 레이블 기반 보호

더 세분화하려면 레이블 기반 워크플로를 사용할 수 있습니다. 예를 들어 이 워크플로를 설치하고 필요한 보호 수준에 따라 보호할 네임스페이스에 여러 레이블 중 하나를 적용하도록 사용자에게 지시할 수 있습니다. 따라서 사용자는 이러한 레이블 중 하나로 네임스페이스를 만들 수 있으며 관리자에게 알릴 필요가 없습니다. 새로운 네임스페이스와 IT 내의 모든 앱이 자동으로 보호됩니다.

모든 네임스페이스의 예약된 백업을 생성합니다

전체 클러스터 백업 워크플로우를 사용하여 클러스터의 모든 네임스페이스에 대해 예약된 백업을 생성할 수 있습니다.

단계

1. 클러스터에 대한 네트워크 액세스 권한이 있는 시스템에 다음 파일을 다운로드합니다.
    - "components.yaml CRD 파일"
    - "protectCluster.py Python 스크립트"
2. 툴킷을 구성하고 설치하려면 "포함된 지침을 따릅니다".

특정 네임스페이스의 예약된 백업을 생성합니다

레이블 기반 백업 워크플로우를 사용하여 레이블을 기준으로 특정 네임스페이스의 예약된 백업을 만들 수 있습니다.

단계

1. 클러스터에 대한 네트워크 액세스 권한이 있는 시스템에 다음 파일을 다운로드합니다.
    - "components.yaml CRD 파일"
    - "protectCluster.py Python 스크립트"
2. 툴킷을 구성하고 설치하려면 "포함된 지침을 따릅니다".

앱 복원

Astra Control은 스냅샷 또는 백업에서 애플리케이션을 복원할 수 있습니다. 애플리케이션을 동일한 클러스터로 복구할 경우 기존 스냅샷에서 복구하는 속도가 빨라집니다. Astra Control UI 또는 를 사용할 수 있습니다 "Astra Control API" 앱을 복원합니다.

> 복원 또는 클론 작업 후에 실행되는 실행 후크에 네임스페이스 필터를 추가하고 복원 또는 클론 소스와 대상이 서로 다른 네임스페이스에 있는 경우 네임스페이스 필터는 대상 네임스페이스에만 적용됩니다.

시작하기 전에

- * 앱을 먼저 보호 *: 복원하기 전에 응용 프로그램의 스냅샷 또는 백업을 수행하는 것이 좋습니다. 이렇게 하면 복구가 실패할 경우 스냅샷 또는 백업에서 클론을 생성할 수 있습니다.
- * 대상 볼륨 확인 *: 다른 스토리지 클래스로 복원하는 경우 스토리지 클래스가 동일한 영구 볼륨 액세스 모드(예: ReadWriteMany)를 사용하는지 확인합니다. 대상 영구 볼륨 액세스 모드가 다르면 복원 작업이 실패합니다. 예를 들어, 소스 영구 볼륨에서 rwx 액세스 모드를 사용하는 경우 Azure Managed Disks, AWS EBS, Google

Persistent Disk 또는 와 같이 rwx를 제공할 수 없는 대상 스토리지 클래스를 선택합니다 `ontap-san`, 복구 작업이 실패합니다. 영구 볼륨 액세스 모드에 대한 자세한 내용은 를 참조하십시오 "쿠버네티스" 문서화:

- * 공간 요구사항 계획 *: NetApp ONTAP 스토리지를 사용하는 애플리케이션의 데이터 이동 없이 복원을 수행할 경우 복원된 앱에서 사용하는 공간이 두 배로 증가할 수 있습니다. 데이터 이동 없이 복구를 수행한 후 복구된 애플리케이션에서 원치 않는 스냅샷을 모두 제거하여 스토리지 공간을 확보합니다.
- * 지원되는 스토리지 클래스 드라이버 *: Astra Control은 다음 드라이버로 지원되는 스토리지 클래스를 사용하여 백업 복원을 지원합니다.
  - `ontap-nas`
  - `ontap-nas-economy`
  - `ontap-san`
  - `ontap-san-economy`
- * (ONTAP-NAS-이코노미 드라이버만 해당) 백업 및 복원 *: 에서 지원하는 스토리지 클래스를 사용하는 앱을 백업 또는 복원하기 전에 `ontap-nas-economy` 드라이버에서 을 확인합니다 "ONTAP 스토리지 백엔드의 스냅샷 디렉토리가 숨겨집니다". 이 디렉토리를 숨기지 않으면 특히 NFSv3을 사용하는 경우에는 애플리케이션에 대한 액세스가 손실될 수 있습니다.

> 다른 앱과 리소스를 공유하는 앱에서 데이터 이동 없이 복원 작업을 수행하면 의도하지 않은 결과가 발생할 수 있습니다. 앱 간에 공유되는 모든 리소스는 앱 중 하나에서 데이터 이동 없이 복원이 수행될 때 교체됩니다.

단계

1. 응용 프로그램 * 을 선택한 다음 앱 이름을 선택합니다.
2. 작업 열의 옵션 메뉴에서 * 복원 * 을 선택합니다.
3. 복원 유형 선택:
   - * 원래 네임스페이스로 복원 *: 이 절차를 사용하여 원래 클러스터로 응용 프로그램을 원래 상태로 복원할 수 있습니다.
     i. 앱을 원래 상태로 복원하는 데 사용할 스냅샷 또는 백업을 선택합니다. 그러면 앱이 이전 버전으로 되돌아갑니다.
     ii. 다음 * 을 선택합니다.

     > 이전에 삭제된 네임스페이스에 복원하는 경우 복원 프로세스의 일부로 동일한 이름의 새 네임스페이스가 만들어집니다. 이전에 삭제된 네임스페이스에서 앱을 관리할 권한이 있는 사용자는 새로 다시 생성된 네임스페이스에 대한 권한을 수동으로 복원해야 합니다.

   - * 새 네임스페이스로 복원 *: 이 절차를 사용하여 응용 프로그램을 다른 클러스터나 소스의 다른 네임스페이스로 복원할 수 있습니다. 이 절차를 사용하여 앱을 다른 스토리지 클래스로 마이그레이션할 수도 있습니다.
     i. 복원된 앱의 이름을 지정합니다.
     ii. 복원하려는 앱의 대상 클러스터를 선택합니다.
     iii. 앱과 연결된 각 소스 네임스페이스의 대상 네임스페이스를 입력합니다.

     > Astra Control은 이 복원 옵션의 일부로 새 대상 네임스페이스를 만듭니다. 지정한 대상 네임스페이스가 대상 클러스터에 이미 있으면 안 됩니다.

iv. 다음 * 을 선택합니다.

v. 앱을 복원하는 데 사용할 스냅샷 또는 백업을 선택합니다.

vi. 다음 * 을 선택합니다.

vii. 다음 중 하나를 선택합니다.

- * 원래 스토리지 클래스를 사용하여 복원 *: 대상 클러스터에 없는 경우 응용 프로그램은 원래 연결된 스토리지 클래스를 사용합니다. 이 경우 클러스터의 기본 스토리지 클래스가 사용됩니다.
- * 다른 스토리지 클래스를 사용하여 복구 *: 타겟 클러스터에 존재하는 스토리지 클래스를 선택합니다. 원래 연결된 스토리지 클래스에 관계없이 모든 애플리케이션 볼륨은 복구의 일부로 이 서로 다른 스토리지 클래스로 마이그레이션됩니다.

viii. 다음 * 을 선택합니다.

4. 필터링할 자원 선택:
   - * 모든 리소스 복원 *: 원래 앱과 연결된 모든 리소스를 복원합니다.
   - * 필터 리소스 *: 원래 응용 프로그램 리소스의 하위 집합을 복원하는 규칙을 지정합니다.

     i. 복원된 응용 프로그램에서 리소스를 포함하거나 제외하도록 선택합니다.

     ii. 포함 규칙 추가 * 또는 * 제외 규칙 추가 * 를 선택하고 응용 프로그램 복원 중에 올바른 리소스를 필터링하도록 규칙을 구성합니다. 규칙을 편집하거나 제거하고 구성이 올바른지 때까지 규칙을 다시 만들 수 있습니다.

> 포함 및 제외 규칙 구성에 대한 자세한 내용은 을 참조하십시오 응용 프로그램 복원 중에 리소스를 필터링합니다.

5. 다음 * 을 선택합니다.
6. 복원 작업에 대한 세부 정보를 주의 깊게 검토하고 "restore"를 입력하고(메시지가 나타나면) * Restore * 를 선택합니다.

결과

Astra Control은 사용자가 제공한 정보를 기반으로 앱을 복원합니다. 앱을 제자리에 복원한 경우 기존 영구 볼륨의 콘텐츠가 복원된 앱의 영구 볼륨 콘텐츠로 바뀝니다.

> 데이터 보호 작업(클론, 백업 또는 복원)과 후속 영구 볼륨 크기 조정 후 웹 UI에 새 볼륨 크기가 표시되기까지 최대 20분이 지연됩니다. 데이터 보호 작업이 몇 분 내에 성공적으로 완료되며 스토리지 백엔드에 관리 소프트웨어를 사용하여 볼륨 크기 변경을 확인할 수 있습니다.

> 네임스페이스 이름/ID 또는 네임스페이스 레이블에 의해 네임스페이스 제한이 있는 구성원 사용자는 동일한 클러스터 또는 조직 계정의 다른 클러스터에 있는 새 네임스페이스에 앱을 클론 복제하거나 복원할 수 있습니다. 그러나 동일한 사용자가 새 네임스페이스에서 복제되거나 복원된 앱에 액세스할 수 없습니다. 클론 또는 복원 작업에서 새 네임스페이스를 생성한 후 계정 관리자/소유자는 구성원 사용자 계정을 편집하고 영향을 받는 사용자의 역할 제약 조건을 업데이트하여 새 네임스페이스에 대한 액세스 권한을 부여할 수 있습니다.

응용 프로그램 복원 중에 리소스를 필터링합니다

에 필터 규칙을 추가할 수 있습니다 "복원" 복원된 응용 프로그램에서 포함하거나 제외할 기존 응용 프로그램 리소스를 지정하는 작업입니다. 지정된 네임스페이스, 레이블 또는 GVK(GroupVersionKind)를 기반으로 리소스를 포함하거나

제외할 수 있습니다.

포함 및 제외 시나리오에 대해 자세히 알아보세요

- * 원본 네임스페이스가 있는 포함 규칙(원본 위치 복원) * 을 선택합니다. 규칙에 정의된 기존 응용 프로그램 리소스는 삭제되며 복구에 사용하는 선택한 스냅숏 또는 백업의 리소스로 대체됩니다. 포함 규칙에 지정하지 않은 모든 리소스는 변경되지 않습니다.
- * 새 네임스페이스가 있는 포함 규칙 선택 *: 이 규칙을 사용하여 복원된 응용 프로그램에서 원하는 특정 리소스를 선택합니다. 포함 규칙에 지정하지 않은 리소스는 복원된 응용 프로그램에 포함되지 않습니다.
- * 원본 네임스페이스가 있는 제외 규칙(원본 위치 복원) * 선택: 제외하도록 지정한 리소스는 복원되지 않고 변경되지 않습니다. 제외하도록 지정하지 않은 리소스는 스냅샷 또는 백업에서 복구됩니다. 해당 StatefulSet 이 필터링된 리소스의 일부인 경우 영구 볼륨의 모든 데이터가 삭제되고 다시 생성됩니다.
- * 새 네임스페이스가 있는 제외 규칙을 선택합니다. * : 규칙을 사용하여 복원된 응용 프로그램에서 제거할 특정 리소스를 선택합니다. 제외하도록 지정하지 않은 리소스는 스냅샷 또는 백업에서 복구됩니다.

규칙은 포함 또는 제외 유형입니다. 자원 포함과 제외 를 결합하는 규칙은 사용할 수 없습니다.

단계

1. 리소스를 필터링하도록 선택하고 앱 복원 마법사에서 포함 또는 제외 옵션을 선택한 후 * 포함 규칙 추가 * 또는 * 제외 규칙 추가 * 를 선택합니다.

   Astra Control에 의해 자동으로 포함되는 클러스터 범위 리소스는 제외할 수 없습니다.

2. 필터 규칙 구성:

   적어도 하나의 네임스페이스, 레이블 또는 GVK를 지정해야 합니다. 필터 규칙을 적용한 후 유지하는 리소스가 복원된 응용 프로그램을 양호한 상태로 유지하는 데 충분한지 확인합니다.

   a. 규칙의 특정 네임스페이스를 선택합니다. 선택하지 않으면 모든 네임스페이스가 필터에 사용됩니다.

      응용 프로그램에 원래 여러 네임스페이스가 포함되어 있고 이를 새 네임스페이스로 복원하면 리소스에 포함되지 않은 네임스페이스도 모두 만들어집니다.

   b. (선택 사항) 리소스 이름을 입력합니다.

   c. (선택 사항) * 라벨 선택기 *: 포함 "라벨 선택기" 규칙에 추가합니다. 레이블 선택기는 선택한 레이블과 일치하는 자원만 필터링하는 데 사용됩니다.

   d. (선택 사항) 추가 필터링 옵션을 사용하려면 GVK(GroupVersionKind) SET * 를 선택하여 리소스 * 를 필터링합니다.

      GVK 필터를 사용하는 경우 버전 및 종류를 지정해야 합니다.

      i. (선택 사항) * Group *: 드롭다운 목록에서 Kubernetes API 그룹을 선택합니다.

      ii. * Kind *: 드롭다운 목록에서 필터에 사용할 Kubernetes 리소스 유형에 대한 오브젝트 스키마를 선택합니다.

      iii. * 버전 *: Kubernetes API 버전을 선택합니다.

3. 항목에 따라 만들어진 규칙을 검토합니다.
4. 추가 * 를 선택합니다.

> 원하는 만큼 리소스 포함 및 제외 규칙을 만들 수 있습니다. 작업을 시작하기 전에 복원 애플리케이션 요약에 규칙이 나타납니다.

애플리케이션 클론 복제 및 마이그레이션

기존 앱을 클론 복제하여 동일한 Kubernetes 클러스터 또는 다른 클러스터에 중복 앱을 생성할 수 있습니다. Astra Control은 앱을 클론할 때 애플리케이션 구성 및 영구 스토리지의 클론을 생성합니다.

