Los geht's

Astra Control Service

NetApp
July 29, 2024

Inhalt

Los geht’s

Weitere Informationen zu Astra Control

Astra Control ist eine Kubernetes-Lösung für das Lifecycle-Management von Applikationsdaten, die den Betrieb zustandsorientierte Applikationen vereinfacht. Schutz, Backup und Migration von Kubernetes-Workloads und sofortige Erstellung von Applikationsklonen

Funktionen

Astra Control bietet entscheidende Funktionen für das Lifecycle Management von Kubernetes-Applikationsdaten:

- Automatisches Management von persistentem Storage
- Erstellen Sie applikationsorientierte Snapshots und Backups nach Bedarf
- Automatisierung von richtlinienbasierten Snapshot- und Backup-Vorgängen
- Migrieren Sie Applikationen und Daten von einem Kubernetes-Cluster zu einem anderen
- Replizieren von Applikationen auf ein Remote-System mit NetApp SnapMirror Technologie (Astra Control Center)
- Klonen von Applikationen von Staging hin zur Produktion
- Darstellung des Anwendungszustands und des Schutzstatus
- Verwenden Sie eine Web-Oberfläche oder eine API zur Implementierung Ihrer Backup- und Migration-Workflows

Implementierungsmodelle

Astra Control ist in zwei Implementierungsmodellen erhältlich:

- **Astra Control Service**: Ein von NetApp gemanagter Service, der applikationskonsistentes Datenmanagement von Kubernetes Clustern in Umgebungen mehrerer Cloud-Provider sowie selbst gemanagte Kubernetes Cluster bietet.
- **Astra Control Center**: Gemanagte Software für applikationsgerechtes Datenmanagement von Kubernetes-Clustern, die in Ihrer On-Premises-Umgebung ausgeführt werden. Astra Control Center kann auch auf mehreren Cloud-Provider-Umgebungen mit einem NetApp Cloud Volumes ONTAP Storage-Backend installiert werden.

	Astra Control Service	**Astra Control Center**
Wie wird das angeboten?	Vollständig gemanagter Cloud-Service von NetApp	Als Software, die Sie herunterladen, installieren und verwalten können
Wo wird sie gehostet?	In einer Public Cloud von NetApp ihrer Wahl	In Ihrem eigenen Kubernetes-Cluster
Wie wird sie aktualisiert?	Gemanagt von NetApp	Sie verwalten jegliche Updates

	Astra Control Service	Astra Control Center
Welche Kubernetes-Distributionen werden unterstützt?	• **Cloud-Provider** ◦ Amazon Web Services ▪ Amazon Elastic Kubernetes Service (EKS) ◦ Google Cloud ▪ Google Kubernetes Engine (GKE) ◦ Microsoft Azure ▪ Azure Kubernetes-Service (AKS) • **Selbstverwaltete Cluster** ◦ Kubernetes (Vorgelagert) ◦ Rancher Kubernetes Engine (RKE) ◦ Red hat OpenShift Container Platform • **On-Premises-Cluster** ◦ Lokale Red hat OpenShift Container-Plattform	• Azure Kubernetes Service für Azure Stack HCI • Google Anthos • Kubernetes (Vorgelagert) • Rancher Kubernetes Engine (RKE) • Red hat OpenShift Container Platform

	Astra Control Service	Astra Control Center
Welche Storage-Back-Ends werden unterstützt?	• **Cloud-Provider** ◦ Amazon Web Services ▪ Amazon EBS ▪ Amazon FSX für NetApp ONTAP ▪ "Cloud Volumes ONTAP" ◦ Google Cloud ▪ Google Persistent Disk ▪ NetApp Cloud Volumes Service ▪ "Cloud Volumes ONTAP" ◦ Microsoft Azure ▪ Über Azure Gemanagte Festplatten ▪ Azure NetApp Dateien ▪ "Cloud Volumes ONTAP" • **Selbstverwaltete Cluster** ◦ Amazon EBS ◦ Über Azure Gemanagte Festplatten ◦ Google Persistent Disk ◦ "Cloud Volumes ONTAP" ◦ NetApp MetroCluster ◦ "Longhorn" • **On-Premises-Cluster** ◦ NetApp MetroCluster ◦ NetApp ONTAP AFF und FAS Systeme ◦ NetApp ONTAP Select ◦ "Cloud Volumes ONTAP" ◦ "Longhorn"	• NetApp ONTAP AFF und FAS Systeme • NetApp ONTAP Select • "Cloud Volumes ONTAP" • "Longhorn"

Funktionsweise des Astra Control Service

Astra Control Service ist ein von NetApp gemanagter Cloud-Service, der ständig verfügbar und mit den neuesten Funktionen aktualisiert ist. Verschiedene Komponenten unterstützen das Lifecycle-Management von Applikationsdaten.

Astra Control Service funktioniert auf hohem Niveau wie folgt:

- Starten Sie mit Astra Control Service, indem Sie Ihren Cloud-Provider einrichten und einen Astra Account anfordern.

+ ** für GKE Cluster, Astra Control Service verwendet "NetApp Cloud Volumes Service für Google Cloud" Oder Google Persistent Disks als Storage-Backend für Ihre persistenten Volumes.

+ ** für AKS-Cluster, Astra Control Service verwendet "Azure NetApp Dateien" Oder von Azure gemanagte Festplatten als Storage-Backend für Ihre persistenten Volumes.

+ ** für Amazon EKS Cluster, Astra Control Service verwendet "Amazon Elastic Block Store" Oder "Amazon FSX für NetApp ONTAP" Das Storage-Backend für Ihre persistenten Volumes

- Sie fügen Ihre ersten Kubernetes-Computing-Ressourcen in den Astra Control Service ein. Astra Control Service übernimmt dann Folgendes:
  - Erstellung eines Objektspeicher in Ihrem Cloud-Provider-Konto, an dem Backup-Kopien gespeichert werden

+ in Azure erstellt Astra Control Service außerdem eine Ressourcengruppe, ein Storage-Konto und Schlüssel für den Blob-Container.

- Erstellt eine neue Administratorrolle und ein Kubernetes-Servicekonto auf dem Cluster.
- Verwendet diese neue Administratorrolle, um den Link../concepts/architecture#astra-control-components[Astra Control Provisioner] auf dem Cluster zu installieren und eine oder mehrere Storage-Klassen zu erstellen.
- Wenn Sie ein Storage-Angebot mit NetApp Cloud-Services als Storage-Back-End verwenden, stellt Astra Control Service persistente Volumes für Ihre Applikationen bereit. Wenn Sie von Amazon EBS oder Azure gemanagte Festplatten als Storage-Backend verwenden, müssen Sie einen Provider-spezifischen CSI-Treiber installieren. Installationsanweisungen finden Sie in "Einrichten von Amazon Web Services" Und "Richten Sie Microsoft Azure mit von Azure gemanagten Festplatten ein".
  - An diesem Punkt können Sie Apps aus Ihrem Cluster definieren. Persistente Volumes werden auf dem Storage-Back-End über die neue Standard-Storage-Klasse bereitgestellt.
  - Anschließend verwalten Sie diese Applikationen mithilfe des Astra Control Service und erstellen Snapshots, Backups und Klone.

Mit dem kostenlosen Plan von Astra Control können Sie bis zu 10 Namespaces in Ihrem Konto verwalten. Wenn Sie mehr als 10 Namespaces verwalten möchten, müssen Sie die Abrechnung durch ein Upgrade vom kostenlosen Plan auf den Premium-Plan einrichten.

So funktioniert Astra Control Center

Astra Control Center wird lokal in Ihrer eigenen Private Cloud ausgeführt.

Astra Control Center unterstützt Kubernetes-Cluster mit einer für die Astra Control Provisioner konfigurierten Storage-Klasse mit einem ONTAP Storage-Back-End.

Astra Control Center ist vollständig in das AutoSupport und Active IQ Ecosystem integriert, damit Benutzer und NetApp Support Fehlerbehebungs- und Verwendungsinformationen liefern können.

Sie können Astra Control Center mit einer 90-Tage-Evaluierungslizenz ausprobieren. Die Evaluierungsversion wird durch E-Mail- und Community-Optionen unterstützt. Zudem haben Sie über das Dashboard für den Produktsupport Zugriff auf Knowledgebase-Artikel und -Dokumentation.

Um Astra Control Center zu installieren und zu verwenden, müssen Sie sicher sein "Anforderungen".

Astra Control Center funktioniert auf hohem Niveau wie folgt:

- Sie installieren Astra Control Center in Ihrer lokalen Umgebung. Erfahren Sie mehr darüber, wie Sie "Installieren Sie Astra Control Center".
- Sie führen einige Setup-Aufgaben wie die folgenden aus:
  - Lizenzierung einrichten.
  - Fügen Sie den ersten Cluster hinzu.
  - Fügen Sie ein Storage-Back-End hinzu, das beim Hinzufügen des Clusters erkannt wird.
  - Fügen Sie einen Objektspeicher-Bucket hinzu, der Ihre Applikations-Backups speichert.

Erfahren Sie mehr darüber, wie Sie "Einrichten des Astra Control Center".

Sie können Applikationen zu Ihrem Cluster hinzufügen. Wenn auch einige Applikationen bereits im Cluster gemanagt werden, können Sie sie mit Astra Control Center managen. Nutzen Sie dann das Astra Control Center, um Snapshots, Backups, Klone und Replizierungsbeziehungen zu erstellen.

Finden Sie weitere Informationen

- "Dokumentation zur NetApp Astra Produktfamilie"
- "Astra Control Center-Dokumentation"
- "Astra Control API-Dokumentation"
- "Astra Trident-Dokumentation"
- "ONTAP-Dokumentation"

Unterstützte Kubernetes-Implementierungen

Astra Control Service managt Applikationen, die auf einem gemanagten Kubernetes-Cluster in Amazon Elastic Kubernetes Service (EKS) ausgeführt werden, sowie Cluster, die Sie selbst managen.

Astra Control Service kann sowohl auf einem gemanagten Kubernetes-Cluster in der Google Kubernetes Engine (GKE) als auch auf Clustern, die Sie selbst managen, ausgeführte Applikationen managen.

Astra Control Service kann Apps managen, die auf einem gemanagten Kubernetes-Cluster in Azure Kubernetes Service (AKS) ausgeführt werden, sowie Cluster, die Sie selbst managen.

- "Erfahren Sie, wie Sie Amazon Web Services für Astra Control Service einrichten".
- "Erfahren Sie, wie Sie Google Cloud für Astra Control Service einrichten".
- "Erfahren Sie, wie Sie Microsoft Azure mit Azure NetApp Files für Astra Control Service einrichten".
- "Erfahren Sie, wie Sie Microsoft Azure mit gemanagten Azure Festplatten für den Astra Control Service einrichten".
- "Bereiten Sie selbst gemanagte Cluster vor, bevor Sie sie in den Astra Control Service hinzufügen".

Schnellstart für den Astra Control Service

Diese Seite bietet einen grundlegenden Überblick über die Schritte, die Sie für den

Einstieg in den Astra Control Service benötigen. Die Links in den einzelnen Schritten führen zu einer Seite, die weitere Details enthält.

[Eins] Richten Sie Ihren Cloud-Provider ein

1. Google Cloud:
   - Google Kubernetes Engine-Cluster-Anforderungen prüfen.
   - Kaufen Sie Cloud Volumes Service für Google Cloud über den Google Cloud Marketplace.
   - Aktivieren Sie die erforderlichen APIs.
   - Erstellen eines Servicekontos und eines Servicekontenschlüssels.
   - Netzwerk-Peering von Ihrem VPC zu Cloud Volumes Service für Google Cloud einrichten.

     "Erfahren Sie mehr über die Google Cloud Anforderungen".
2. Amazon Web Services:
   - Amazon Web Services-Cluster-Anforderungen prüfen.
   - Erstellen Sie ein Amazon-Konto.
   - Installieren Sie die Amazon Web Services-CLI.
   - Erstellen Sie einen IAM-Benutzer.
   - Erstellen Sie eine Berechtigungsrichtlinie und fügen Sie sie an.
   - Speichern Sie die Anmeldeinformationen für den IAM-Benutzer.

     "Erfahren Sie mehr über die Anforderungen von Amazon Web Services".
3. Microsoft Azure:
   - Azure Kubernetes Service-Cluster-Anforderungen für das Storage-Back-End prüfen, das Sie verwenden möchten.

     "Erfahren Sie mehr über Microsoft Azure und Azure NetApp Files Anforderungen".

   "Erfahren Sie mehr über die von Microsoft Azure und Azure gemanagten Festplattenanforderungen".

Wenn Sie ein eigenes Cluster managen und nicht von einem Cloud-Provider gehostet werden, prüfen Sie die Anforderungen für Self-Managed Cluster.
"Erfahren Sie mehr über Self-Managed-Cluster-Anforderungen".

[Zwei] Schließen Sie die Registrierung für den Astra Control ab

1. Erstellen Sie ein "NetApp BlueXP" Konto.
2. Geben Sie bei der Erstellung Ihres Astra Control Kontos Ihre NetApp BlueXP E-Mail-ID an "Auf der Astra Control Produktseite aus".

"Erfahren Sie mehr über den Registrierungsprozess".

[Drittens] Fügen Sie Cluster zum Astra Control hinzu

Nachdem Sie sich angemeldet haben, wählen Sie **Cluster hinzufügen**, um das Cluster mit Astra Control zu

verwalten.

"Erfahren Sie mehr über das Hinzufügen von Clustern".

Richten Sie Ihren Cloud-Provider ein

Einrichten von Amazon Web Services

Zur Vorbereitung Ihres Amazon Web Services Projekts sind einige Schritte erforderlich, bevor Sie Amazon Elastic Kubernetes Service (EKS) Cluster mit Astra Control Service managen können.

Schnellstart für die Einrichtung von Amazon Web Services

Führen Sie diese Schritte schnell durch, oder scrollen Sie nach unten zu den verbleibenden Abschnitten, um ausführliche Informationen zu erhalten.

[Eins] Astra Control Service-Anforderungen für Amazon Web Services überprüfen

Stellen Sie sicher, dass die Cluster ordnungsgemäß sind und eine unterstützte Version von Kubernetes ausführen, dass Worker-Nodes unter anderem Linux oder Windows online sind. Erfahren Sie mehr zu diesem Schritt.

[Zwei] Erstellen Sie ein Amazon-Konto

Wenn Sie noch kein Amazon-Konto haben, müssen Sie ein Konto erstellen, damit Sie EKS verwenden können. Erfahren Sie mehr zu diesem Schritt.

[Drittens] Installieren Sie die Amazon Web Services-CLI

Installieren Sie die AWS CLI, sodass Sie AWS über die Befehlszeile managen können. Befolgen Sie die Schritt-für-Schritt-Anweisungen.

[Vier] Optional: Erstellen Sie einen IAM-Benutzer

Erstellen Sie einen Amazon IAM-Benutzer (Identity and Access Management). Sie können diesen Schritt auch überspringen und einen vorhandenen IAM-Benutzer mit Astra Control Service verwenden.

Lesen Sie Schritt-für-Schritt-Anleitungen.

[Fünf] Erstellen Sie eine Berechtigungsrichtlinie und fügen Sie sie an

Erstellen einer Richtlinie mit den erforderlichen Berechtigungen für den Astra Control Service zur Interaktion mit Ihrem AWS Konto

Lesen Sie Schritt-für-Schritt-Anleitungen.

[Sechs] Speichern Sie die Anmeldeinformationen für den IAM-Benutzer

Speichern Sie die Anmeldeinformationen für den IAM-Benutzer, damit Sie die Anmeldeinformationen in den Astra Control Service importieren können.

Lesen Sie Schritt-für-Schritt-Anleitungen.

EKS-Clusteranforderungen

Ein Kubernetes-Cluster muss folgende Anforderungen erfüllen, damit Sie ihn über den Astra Control Service erkennen und managen können.

Kubernetes-Version
: Auf einem Cluster muss eine Kubernetes-Version im Bereich von 1.25 bis 1.28 ausgeführt werden.

Bildtyp
: Der Bildtyp für jeden Arbeiterknoten muss Linux sein.

Der Cluster-Status
: Cluster müssen in einem ordnungsgemäßen Zustand ausgeführt werden und mindestens einen Online-Worker-Node ohne „Worker“-Nodes im ausgefallenen Status aufweisen.

Astra Control Provisioner
: Für den Betrieb mit Storage-Back-Ends sind die Provisionierung von Astra Control und ein externer Snapshot Controller erforderlich. Gehen Sie wie folgt vor, um diese Vorgänge zu aktivieren:

1. "Installieren Sie die Snapshot-CRDs und den Snapshot-Controller".
2. "Astra Control Provisioner Aktivieren".
3. "Erstellen Sie eine VolumeSnapshotClass".

CSI-Treiber für Amazon Elastic Block Store (EBS)
: Wenn Sie das Amazon EBS Storage-Backend verwenden, müssen Sie den Container Storage Interface (CSI)-Treiber für EBS installieren (dieser wird nicht automatisch installiert).

  Anweisungen zur Installation des CSI-Treibers finden Sie in den Schritten.

Installieren Sie einen externen Schnappschussfilter

Falls noch nicht geschehen, "Installieren Sie die Snapshot-CRDs und den Snapshot-Controller".