Kubernetes 클러스터 간에 애플리케이션 및 스토리지를 이동해야 하는 경우 클로닝에 도움이 될 수 있습니다. 예를 들어, CI/CD 파이프라인과 Kubernetes 네임스페이스 전체에서 워크로드를 이동할 수 있습니다.

> 복원 또는 클론 작업 후에 실행되는 실행 후크에 네임스페이스 필터를 추가하고 복원 또는 클론 소스와 대상이 서로 다른 네임스페이스에 있는 경우 네임스페이스 필터는 대상 네임스페이스에만 적용됩니다.

시작하기 전에

- * 대상 볼륨 확인 *: 다른 스토리지 클래스에 클론을 생성하는 경우 스토리지 클래스가 동일한 영구 볼륨 액세스 모드(예: ReadWriteMany)를 사용하는지 확인합니다. 대상 영구 볼륨 액세스 모드가 다르면 클론 작업이 실패합니다. 예를 들어, 소스 영구 볼륨에서 rwx 액세스 모드를 사용하는 경우 Azure Managed Disks, AWS EBS, Google Persistent Disk 또는 와 같이 rwx를 제공할 수 없는 대상 스토리지 클래스를 선택합니다 `ontap-san`, 클론 작업이 실패합니다. 영구 볼륨 액세스 모드에 대한 자세한 내용은 를 참조하십시오 "쿠버네티스" 문서화:
- 앱을 다른 클러스터에 클론 복제하려면 소스 클러스터가 포함된 클라우드 인스턴스에 대한 기본 버킷을 할당해야 합니다. 소스 클라우드 인스턴스에 기본 버킷 세트가 없으면 클러스터 간 클론 작업이 실패합니다.
- 클론 작업 중에 IngressClass 리소스 또는 Webhook가 필요한 애플리케이션에는 대상 클러스터에 이미 정의된 리소스가 없어야 합니다.

클론 제한 사항

- * 명시적 스토리지 클래스 *: 스토리지 클래스가 명시적으로 설정된 앱을 배포하고 앱을 복제해야 하는 경우 타겟 클러스터에 원래 지정된 스토리지 클래스가 있어야 합니다. 명시적으로 설정된 스토리지 클래스를 가진 애플리케이션을 동일한 스토리지 클래스가 없는 클러스터로 클론 복제하면 실패합니다.
- * ONTAP - NAS - 경제적인 응용 프로그램 *: 응용 프로그램의 저장소 클래스가 에서 지원될 경우 복제 작업을 사용할 수 없습니다 `ontap-nas-economy` 드라이버. 그러나, "ONTAP - NAS - 경제성 작업을 위한 백업 및 복원 지원".
- * 클론 및 사용자 제약 조건 *: 네임스페이스 이름/ID 또는 네임스페이스 레이블에 의해 네임스페이스 제한이 있는 구성원 사용자는 동일한 클러스터의 새 네임스페이스 또는 조직 계정의 다른 클러스터에 앱을 클론 복제하거나 복원할 수 있습니다. 그러나 동일한 사용자가 새 네임스페이스에서 복제되거나 복원된 앱에 액세스할 수 없습니다. 클론 또는 복원 작업에서 새 네임스페이스를 생성한 후 계정 관리자/소유자는 구성원 사용자 계정을 편집하고 영향을 받는 사용자의 역할 제약 조건을 업데이트하여 새 네임스페이스에 대한 액세스 권한을 부여할 수 있습니다.
- * 클론은 기본 버킷 사용 *:
  - 애플리케이션 백업 또는 애플리케이션 복원 중에 사용할 버킷을 지정할 수 있습니다. 클러스터 간에 클론을 생성할 때 기본 버킷을 지정해야 하지만 같은 클러스터 내에서 클론을 생성할 때는 버킷을 지정할 필요가 없습니다.

- 클러스터 간에 클론을 생성할 때 클론 작업의 소스 클러스터가 포함된 클라우드 인스턴스에 기본 버킷 세트가 있어야 합니다.
- 클론의 버킷을 변경할 수 있는 옵션은 없습니다. 어떤 버킷이 사용되는지 제어하려는 경우 이 두 가지 방법을 사용할 수 있습니다 "버킷 기본값을 변경합니다" 또는 을 수행합니다 "백업" 뒤에 가 있습니다 "복원" 별도.

* * Jenkins CI * 사용: Jenkins CI의 운영자 배포 인스턴스를 복제하는 경우 영구 데이터를 수동으로 복원해야 합니다. 이는 앱 배포 모델의 제한 사항입니다.

단계

1. 응용 프로그램 * 을 선택합니다.
2. 다음 중 하나를 수행합니다.
   - 원하는 앱의 * Actions * 열에서 Options 메뉴를 선택합니다.
   - 원하는 앱의 이름을 선택하고 페이지 오른쪽 상단의 상태 드롭다운 목록을 선택합니다.
3. 클론 * 을 선택합니다.
4. 클론의 세부 정보 지정:
   - 이름을 입력합니다.
   - 클론의 대상 클러스터를 선택합니다.
   - 클론의 대상 네임스페이스를 입력합니다. 앱과 연결된 각 소스 네임스페이스는 대상 네임스페이스에 매핑됩니다.

     Astra Control은 클론 작업의 일부로 새 대상 네임스페이스를 생성합니다. 지정한 대상 네임스페이스가 대상 클러스터에 이미 있으면 안 됩니다.

   - 다음 * 을 선택합니다.
   - 앱과 연결된 원래 저장소 클래스를 유지하거나 다른 저장소 클래스를 선택합니다.

     앱의 스토리지 클래스를 기본 클라우드 공급자 스토리지 클래스나 기타 지원되는 스토리지 클래스로 마이그레이션하고 에서 지원하는 스토리지 클래스에서 앱을 마이그레이션할 수 있습니다 `ontap-nas-economy` 에서 지원하는 스토리지 클래스로 `ontap-nas` 또는 에서 지원하는 저장소 클래스가 있는 다른 클러스터로 앱을 복사합니다 `ontap-nas-economy` 드라이버.

   다른 스토리지 클래스를 선택했고 복원 시 이 스토리지 클래스가 존재하지 않는 경우 오류가 반환됩니다.

5. 다음 * 을 선택합니다.
6. 클론에 대한 정보를 검토하고 * Clone * 을 선택합니다.

결과

Astra Control은 사용자가 제공한 정보를 기반으로 앱을 복제합니다. 새 애플리케이션 클론이 에 있을 때 클론 작업이 성공적으로 수행됩니다 `Healthy` 상태를 표시합니다.

클론 또는 복원 작업에서 새 네임스페이스를 생성한 후 계정 관리자/소유자는 구성원 사용자 계정을 편집하고 영향을 받는 사용자의 역할 제약 조건을 업데이트하여 새 네임스페이스에 대한 액세스 권한을 부여할 수 있습니다.

앱 실행 후크 관리

실행 후크는 관리되는 앱의 데이터 보호 작업과 함께 실행되도록 구성할 수 있는 사용자 지정 작업입니다. 예를 들어 데이터베이스 앱이 있는 경우 실행 후크를 사용하여 스냅샷 전에 모든 데이터베이스 트랜잭션을 일시 중지하고 스냅샷이 완료된 후 트랜잭션을 다시 시작할 수 있습니다. 따라서 애플리케이션 정합성이 보장되는 스냅샷이 보장됩니다.

실행 후크 유형

Astra Control Service는 실행 가능한 시점을 기준으로 다음과 같은 유형의 실행 후크를 지원합니다.

- 사전 스냅샷
- 사후 스냅샷
- 사전 백업
- 백업 후
- 사후 복원

실행 후크 필터

실행 후크를 응용 프로그램에 추가하거나 편집할 때 실행 후크에 필터를 추가하여 후크가 일치시킬 컨테이너를 관리할 수 있습니다. 필터는 모든 컨테이너에서 동일한 컨테이너 이미지를 사용하는 응용 프로그램에 유용하지만 각 이미지를 다른 용도(예: Elasticsearch)로 사용할 수 있습니다. 필터를 사용하면 실행 후크가 실행되는 시나리오를 만들 수 있습니다. 단, 모든 동일한 컨테이너를 실행하는 것은 아닙니다. 단일 실행 후크에 대해 여러 개의 필터를 만들면 논리적 필터 및 연산자와 결합됩니다. 실행 후크당 최대 10개의 활성 필터를 사용할 수 있습니다.

실행 후크에 추가하는 각 필터는 클러스터의 컨테이너와 일치시키기 위해 정규식을 사용합니다. 후크가 컨테이너와 일치하면 후크는 해당 컨테이너에서 연결된 스크립트를 실행합니다. 필터에 대한 정규식은 정규식 2(RE2) 구문을 사용합니다. 이 구문은 일치 목록에서 컨테이너를 제외하는 필터를 만드는 것을 지원하지 않습니다. Astra Control이 실행 후크 필터의 정규식에 대해 지원하는 구문에 대한 자세한 내용은 을 참조하십시오 "정규식 2(RE2) 구문 지원".

> 복원 또는 클론 작업 후에 실행되는 실행 후크에 네임스페이스 필터를 추가하고 복원 또는 클론 소스와 대상이 서로 다른 네임스페이스에 있는 경우 네임스페이스 필터는 대상 네임스페이스에만 적용됩니다.

사용자 정의 실행 후크에 대한 중요 참고 사항

앱에 대한 실행 후크를 계획할 때 다음 사항을 고려하십시오.

> 실행 후크는 실행 중인 응용 프로그램의 기능을 줄이거나 완전히 비활성화하기 때문에 사용자 지정 실행 후크가 실행되는 시간을 최소화해야 합니다.
> 연결된 실행 후크와 함께 백업 또는 스냅샷 작업을 시작한 다음 취소하면 백업 또는 스냅샷 작업이 이미 시작된 경우에도 후크를 실행할 수 있습니다. 즉, 백업 후 실행 후크에 사용되는 논리는 백업이 완료된 것으로 가정할 수 없습니다.

- 새 Astra Control 구축 환경에서는 실행 후크 기능이 기본적으로 비활성화되어 있습니다.
  - 실행 후크를 사용하려면 먼저 실행 후크 기능을 활성화해야 합니다.
  - 소유자 또는 관리자 사용자는 현재 Astra Control 계정에 정의된 모든 사용자에 대해 실행 후크 기능을 활성화하거나 비활성화할 수 있습니다. 을 참조하십시오 실행 후크 기능을 활성화합니다 및 실행 후크 기능을

비활성화합니다 를 참조하십시오.

  - Astra Control 업그레이드 중에 기능 지원 상태가 유지됩니다.

- 실행 후크는 스크립트를 사용하여 작업을 수행해야 합니다. 많은 실행 후크가 동일한 스크립트를 참조할 수 있습니다.
- Astra Control에는 실행 후크가 실행 가능한 셸 스크립트 형식으로 기록하는 데 사용하는 스크립트가 필요합니다.
- 스크립트 크기는 96KB로 제한됩니다.
- Astra Control은 실행 후크 설정과 모든 일치 기준을 사용하여 스냅샷, 백업 또는 복구 작업에 적용할 수 있는 후크를 결정합니다.
- 모든 실행 후크 장애는 소프트 장애이며, 후크가 실패하더라도 다른 후크와 데이터 보호 작업은 계속 시도됩니다. 그러나 후크가 실패하면 * Activity * 페이지 이벤트 로그에 경고 이벤트가 기록됩니다.
- 실행 후크를 생성, 편집 또는 삭제하려면 소유자, 관리자 또는 구성원 권한이 있는 사용자여야 합니다.
- 실행 후크를 실행하는 데 25분 이상 걸리는 경우 후크에 장애가 발생하고 반환 코드가 "N/A"인 이벤트 로그 항목이 생성됩니다. 영향을 받는 모든 스냅샷은 시간 초과되어 실패로 표시되며, 그 결과 이벤트 로그 항목이 시간 초과를 나타냅니다.
- 임시 데이터 보호 작업의 경우 모든 후크 이벤트가 생성되고 * Activity * 페이지 이벤트 로그에 저장됩니다. 그러나 예약된 데이터 보호 작업의 경우 후크 장애 이벤트만 이벤트 로그에 기록됩니다(예약된 데이터 보호 작업 자체에서 생성되는 이벤트는 계속 기록됨).

실행 순서

데이터 보호 작업이 실행되면 실행 후크 이벤트가 다음 순서로 발생합니다.

1. 해당되는 모든 사용자 정의 사전 작업 실행 후크는 해당 컨테이너에서 실행됩니다. 필요한 만큼 사용자 지정 사전 작업 후크를 만들고 실행할 수 있지만, 이 후크의 실행 순서는 보장되거나 구성할 수 없습니다.
2. 데이터 보호 작업이 수행됩니다.
3. 해당되는 모든 사용자 지정 작업 후 실행 후크는 해당 컨테이너에서 실행됩니다. 필요한 만큼 사용자 지정 사후 작업 후크를 만들고 실행할 수 있지만 작업 후 후크의 실행 순서는 보장되거나 구성할 수 없습니다.

같은 유형의 실행 후크를 여러 개 생성하는 경우(예: 사전 스냅샷) 해당 후크의 실행 순서는 보장되지 않습니다. 그러나 다른 유형의 후크를 실행하는 순서는 보장됩니다. 예를 들어, 서로 다른 모든 유형의 후크가 있는 구성의 실행 순서는 다음과 같습니다.

1. 예비 후크가 실행되었습니다
2. 사전 스냅샷 후크가 실행되었습니다
3. 사후 스냅샷 후크가 실행되었습니다
4. 백업 후 후크가 실행되었습니다
5. 복원 후 후크가 실행되었습니다

시나리오 번호 2에서 이 구성의 예를 볼 수 있습니다 후크가 실행될지 여부를 결정합니다.

> 운영 환경에서 실행 후크 스크립트를 사용하려면 항상 해당 스크립트를 테스트해야 합니다. 'kubbeck exec' 명령을 사용하여 스크립트를 편리하게 테스트할 수 있습니다. 운영 환경에서 실행 후크를 사용하도록 설정한 후 결과 스냅샷과 백업을 테스트하여 정합성이 보장되는지 확인합니다. 앱을 임시 네임스페이스에 클론 복제하고, 스냅샷 또는 백업을 복원한 다음 앱을 테스트하여 이 작업을 수행할 수 있습니다.

다음 표를 사용하여 사용자 지정 실행 후크가 앱에 대해 실행되는지 여부를 확인할 수 있습니다.

모든 상위 수준 앱 작업은 스냅샷, 백업 또는 복원의 기본 작업 중 하나를 실행하는 것으로 구성됩니다. 시나리오에 따라 클론 작업은 이러한 작업의 다양한 조합으로 구성되므로 클론 작업이 실행되는 실행 후크는 달라집니다.

데이터 이동 없이 복원 작업을 수행하려면 기존 스냅샷 또는 백업이 필요하므로 이러한 작업은 스냅샷 또는 백업 후크를 실행하지 않습니다.

> 를 시작한 다음 스냅샷이 포함된 백업을 취소하고 연결된 실행 후크가 있는 경우 일부 후크가 실행될 수 있고 그렇지 않은 백업이 있을 수 있습니다. 즉, 백업 후 실행 후크는 백업이 완료된 것으로 가정할 수 없습니다. 연결된 실행 후크와 함께 취소된 백업의 경우 다음 사항에 유의하십시오.
>
> - 예비 백업 및 예비 후크는 항상 실행됩니다.
> - 백업에 새 스냅샷이 포함되어 있고 스냅샷이 시작된 경우 사전 스냅샷 및 사후 스냅샷 후크가 실행됩니다.
> - 스냅샷을 시작하기 전에 백업을 취소하면 사전 스냅샷 및 사후 스냅샷 후크가 실행되지 않습니다.

시나리오	작동	기존 스냅샷	더 많은 워크로드 추가/제거	네임스페이스	클러스터	스냅샷 후크가 실행됩니다	백업 후크가 실행됩니다	후크 실행을 복원합니다
1	복제	해당 없음	해당 없음	신규	동일합니다	예	해당 없음	예
2	복제	해당 없음	해당 없음	신규	다릅니다	예	예	예
3	복제 또는 복원	예	해당 없음	신규	동일합니다	해당 없음	해당 없음	예
4	복제 또는 복원	해당 없음	예	신규	동일합니다	해당 없음	해당 없음	예
5	복제 또는 복원	예	해당 없음	신규	다릅니다	해당 없음	해당 없음	예
6	복제 또는 복원	해당 없음	예	신규	다릅니다	해당 없음	해당 없음	예
7	복원	예	해당 없음	기존	동일합니다	해당 없음	해당 없음	예
8	복원	해당 없음	예	기존	동일합니다	해당 없음	해당 없음	예
9	스냅샷	해당 없음	해당 없음	해당 없음	해당 없음	예	해당 없음	해당 없음
10	백업	해당 없음	해당 없음	해당 없음	해당 없음	예	예	해당 없음
11	백업	예	해당 없음	해당 없음	해당 없음	해당 없음	해당 없음	해당 없음

실행 후크 예

를 방문하십시오 "NetApp Verda GitHub 프로젝트" Apache Cassandra 및 Elasticsearch와 같은 인기 있는 앱의 실제 실행 후크를 다운로드하려면 다음을 수행합니다. 예제를 보고 사용자 지정 실행 후크를 구조화하는 아이디어를 얻을 수도 있습니다.

실행 후크 기능을 활성화합니다

소유자 또는 관리자 사용자인 경우 실행 후크 기능을 활성화할 수 있습니다. 이 기능을 활성화하면 이 Astra Control 계정에 정의된 모든 사용자가 실행 후크를 사용하고 기존 실행 후크와 후크 스크립트를 볼 수 있습니다.

단계

1. 응용 프로그램 * 으로 이동한 다음 관리되는 응용 프로그램의 이름을 선택합니다.
2. Execution hook * 탭을 선택합니다.
3. 실행 후크 활성화 * 를 선택합니다.

   계정 * > * 기능 설정 * 탭이 나타납니다.

4. Execution Hooks * 창에서 설정 메뉴를 선택합니다.
5. 활성화 * 를 선택합니다.
6. 나타나는 보안 경고를 확인합니다.
7. Yes, enable execution hook * 를 선택합니다.

실행 후크 기능을 비활성화합니다

소유자 또는 관리자 사용자인 경우 이 Astra Control 계정에 정의된 모든 사용자에 대해 실행 후크 기능을 비활성화할 수 있습니다. 실행 후크 기능을 비활성화하려면 먼저 기존 실행 후크를 모두 삭제해야 합니다. 을 참조하십시오 실행 후크를 삭제합니다 기존 실행 후크를 삭제하는 방법에 대한 지침은 을 참조하십시오.

단계

1. 계정 * 으로 이동한 다음 * 기능 설정 * 탭을 선택합니다.
2. Execution hook * 탭을 선택합니다.
3. Execution Hooks * 창에서 설정 메뉴를 선택합니다.
4. 비활성화 * 를 선택합니다.
5. 나타나는 경고를 확인합니다.
6. 유형 `disable` 모든 사용자에 대해 이 기능을 사용하지 않도록 설정할 것인지 확인합니다.
7. 예, 사용 안 함 * 을 선택합니다.

기존 실행 후크를 봅니다

앱의 기존 사용자 지정 실행 후크를 볼 수 있습니다.

단계

1. 응용 프로그램 * 으로 이동한 다음 관리되는 응용 프로그램의 이름을 선택합니다.
2. Execution hook * 탭을 선택합니다.

   결과 목록에서 사용 가능하거나 비활성화된 실행 후크를 모두 볼 수 있습니다. 후크의 상태, 일치하는 컨테이너 수, 생성 시간 및 실행 시간(사전 또는 사후 작업)을 확인할 수 있습니다. 를 선택할 수 있습니다 + 실행할 컨테이너 목록을 확장하려면 후크 이름 옆에 있는 아이콘을 클릭합니다. 이 응용 프로그램의 실행 후크를 둘러싼 이벤트 로그를 보려면 * Activity * 탭으로 이동하십시오.

기존 스크립트 보기

업로드된 기존 스크립트를 볼 수 있습니다. 또한 이 페이지에서 사용 중인 스크립트와 해당 스크립트를 사용하는 후크를 확인할 수 있습니다.

단계

1. 계정 * 으로 이동합니다.
2. 스크립트 * 탭을 선택합니다.

   이 페이지에서는 업로드된 기존 스크립트 목록을 볼 수 있습니다. Used By* 열에는 각 스크립트를 사용하는 실행 후크가 표시됩니다.

스크립트를 추가합니다

각 실행 후크는 스크립트를 사용하여 작업을 수행해야 합니다. 실행 후크가 참조할 수 있는 스크립트를 하나 이상 추가할 수 있습니다. 많은 실행 후크가 동일한 스크립트를 참조할 수 있으므로 하나의 스크립트만 변경하여 여러 실행 후크를 업데이트할 수 있습니다.

단계

1. 실행 후크 기능이 인지 확인합니다 활성화됨.
2. 계정 * 으로 이동합니다.
3. 스크립트 * 탭을 선택합니다.
4. 추가 * 를 선택합니다.
5. 다음 중 하나를 수행합니다.
   - 사용자 지정 스크립트를 업로드합니다.
     i. 파일 업로드 * 옵션을 선택합니다.
     ii. 파일을 찾아 업로드합니다.
     iii. 스크립트에 고유한 이름을 지정합니다.
     iv. (선택 사항) 다른 관리자가 스크립트에 대해 알아야 하는 참고 사항을 입력합니다.
     v. Save script * 를 선택합니다.
   - 클립보드에서 사용자 정의 스크립트를 붙여 넣습니다.
     i. 붙여넣기 또는 형식 * 옵션을 선택합니다.
     ii. 텍스트 필드를 선택하고 필드에 스크립트 텍스트를 붙여 넣습니다.
     iii. 스크립트에 고유한 이름을 지정합니다.
     iv. (선택 사항) 다른 관리자가 스크립트에 대해 알아야 하는 참고 사항을 입력합니다.
6. Save script * 를 선택합니다.

결과

새 스크립트가 * 스크립트 * 탭의 목록에 나타납니다.

스크립트를 삭제합니다

스크립트가 더 이상 필요하지 않고 실행 후크에서 사용되지 않는 경우 시스템에서 스크립트를 제거할 수 있습니다.

단계

1. 계정 * 으로 이동합니다.
2. 스크립트 * 탭을 선택합니다.
3. 제거할 스크립트를 선택하고 * Actions * 열에서 메뉴를 선택합니다.
4. 삭제 * 를 선택합니다.

> 스크립트가 하나 이상의 실행 후크에 연결되어 있으면 * 삭제 * 작업을 사용할 수 없습니다. 스크립트를 삭제하려면 먼저 연결된 실행 후크를 편집하여 다른 스크립트에 연결합니다.

사용자 지정 실행 후크를 만듭니다

앱에 대한 사용자 정의 실행 후크를 생성하여 Astra Control에 추가할 수 있습니다. 을 참조하십시오 실행 후크 예 후크 예 실행 후크를 만들려면 소유자, 관리자 또는 구성원 권한이 있어야 합니다.

> 실행 후크로 사용할 사용자 정의 쉘 스크립트를 작성할 때는 특정 명령을 실행하거나 실행 파일에 대한 전체 경로를 제공하지 않는 한 파일 시작 부분에 적절한 셸을 지정해야 합니다.

단계

1. 실행 후크 기능이 인지 확인합니다 활성화됨.
2. 응용 프로그램 * 을 선택한 다음 관리되는 응용 프로그램의 이름을 선택합니다.
3. Execution hook * 탭을 선택합니다.
4. 추가 * 를 선택합니다.
5. 후크 세부 정보 * 영역에서:
   a. 작업 * 드롭다운 메뉴에서 작업 유형을 선택하여 후크를 언제 실행해야 하는지 결정합니다.
   b. 후크의 고유한 이름을 입력합니다.
   c. (선택 사항) 실행 중에 후크에 전달할 인수를 입력하고 각 인수 뒤에 Enter 키를 눌러 각 인수를 기록합니다.
6. (선택 사항) * Hook Filter Details * 영역에서 실행 후크가 실행되는 컨테이너를 제어하는 필터를 추가할 수 있습니다.
   a. 필터 추가 * 를 선택합니다.
   b. Hook filter type * 열의 드롭다운 메뉴에서 필터링할 특성을 선택합니다.
   c. Regex * 열에 필터로 사용할 정규식을 입력합니다. Astra Control은 를 사용합니다 "정규식 2(RE2) regex 구문".

      > 정규식 필드에 다른 텍스트가 없는 특성(예: pod 이름)의 정확한 이름을 필터링하면 부분 문자열 일치가 수행됩니다. 정확한 이름과 해당 이름만 일치시키려면 정확한 문자열 일치 구문(예: `^exact_podname$`)를 클릭합니다.

   d. 필터를 더 추가하려면 * 필터 추가 * 를 선택합니다.

> 실행 후크에 대한 여러 필터가 논리 및 연산자와 결합됩니다. 실행 후크당 최대 10개의 활성 필터를 사용할 수 있습니다.

7. 완료되면 * Next * 를 선택합니다.
8. Script * 영역에서 다음 중 하나를 수행합니다.
   - 새 스크립트를 추가합니다.
     i. 추가 * 를 선택합니다.
     ii. 다음 중 하나를 수행합니다.
       - 사용자 지정 스크립트를 업로드합니다.
         I. 파일 업로드 * 옵션을 선택합니다.
         II. 파일을 찾아 업로드합니다.
         III. 스크립트에 고유한 이름을 지정합니다.
         IV. (선택 사항) 다른 관리자가 스크립트에 대해 알아야 하는 참고 사항을 입력합니다.
         V. Save script * 를 선택합니다.
       - 클립보드에서 사용자 정의 스크립트를 붙여 넣습니다.
         I. 붙여넣기 또는 형식 * 옵션을 선택합니다.
         II. 텍스트 필드를 선택하고 필드에 스크립트 텍스트를 붙여 넣습니다.
         III. 스크립트에 고유한 이름을 지정합니다.
         IV. (선택 사항) 다른 관리자가 스크립트에 대해 알아야 하는 참고 사항을 입력합니다.
   - 목록에서 기존 스크립트를 선택합니다.

     이렇게 하면 실행 후크에 이 스크립트를 사용하도록 지시합니다.
9. 다음 * 을 선택합니다.
10. 실행 후크 구성을 검토합니다.
11. 추가 * 를 선택합니다.

실행 후크의 상태를 확인합니다

스냅샷, 백업 또는 복원 작업이 실행된 후에 작업의 일부로 실행된 실행 후크의 상태를 확인할 수 있습니다. 이 상태 정보를 사용하여 실행 후크를 유지할지, 수정하거나 삭제할 것인지 결정할 수 있습니다.

단계

1. 응용 프로그램 * 을 선택한 다음 관리되는 응용 프로그램의 이름을 선택합니다.
2. 데이터 보호 * 탭을 선택합니다.
3. 스냅샷 * 을 선택하여 실행 중인 스냅샷을 보거나 * 백업 * 을 선택하여 실행 중인 백업을 확인합니다.

   후크 상태 * 는 작업이 완료된 후 실행 후크의 상태를 표시합니다. 상태 위로 마우스를 가져가면 자세한 정보를 볼 수 있습니다. 예를 들어, 스냅샷 중에 실행 후크 오류가 발생한 경우 해당 스냅샷의 후크 상태 위로 마우스를 이동하면 실패한 실행 후크 목록이 표시됩니다. 각 오류의 원인을 확인하려면 왼쪽 탐색 영역의 * Activity * 페이지를 확인하십시오.

스크립트 사용을 봅니다

Astra Control 웹 UI에서 특정 스크립트를 사용하는 실행 후크를 확인할 수 있습니다.

단계

1. 계정 * 을 선택합니다.
2. 스크립트 * 탭을 선택합니다.

   스크립트 목록의 * Used By * 열에 목록의 각 스크립트를 사용하는 후크에 대한 세부 정보가 포함되어 있습니다.

3. 관심 있는 스크립트에 대해 * Used By *(사용 대상 *) 열에서 정보를 선택합니다.