Den CSI-Treiber als Amazon EKS-Add-On installieren

1. Erstellen der IAM-Rolle des Amazon EBS CSI-Treibers für Service-Konten Befolgen Sie die Anweisungen "In der Amazon-Dokumentation", Verwenden der AWS CLI-Befehle in den Anweisungen.
2. Fügen Sie das Amazon EBS CSI-Add-on mit dem folgenden AWS-CLI-Befehl hinzu und ersetzen Sie Informationen in Klammern <> durch Werte speziell für Ihre Umgebung. Ersetzen Sie <DRIVER_ROLE> durch den Namen der EBS CSI-Treiberrolle, die Sie im vorherigen Schritt erstellt haben:

```
aws eks create-addon \
  --cluster-name <CLUSTER_NAME> \
  --addon-name aws-ebs-csi-driver \
  --service-account-role-arn
arn:aws:iam::<ACCOUNT_ID>:role/<DRIVER_ROLE>
```

Konfigurieren der EBS Storage-Klasse

1. Klonen Sie das GitHub Repository des Amazon EBS CSI-Treibers auf Ihrem System.

```
git clone https://github.com/kubernetes-sigs/aws-ebs-csi-
driver.git
```

2. Navigieren Sie zum Beispielverzeichnis für dynamische Bereitstellung.

```
cd aws-ebs-csi-driver/examples/kubernetes/dynamic-provisioning/
```

3. Implementierung der ebs-sc-Storage-Klasse und der ebs-Claim Persistent Volume Claim aus dem Manifeste Verzeichnis

```
kubectl apply -f manifests/storageclass.yaml
kubectl apply -f manifests/claim.yaml
```

4. ebs-sc Storage-Klasse beschreiben

```
kubectl describe storageclass ebs-sc
```

Sie sollten die Ausgabe sehen, in der die Attribute der Storage-Klasse beschrieben werden.

Erstellen Sie ein Amazon-Konto

Wenn Sie noch kein Amazon-Konto besitzen, müssen Sie ein Konto erstellen, um die Abrechnung für Amazon EKS zu aktivieren.

Schritte

1. Wechseln Sie zum "Amazon Homepage" Wählen Sie oben rechts **Anmelden** und wählen Sie **Hier starten**.
2. Befolgen Sie die Anweisungen, um ein Konto zu erstellen.

Installieren Sie die Amazon Web Services-CLI

Installieren Sie die AWS CLI, sodass Sie AWS Ressourcen über die Befehlszeile managen können.

Schritt

1. Gehen Sie zu "Erste Schritte mit der AWS CLI" Und befolgen Sie die Anweisungen zur Installation der CLI.

Optional: Erstellen Sie einen IAM-Benutzer

Erstellen Sie einen IAM-Benutzer, damit Sie AWS Services und Ressourcen mit erhöhter Sicherheit nutzen und managen können. Sie können diesen Schritt auch überspringen und einen vorhandenen IAM-Benutzer mit Astra Control Service verwenden.

Schritt

1. Gehen Sie zu "IAM-Benutzer werden erstellt" Und befolgen Sie die Anweisungen zum Erstellen eines IAM-Benutzers.

Erstellen Sie eine Berechtigungsrichtlinie und fügen Sie sie an

Erstellen einer Richtlinie mit den erforderlichen Berechtigungen für den Astra Control Service zur Interaktion mit Ihrem AWS Konto

Schritte

1. Erstellen Sie eine neue Datei mit dem Namen `policy.json`.
2. Kopieren Sie den folgenden JSON-Inhalt in die Datei:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "cloudwatch:GetMetricData",
                "fsx:DescribeVolumes",
                "ec2:DescribeRegions",
                "s3:CreateBucket",
                "s3:ListBucket",
                "s3:PutObject",
                "s3:GetObject",
                "iam:SimulatePrincipalPolicy",
                "s3:ListAllMyBuckets",
                "eks:DescribeCluster",
                "eks:ListNodegroups",
                "eks:DescribeNodegroup",
                "eks:ListClusters",
                "iam:GetUser",
                "s3:DeleteObject",
                "s3:DeleteBucket",
                "autoscaling:DescribeAutoScalingGroups"
            ],
            "Resource": "*"
        }
    ]
}
```

3. Erstellen der Richtlinie:

```
POLICY_ARN=$(aws iam create-policy  --policy-name <policy-name> --policy
-document file://policy.json  --query='Policy.Arn' --output=text)
```

4. Hängen Sie die Richtlinie an den IAM-Benutzer an. Austausch `<IAM-USER-NAME>` Entweder mit dem Benutzernamen des von Ihnen erstellten IAM-Benutzers oder mit einem vorhandenen IAM-Benutzer:

```
aws iam attach-user-policy --user-name <IAM-USER-NAME> --policy-arn
=$POLICY_ARN
```

Speichern Sie die Anmeldeinformationen für den IAM-Benutzer

Speichern Sie die Anmeldeinformationen für den IAM-Benutzer, damit Sie den Astra Control Service auf den Benutzer aufmerksam machen können.

Schritte

1. Anmeldedaten herunterladen Austausch `<IAM-USER-NAME>` Mit dem Benutzernamen des IAM-Benutzers, den Sie verwenden möchten:

```
aws iam create-access-key --user-name <IAM-USER-NAME> --output json >
credential.json
```

Ergebnis

Der `credential.json` Datei ist erstellt, und Sie können die Anmeldeinformationen in Astra Control Service importieren.

Google Cloud einrichten

Zur Vorbereitung Ihres Google Cloud-Projekts sind einige Schritte erforderlich, bevor Sie Google Kubernetes Engine-Cluster mit Astra Control Service verwalten können.

> Wenn Sie Google Cloud Volumes Service for Google Cloud nicht als Speicher-Backend nutzen, sondern zu einem späteren Zeitpunkt nutzen möchten, sollten Sie die notwendigen Schritte ausführen, um Google Cloud Volumes Service für Google Cloud jetzt zu konfigurieren. Das Erstellen eines Service-Kontos im späteren Verlauf bedeutet, dass der Zugriff auf die vorhandenen Storage-Buckets verloren geht.

Schnellstart für die Einrichtung von Google Cloud

Führen Sie diese Schritte schnell durch, oder scrollen Sie nach unten zu den verbleibenden Abschnitten, um ausführliche Informationen zu erhalten.

[Eins] Astra Control Service-Anforderungen für Google Kubernetes Engine prüfen

Stellen Sie sicher, dass die Cluster ordnungsgemäß sind und eine unterstützte Kubernetes-Version ausführen, dass Worker-Nodes online sind und einen unterstützten Bildtyp ausführen, und vieles mehr. Erfahren Sie mehr zu diesem Schritt.

[Zwei] (Optional): Kaufen Sie Cloud Volumes Service für Google Cloud

Wenn Sie Cloud Volumes Service für Google Cloud als Storage-Back-End verwenden möchten, gehen Sie zur NetApp Cloud Volumes Service Seite im Google Cloud Marketplace und wählen Sie „Kaufen“. Erfahren Sie mehr zu diesem Schritt.

[Drittens] Aktivieren Sie APIs in Ihrem Google Cloud-Projekt

Aktivieren Sie die folgenden Google Cloud APIs:

- Google Kubernetes Engine
- Cloud-Storage

- Cloud Storage JSON API
- Nutzung Von Services
- Cloud Resource Manager API
- NetApp Cloud Volumes Service
  - Für Cloud Volumes Service für Google Cloud erforderlich
  - Optional (aber empfohlen) für Google Persistent Disk
- Service Consumer Management API
- Service Networking API
- Service-Management-API

Befolgen Sie die Schritt-für-Schritt-Anweisungen.

[Vier] Erstellen Sie ein Dienstkonto mit den erforderlichen Berechtigungen

Erstellen Sie ein Google Cloud-Servicekonto mit folgenden Berechtigungen:

- Kubernetes Engine-Administrator
- NetApp Cloud Volumes Admin
  - Für Cloud Volumes Service für Google Cloud erforderlich
  - Optional (aber empfohlen) für Google Persistent Disk
- Storage-Admin
- Viewer Für Die Nutzung Des Dienstes
- Network Viewer Für Computing

Lesen Sie Schritt-für-Schritt-Anleitungen.

[Fünf] Erstellen eines Service-Kontokonschlüssels

Erstellen Sie einen Schlüssel für das Servicekonto, und speichern Sie die Schlüsseldatei an einem sicheren Speicherort. Befolgen Sie die Schritt-für-Schritt-Anweisungen.

[Sechs] (Optional): Netzwerk-Peering für Ihr VPC einrichten

Wenn Sie Cloud Volumes Service für Google Cloud als Storage-Back-End verwenden möchten, richten Sie Netzwerk-Peering von Ihrem VPC zu Cloud Volumes Service für Google Cloud ein. Befolgen Sie die Schritt-für-Schritt-Anweisungen.

GKE-Clusteranforderungen

Ein Kubernetes-Cluster muss folgende Anforderungen erfüllen, damit Sie ihn über den Astra Control Service erkennen und managen können. Einige dieser Anforderungen gelten nur, wenn Sie Cloud Volumes Service für Google Cloud als Storage-Backend verwenden möchten.

Kubernetes-Version

Auf einem Cluster muss eine Kubernetes-Version im Bereich von 1.26 bis 1.28 ausgeführt werden.

Bildtyp

Der Bildtyp für jeden Arbeiterknoten muss sein `COS_CONTAINERD`.

Der Cluster-Status

Cluster müssen in einem ordnungsgemäßen Zustand ausgeführt werden und mindestens einen Online-Worker-Node ohne „Worker“-Nodes im ausgefallenen Status aufweisen.

Google Cloud-Region

Wenn Sie Cloud Volumes Service für Google Cloud als Storage-Back-End verwenden möchten, müssen Cluster in einem ausgeführt werden "Google Cloud-Region, in der Cloud Volumes Service für Google Cloud unterstützt wird." Der Astra Control Service unterstützt beide Servicetypen: CVS und CVS-Performance. Als Best Practice sollten Sie eine Region wählen, die Cloud Volumes Service für Google Cloud unterstützt, auch wenn Sie sie nicht als Storage-Backend verwenden. Dies vereinfacht die Verwendung von Cloud Volumes Service für Google Cloud als Storage-Backend, wenn sich Ihre Performance-Anforderungen ändern.

Netzwerkbetrieb

Wenn Sie Cloud Volumes Service für Google Cloud als Storage-Back-End verwenden möchten, muss der Cluster in einer VPC oder der mit Cloud Volumes Service für Google Cloud Peering durchgeführt werden. Dieser Schritt wird im Folgenden beschrieben.

Private Cluster

Wenn das Cluster privat ist, gilt das "Autorisierte Netzwerke" Die Astra Control Service-IP-Adresse muss zugelassen werden:

52.188.218.166/32

Betriebsmodus für ein GKE-Cluster

Sie sollten den Standardbetriebsmodus verwenden. Der Autopilot-Modus wurde derzeit nicht getestet. "Erfahren Sie mehr über Betriebsmodi".

Storage-Pools

Wenn Sie NetApp Cloud Volumes Service als Storage-Backend mit dem CVS Servicetyp verwenden, müssen Sie Speicherpools konfigurieren, bevor Sie Volumes bereitstellen können. Siehe "Servicetyp, Speicherklassen und PV-Größe für GKE-Cluster" Finden Sie weitere Informationen.

Optional: Kauf von Cloud Volumes Service für Google Cloud

Astra Control Service kann Cloud Volumes Service für Google Cloud als Storage-Backend für Ihre persistenten Volumes nutzen. Wenn Sie diesen Service nutzen möchten, müssen Sie Cloud Volumes Service für Google Cloud über Google Cloud Marketplace erwerben, um die Abrechnung für persistente Volumes zu ermöglichen.

Schritt

1. Wechseln Sie zum "NetApp Cloud Volumes Service Seite" Wählen Sie im Google Cloud Marketplace die Option **Einkauf** aus, und folgen Sie den Anweisungen.

   "Befolgen Sie die Schritt-für-Schritt-Anweisungen in der Google Cloud-Dokumentation, um den Service zu erwerben und zu aktivieren".

Aktivieren Sie APIs in Ihrem Projekt

Für Ihr Projekt sind Berechtigungen erforderlich, um auf bestimmte Google Cloud-APIs zuzugreifen. APIs

werden für die Interaktion mit Google Cloud-Ressourcen eingesetzt, beispielsweise mit Google Kubernetes Engine-Clustern (GKE) und NetApp Cloud Volumes Service Storage.

Schritt

1. "Verwenden Sie die Google Cloud-Konsole oder die gcloudbasierte CLI, um die folgenden APIs zu aktivieren":
   - Google Kubernetes Engine
   - Cloud-Storage
   - Cloud Storage JSON API
   - Nutzung Von Services
   - Cloud Resource Manager API
   - NetApp Cloud Volumes Service (für Cloud Volumes Service für Google Cloud erforderlich)
   - Service Consumer Management API
   - Service Networking API
   - Service-Management-API

Das folgende Video zeigt, wie die APIs über die Google Cloud-Konsole aktiviert werden.

► https://docs.netapp.com/de-de/astra-control-service/media/get-started/video-enable-gcp-apis.mp4 *(video)*

Erstellen eines Dienstkontos

Astra Control Service nutzt ein Google Cloud-Service-Konto, um das Management von Kubernetes-Applikationsdaten in Ihrem Auftrag zu vereinfachen.

Schritte

1. Besuchen Sie Google Cloud und "Erstellen Sie ein Servicekonto, indem Sie die Konsole, den gcloudbasierten Befehl oder eine andere bevorzugte Methode verwenden".
2. Gewähren Sie dem Dienstkonto die folgenden Rollen:
   - **Kubernetes Engine Admin** - wird verwendet, um Cluster aufzulisten und Administratorzugriff zum Verwalten von Apps zu erstellen.
   - **NetApp Cloud Volumes Admin** - wird für das Management von persistentem Storage für Applikationen verwendet.
   - **Storage Admin** - zur Verwaltung von Buckets und Objekten für Backups von Apps.
   - **Service Usage Viewer** - wird verwendet, um zu überprüfen, ob die erforderlichen Cloud Volumes Service für Google Cloud APIs aktiviert sind.
   - **Computing Network Viewer** - wird verwendet, um zu prüfen, ob die Kubernetes VPC erlaubt ist, Cloud Volumes Service für Google Cloud zu erreichen.

Wenn Sie gcloudbasierte Lösungen verwenden möchten, können Sie im Astra Control Interface die gewünschten Schritte ausführen. Wählen Sie **Konto > Anmeldeinformationen > Anmeldeinformationen hinzufügen**, und wählen Sie dann **Anweisungen** aus.

Wenn Sie die Google Cloud-Konsole verwenden möchten, wird im folgenden Video gezeigt, wie Sie das Servicekonto über die Konsole erstellen.

► https://docs.netapp.com/de-de/astra-control-service/media/get-started/video-create-gcp-service-

account.mp4 *(video)*

Konfigurieren des Service-Kontos für eine gemeinsame VPC

Um GKE-Cluster zu verwalten, die sich in einem Projekt befinden, aber ein VPC aus einem anderen Projekt (ein gemeinsames VPC) zu verwenden, müssen Sie das Astra-Servicekonto als Mitglied des Hostprojekts mit der Rolle **Compute Network Viewer** angeben.

Schritte

1. Wählen Sie von der Google Cloud-Konsole aus die Option **IAM & Admin** aus und wählen Sie **Servicekonten** aus.
2. Finden Sie das Astra-Servicekonto mit "Die erforderlichen Berechtigungen" Und dann kopieren Sie die E-Mail-Adresse.
3. Gehen Sie zu Ihrem Hostprojekt und wählen Sie dann **IAM & Admin** > **IAM**.
4. Wählen Sie **Hinzufügen** und fügen Sie einen Eintrag für das Servicekonto hinzu.
   a. **Neue Mitglieder**: Geben Sie die E-Mail-Adresse für das Service-Konto ein.
   b. **Rolle**: Wählen Sie **Compute Network Viewer**.
   c. Wählen Sie **Speichern**.

Ergebnis

Das Hinzufügen eines GKE-Clusters mithilfe einer gemeinsamen VPC wird mit Astra vollständig funktionieren.

Erstellen eines Service-Kontokonschlüssels

Statt dem Astra Control Service einen Benutzernamen und ein Passwort anzugeben, stellen Sie beim Hinzufügen des ersten Clusters einen Service-Account-Schlüssel bereit. Astra Control Service verwendet den Service-Account-Schlüssel, um die Identität des Service-Kontos zu ermitteln, das Sie gerade eingerichtet haben.

Der Dienstkontenschlüssel ist Klartext im JavaScript Object Notation (JSON) Format gespeichert. Es enthält Informationen zu den GCP-Ressourcen, auf die Sie Zugriff haben.

Sie können die JSON-Datei nur anzeigen oder herunterladen, wenn Sie den Schlüssel erstellen. Sie können jedoch jederzeit einen neuen Schlüssel erstellen.

Schritte

1. Besuchen Sie Google Cloud und "Erstellen Sie einen Service-Kontokschlüssel über die Konsole, den gcloudbasierten Befehl oder eine andere bevorzugte Methode".
2. Wenn Sie dazu aufgefordert werden, speichern Sie die Servicekontoschlüsseldatei an einem sicheren Ort.

Das folgende Video zeigt, wie der Service-Kontokschlüssel über die Google Cloud-Konsole erstellt wird.

► https://docs.netapp.com/de-de/astra-control-service/media/get-started/video-create-gcp-service-account-

key.mp4 *(video)*

Optional: Netzwerk-Peering für Ihr VPC einrichten

Wenn Sie Cloud Volumes Service für Google Cloud als Storage-Backend-Service nutzen möchten, besteht der letzte Schritt darin, Netzwerk-Peering von Ihrem VPC zum Cloud Volumes Service für Google Cloud einzurichten.