   스크립트를 사용하는 후크의 이름 및 스크립트를 실행하도록 구성된 작업 유형과 함께 더 자세한 목록이 나타납니다.

실행 후크를 편집합니다

실행 후크를 편집하여 속성, 필터 또는 사용하는 스크립트를 변경할 수 있습니다. 실행 후크를 편집하려면 소유자, 관리자 또는 구성원 권한이 있어야 합니다.

단계

1. 응용 프로그램 * 을 선택한 다음 관리되는 응용 프로그램의 이름을 선택합니다.
2. Execution hook * 탭을 선택합니다.
3. 편집할 후크의 경우 * Actions * 열에서 옵션 메뉴를 선택합니다.
4. 편집 * 을 선택합니다.
5. 필요한 사항을 변경하고 각 섹션을 완료한 후 * 다음 * 을 선택합니다.
6. 저장 * 을 선택합니다.

실행 후크를 비활성화합니다

앱 스냅샷 전후에 실행 후크가 실행되지 않도록 임시로 설정하려면 실행 후크를 사용하지 않도록 설정할 수 있습니다. 실행 후크를 비활성화하려면 소유자, 관리자 또는 구성원 권한이 있어야 합니다.

단계

1. 응용 프로그램 * 을 선택한 다음 관리되는 응용 프로그램의 이름을 선택합니다.
2. Execution hook * 탭을 선택합니다.
3. 비활성화할 후크의 경우 * Actions * 열에서 옵션 메뉴를 선택합니다.
4. 비활성화 * 를 선택합니다.

실행 후크를 삭제합니다

더 이상 필요 없는 경우 실행 후크를 완전히 제거할 수 있습니다. 실행 후크를 삭제하려면 소유자, 관리자 또는 구성원 권한이 있어야 합니다.

단계

1. 응용 프로그램 * 을 선택한 다음 관리되는 응용 프로그램의 이름을 선택합니다.

2. Execution hook * 탭을 선택합니다.
3. 삭제할 후크의 경우 * Actions * 열에서 옵션 메뉴를 선택합니다.
4. 삭제 * 를 선택합니다.
5. 결과 대화 상자에 "delete"를 입력하여 확인합니다.
6. 예, 실행 후크 삭제 * 를 선택합니다.

를 참조하십시오

- "NetApp Verda GitHub 프로젝트"

앱 및 컴퓨팅 상태 보기

앱 및 클러스터 상태 요약 보기

대시보드 * 를 클릭하여 앱, 클러스터 및 해당 상태를 한눈에 파악할 수 있습니다.

앱 타일은 다음을 식별하는 데 도움이 됩니다.

- 현재 관리하고 있는 앱 수
- 관리된 앱이 정상 상태인지 여부
- 애플리케이션이 완전히 보호되는지 여부(최근 백업을 사용할 수 있는 경우 보호됨)

이러한 항목은 단순한 숫자 또는 상태가 아니라 각 상태별로 드릴다운할 수 있습니다. 예를 들어 앱이 완전히 보호되지 않은 경우 아이콘 위로 마우스를 가져가면 완전히 보호되지 않은 앱을 확인할 수 있습니다. 여기에는 이유가 포함됩니다.

클러스터 타일은 클러스터 상태에 대한 유사한 세부 정보를 제공하며 앱을 사용하는 것처럼 드릴다운하여 자세한 정보를 얻을 수 있습니다.

클러스터의 상태 및 세부 정보를 봅니다

Astra Control에 Kubernetes 클러스터를 추가한 후에는 클러스터의 위치, 작업자 노드, 영구 볼륨 및 스토리지 클래스 등과 같은 클러스터에 대한 세부 정보를 볼 수 있습니다.

단계

1. Astra Control Service UI에서 * Clusters * 를 선택합니다.
2. 클러스터 * 페이지에서 세부 정보를 확인할 클러스터를 선택합니다.

> 클러스터가 "이동된" 상태에 있지만 클러스터 및 네트워크 연결이 양호한 것으로 나타나는 경우(Kubernetes API를 사용하여 클러스터에 액세스하려는 외부 시도가 성공한 경우), Astra Control에 제공한 kubeconfig는 더 이상 유효하지 않을 수 있습니다. 클러스터의 인증서 순환 또는 만료 때문일 수 있습니다. 이 문제를 해결하려면 을 사용하여 Astra Control의 클러스터와 연결된 자격 증명을 업데이트하십시오 "Astra Control API를 참조하십시오".

3. Overview *, * Storage * 및 * Activity * 탭에서 원하는 정보를 확인할 수 있습니다.
   - * 개요 *: 해당 상태를 포함한 작업자 노드에 대한 세부 정보.

- * 스토리지 *: 스토리지 클래스 및 상태를 비롯하여 컴퓨팅과 연관된 영구 볼륨입니다.
- * 활동 *: 클러스터와 관련된 활동

> Astra Control Service * Dashboard * 부터 클러스터 정보를 볼 수도 있습니다. 리소스 요약 * 의 * 클러스터 * 탭에서 * 클러스터 * 페이지로 이동하는 관리 클러스터를 선택할 수 있습니다. 클러스터 * 페이지로 이동한 후 위에 설명된 단계를 따릅니다.

앱의 상태 및 세부 정보를 봅니다

앱 관리를 시작하면 Astra Control은 앱에 대한 세부 정보를 제공하여 통신 상태(Astra Control이 앱과 통신할 수 있는지 여부), 보호 상태(장애 발생 시 완전히 보호되는지 여부), Pod, 영구 스토리지 등을 식별할 수 있도록 합니다.

단계

1. 응용 프로그램 * 을 선택한 다음 앱 이름을 선택합니다.
2. 원하는 정보를 찾습니다.

앱 상태

Astra Control이 애플리케이션과 통신할 수 있는지 여부를 나타내는 상태를 제공한다.

앱 보호 상태

앱이 얼마나 잘 보호되는지 상태를 제공합니다.

- * 완전 보호 *: 이 앱에는 활성 백업 스케줄과 1주일 미만의 성공적인 백업이 있습니다
- * 부분 보호됨 *: 응용 프로그램에 활성 백업 일정, 활성 스냅샷 일정 또는 백업 또는 스냅샷이 있습니다
- * 보호되지 않음 *: 완전히 보호되거나 부분적으로 보호되지 않는 앱

  _최근 백업 _ 이(가) 있을 때까지 완전히 보호할 수 없습니다. 백업은 영구 볼륨으로부터 멀리 떨어진 개체 저장소에 저장되기 때문에 이 작업이 중요합니다. 장애 또는 사고로 인해 클러스터가 삭제되며 영구적 저장소인 경우 복구할 백업이 필요합니다. 스냅샷을 사용하면 복구할 수 없습니다.

개요

앱과 연결된 Pod의 상태에 대한 정보입니다.

데이터 보호

데이터 보호 정책을 구성하고 기존 스냅샷 및 백업을 볼 수 있습니다.

스토리지

에는 애플리케이션 레벨의 영구 볼륨이 나와 있습니다. 영구 볼륨의 상태는 Kubernetes 클러스터의 관점에서 나옵니다.

리소스

백업 및 관리되는 리소스를 확인할 수 있습니다.

활동입니다

앱과 관련된 Astra Control 활동입니다.

버킷을 관리합니다

Astra가 백업 및 클론에 사용하는 버킷을 관리할 수 있습니다. 클라우드 인스턴스에서 추가 버킷을 추가하고, 기존 버킷을 제거하고, Kubernetes 클러스터의 기본 버킷을 변경할 수 있습니다.

소유자 및 관리자만 버킷을 관리할 수 있습니다.

Astra Control이 버킷을 사용하는 방식

클라우드 인스턴스를 위한 첫 번째 Kubernetes 클러스터 관리를 시작할 때 Astra Control Service는 이를 위한 초기 버킷을 생성합니다 "클라우드 인스턴스".

수동으로 버킷을 클라우드 인스턴스의 기본 버킷으로 지정할 수 있습니다. 이 경우 Astra Control Service는 해당 클라우드 인스턴스의 관리되는 클러스터에서 생성한 백업 및 클론에 대해 기본적으로 이 버킷을 사용합니다(백업에 다른 버킷을 선택할 수 있음). 클라우드 인스턴스의 관리되는 클러스터에서 다른 클러스터로 애플리케이션의 라이브 클론을 수행하는 경우 Astra Control Service는 소스 클라우드 인스턴스에 대한 기본 버킷을 사용하여 클론 작업을 수행합니다.

여러 클라우드 인스턴스의 기본 버킷과 동일한 버킷을 설정할 수 있습니다.

보호 정책을 생성하거나 임시 백업을 시작할 때 모든 버킷에서 선택할 수 있습니다.

> Astra Control Service는 백업 또는 클론을 시작하기 전에 대상 버킷에 액세스할 수 있는지 여부를 확인합니다.

기존 버킷을 봅니다

Astra Control Service에서 사용할 수 있는 버킷 목록을 보고 해당 상태를 확인하고 클라우드 인스턴스의 기본 버킷(정의된 경우)을 식별합니다.

버킷은 다음 상태 중 하나를 가질 수 있습니다.

보류 중
: 버킷을 추가하면 보류 중인 상태에서 시작되고 Astra Control이 이를 검색합니다.

사용 가능
: 이 버킷은 Astra Control에서 사용할 수 있습니다.

제거되었습니다
: 현재는 버킷이 작동하지 않습니다. 상태 아이콘 위에 마우스를 올려 놓으면 문제가 무엇인지 확인할 수 있습니다.

  버킷이 제거된 상태인 경우에도 기본 버킷으로 설정하고 보호 일정에 할당할 수 있습니다. 하지만 데이터 보호 작업이 시작될 때 버킷이 가용 상태가 아니면 해당 작업이 실패합니다.

단계

1. 버킷 * 으로 이동합니다.

   Astra Control Service에서 사용할 수 있는 버킷 목록이 표시됩니다.

추가 버킷을 추가합니다

언제든지 추가 버킷을 추가할 수 있습니다. 따라서 보호 정책을 생성하거나 임시 백업을 시작할 때 버킷 중에서 선택할 수 있으며, 클라우드 인스턴스에서 사용하는 기본 버킷을 변경할 수 있습니다.

다음 유형의 버킷을 추가할 수 있습니다.

- Amazon Web Services에서 직접 지원합니다
- 일반 S3
- Google 클라우드 플랫폼
- Microsoft Azure를 참조하십시오
- NetApp ONTAP S3
- NetApp StorageGRID S3

시작하기 전에

- 기존 버킷의 이름을 알고 있어야 합니다.
- Astra Control에 버킷 관리에 필요한 권한을 제공하는 버킷에 대한 자격 증명이 있어야 합니다.
- 버킷이 Microsoft Azure에 있는 경우:
  - 버킷은 _Astra-backup-rg_이라는 리소스 그룹에 속해야 합니다.
  - Azure 저장소 계정 인스턴스 성능 설정이 "프리미엄"으로 설정된 경우 "프리미엄 계정 유형" 설정을 "Blob 차단 "으로 설정해야 합니다.

단계

1. 버킷 * 으로 이동합니다.
2. 추가 * 를 선택하고 프롬프트에 따라 버킷을 추가합니다.
   - * 유형 *: 클라우드 공급자를 선택하십시오.
   - * 기존 버킷 이름 *: 버킷의 이름을 입력합니다.
   - * 설명 *: 버킷에 대한 설명을 선택적으로 입력합니다.
     - * 저장소 계정 * (Azure에만 해당): Azure 저장소 계정의 이름을 입력합니다. 이 버킷은 이름이 _Astra-backup-rg_인 리소스 그룹에 속해야 합니다.
     - * S3 서버 이름 또는 IP 주소 * (AWS 및 S3 버킷 유형만 해당): 해당 지역에 해당하는 S3 엔드포인트의 정규화된 도메인 이름을 에 입력합니다 `https://`. 을 참조하십시오 "아마존 문서" 를 참조하십시오.
     - * 자격 증명 선택 *: Astra Control Service에 버킷 관리에 필요한 권한을 제공하는 자격 증명을 입력합니다. 제공해야 하는 정보는 버킷 유형에 따라 다릅니다.
       a. 버킷을 추가하려면 * 추가 * 를 선택합니다.

결과

Astra Control Service가 버킷을 추가합니다. 이제 보호 정책을 생성하거나 임시 백업을 수행할 때 이 버킷을 선택할 수 있습니다. 이 버킷을 클라우드 인스턴스의 기본 버킷으로 설정할 수도 있습니다.

기본 버킷을 변경합니다

클라우드 인스턴스의 기본 버킷을 변경할 수 있습니다. Astra Control Service는 기본적으로 백업 및 클론에 이 버킷을 사용합니다. 각 클라우드 인스턴스에는 고유한 기본 버킷이 있습니다.

> Astra Control은 클라우드 인스턴스에 대해 기본 버킷을 자동으로 할당하지 않습니다. 두 클러스터 간에 애플리케이션 클론 작업을 수행하기 전에 클라우드 인스턴스의 기본 버킷을 수동으로 설정해야 합니다.

단계

1. 클라우드 인스턴스 * 로 이동합니다.
2. 편집할 클라우드 인스턴스의 * 작업 * 열에서 구성 메뉴를 선택합니다.
3. 편집 * 을 선택합니다.
4. 버킷 목록에서 이 클라우드 인스턴스의 기본 버킷을 만들 버킷을 선택합니다.
5. Update * 를 선택합니다.

버킷을 탈거하십시오

더 이상 사용하지 않거나 상태가 불량한 버킷을 제거할 수 있습니다. 오브젝트 저장소 구성을 단순하고 최신 상태로 유지하기 위해 이 작업을 수행할 수 있습니다.

> - 기본 버킷을 제거할 수 없습니다. 해당 버킷을 제거하려면 먼저 다른 버킷을 기본값으로 선택하십시오.
> - 버킷의 클라우드 공급자 보존 기간이 만료되기 전에는 WORM(Write Once Read Many) 버킷을 제거할 수 없습니다. 웜 버킷은 버킷 이름 옆에 "잠김"으로 표시됩니다.

시작하기 전에

- 시작하기 전에 이 버킷에 대해 실행 중이거나 완료된 백업이 없는지 확인해야 합니다.
- 예약된 백업에 버킷이 사용되지 않는지 확인해야 합니다.

있는 경우 계속할 수 없습니다.

단계

1. 버킷 * 으로 이동합니다.
2. Actions * 메뉴에서 * Remove * 를 선택합니다.

> Astra Control은 먼저 버킷에 백업을 사용하는 스케줄 정책이 없고 제거할 버킷에 활성 백업이 없음을 보장합니다.

3. 작업을 확인하려면 "remove"를 입력합니다.
4. 예, 버킷 제거 * 를 선택합니다.

자세한 내용을 확인하십시오

- "Astra Control API를 사용합니다"

실행 중인 작업을 모니터링합니다

지난 24시간 동안 Astra Control에서 완료, 실패 또는 취소된 작업 및 실행 작업에 대한 세부 정보를 볼 수 있습니다. 예를 들어 실행 중인 백업, 복원 또는 클론 작업의 상태를 보고 완료율 및 남은 예상 시간과 같은 세부 정보를 볼 수 있습니다. 가 실행된 예약된 작업의 상태 또는 수동으로 시작한 작업을 볼 수 있습니다.

실행 중이거나 완료된 작업을 보는 동안 작업 세부 정보를 확장하여 각 하위 작업의 상태를 볼 수 있습니다. 진행 중이거나 완료된 작업의 경우 작업 진행률 표시줄이 녹색이고, 취소된 작업의 경우 파란색이고, 오류로 인해 실패한 작업의 경우 빨간색입니다.

> 클론 작업의 경우 작업 하위 작업은 스냅샷과 스냅샷 복구 작업으로 구성됩니다.

실패한 작업에 대한 자세한 내용은 을 참조하십시오 "계정 활동을 모니터링합니다".

단계

1. 작업을 실행하는 동안 * 응용 프로그램 * 으로 이동합니다.
2. 목록에서 응용 프로그램의 이름을 선택합니다.
3. 응용 프로그램의 세부 정보에서 * 작업 * 탭을 선택합니다.

   현재 또는 과거 작업의 세부 정보를 보고 작업 상태별로 필터링할 수 있습니다.

> 태스크는 최대 24시간 동안 * 작업 * 목록에 유지됩니다. 을 사용하여 이 제한 및 기타 작업 모니터 설정을 구성할 수 있습니다 "Astra Control API를 참조하십시오".

계정을 관리합니다

대금 청구를 설정합니다

Astra Control Service 계정 청구를 관리하기 위해 두 가지 이상의 방법을 사용할 수 있습니다. Azure 또는 Amazon AWS를 사용하는 경우 Microsoft Azure Marketplace 또는 AWS Marketplace를 통해 Astra Control Service 계획을 신청할 수 있습니다. 이렇게 하면 마켓플레이스를 통해 청구 세부 정보를 관리할 수 있습니다. 또는 NetApp을 통해 직접 가입할 수 있습니다. NetApp에 직접 구독하면 Astra Control Service를 통해 청구 세부 정보를 관리할 수 있습니다. 가입 없이 Astra Control Service를 사용하는 경우 무료 요금제에 자동으로 가입됩니다.

Astra Control Service Free Plan을 사용하면 계정에서 최대 10개의 네임스페이스를 관리할 수 있습니다. 10개 이상의 네임스페이스를 관리하려면 무료 요금제로 업그레이드할 때 또는 Azure Marketplace 또는 AWS Marketplace를 통해 구독하여 청구를 설정해야 합니다.

대금 청구 개요

Astra Control Service와 관련된 비용에는 두 가지 유형이 있습니다. Astra Control Service에는 NetApp이 비용을 청구하며, 영구 볼륨 및 오브젝트 스토리지에 대해서는 클라우드 공급자가 비용을 청구합니다.

Astra Control 서비스 청구

Astra Control Service는 다음과 같은 세 가지 계획을 제공합니다.

무료 플랜

최대 10개의 네임스페이스를 무료로 관리할 수 있습니다.

프리미엄 페이고

네임스페이스당 특정 속도로 무제한의 네임스페이스 관리

프리미엄 구독

namespace pack _ 당 최대 20개의 네임스페이스를 관리할 수 있는 연간 구독으로 할인된 요금으로 선결제합니다. NetApp 세일즈 팀에 문의하여 조직에 필요한 만큼 팩을 구매합니다. 예를 들어 Astra Control Service에서 60개의 네임스페이스를 관리하려면 3개의 팩을 구입해야 합니다. 연간 구독에서 허용하는 것보다 많은 네임스페이스를 관리하는 경우 추가 네임스페이스당 구독 종속 초과 요금이 부과됩니다. 아직 Astra Control 계정이 없는 경우 Premium Subscription을 구매하면 자동으로 Astra Control 계정이 생성됩니다. 기존 무료 플랜이 있는 경우 프리미엄 구독으로 자동 변환됩니다.

Astra Control 계정을 만들면 무료 플랜이 자동으로 가입됩니다. Astra Control의 대시보드는 현재 허용 가능한 10개의 네임스페이스 중 관리하고 있는 네임스페이스 수를 보여 줍니다. 네임스페이스가 포함된 첫 번째 앱이 관리되는 경우 네임스페이스에 대한 청구가 시작되고 네임스페이스가 포함된 마지막 앱이 관리되지 않을 때 해당 네임스페이스에 대한 청구가 중지됩니다.

11번째 네임스페이스를 관리하려고 하면 Astra Control이 Free Plan의 한계에 도달했음을 알려줍니다. 그런 다음 무료 플랜에서 프리미엄 플랜으로 업그레이드하라는 메시지가 표시됩니다. 추가 네임스페이스당 구독 종속 초과 요금이 부과됩니다.

언제든지 프리미엄 요금제로 업그레이드할 수 있습니다. 업그레이드 후 Astra Control은 계정의 _ALL_Namespaces에 대해 귀하에게 요금을 부과하기 시작합니다. 처음 10개 네임스페이스는 무료 요금제에 머무르고 있지 않습니다.

Google Cloud 청구

영구 볼륨은 NetApp Cloud Volumes Service를 통해 지원되며, 앱의 백업은 Google 클라우드 스토리지 버킷에 저장됩니다.

- "Cloud Volumes Service에 대한 가격 세부 정보를 봅니다".

  Astra Control Service는 모든 서비스 유형과 서비스 수준을 지원합니다. 사용하는 서비스 유형은 에 따라 다릅니다 "Google Cloud 지역".

- "Google Cloud 스토리지 버킷의 가격 세부 정보를 확인하십시오".

Microsoft Azure 청구

영구 볼륨은 Azure NetApp Files에 의해 백업되고 앱 백업은 Azure Blob 컨테이너에 저장됩니다.

- "Azure NetApp Files에 대한 가격 세부 정보를 봅니다".
- "Microsoft Azure Blob 스토리지의 가격 세부 정보를 봅니다".
- "Azure 마켓플레이스에서 Astra Control Service 계획 및 가격을 확인하십시오"

> Astra Control Service에 대한 Azure 청구 요금은 시간당 청구되며, 사용 시간의 29분이 경과한 후 새로운 청구 시간이 시작됩니다.

Amazon Web Services 청구

영구 볼륨은 EBS 또는 FSx for NetApp ONTAP에 의해 백업되고 앱 백업은 AWS 버킷에 저장됩니다.

- "Amazon Web Services에 대한 가격 세부 정보를 봅니다".

Azure Marketplace에서 **Astra Control Service**를 구독하십시오

Azure Marketplace를 사용하여 Astra Control Service에 가입할 수 있습니다. 계정 및 청구 세부 정보는 마켓플레이스를 통해 관리됩니다.

> Azure Marketplace 구독 프로세스에 대한 비디오 안내를 보려면 를 방문하십시오 "NetApp TV를 참조하십시오".

단계

1. 로 이동합니다 "Azure 마켓플레이스 를 참조하십시오".
2. 지금 다운로드 * 를 선택합니다.
3. 지침에 따라 계획을 구독합니다.

AWS Marketplace에서 **Astra Control Service**를 구독하십시오

AWS Marketplace를 사용하여 Astra Control Service를 구독할 수 있습니다. 계정 및 청구 세부 정보는 마켓플레이스를 통해 관리됩니다.

단계

1. 로 이동합니다 "AWS 마켓플레이스 를 참조하십시오".
2. 구매 옵션 보기 * 를 선택합니다.
3. 메시지가 표시되면 AWS 계정에 로그인하거나 새 계정을 만듭니다.
4. 지침에 따라 계획을 구독합니다.

NetApp에 직접 **Astra Control Service**를 구독하십시오

Astra Control Service UI 내에서 또는 NetApp 세일즈 팀에 연락하여 Astra Control Service를 구독할 수 있습니다.

무료 요금제의 프리미엄 페이고 요금제로 업그레이드하세요

언제든지 결제 계획을 업그레이드하여 Astra Control에서 10개 이상의 네임스페이스를 관리할 수 있습니다. 유효한 신용 카드만 있으면 됩니다.

단계

1. 계정 * 을 선택한 다음 * 청구 * 를 선택합니다.
2. 요금제 * 에서 * Premium PayGo * 로 이동하여 * 지금 업그레이드 * 를 선택합니다.
3. 유효한 신용 카드에 대한 결제 세부 정보를 제공하고 * 프리미엄 요금제로 업그레이드 * 를 선택하십시오.

Astra Control은 신용 카드 만료일이 다가오면 이메일로 발송합니다.

결과

이제 10개 이상의 네임스페이스를 관리할 수 있습니다. Astra Control은 현재 관리하고 있는 _ all_namespaces에 대해 충전 작업을 시작합니다.

무료 요금제로부터 프리미엄 요금제로 업그레이드하세요

NetApp 세일즈 팀에 연락하여 연간 구독과 함께 할인된 요금으로 사전 결제할 수 있습니다.

단계

1. 계정 * 을 선택한 다음 * 청구 * 를 선택합니다.
2. 요금제 * 에서 * 프리미엄 이용 * 으로 이동하여 * 판매 연락처 * 를 선택합니다.
3. 프로세스를 시작할 수 있도록 세일즈 팀에 세부 정보를 제공합니다.

결과

NetApp 세일즈 담당자가 구매 주문서를 처리하기 위해 연락을 드릴 것입니다. 주문이 완료되면 Astra Control은 * 청구 * 탭에 현재 계획을 반영합니다.

현재 비용 및 청구 기록을 봅니다

Astra Control은 현재 월별 비용 및 네임스페이스별 상세 청구 내역을 보여 줍니다. 마켓플레이스를 통해 요금제를 구독한 경우 청구 내역이 표시되지 않지만 마켓플레이스에 로그인하여 확인할 수 있습니다.

단계

1. 계정 * 을 선택한 다음 * 청구 * 를 선택합니다.

   현재 비용이 청구 개요 아래에 표시됩니다.

2. 네임스페이스로 청구 내역을 보려면 * 청구 내역 * 을 선택합니다.

   Astra Control은 각 네임스페이스의 사용 시간 및 비용을 보여 줍니다. 사용 시간은 Astra Control이 청구 기간 동안 네임스페이스를 관리하는 시간(분)입니다.

3. 드롭다운 목록을 선택하여 이전 달을 선택합니다.

Premium PayGo의 신용 카드를 변경합니다

필요한 경우 Astra Control이 청구하기 위해 파일에 가지고 있는 신용 카드를 변경할 수 있습니다.

단계

1. 계정 > 청구 > 결제 방법 * 을 선택합니다.
2. 구성 아이콘을 선택합니다.
3. 신용 카드를 수정합니다.

중요 참고 사항

- 귀하의 청구 계획은 Astra Control 계정입니다.

  계정이 여러 개인 경우 각 계정마다 자체 청구 계획이 있습니다.

- Astra Control 청구서에는 네임스페이스 관리에 대한 비용이 포함되어 있습니다. 영구 볼륨의 스토리지 백엔드는 클라우드 공급자가 별도로 요금을 부과합니다.

  "Astra Control 가격에 대해 자세히 알아보십시오".

- 각 청구 기간은 해당 월의 마지막 날에 종료됩니다.
- 프리미엄 요금제의 경우 무료 요금제로 다운그레이드할 수 없습니다.

사용자를 초대하고 제거합니다

Astra Control 계정에 가입하도록 사용자를 초대하고 더 이상 계정에 액세스할 수 없는 사용자를 제거합니다.

사용자를 초대합니다

계정 소유자와 관리자는 다른 사용자를 Astra Control 계정에 가입하도록 초대할 수 있습니다.

단계

1. 사용자에게 가 있는지 확인합니다 "BlueXP 로그인".
2. 계정 * 을 선택합니다.
3. 사용자 * 탭에서 * 초대 * 를 선택합니다.
4. 사용자의 이름, 이메일 주소 및 역할을 입력합니다.

   다음 사항에 유의하십시오.

   - 이메일 주소는 사용자가 BlueXP에 등록하는 데 사용한 이메일 주소와 일치해야 합니다.
   - 각 역할은 다음과 같은 권한을 제공합니다.
     - 소유자 * 는 관리자 권한을 가지며 계정을 삭제할 수 있습니다.
     - Admin * 은 구성원 권한을 가지며 다른 사용자를 초대할 수 있습니다.
     - 회원 * 은 앱과 클러스터를 완벽하게 관리할 수 있습니다.
     - Viewer * 는 리소스를 볼 수 있습니다.
5. 멤버 또는 뷰어 역할이 있는 사용자에게 제약 조건을 추가하려면 * 제약 조건으로 역할 제한 * 확인란을 활성화합니다.

   제약 조건 추가에 대한 자세한 내용은 을 참조하십시오 "역할을 관리합니다".

6. 다른 사용자를 초대하려면 * 다른 사용자 추가 * 를 선택하고 새 사용자에 대한 정보를 입력합니다.

   한 번에 최대 10명의 사용자를 초대할 수 있습니다. 사용자 초대 * 대화 상자의 왼쪽에서 초대하는 사용자 사이를 탐색할 수 있습니다.

7. 사용자 초대 * 를 선택합니다.

결과

사용자 또는 사용자는 계정에 가입하도록 초대하는 이메일을 받게 됩니다.

사용자의 역할을 변경합니다

계정 소유자는 모든 사용자의 역할을 변경할 수 있고 계정 관리자는 관리자, 구성원 또는 뷰어 역할을 가진 사용자의 역할을 변경할 수 있습니다.