Die einfachste Möglichkeit, Netzwerk-Peering einzurichten, besteht darin, die gcloudbefehle direkt von Cloud Volumes Service zu beziehen. Die Befehle sind über Cloud Volumes Service verfügbar, wenn ein neues Dateisystem erstellt wird.

Schritte

1. "Wechseln Sie zu den Zuordnungen von NetApp BlueXP Regionen weltweit" Und geben Sie den Servicetyp an, den Sie in der Region Google Cloud verwenden möchten, in der sich Ihr Cluster befindet.

   Cloud Volumes Service bietet zwei Arten von Services: CVS und CVS-Performance. "Erfahren Sie mehr über diese Service-Typen".

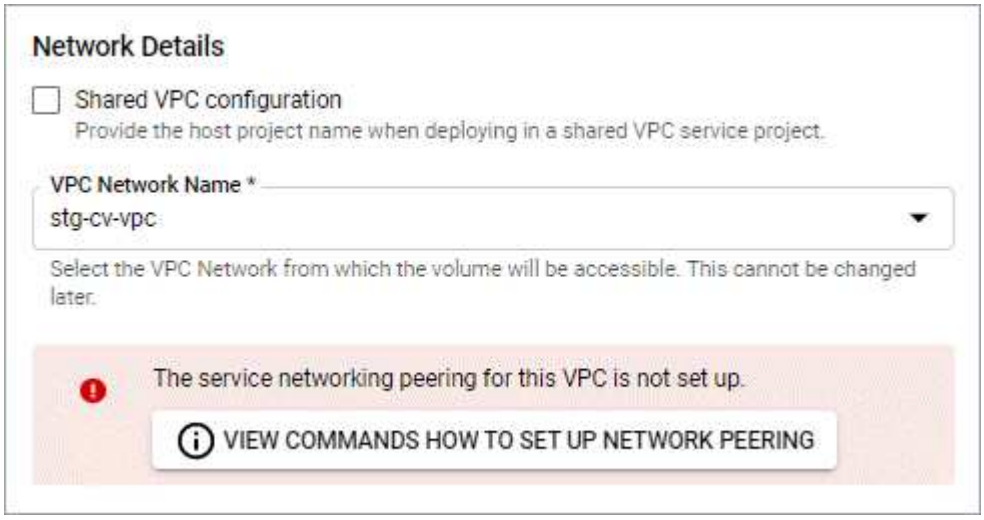
2. "Wechseln Sie zu Cloud Volumes in der Google Cloud Platform".
3. Wählen Sie auf der Seite **Bände** die Option **Erstellen**.
4. Wählen Sie unter **Diensttyp** entweder **CVS** oder **CVS-Performance** aus.

   Sie müssen den richtigen Servicetyp für Ihre Google Cloud-Region auswählen. Dies ist der Service-Typ, den Sie in Schritt 1 identifiziert haben. Nachdem Sie einen Servicetyp ausgewählt haben, wird die Liste der Regionen auf der Seite mit den Regionen aktualisiert, in denen dieser Servicetyp unterstützt wird.

   Nach diesem Schritt müssen Sie nur Ihre Netzwerkinformationen eingeben, um die Befehle abzurufen.

5. Wählen Sie unter **Region** Ihre Region und Zone aus.
6. Wählen Sie unter **Netzwerkdetails** die VPC aus.

   Wenn Sie Netzwerk-Peering nicht eingerichtet haben, sehen Sie die folgende Benachrichtigung:



7. Wählen Sie die Schaltfläche aus, um die Befehle zum Einrichten von Netzwerk-Peering anzuzeigen.
8. Kopieren Sie die Befehle und führen Sie sie in Cloud Shell aus.

Weitere Informationen zur Verwendung dieser Befehle finden Sie im "QuickStart for Cloud Volumes Service for GCP".

"Erfahren Sie mehr über die Konfiguration des Zugriffs auf private Services und die Einrichtung von Netzwerk-Peering".

9. Nachdem Sie fertig sind, können Sie auf der Seite **Dateisystem erstellen** Abbrechen auswählen.

   Wir haben mit dem Erstellen dieses Volumes nur begonnen, um die Befehle für Netzwerk-Peering zu erhalten.

Microsoft Azure mit Azure NetApp Files einrichten

Einige Schritte sind zur Vorbereitung Ihres Microsoft Azure Abonnements erforderlich, bevor Sie Azure Kubernetes Service-Cluster mit Astra Control Service managen können. Folgen Sie diesen Anweisungen, wenn Sie Azure NetApp Files als Storage-Back-End verwenden möchten.

Schnellstart für die Einrichtung von Azure

Führen Sie diese Schritte schnell durch, oder scrollen Sie nach unten zu den verbleibenden Abschnitten, um ausführliche Informationen zu erhalten.

[Eins] Astra Control Service-Anforderungen für Azure Kubernetes Service prüfen

Vergewissern Sie sich, dass die Cluster ordnungsgemäß sind und eine unterstützte Version von Kubernetes ausführen, dass Node-Pools unter Linux verfügbar sind und unter anderem. Erfahren Sie mehr zu diesem Schritt.

[Zwei] Melden Sie sich für Microsoft Azure an

Erstellen Sie ein Microsoft Azure Konto. Erfahren Sie mehr zu diesem Schritt.

[Drittens] Für Azure NetApp Files anmelden

Registrieren Sie den NetApp Resource Provider. Erfahren Sie mehr zu diesem Schritt.

[Vier] Erstellen Sie einen NetApp Account

Erstellen Sie im Azure-Portal unter Azure NetApp Files einen NetApp Account. Erfahren Sie mehr zu diesem Schritt.

[Fünf] Einrichten von Kapazitäts-Pools

Richten Sie einen oder mehrere Kapazitäts-Pools für Ihre persistenten Volumes ein. Erfahren Sie mehr zu diesem Schritt.

[Sechs] Delegieren eines Subnetzes an Azure NetApp Files

Delegieren Sie ein Subnetz an Azure NetApp Files, damit der Astra Control Service persistente Volumes in diesem Subnetz erstellen kann. Erfahren Sie mehr zu diesem Schritt.

[Sieben] Erstellen Sie einen Azure Service Principal

Erstellen Sie einen Azure-Serviceprincipal mit der Rolle „Contributor“. Erfahren Sie mehr zu diesem Schritt.

[Acht] Optional: Redundanz für Azure Backup Buckets konfigurieren

Standardmäßig verwendet der Buckets Astra Control Service für das Speichern von Azure Kubernetes Service-Backups die LRS-Redundanzoption (lokal Redundant Storage). Als optionaler Schritt können Sie einen langlebigen Grad an Redundanz für Azure Buckets konfigurieren. Erfahren Sie mehr zu diesem Schritt.

Anforderungen für den Azure Kubernetes Service-Cluster

Ein Kubernetes-Cluster muss folgende Anforderungen erfüllen, damit Sie ihn über den Astra Control Service erkennen und managen können.

Kubernetes-Version
: Auf Clustern muss Kubernetes Version 1.26 bis 1.28 ausgeführt werden.

Bildtyp
: Der Image-Typ für alle Node-Pools muss Linux sein.

Der Cluster-Status
: Cluster müssen in einem ordnungsgemäßen Zustand ausgeführt werden und mindestens einen Online-Worker-Node ohne „Worker“-Nodes im ausgefallenen Status aufweisen.

Azure Region
: Cluster müssen in einer Region residieren, in der Azure NetApp Files verfügbar ist. "Hier finden Sie Azure Produkte nach Region".

Abonnement
: Cluster müssen in einem Abonnement gespeichert werden, in dem Azure NetApp Files aktiviert ist. Sie wählen ein Abonnement, wenn Sie für Azure NetApp Files anmelden.

Vnet
: Folgende vnet-Anforderungen sind zu berücksichtigen:

- Cluster müssen sich in einem vnet befinden, das direkten Zugriff auf ein für Azure NetApp Files delegiertes Subnetz hat. Erfahren Sie, wie Sie ein delegiertes Subnetz einrichten.
- Wenn sich Ihre Kubernetes Cluster in einem vnet befinden, das über das von Azure NetApp Files delegierte Subnetz in einem anderen vnet verfügt, müssen beide Seiten der Peering-Verbindung online sein.
- Beachten Sie, dass die Standardgrenze für die Anzahl der IP-Adressen, die in einem vnet (einschließlich sofort gepedierter VNets) mit Azure NetApp Files verwendet werden, 1,000 ist. "Zeigen Sie Einschränkungen für Azure NetApp Files-Ressourcen an".

  Wenn Sie nahe am Limit sind, haben Sie zwei Möglichkeiten:

  - Das können Sie "Senden Sie eine Anfrage für eine Grenzerhöhung". Wenden Sie sich an Ihren NetApp Ansprechpartner, wenn Sie Hilfe benötigen.
  - Geben Sie bei der Erstellung eines neuen Amazon Kubernetes Service (AKS)-Clusters ein neues Netzwerk für den Cluster an. Sobald das neue Netzwerk erstellt wurde, stellen Sie ein neues Subnetz bereit und delegieren Sie das Subnetz an Azure NetApp Files.

Melden Sie sich für Microsoft Azure an

Wenn Sie kein Microsoft Azure Konto haben, melden Sie sich zunächst bei Microsoft Azure an.

Schritte

1. Wechseln Sie zum "Azure-Abonnementseite" Um den Azure Service zu abonnieren.
2. Wählen Sie einen Plan aus, und befolgen Sie die Anweisungen, um das Abonnement abzuschließen.

Für Azure NetApp Files anmelden

Erhalten Sie Zugriff auf Azure NetApp Files, indem Sie den NetApp Resource Provider registrieren.

Schritte

1. Melden Sie sich beim Azure Portal an.
2. "Registrieren Sie den NetApp Ressourcenanbieter mithilfe der Azure NetApp Files Dokumentation".

Erstellen Sie einen NetApp Account

Erstellen Sie einen NetApp Account in Azure NetApp Files.

Schritt

1. "Erstellen Sie mit der Azure NetApp Files Dokumentation ein NetApp Konto aus dem Azure Portal".

Richten Sie einen Kapazitäts-Pool ein

Ein oder mehrere Kapazitäts-Pools sind erforderlich, damit der Astra Control Service persistente Volumes in einem Kapazitäts-Pool bereitstellen kann. Astra Control Service erstellt keine Kapazitäts-Pools.

Berücksichtigen Sie bei der Einrichtung von Kapazitäts-Pools für Ihre Kubernetes-Applikationen folgende Punkte:

- Die Kapazitätspools müssen in derselben Region Azure erstellt werden, in der die AKS-Cluster mit Astra Control Service verwaltet werden.
- Ein Kapazitäts-Pool kann ein Ultra-, Premium- oder Standard-Service-Level haben. Jedes dieser Service-Level ist für unterschiedliche Performance-Anforderungen konzipiert. Astra Control Service unterstützt alle drei.

  Sie müssen für jedes Service-Level, das Sie mit Ihren Kubernetes Clustern verwenden möchten, einen Kapazitäts-Pool einrichten.

  "Erfahren Sie mehr über Service-Level für Azure NetApp Files".

- Bevor Sie einen Kapazitäts-Pool für die Applikationen erstellen, die Sie mit dem Astra Control Service schützen möchten, wählen Sie die erforderliche Performance und Kapazität für diese Anwendungen.

  Durch die Bereitstellung der richtigen Kapazität wird sichergestellt, dass Benutzer persistente Volumes nach Bedarf erstellen können. Wenn keine Kapazität verfügbar ist, können die persistenten Volumes nicht bereitgestellt werden.

- Ein Azure NetApp Files-Kapazitäts-Pool kann den manuellen oder automatischen QoS-Typ verwenden. Astra Control Service unterstützt automatische QoS-Kapazitäts-Pools. Manuelle QoS-Kapazitätspools werden nicht unterstützt.

Schritt

1. "Folgen Sie der Azure NetApp Files Dokumentation, um einen automatischen QoS-Kapazitätspool einzurichten".

Delegieren eines Subnetzes an Azure NetApp Files

Sie müssen ein Subnetz an Azure NetApp Files delegieren, damit der Astra Control Service persistente Volumes in diesem Subnetz erstellen kann. Beachten Sie, dass Sie mit Azure NetApp Files nur ein delegiertes Subnetz in einem vnet haben können.

Wenn Sie Peered VNets verwenden, müssen beide Seiten der Peering-Verbindung online sein: Die vnet, in der sich Ihre Kubernetes-Cluster befinden, und das vnet mit dem Azure NetApp Files delegierten Subnetz.

Schritt

1. "Folgen Sie der Azure NetApp Files-Dokumentation, um ein Subnetz an Azure NetApp Files zu delegieren".

Nachdem Sie fertig sind

Warten Sie ungefähr 10 Minuten, bevor Sie den im delegierten Subnetz ausgeführten Cluster ermitteln.

Erstellen Sie einen Azure Service Principal

Astra Control Service erfordert einen Azure-Service-Principal, dem die Rolle „Contributor“ zugewiesen wird. Astra Control Service nutzt diesen Service-Principal, um das Management von Kubernetes-Applikationsdaten in Ihrem Auftrag zu vereinfachen.

Ein Service-Principal ist eine Identität, die speziell für die Verwendung mit Anwendungen, Services und Tools erstellt wurde. Durch die Zuweisung einer Rolle zum Service-Principal wird der Zugriff auf bestimmte Azure-Ressourcen beschränkt.

Führen Sie die folgenden Schritte aus, um einen Service-Principal mithilfe der Azure CLI zu erstellen. Sie müssen die Ausgabe in einer JSON-Datei speichern und später den Astra Control Service bereitstellen. "Weitere Details zur Verwendung der CLI finden Sie in der Azure Dokumentation".

Bei den folgenden Schritten wird davon ausgegangen, dass Sie die Berechtigung zum Erstellen eines Service-Principal haben und dass das Microsoft Azure SDK (az-Befehl) auf Ihrem Computer installiert ist.

Anforderungen

- Der Service-Principal muss die regelmäßige Authentifizierung verwenden. Zertifikate werden nicht unterstützt.
- Dem Service Principal muss ein Zugriff auf Ihr Azure Abonnement für Mitarbeiter oder Eigentümer gewährt werden.
- Das Abonnement oder die Ressourcengruppe, die Sie für den Umfang auswählen, muss die AKS-Cluster und Ihr Azure NetApp Files-Konto enthalten.

Schritte

1. Geben Sie die Abonnement- und Mandanten-ID an, in der sich Ihre AKS-Cluster befinden (dies sind die Cluster, die Sie im Astra Control Service verwalten möchten).