단계

1. 계정 * 을 선택합니다.
2. 사용자 * 탭의 * 작업 * 열에서 해당 사용자의 메뉴를 선택합니다.
3. 역할 편집 * 을 선택합니다.
4. 새 역할을 선택합니다.
5. 멤버 또는 뷰어 역할이 있는 사용자에게 제약 조건을 추가하려면 * 제약 조건으로 역할 제한 * 확인란을 활성화합니다.

   제약 조건 추가에 대한 자세한 내용은 을 참조하십시오 "역할을 관리합니다".

6. Confirm * 을 선택합니다.

결과

Astra Control은 선택한 새 역할에 따라 사용자의 권한을 업데이트합니다.

사용자를 제거합니다

소유자 역할을 가진 사용자는 언제든지 계정에서 다른 사용자를 제거할 수 있습니다.

단계

1. 계정 * 을 선택합니다.
2. 사용자 * 탭에서 제거할 사용자를 선택합니다.
3. Actions * 열에서 메뉴를 선택하고 * Remove user * 를 선택합니다.
4. 메시지가 표시되면 "remove"를 입력하여 삭제를 확인한 다음 * 예, 사용자 제거 * 를 선택합니다.

결과

Astra Control은 사용자를 계정에서 제거합니다.

역할을 관리합니다

네임스페이스 제약 조건을 추가하고 이러한 제약 조건에 대한 사용자 역할을 제한하여 역할을 관리할 수 있습니다. 이렇게 하면 조직 내의 리소스에 대한 액세스를 제어할 수 있습니다. Astra Control UI 또는 를 사용할 수 있습니다 "Astra Control API" 역할을 관리합니다.

역할에 네임스페이스 제약 조건을 추가합니다

관리자 또는 소유자 사용자는 네임스페이스 제약 조건을 추가할 수 있습니다.

단계

1. 계정 관리 * 탐색 영역에서 * 계정 * 을 선택합니다.
2. 사용자 * 탭을 선택합니다.
3. Actions * 열에서 Member 또는 Viewer 역할을 가진 사용자의 메뉴 버튼을 선택합니다.
4. 역할 편집 * 을 선택합니다.
5. 제약 조건으로 역할 제한 * 확인란을 활성화합니다.

   이 확인란은 구성원 또는 뷰어 역할에만 사용할 수 있습니다. 역할 * 드롭다운 목록에서 다른 역할을 선택할 수 있습니다.

6. 구속 조건 추가 * 를 선택합니다.

   네임스페이스 또는 네임스페이스 레이블별로 사용 가능한 제약 조건 목록을 볼 수 있습니다.

7. 네임스페이스 구성 방법에 따라 * 제약 조건 유형 * 드롭다운 목록에서 * Kubernetes 네임스페이스 * 또는 * Kubernetes 네임스페이스 레이블 * 을 선택합니다.
8. 목록에서 하나 이상의 네임스페이스 또는 레이블을 선택하여 해당 네임스페이스로 역할을 제한하는 제약 조건을 구성합니다.
9. Confirm * 을 선택합니다.

   역할 편집 * 페이지에는 이 역할에 대해 선택한 제약 조건 목록이 표시됩니다.

10. Confirm * 을 선택합니다.

    계정 * 페이지의 * 역할 * 열에서 구성원 또는 뷰어 역할에 대한 제약 조건을 볼 수 있습니다.

> 역할에 대한 제약 조건을 설정하고 제약 조건을 추가하지 않고 * 확인 * 을 선택하면 역할이 전체 제한 사항으로 간주됩니다(역할에 네임스페이스가 할당된 리소스에 대한 액세스가 거부됨).

역할에서 네임스페이스 제약 조건을 제거합니다

관리자 또는 소유자 사용자는 역할에서 네임스페이스 제약 조건을 제거할 수 있습니다.

단계

1. 계정 관리 * 탐색 영역에서 * 계정 * 을 선택합니다.
2. 사용자 * 탭을 선택합니다.
3. Actions * 열에서 활성 제약 조건이 있는 Member 또는 Viewer 역할을 가진 사용자의 메뉴 버튼을 선택합니다.
4. 역할 편집 * 을 선택합니다.

   역할 편집 * 대화 상자에 해당 역할에 대한 활성 제약 조건이 표시됩니다.

5. 제거할 구속 조건의 오른쪽에 있는 * X * 를 선택합니다.

6. Confirm * 을 선택합니다.

를 참조하십시오

- "사용자 역할 및 네임스페이스"

자격 증명을 추가 및 제거합니다

언제든지 계정에서 클라우드 공급자 자격 증명을 추가 및 제거할 수 있습니다. Astra Control은 이러한 자격 증명을 사용하여 Kubernetes 클러스터, 클러스터에 있는 앱을 검색하고 대신 리소스를 프로비저닝합니다.

Astra Control의 모든 사용자는 동일한 자격 증명 세트를 공유합니다.

자격 증명을 추가합니다

Astra Control에 자격 증명을 추가하는 가장 일반적인 방법은 클러스터를 관리하는 것이지만 계정 페이지에서 자격 증명을 추가할 수도 있습니다. 그러면 추가 Kubernetes 클러스터를 관리할 때 자격 증명을 선택할 수 있습니다.

시작하기 전에

- Amazon Web Services의 경우 클러스터를 생성하는 데 사용되는 IAM 계정에 대한 자격 증명의 JSON 출력이 있어야 합니다. "IAM 사용자 설정 방법을 알아봅니다".
- GKE의 경우 필요한 권한이 있는 서비스 계정에 대한 서비스 계정 키 파일이 있어야 합니다. "서비스 계정 설정 방법에 대해 알아보십시오".
- AKS의 경우 서비스 보안 주체를 생성할 때 Azure CLI의 출력이 포함된 JSON 파일이 있어야 합니다. "서비스 보안 주체를 설정하는 방법에 대해 알아봅니다".

  JSON 파일에 추가하지 않은 경우 Azure 구독 ID도 필요합니다.

단계

1. 계정 > 자격 증명 * 을 선택합니다.
2. 자격 증명 추가 * 를 선택합니다.
3. Microsoft Azure * 를 선택합니다.
4. Google Cloud Platform * 을 선택합니다.
5. Amazon Web Services * 를 선택합니다.
6. Astra Control의 다른 자격 증명과 구별되는 자격 증명의 이름을 입력합니다.
7. 필요한 자격 증명을 입력합니다.
8. * Microsoft Azure *: JSON 파일을 업로드하거나 클립보드에서 해당 JSON 파일의 내용을 붙여넣어 Azure 서비스 교장에게 자세한 정보를 Astra Control에 제공합니다.

   JSON 파일에는 서비스 보안 주체를 생성할 때 Azure CLI의 출력이 포함되어야 합니다. 또한 구독 ID를 포함할 수 있으므로 Astra Control에 자동으로 추가됩니다. 그렇지 않으면 JSON을 제공한 후 ID를 수동으로 입력해야 합니다.

9. * Google Cloud Platform *: 파일을 업로드하거나 클립보드의 콘텐츠를 붙여넣어 Google Cloud 서비스 계정 키 파일을 제공합니다.

10. * Amazon Web Services *: 파일을 업로드하거나 클립보드의 콘텐츠를 붙여넣어 Amazon Web Services IAM 사용자 자격 증명을 제공합니다.
11. 자격 증명 추가 * 를 선택합니다.

결과

이제 Astra Control에 클러스터를 추가할 때 자격 증명을 선택할 수 있습니다.

자격 증명을 제거합니다

언제든지 계정에서 자격 증명을 제거합니다. 자격 증명은 이후에 제거해야 합니다 "모든 클러스터 관리를 취소합니다 "자격 증명을 순환하지 않는 경우(참조 자격 증명 회전)를 클릭합니다.

> Astra Control에 추가하는 첫 번째 자격 증명 세트는 항상 사용 중입니다. Astra Control은 자격 증명을 사용하여 백업 버킷에 인증하기 때문입니다. 이러한 자격 증명을 제거하지 않는 것이 좋습니다.

단계

1. 계정 > 자격 증명 * 을 선택합니다.
2. 제거할 자격 증명에 대한 * 상태 * 열의 드롭다운 목록을 선택합니다.
3. 제거 * 를 선택합니다.
4. 삭제할 자격 증명의 이름을 입력한 다음 * 예, 자격 증명 제거 * 를 선택합니다.

결과

Astra Control은 계정에서 자격 증명을 제거합니다.

자격 증명 회전

계정에서 자격 증명을 회전할 수 있습니다. 자격 증명을 회전하는 경우 백업이 진행 중인 상태(예약 또는 필요 시)가 없을 때 유지 관리 창에서 자격 증명을 회전합니다.

단계

1. 의 단계에 따라 기존 자격 증명을 제거합니다 자격 증명을 제거합니다.
2. 의 단계에 따라 새 자격 증명을 추가합니다 자격 증명을 추가합니다.
3. 새 자격 증명을 사용하도록 모든 버킷을 업데이트합니다.
    a. 왼쪽 탐색 창에서 * Bucket * 을 선택합니다.
    b. 편집할 버킷의 * Actions * 열에서 드롭다운 목록을 선택합니다.
    c. 편집 * 을 선택합니다.
    d. 자격 증명 선택 * 섹션에서 Astra Control에 추가한 새 자격 증명을 선택합니다.
    e. Update * 를 선택합니다.
    f. 시스템에 남아 있는 버킷에 대해 * b * ~ * e * 단계를 반복합니다.

결과

Astra Control은 새로운 클라우드 공급자 자격 증명을 사용하기 시작합니다.

계정 활동을 모니터링합니다

Astra Control 계정의 활동에 대한 세부 정보를 볼 수 있습니다. 예를 들어, 새 사용자를 초대하거나, 클러스터를 추가하거나, 스냅샷을 생성할 때 사용할 수 있습니다. 계정 활동을 CSV 파일로 내보낼 수도 있습니다.

Astra Control에서 모든 계정 활동을 봅니다

1. Activity * 를 선택합니다.
2. 필터를 사용하여 활동 목록의 범위를 좁히거나 검색 상자를 사용하여 원하는 항목을 정확하게 찾을 수 있습니다.
3. CSV로 내보내기 * 를 선택하여 계정 활동을 CSV 파일로 다운로드합니다.

특정 앱의 계정 활동을 봅니다

1. 응용 프로그램 * 을 선택한 다음 앱 이름을 선택합니다.
2. Activity * 를 선택합니다.

클러스터의 계정 활동을 봅니다

1. 클러스터 * 를 선택한 다음 클러스터 이름을 선택합니다.
2. Activity * 를 선택합니다.

알림을 보고 관리합니다

Astra Control은 작업이 완료되거나 실패했을 때 알려줍니다. 예를 들어, 앱 백업이 성공적으로 완료되면 알림이 표시됩니다.

인터페이스의 오른쪽 상단에서 읽지 않은 알림 수를 사용할 수 있습니다.

이러한 알림을 보고 읽은 상태로 표시할 수 있습니다. 이렇게 하면 읽지 않은 알림을 지우는 것이 편리합니다.

단계

1. 오른쪽 상단에서 읽지 않은 알림 수를 선택합니다.
2. 알림을 검토한 후 * 읽은 상태로 표시 * 또는 * 모든 알림 표시 * 를 선택합니다.

   모든 알림 표시 * 를 선택한 경우 알림 페이지가 로드됩니다.

3. 알림 * 페이지에서 알림을 보고 읽음으로 표시할 알림을 선택하고 * 작업 * 을 선택한 다음 * 읽음으로 표시 * 를 선택합니다.

계정을 닫습니다

Astra Control 계정이 더 이상 필요하지 않으면 언제든지 계정을 종료할 수 있습니다.

> 계정을 닫으면 Astra Control이 자동으로 생성한 버킷이 자동으로 삭제됩니다.

단계

1. "모든 앱 및 클러스터 관리를 취소합니다".

2. "Astra Control에서 자격 증명을 제거합니다".
3. 계정 > 청구 > 결제 방법 * 을 선택합니다.
4. 계정 닫기 * 를 선택합니다.
5. 계정 이름을 입력하고 확인 을 클릭하여 계정을 닫습니다.

클라우드 인스턴스 관리

클라우드 인스턴스는 클라우드 공급자 내의 고유한 도메인입니다. 각 클라우드 공급자에 대해 여러 클라우드 인스턴스를 생성할 수 있으며 각 클라우드 인스턴스에는 고유한 이름, 자격 증명 및 관련 클러스터가 있습니다.

Astra Control에 새 클러스터를 추가할 때 클라우드 인스턴스를 생성합니다. Astra Control UI를 사용하여 클라우드 인스턴스를 편집하여 이름 또는 기본 버킷을 변경하고, Astra Control API를 사용하여 클라우드 인스턴스에서 다른 작업을 수행할 수 있습니다.

클라우드 인스턴스를 추가합니다

Astra Control에 새 클러스터를 추가할 때 새 클라우드 인스턴스를 추가할 수 있습니다. 을 참조하십시오 "Astra Control Service에서 Kubernetes 클러스터 관리를 시작합니다" 를 참조하십시오.

클라우드 인스턴스를 편집합니다

클라우드 공급자의 기존 클라우드 인스턴스를 수정할 수 있습니다.

단계

1. 클라우드 인스턴스 * 로 이동합니다.
2. 클라우드 인스턴스 목록에서 편집하려는 클라우드 인스턴스의 * 작업 * 메뉴를 선택합니다.
3. 편집 * 을 선택합니다.

   이 페이지에서 클라우드 인스턴스의 이름과 기본 버킷을 업데이트할 수 있습니다.

   Astra Control의 각 클라우드 인스턴스는 고유한 이름을 가져야 합니다.

클라우드 인스턴스의 자격 증명을 회전합니다

Astra Control API를 사용하여 클라우드 인스턴스의 자격 증명을 회전할 수 있습니다. 자세한 내용은 "Astra 자동화 문서로 이동합니다".

클라우드 인스턴스를 제거합니다

Astra Control API를 사용하여 클라우드 공급자에서 클라우드 인스턴스를 제거할 수 있습니다. 자세한 내용은 "Astra 자동화 문서로 이동합니다".