```
az configure --list-defaults
az account list --output table
```

2. Führen Sie einen der folgenden Schritte aus, je nachdem, ob Sie ein gesamtes Abonnement oder eine Ressourcengruppe verwenden:
   - Erstellen Sie den Service-Principal, weisen Sie die Rolle Contributor zu und geben Sie den Umfang dem gesamten Abonnement an, in dem sich die Cluster befinden.

     ```
     az ad sp create-for-rbac --name service-principal-name --role
     contributor --scopes /subscriptions/SUBSCRIPTION-ID
     ```

   - Erstellen Sie den Service-Principal, weisen Sie die Contributor-Rolle zu und geben Sie die Ressourcengruppe an, in der sich die Cluster befinden.

     ```
     az ad sp create-for-rbac --name service-principal-name --role
     contributor --scopes /subscriptions/SUBSCRIPTION-
     ID/resourceGroups/RESOURCE-GROUP-ID
     ```

3. Speichern Sie die resultierende Azure CLI-Ausgabe als JSON-Datei.

   Sie müssen diese Datei bereitstellen, damit Astra Control Service Ihre AKS-Cluster erkennen und Kubernetes-Datenmanagement-Vorgänge managen kann. "Erfahren Sie mehr über das Management von Anmeldeinformationen im Astra Control Service".

4. Optional: Fügen Sie die Abonnement-ID der JSON-Datei hinzu, damit der Astra Control Service beim Auswählen der Datei automatisch die ID füllt.

   Andernfalls müssen Sie die Abonnement-ID in Astra Control Service eingeben, wenn Sie dazu aufgefordert werden.

   Beispiel

   ```
   {
     "appId": "0db3929a-bfb0-4c93-baee-aaf8",
     "displayName": "sp-example-dev-sandbox",
     "name": "http://sp-example-dev-sandbox",
     "password": "mypassword",
     "tenant": "011cdf6c-7512-4805-aaf8-7721afd8ca37",
     "subscriptionId": "99ce999a-8c99-99d9-a9d9-99cce99f99ad"
   }
   ```

5. Optional: Testen Sie Ihren Service-Principal. Wählen Sie je nach Umfang, den Ihr Service Principal verwendet, die folgenden Beispielbefehle aus.

Abonnement-Umfang