Astra Control Provisioner를 활성화합니다

Astra Trident 버전 23.10 이상에는 라이센스를 보유한 Astra Control 사용자가 고급 스토리지 프로비저닝 기능에 액세스할 수 있도록 Astra Control Provisioner를 사용하는 옵션이 포함되어 있습니다. Astra Control Provisioner는 표준 Astra Trident CSI 기반 기능과 더불어 이 확장 기능을 제공합니다. 이 절차를 사용하여 Astra Control Provisioner를 활성화하고 설치할 수 있습니다.

Astra Control Service 구독에는 Astra Control Provisioner 사용에 대한 라이센스가 자동으로 포함됩니다.

향후 Astra Control 업데이트에서 Astra Control Provisioner는 Astra Trident를 스토리지 프로비저닝 및 오케스트레이터로 대체하며 Astra Control을 사용하려면 필수입니다. 따라서 Astra Control 사용자가 Astra Control Provisioner를 활성화하는 것이 좋습니다. Astra Trident는 오픈 소스를 계속 유지하며, NetApp의 새로운 CSI 및 기타 기능으로 릴리즈, 유지, 지원 및 업데이트될 것입니다.

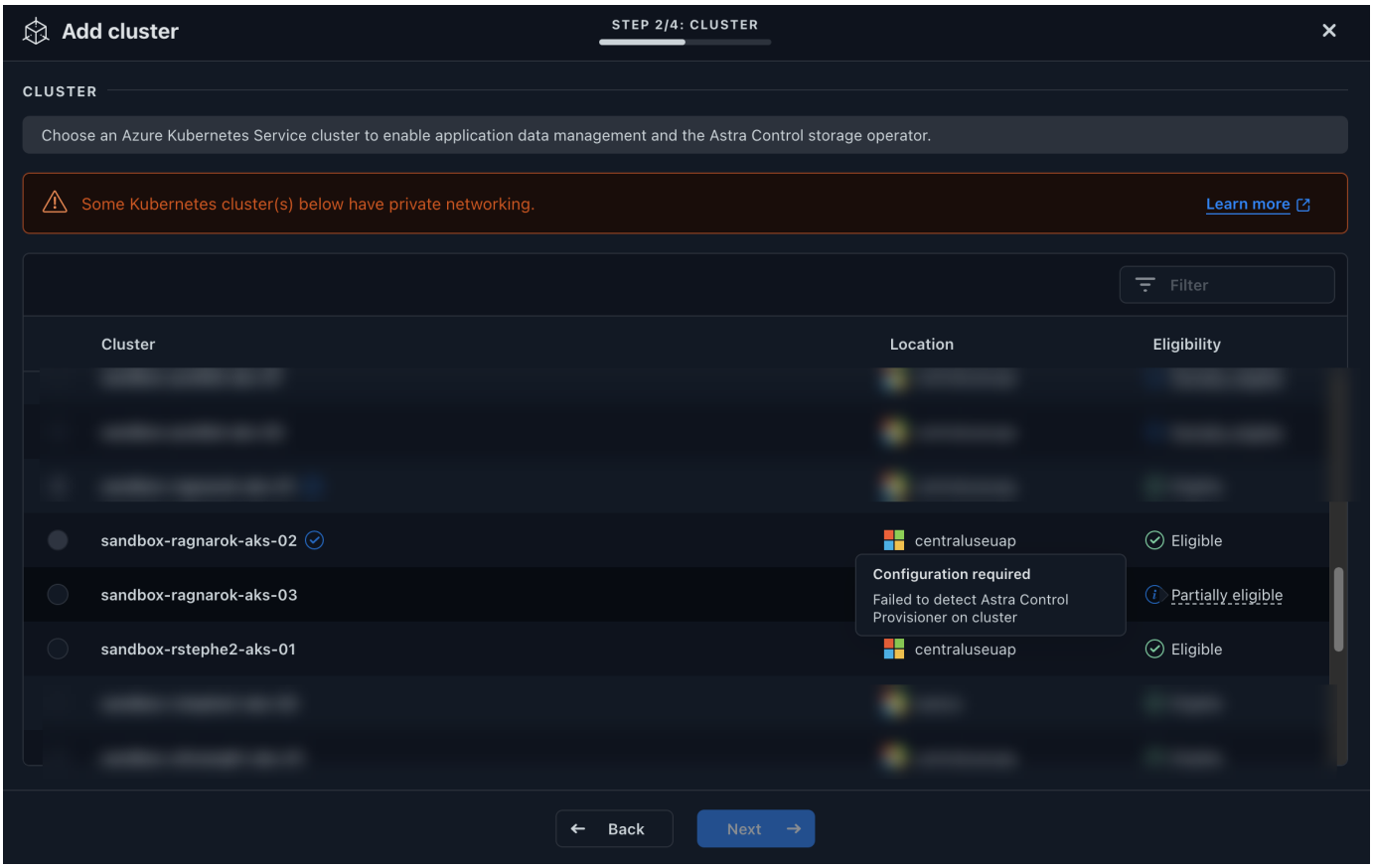
Astra Control Provisioner를 활성화해야 하는지 어떻게 알 수 있습니까**?**

이전에 Astra Trident를 설치하지 않은 클러스터를 Astra Control Service에 추가하면 클러스터가 로 표시됩니다 `Eligible`. 먼저 해 "클러스터를 Astra Control에 추가합니다", Astra Control Provisioner가 자동으로 활성화됩니다.

클러스터가 표시되어 있지 않은 경우 `Eligible`표시됩니다 `Partially eligible` 다음 중 하나로 인해 발생:

- 이전 버전의 Astra Trident를 사용하고 있습니다
- Provisioner 옵션이 아직 활성화되지 않은 Astra Trident 23.10을 사용하고 있습니다
- 자동 활성화를 허용하지 않는 클러스터 유형입니다

용 `Partially eligible` 경우 다음 지침을 사용하여 클러스터에 Astra Control Provisioner를 수동으로 활성화하십시오.



Astra Control Provisioner를 활성화하기 전에

Astra Control Provisioner가 없는 기존 Astra Trident가 있고 Astra Control Provisioner를 활성화하려면 먼저 다음을 수행합니다.

- * Astra Trident를 설치한 경우 해당 버전이 4개의 릴리즈 창 내에 있는지 확인 *: Astra Trident가 버전 24.02의 4개의 릴리즈 창 내에 있는 경우 Astra Control Provisioner를 사용하여 Astra Trident 24.02로 직접 업그레이드할 수 있습니다. 예를 들어, Astra Trident 23.04에서 24.02로 직접 업그레이드할 수 있습니다.
- * 클러스터에 AMD64 시스템 아키텍처가 있는지 확인 *: Astra Control Provisioner 이미지는 AMD64 및 ARM64 CPU 아키텍처 모두에서 제공되지만 Astra Control에서는 AMD64만 지원됩니다.

단계

1. NetApp Astra Control 이미지 레지스트리에 액세스:
   a. Astra Control Service UI에 로그온하고 Astra Control 계정 ID를 기록합니다.
      i. 페이지 오른쪽 상단의 그림 아이콘을 선택합니다.
      ii. API 액세스 * 를 선택합니다.
      iii. 계정 ID를 기록합니다.
   b. 같은 페이지에서 * API 토큰 생성 * 을 선택하고 API 토큰 문자열을 클립보드에 복사하여 편집기에 저장합니다.
   c. 원하는 방법을 사용하여 Astra Control 레지스트리에 로그인합니다.

```
docker login cr.astra.netapp.io -u <account-id> -p <api-token>
```

```
crane auth login cr.astra.netapp.io -u <account-id> -p <api-token>
```

2. (사용자 지정 레지스트리에만 해당) 이미지를 사용자 지정 레지스트리로 이동하려면 다음 단계를 수행하십시오. 레지스트리를 사용하지 않는 경우 의 Trident 운영자 단계를 따르십시오 다음 섹션을 참조하십시오.

> 다음 명령에 Docker 대신 Podman을 사용할 수 있습니다. Windows 환경을 사용하는 경우 PowerShell을 사용하는 것이 좋습니다.

Docker 를 참조하십시오

a. 레지스트리에서 Astra Control Provisioner 이미지를 가져옵니다.

> 가져온 이미지는 여러 플랫폼을 지원하지 않으며 Linux AMD64와 같이 이미지를 가져온 호스트와 동일한 플랫폼만 지원합니다.

```
docker pull cr.astra.netapp.io/astra/trident-acp:24.02.0
--platform <cluster platform>
```

예:

```
docker pull cr.astra.netapp.io/astra/trident-acp:24.02.0
--platform linux/amd64
```

b. 이미지에 태그 지정:

```
docker tag cr.astra.netapp.io/astra/trident-acp:24.02.0
<my_custom_registry>/trident-acp:24.02.0
```

c. 이미지를 사용자 지정 레지스트리에 푸시합니다.

```
docker push <my_custom_registry>/trident-acp:24.02.0
```

크레인

a. Astra Control Provisioner 매니페스트를 사용자 지정 레지스트리에 복사합니다.

```
crane copy cr.astra.netapp.io/astra/trident-acp:24.02.0
<my_custom_registry>/trident-acp:24.02.0
```

3. 원래의 Astra Trident 설치 방법에 가 있는지 확인합니다.
4. 원래 사용한 설치 방법을 사용하여 Astra Trident에서 Astra Control Provisioner를 활성화합니다.

Astra Trident 운영자

a. "Astra Trident 설치 프로그램을 다운로드하여 압축을 풉니다".

b. Astra Trident를 아직 설치하지 않았거나 원본 Astra Trident 구축에서 연산자를 제거한 경우 다음 단계를 완료하십시오.

   i. CRD 생성:

```
kubectl create -f
deploy/crds/trident.netapp.io_tridentorchestrators_crd_post1.1
6.yaml
```

   ii. 트라이덴트 네임스페이스를 만듭니다 (`kubectl create namespace trident`) 또는 트리덴트 네임스페이스가 여전히 존재하는지 확인합니다 (`kubectl get all -n trident`)를 클릭합니다. 네임스페이스가 제거된 경우 다시 만듭니다.

c. Astra Trident를 24.02.0으로 업데이트:

> Kubernetes 1.24 이하 버전을 실행하는 클러스터의 경우, 를 사용합니다 `bundle_pre_1_25.yaml`. Kubernetes 1.25 이상을 실행하는 클러스터의 경우 를 사용합니다 `bundle_post_1_25.yaml`.

```
kubectl -n trident apply -f trident-installer/deploy/<bundle-
name.yaml>
```

d. Astra Trident가 실행 중인지 확인합니다.

```
kubectl get torc -n trident
```

응답:

```
NAME       AGE
trident    21m
```

e.] 비밀을 사용하는 레지스트리가 있는 경우 Astra Control Provisioner 이미지를 가져오는 데 사용할 비밀을 만듭니다.

```
kubectl create secret docker-registry <secret_name> -n trident
--docker-server=<my_custom_registry> --docker-username=<username>
--docker-password=<token>
```

f. TridentOrchestrator CR을 편집하고 다음과 같이 편집합니다.

```
kubectl edit torc trident -n trident
```

i. Astra Trident 이미지에 대한 사용자 지정 레지스트리 위치를 설정하거나 Astra Control 레지스트리에서 가져옵니다 (`tridentImage: <my_custom_registry>/trident:24.02.0` 또는 `tridentImage: netapp/trident:24.02.0`)를 클릭합니다.

ii. Astra Control Provisioner를 활성화합니다 (`enableACP: true`)를 클릭합니다.

iii. Astra Control Provisioner 이미지의 사용자 지정 레지스트리 위치를 설정하거나 Astra Control 레지스트리에서 가져옵니다 (`acpImage: <my_custom_registry>/trident-acp:24.02.0` 또는 `acpImage: cr.astra.netapp.io/astra/trident-acp:24.02.0`)를 클릭합니다.

iv. 를 설정했는지 확인합니다 이미지 풀 암호 이 절차의 앞부분에서 여기에서 설정할 수 있습니다 (`imagePullSecrets: - <secret_name>`)를 클릭합니다. 이전 단계에서 설정한 것과 동일한 이름 암호 이름을 사용합니다.

```
apiVersion: trident.netapp.io/v1
kind: TridentOrchestrator
metadata:
  name: trident
spec:
  debug: true
  namespace: trident
  tridentImage: <registry>/trident:24.02.0
  enableACP: true
  acpImage: <registry>/trident-acp:24.02.0
  imagePullSecrets:
  - <secret_name>
```

g. 파일을 저장하고 종료합니다. 배포 프로세스가 자동으로 시작됩니다.

h. 운영자, 배포 및 복제 세트가 생성되었는지 확인합니다.

```
kubectl get all -n trident
```

> Kubernetes 클러스터에는 운영자의 인스턴스 * 하나가 있어야 합니다. Astra Trident 연산자를 여러 번 구축해서는 안 됩니다.

i. 를 확인합니다 `trident-acp` 컨테이너가 실행 중이며 `acpVersion` 있습니다 `24.02.0` 의 상태입니다 `Installed`:

```
kubectl get torc -o yaml
```

응답:

```
status:
  acpVersion: 24.02.0
  currentInstallationParams:
    ...
    acpImage: <registry>/trident-acp:24.02.0
    enableACP: "true"
    ...
  ...
  status: Installed
```

tridentctl 을 선택합니다

a. "Astra Trident 설치 프로그램을 다운로드하여 압축을 풉니다".

b. "기존 Astra Trident가 있는 경우 이를 호스팅하는 클러스터에서 제거합니다".

c. Astra Control Provisioner를 사용하도록 설정된 Astra Trident를 설치합니다 (`--enable-acp=true`):

```
./tridentctl -n trident install --enable-acp=true --acp
-image=mycustomregistry/trident-acp:24.02
```

d. Astra Control Provisioner가 활성화되었는지 확인합니다.

```
./tridentctl -n trident version
```

응답:

```
+----------------+----------------+-------------+ | SERVER
VERSION | CLIENT VERSION | ACP VERSION | +----------------
+----------------+-------------+ | 24.02.0 | 24.02.0 | 24.02.0. |
+----------------+----------------+-------------+
```

헬름

a. Astra Trident 23.07.1 이하를 설치한 경우 "설치 제거" 작업자 및 기타 구성품

b. Kubernetes 클러스터에서 1.24 이전 버전을 실행 중인 경우 psp:

```
kubectl delete psp tridentoperatorpod
```

c. Astra Trident Helm 리포지토리를 추가합니다.

```
helm repo add netapp-trident https://netapp.github.io/trident-
helm-chart
```

d. 제어 차트 업데이트:

```
helm repo update netapp-trident
```

응답:

```
Hang tight while we grab the latest from your chart
repositories...
...Successfully got an update from the "netapp-trident" chart
repository
Update Complete. □Happy Helming!□
```

e. 영상을 나열합니다.

```
./tridentctl images -n trident
```

응답:

```
| v1.28.0            | netapp/trident:24.02.0|
|                    | docker.io/netapp/trident-
autosupport:24.02|
|                    | registry.k8s.io/sig-storage/csi-
provisioner:v4.0.0|
|                    | registry.k8s.io/sig-storage/csi-
attacher:v4.5.0|
|                    | registry.k8s.io/sig-storage/csi-
resizer:v1.9.3|
|                    | registry.k8s.io/sig-storage/csi-
snapshotter:v6.3.3|
|                    | registry.k8s.io/sig-storage/csi-node-
driver-registrar:v2.10.0 |
|                    | netapp/trident-operator:24.02.0 (optional)
```

f. 트라이덴트 - 운전자 24.02.0을 사용할 수 있는지 확인합니다.

```
helm search repo netapp-trident/trident-operator --versions
```

응답:

```
NAME                            CHART VERSION   APP VERSION
DESCRIPTION
netapp-trident/trident-operator 100.2402.0      24.02.0         A
```

g. 사용 `helm install` 을 클릭하고 다음 설정을 포함하는 옵션 중 하나를 실행합니다.

- 배포 위치의 이름입니다
- Astra Trident 버전
- Astra Control Provisioner 이미지의 이름
- Provisioner를 활성화하는 플래그입니다
- (선택 사항) 로컬 레지스트리 경로입니다. 로컬 레지스트리를 사용하는 경우, 을(를) 참조하십시오 "Trident 이미지" 하나의 레지스트리 또는 다른 레지스트리에 있을 수 있지만 모든 CSI 이미지는 동일한 레지스트리에 있어야 합니다.
- Trident 네임스페이스

옵션

- 레지스트리가 없는 이미지

```
helm install trident netapp-trident/trident-operator --version
100.2402.0 --set acpImage=cr.astra.netapp.io/astra/trident-
acp:24.02.0 --set enableACP=true --set operatorImage=netapp/trident-
operator:24.02.0 --set
tridentAutosupportImage=docker.io/netapp/trident-autosupport:24.02
--set tridentImage=netapp/trident:24.02.0 --namespace trident
```

- 하나 이상의 레지스트리에 있는 이미지

```
helm install trident netapp-trident/trident-operator --version
100.2402.0 --set acpImage=<your-registry>:<acp image> --set
enableACP=true --set imageRegistry=<your-registry>/sig-storage --set
operatorImage=netapp/trident-operator:24.02.0 --set
tridentAutosupportImage=docker.io/netapp/trident-autosupport:24.02
--set tridentImage=netapp/trident:24.02.0 --namespace trident
```

을 사용할 수 있습니다 `helm list` 이름, 네임스페이스, 차트, 상태, 앱 버전과 같은 설치 세부 정보를 검토하려면 수정본 번호.

Helm을 사용하여 Trident를 구축하는 데 문제가 있는 경우 다음 명령을 실행하여 Astra Trident를 완전히 제거합니다.

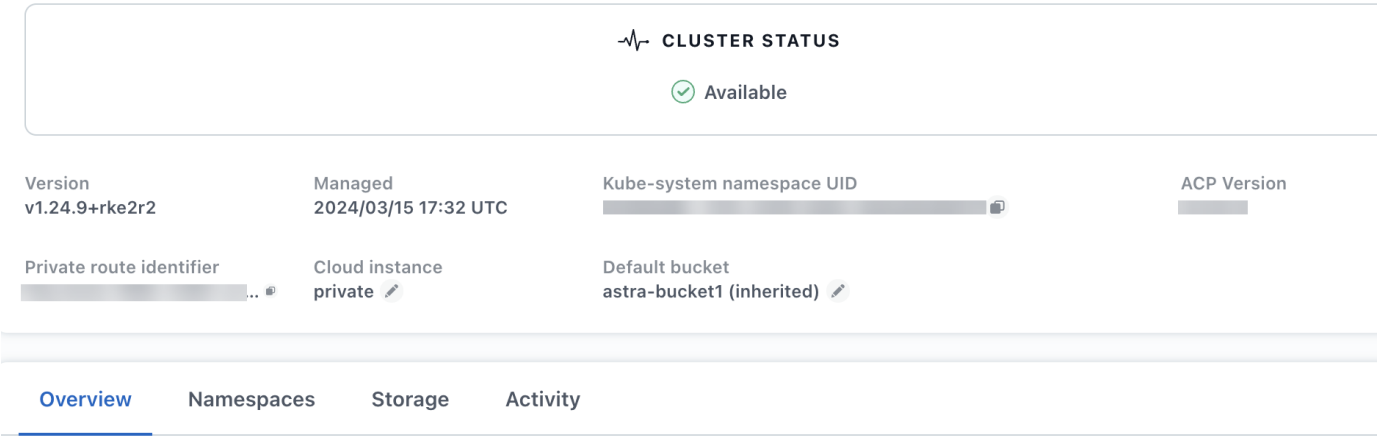
```
./tridentctl uninstall -n trident
```

- 하지 마십시오 * "Astra Trident CRD를 완전히 제거합니다" 설치 제거의 일부로 Astra Control Provisioner를 다시 활성화하려고 합니다.

결과

Astra Control Provisioner 기능이 활성화되어 있으며 실행 중인 버전에 제공되는 모든 기능을 사용할 수 있습니다.

Astra Control Provisioner를 설치하면 Astra Control UI에서 Provisioner를 호스팅하는 클러스터에 가 표시됩니다 `ACP version` 을 사용하지 마십시오 `Trident version` 필드 및 현재 설치된 버전 번호



를 참조하십시오

- "Astra Trident 업그레이드 설명서"

앱 및 클러스터 관리를 취소합니다

Astra Control에서 더 이상 관리하지 않으려는 응용 프로그램 또는 클러스터를 제거합니다.

앱 관리를 중지합니다

Astra Control에서 더 이상 백업, 스냅샷 또는 클론 복제하지 않을 애플리케이션 관리를 중지합니다.

앱 관리를 취소하는 경우:

- 기존 백업 및 스냅샷이 삭제됩니다.
- 애플리케이션과 데이터는 사용 가능한 상태로 유지됩니다.

단계

1. 왼쪽 탐색 모음에서 * 응용 프로그램 * 을 선택합니다.
2. 앱을 선택합니다.
3. 작업 열의 옵션 메뉴에서 * 관리 취소 * 를 선택합니다.

4. 정보를 검토합니다.
5. "unmanage"를 입력하여 확인합니다.
6. 예, 응용 프로그램 관리 취소 * 를 선택합니다.

결과

Astra Control은 앱 관리를 중지합니다.

클러스터 관리를 중지합니다

Astra Control에서 더 이상 관리하지 않으려는 클러스터 관리를 중지합니다.

> 클러스터를 관리하기 전에 클러스터와 연결된 앱의 관리를 해제해야 합니다.

모범 사례로서, GCP를 통해 클러스터를 삭제하기 전에 Astra Control에서 클러스터를 삭제하는 것이 좋습니다.

클러스터 관리를 취소하는 경우:

- 이 작업을 수행하면 Astra Control에서 클러스터를 관리할 수 없습니다. 클러스터 구성을 변경하지 않고 클러스터를 삭제하지 않습니다.
- Astra Control Provisioner 또는 Astra Trident는 클러스터에서 제거되지 않습니다. "Astra Trident를 제거하는 방법을 알아보십시오".

단계

1. 클러스터 * 를 선택합니다.
2. 더 이상 관리하지 않으려는 클러스터의 확인란을 선택합니다.
3. Actions * 열의 Options 메뉴에서 * Unmanage * 를 선택합니다.
4. 클러스터 관리를 해제할지 확인한 다음 * 예, 관리 취소 * 를 선택합니다.

결과

클러스터의 상태가 * Removing * 으로 변경됩니다. 그 이후에는 클러스터가 * Clusters * 페이지에서 제거되고 Astra Control에서 더 이상 관리되지 않습니다.

클라우드 공급자에서 클러스터 삭제

NetApp 스토리지 클래스에 상주하는 PV(영구적 볼륨)가 있는 Kubernetes 클러스터를 삭제하려면 먼저 아래 방법 중 하나를 따라 PVC(영구 볼륨 클레임)를 삭제해야 합니다. 클러스터를 삭제하기 전에 PVC 및 PV를 삭제하면 클라우드 공급자로부터 예상치 못한 청구서를 받지 않게 됩니다.

- * 방법 #1 *: 클러스터에서 애플리케이션 워크로드 네임스페이스를 삭제합니다. Trident 네임스페이스를_삭제하지_마십시오.
- * 방법 #2 *: PVC와 POD를 삭제하거나 PVS가 마운트된 구축을 삭제합니다.

Astra Control에서 Kubernetes 클러스터를 관리할 경우, 해당 클러스터의 애플리케이션은 클라우드 공급자를 영구 볼륨의 스토리지 백엔드로 사용합니다. PVS를 먼저 제거하지 않고 클라우드 공급자에서 클러스터를 삭제하는 경우 백엔드 볼륨은 클러스터와 함께 _NOT_DELETED가 됩니다.

위의 방법 중 하나를 사용하면 클러스터에서 해당 PVS가 삭제됩니다. 삭제하기 전에 클러스터의 NetApp 스토리지

클래스에 PVS가 없는지 확인하십시오.

클러스터를 삭제하기 전에 영구 볼륨을 삭제하지 않은 경우 클라우드 공급자로부터 백엔드 볼륨을 수동으로 삭제해야 합니다.

Astra Control의 자체 관리형 인스턴스를 구축

네트워크 안에 있는 Astra Control의 자체 관리형 인스턴스를 원하는 경우 Astra Control Service에서 Astra Control Center를 직접 구축할 수 있습니다.

단계

1. 대시보드의 Getting Started 영역에서 * Astra Control의 자가 관리형 인스턴스 배포 * 를 선택합니다.
2. 다음 중 하나를 수행합니다.
   - Generate * 를 선택하여 새 API 토큰을 생성합니다.
   - 기존 Astra Control REST API 토큰에 붙여 넣습니다. 을 참조하십시오 "Astra 자동화 문서" API 토큰 생성에 대한 지침을 참조하십시오.
3. Astra Control Center * 배포 창의 지침을 따릅니다.

저작권 정보

Copyright © 2024 NetApp, Inc. All Rights Reserved. 미국에서 인쇄된 본 문서의 어떠한 부분도 저작권 소유자의 사전 서면 승인 없이는 어떠한 형식이나 수단(복사, 녹음, 녹화 또는 전자 검색 시스템에 저장하는 것을 비롯한 그래픽, 전자적 또는 기계적 방법)으로도 복제될 수 없습니다.

NetApp이 저작권을 가진 자료에 있는 소프트웨어에는 아래의 라이센스와 고지사항이 적용됩니다.

본 소프트웨어는 NetApp에 의해 '있는 그대로' 제공되며 상품성 및 특정 목적에의 적합성에 대한 명시적 또는 묵시적 보증을 포함하여(이에 제한되지 않음) 어떠한 보증도 하지 않습니다. NetApp은 대체품 또는 대체 서비스의 조달, 사용 불능, 데이터 손실, 이익 손실, 영업 중단을 포함하여(이에 국한되지 않음), 이 소프트웨어의 사용으로 인해 발생하는 모든 직접 및 간접 손해, 우발적 손해, 특별 손해, 징벌적 손해, 결과적 손해의 발생에 대하여 그 발생 이유, 책임론, 계약 여부, 엄격한 책임, 불법 행위(과실 또는 그렇지 않은 경우)와 관계없이 어떠한 책임도 지지 않으며, 이와 같은 손실의 발생 가능성이 통지되었다 하더라도 마찬가지입니다.

NetApp은 본 문서에 설명된 제품을 언제든지 예고 없이 변경할 권리를 보유합니다. NetApp은 NetApp의 명시적인 서면 동의를 받은 경우를 제외하고 본 문서에 설명된 제품을 사용하여 발생하는 어떠한 문제에도 책임을 지지 않습니다. 본 제품의 사용 또는 구매의 경우 NetApp에서는 어떠한 특허권, 상표권 또는 기타 지적 재산권이 적용되는 라이센스도 제공하지 않습니다.

본 설명서에 설명된 제품은 하나 이상의 미국 특허, 해외 특허 또는 출원 중인 특허로 보호됩니다.

제한적 권리 표시: 정부에 의한 사용, 복제 또는 공개에는 DFARS 252.227-7013(2014년 2월) 및 FAR 52.227-19(2007년 12월)의 기술 데이터-비상업적 품목에 대한 권리(Rights in Technical Data -Noncommercial Items) 조항의 하위 조항 (b)(3)에 설명된 제한사항이 적용됩니다.

여기에 포함된 데이터는 상업용 제품 및/또는 상업용 서비스(FAR 2.101에 정의)에 해당하며 NetApp, Inc.의 독점 자산입니다. 본 계약에 따라 제공되는 모든 NetApp 기술 데이터 및 컴퓨터 소프트웨어는 본질적으로 상업용이며 개인 비용만으로 개발되었습니다. 미국 정부는 데이터가 제공된 미국 계약과 관련하여 해당 계약을 지원하는 데에만 데이터에 대한 전 세계적으로 비독점적이고 양도할 수 없으며 재사용이 불가능하며 취소 불가능한 라이센스를 제한적으로 가집니다. 여기에 제공된 경우를 제외하고 NetApp, Inc.의 사전 서면 승인 없이는 이 데이터를 사용, 공개, 재생산, 수정, 수행 또는 표시할 수 없습니다. 미국 국방부에 대한 정부 라이센스는 DFARS 조항 252.227-7015(b)(2014년 2월)에 명시된 권한으로 제한됩니다.

상표 정보

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