```
az login --service-principal --username APP-ID-SERVICEPRINCIPAL
--password PASSWORD --tenant TENANT-ID
az group list --subscription SUBSCRIPTION-ID
az aks list --subscription SUBSCRIPTION-ID
az storage container list --account-name STORAGE-ACCOUNT-NAME
```

Umfang der Ressourcengruppen

```
az login --service-principal --username APP-ID-SERVICEPRINCIPAL
--password PASSWORD --tenant TENANT-ID
az aks list --subscription SUBSCRIPTION-ID --resource-group RESOURCE-
GROUP-ID
```

Optional: Redundanz für Azure Backup Buckets konfigurieren

Es besteht die Möglichkeit, eine robuenere Redundanzstufe für Azure Backup Buckets zu konfigurieren. Standardmäßig verwendet der Buckets Astra Control Service für das Speichern von Azure Kubernetes Service-Backups die LRS-Redundanzoption (lokal Redundant Storage). Um eine langlebige Redundanzoption für Azure Buckets zu verwenden, müssen Sie Folgendes tun:

Schritte

1. Erstellen Sie ein Azure-Storage-Konto, das die erforderliche Redundanzstufe verwendet "Diese Anweisungen".
2. Erstellen Sie einen Azure-Container auf dem neuen Storage-Konto mit "Diese Anweisungen".
3. Fügen Sie den Container als Eimer zum Astra Control Service hinzu. Siehe "Fügen Sie einen zusätzlichen Bucket hinzu".
4. (Optional) um den neu erstellten Bucket als Standard-Bucket für Azure Backups zu verwenden, setzen Sie ihn als Standard-Bucket für Azure fest. Siehe "Ändern des Standard-Bucket".

Richten Sie Microsoft Azure mit von Azure gemanagten Festplatten ein

Einige Schritte sind zur Vorbereitung Ihres Microsoft Azure Abonnements erforderlich, bevor Sie Azure Kubernetes Service-Cluster mit Astra Control Service managen können. Befolgen Sie diese Anweisungen, wenn Sie die von Azure verwalteten Laufwerke als Storage-Back-End verwenden möchten.

Schnellstart für die Einrichtung von Azure

Führen Sie diese Schritte schnell durch, oder scrollen Sie nach unten zu den verbleibenden Abschnitten, um ausführliche Informationen zu erhalten.

[Eins] Astra Control Service-Anforderungen für Azure Kubernetes Service prüfen

Vergewissern Sie sich, dass die Cluster ordnungsgemäß sind und eine unterstützte Version von Kubernetes ausführen, dass Node-Pools unter Linux verfügbar sind und unter anderem. Erfahren Sie mehr zu diesem Schritt.

[Zwei] Melden Sie sich für Microsoft Azure an

Erstellen Sie ein Microsoft Azure Konto. Erfahren Sie mehr zu diesem Schritt.

[Drittens] Erstellen Sie einen Azure Service Principal

Erstellen Sie einen Azure-Serviceprincipal mit der Rolle „Contributor“. Erfahren Sie mehr zu diesem Schritt.

[Vier] Konfigurieren Sie die Treiberdetails für die Container-Storage-Schnittstelle (CSI)

Sie müssen Ihr Azure-Abonnement und das Cluster konfigurieren, damit Sie mit den CSI-Treibern arbeiten können. Erfahren Sie mehr zu diesem Schritt.

[Fünf] Optional: Redundanz für Azure Backup Buckets konfigurieren

Standardmäßig verwendet der Buckets Astra Control Service für das Speichern von Azure Kubernetes Service-Backups die LRS-Redundanzoption (lokal Redundant Storage). Als optionaler Schritt können Sie einen langlebigen Grad an Redundanz für Azure Buckets konfigurieren. Erfahren Sie mehr zu diesem Schritt.

Anforderungen für den Azure Kubernetes Service-Cluster

Ein Kubernetes-Cluster muss folgende Anforderungen erfüllen, damit Sie ihn über den Astra Control Service erkennen und managen können.

Kubernetes-Version
: Auf Clustern muss Kubernetes Version 1.26 bis 1.28 ausgeführt werden.

Bildtyp
: Der Image-Typ für alle Node-Pools muss Linux sein.

Der Cluster-Status
: Cluster müssen in einem ordnungsgemäßen Zustand ausgeführt werden und mindestens einen Online-Worker-Node ohne „Worker“-Nodes im ausgefallenen Status aufweisen.

Azure Region
: Als Best Practice sollte eine Region gewählt werden, die Azure NetApp Files unterstützt, auch wenn Sie sie nicht als Storage-Backend verwenden. Dadurch ist es einfacher, Azure NetApp Files zukünftig als Storage-Backend zu verwenden, wenn sich Ihre Performance-Anforderungen ändern. "Hier finden Sie Azure Produkte nach Region".

CSI-Treiber
: Auf Clustern müssen die entsprechenden CSI-Treiber installiert sein.

Melden Sie sich für Microsoft Azure an

Wenn Sie kein Microsoft Azure Konto haben, melden Sie sich zunächst bei Microsoft Azure an.

Schritte

1. Wechseln Sie zum "Azure-Abonnementseite" Um den Azure Service zu abonnieren.
2. Wählen Sie einen Plan aus, und befolgen Sie die Anweisungen, um das Abonnement abzuschließen.

Erstellen Sie einen Azure Service Principal

Astra Control Service erfordert einen Azure-Service-Principal, dem die Rolle „Contributor“ zugewiesen wird. Astra Control Service nutzt diesen Service-Principal, um das Management von Kubernetes-Applikationsdaten in Ihrem Auftrag zu vereinfachen.

Ein Service-Principal ist eine Identität, die speziell für die Verwendung mit Anwendungen, Services und Tools erstellt wurde. Durch die Zuweisung einer Rolle zum Service-Principal wird der Zugriff auf bestimmte Azure-Ressourcen beschränkt.

Führen Sie die folgenden Schritte aus, um einen Service-Principal mithilfe der Azure CLI zu erstellen. Sie müssen die Ausgabe in einer JSON-Datei speichern und später den Astra Control Service bereitstellen. "Weitere Details zur Verwendung der CLI finden Sie in der Azure Dokumentation".

Bei den folgenden Schritten wird davon ausgegangen, dass Sie die Berechtigung zum Erstellen eines Service-Principal haben und dass das Microsoft Azure SDK (az-Befehl) auf Ihrem Computer installiert ist.

Anforderungen

- Der Service-Principal muss die regelmäßige Authentifizierung verwenden. Zertifikate werden nicht unterstützt.
- Dem Service Principal muss ein Zugriff auf Ihr Azure Abonnement für Mitarbeiter oder Eigentümer gewährt werden.
- Das Abonnement oder die Ressourcengruppe, die Sie für den Umfang auswählen, muss die AKS-Cluster und Ihr Azure NetApp Files-Konto enthalten.

Schritte

1. Geben Sie die Abonnement- und Mandanten-ID an, in der sich Ihre AKS-Cluster befinden (dies sind die Cluster, die Sie im Astra Control Service verwalten möchten).

   ```
   az configure --list-defaults
   az account list --output table
   ```

2. Führen Sie einen der folgenden Schritte aus, je nachdem, ob Sie ein gesamtes Abonnement oder eine Ressourcengruppe verwenden:
   - Erstellen Sie den Service-Principal, weisen Sie die Rolle Contributor zu und geben Sie den Umfang dem gesamten Abonnement an, in dem sich die Cluster befinden.

     ```
     az ad sp create-for-rbac --name service-principal-name --role
     contributor --scopes /subscriptions/SUBSCRIPTION-ID
     ```

   - Erstellen Sie den Service-Principal, weisen Sie die Contributor-Rolle zu und geben Sie die Ressourcengruppe an, in der sich die Cluster befinden.

     ```
     az ad sp create-for-rbac --name service-principal-name --role
     contributor --scopes /subscriptions/SUBSCRIPTION-
     ID/resourceGroups/RESOURCE-GROUP-ID
     ```

3. Speichern Sie die resultierende Azure CLI-Ausgabe als JSON-Datei.

   Sie müssen diese Datei bereitstellen, damit Astra Control Service Ihre AKS-Cluster erkennen und Kubernetes-Datenmanagement-Vorgänge managen kann. "Erfahren Sie mehr über das Management von Anmeldeinformationen im Astra Control Service".

4. Optional: Fügen Sie die Abonnement-ID der JSON-Datei hinzu, damit der Astra Control Service beim Auswählen der Datei automatisch die ID füllt.

   Andernfalls müssen Sie die Abonnement-ID in Astra Control Service eingeben, wenn Sie dazu aufgefordert werden.

   Beispiel

   ```
   {
     "appId": "0db3929a-bfb0-4c93-baee-aaf8",
     "displayName": "sp-example-dev-sandbox",
     "name": "http://sp-example-dev-sandbox",
     "password": "mypassword",
     "tenant": "011cdf6c-7512-4805-aaf8-7721afd8ca37",
     "subscriptionId": "99ce999a-8c99-99d9-a9d9-99cce99f99ad"
   }
   ```

5. Optional: Testen Sie Ihren Service-Principal. Wählen Sie je nach Umfang, den Ihr Service Principal verwendet, die folgenden Beispielbefehle aus.

   Abonnement-Umfang

   ```
   az login --service-principal --username APP-ID-SERVICEPRINCIPAL
   --password PASSWORD --tenant TENANT-ID
   az group list --subscription SUBSCRIPTION-ID
   az aks list --subscription SUBSCRIPTION-ID
   az storage container list --account-name STORAGE-ACCOUNT-NAME
   ```

   Umfang der Ressourcengruppen

   ```
   az login --service-principal --username APP-ID-SERVICEPRINCIPAL
   --password PASSWORD --tenant TENANT-ID
   az aks list --subscription SUBSCRIPTION-ID --resource-group RESOURCE-
   GROUP-ID
   ```

Konfigurieren Sie die Treiberdetails für die Container-Storage-Schnittstelle (CSI)

Wenn Sie verwaltete Azure-Festplatten mit dem Astra Control Service verwenden möchten, müssen Sie die erforderlichen CSI-Treiber installieren.

Aktivieren Sie die CSI-Treiber-Funktion in Ihrem Azure-Abonnement

Bevor Sie die CSI-Treiber installieren, müssen Sie die CSI-Treiberfunktion in Ihrem Azure-Abonnement aktivieren.

Schritte

1. Öffnen Sie die Azure-Befehlszeilenschnittstelle.
2. Führen Sie den folgenden Befehl aus, um den Treiber zu registrieren:

```
az feature register --namespace "Microsoft.ContainerService" --name
"EnableAzureDiskFileCSIDriver"
```

3. Führen Sie den folgenden Befehl aus, um sicherzustellen, dass die Änderung propagiert wird:

```
az provider register -n Microsoft.ContainerService
```

Sie sollten eine Ausgabe wie die folgende sehen:

```
{
"id": "/subscriptions/b200155f-001a-43be-87be-
3edde83acef4/providers/Microsoft.Features/providers/Microsoft.ContainerSer
vice/features/EnableAzureDiskFileCSIDriver",
"name": "Microsoft.ContainerService/EnableAzureDiskFileCSIDriver",
"properties": {
   "state": "Registering"
},
"type": "Microsoft.Features/providers/features"
}
```

Installieren Sie die von Azure gemanagten CSI-Treiber in Ihrem Azure Kubernetes Service-Cluster

Sie können die Azure CSI Treiber installieren, um Ihre Vorbereitung abzuschließen.

Schritt

1. Gehen Sie zu "Die Microsoft CSI-Treiberdokumentation".
2. Befolgen Sie die Anweisungen zur Installation der erforderlichen CSI-Treiber.

Optional: Redundanz für Azure Backup Buckets konfigurieren

Es besteht die Möglichkeit, eine robuenere Redundanzstufe für Azure Backup Buckets zu konfigurieren. Standardmäßig verwendet der Buckets Astra Control Service für das Speichern von Azure Kubernetes Service-Backups die LRS-Redundanzoption (lokal Redundant Storage). Um eine langlebige Redundanzoption für Azure Buckets zu verwenden, müssen Sie Folgendes tun:

Schritte

1. Erstellen Sie ein Azure-Storage-Konto, das die erforderliche Redundanzstufe verwendet "Diese

Anweisungen".

2. Erstellen Sie einen Azure-Container auf dem neuen Storage-Konto mit "Diese Anweisungen".
3. Fügen Sie den Container als Eimer zum Astra Control Service hinzu. Siehe "Fügen Sie einen zusätzlichen Bucket hinzu".
4. (Optional) um den neu erstellten Bucket als Standard-Bucket für Azure Backups zu verwenden, setzen Sie ihn als Standard-Bucket für Azure fest. Siehe "Ändern des Standard-Bucket".

Registrieren Sie sich für ein Astra Control Service-Konto

Um Astra Control Service nutzen zu können, benötigen Sie ein Astra Control Service Konto, das zu Ihrem NetApp BlueXP Konto verknüpft ist. Füllen Sie den Registrierungsprozess für Astra Control Service aus, und wenn Sie noch kein BlueXP Konto besitzen, melden Sie sich bei BlueXP an, um auf den Astra Control Service zuzugreifen.

Registrieren Sie sich für ein Astra Control Konto

Bevor Sie sich beim Astra Control Service anmelden können, müssen Sie einen Registrierungsvorgang abschließen, um ein Astra Control Service-Konto zu erhalten.

Wenn Sie den Astra Control Service nutzen, verwalten Sie Ihre Apps über ein Konto. Ein Konto umfasst Benutzer, die die Apps im Konto anzeigen und verwalten können, sowie Ihre Rechnungsdaten.

Schritte

1. "Wechseln Sie zur Seite Astra Control bei BlueXP".
2. Wählen Sie **Anmeldung für den kostenlosen Plan**.
3. Geben Sie die erforderlichen Informationen in das Formular ein.

   Beim Ausfüllen des Formulars sind einige wichtige Punkte zu beachten:

   - Ihr Unternehmensname und Ihre Adresse müssen korrekt sein, da wir sie überprüfen, um die Anforderungen der Global Trade Compliance zu erfüllen.
   - Der **Astra-Kundenname** ist der Name Ihres Astra Control Service-Kontos. Diesen Namen sehen Sie in der Benutzeroberfläche des Astra Control Service. Beachten Sie, dass Sie bei Bedarf weitere Konten (bis zu 5) erstellen können.
   - Wenn Sie ein NetApp BlueXP Konto haben, geben Sie im Feld **geschäftliche E-Mail-Adresse** die E-Mail ein, die Sie für dieses Konto verwenden. Wenn Sie noch kein NetApp BlueXP Konto haben, verwenden Sie bei Ihrer Anmeldung zu BlueXP die hier eingegebene E-Mail-Adresse.
4. Wählen Sie **Konto Erstellen**.

Melden Sie sich bei BlueXP an

Der Astra Control Service ist in den Authentifizierungsservice von NetApp BlueXP integriert. Sie können sich mit Ihren Zugangsdaten für die BlueXP oder die NetApp Support-Website bei NetApp BlueXP anmelden. Wenn Sie noch kein NetApp BlueXP oder NetApp Support Site Konto haben, melden Sie sich bei BlueXP an. Damit haben Sie Zugriff auf Astra Control Service und weitere Cloud-Services von NetApp. Wenn Sie bereits über ein BlueXP oder NetApp Konto verfügen und die Registrierung abgeschlossen haben, können Sie darauf zugreifen "Astra Control Service" Sie können Ihre Anmeldedaten für BlueXP oder die NetApp Support-Website direkt verwenden.

Sie können sich auch mit Single Sign-On über Anmeldedaten Ihres Unternehmensverzeichnisses bei BlueXP anmelden (föderierte Identität). Weitere Informationen erhalten Sie im "Hilfe-Center" Und wählen Sie dann **Cloud Central Anmelde-Optionen**.

Schritte

1. Gehen Sie zu "NetApp BlueXP".
2. Wählen Sie oben rechts **erste Schritte**.
3. Wählen Sie **Registrieren**.
4. Füllen Sie das Formular aus.

   Stellen Sie sicher, dass die Telefonnummer und die E-Mail-Adresse, die Sie hier eingeben, mit denen übereinstimmen, die Sie im vorherigen Astra Control-Registrierungsformular verwendet haben.

5. Wählen Sie **Registrieren**.

   Die E-Mail-Adresse, die Sie in diese Formulare eingeben, gilt für Ihre NetApp BlueXP Benutzer-ID. Verwenden Sie diese BlueXP Benutzer-ID, wenn Sie sich für ein neues Astra Control Konto anmelden oder wenn ein Astra Control Administrator Sie zu einem vorhandenen Astra Control Konto einlädt.

6. Warten Sie auf eine E-Mail von NetApp BlueXP. Die E-Mail-Adresse stammt von saas.support@netapp.com und kann einige Minuten dauern. Überprüfen Sie Ihren Spam-Ordner.
7. Wenn die E-Mail eintrifft, wählen Sie den Link in der E-Mail aus, um Ihre E-Mail-Adresse zu überprüfen.

Ergebnis

Sie verfügen jetzt über eine aktive BlueXP Benutzeranmeldung.

Da Sie jetzt registriert sind, können Sie mit Ihren BlueXP Zugangsdaten direkt auf Astra Control zugreifen https://astra.netapp.io.

Fügen Sie dem Astra Control Service einen Cluster hinzu

Nach der Einrichtung der Umgebung erstellen Sie sofort einen Kubernetes Cluster und fügen ihn dann dem Astra Control Service hinzu. Auf diese Weise können Sie Astra Control Service zum Schutz Ihrer Anwendungen auf dem Cluster verwenden.

Je nach dem Cluster-Typ, den Sie zum Astra Control Service hinzufügen müssen, müssen Sie den Cluster mit verschiedenen Schritten hinzufügen.

- "Fügen Sie Astra Control Service einen über einen öffentlichen Provider gemanagten Cluster hinzu": Verwenden Sie diese Schritte, um einen Cluster hinzuzufügen, der eine öffentliche IP-Adresse hat und von einem Cloud-Provider verwaltet wird. Sie benötigen das Service Principal-Konto, das Service-Konto oder das Benutzerkonto für den Cloud-Provider.
- "Fügen Sie dem Astra Control Service einen über privaten Provider gemanagten Cluster hinzu": Verwenden Sie diese Schritte, um einen Cluster hinzuzufügen, der eine private IP-Adresse hat und von einem Cloud-Provider verwaltet wird. Sie benötigen das Service Principal-Konto, das Service-Konto oder das Benutzerkonto für den Cloud-Provider.
- "Fügen Sie Astra Control Service einen öffentlichen, selbst gemanagten Cluster hinzu": Verwenden Sie diese Schritte, um einen Cluster hinzuzufügen, der eine öffentliche IP-Adresse hat und von Ihrer

Organisation verwaltet wird. Sie müssen eine kubeconfig-Datei für den Cluster erstellen, den Sie hinzufügen möchten.

- "Fügen Sie Astra Control Service einen privaten, selbst gemanagten Cluster hinzu": Verwenden Sie diese Schritte, um einen Cluster hinzuzufügen, der eine private IP-Adresse hat und von Ihrer Organisation verwaltet wird. Sie müssen eine kubeconfig-Datei für den Cluster erstellen, den Sie hinzufügen möchten.

Astra Connector installieren, um Cluster zu managen

Astra Connector ist eine Software, die sich auf Ihren gemanagten Clustern befindet und die Kommunikation zwischen dem gemanagten Cluster und Astra Control erleichtert. Für Cluster, die mit Astra Control Service verwaltet werden, stehen zwei Versionen von Astra Connector zur Verfügung:

- **Frühere Version des Astra Connectors**: "Installieren Sie die vorherige Version von Astra Connector" In Ihrem Cluster, wenn Sie den Cluster mit nicht-Kubernetes-nativen Workflows managen möchten.
- [Tech Preview] **Declarative Kubernetes Astra Connector**: "Installieren Sie Astra Connector für Cluster, die mit deklarativen Kubernetes-Workflows gemanagt werden" Wenn Sie den Cluster mit deklarativen Kubernetes-Workflows managen möchten, befinden Sie sich in Ihrem Cluster. Nachdem Sie den Astra Connector auf Ihrem Cluster installiert haben, wird der Cluster automatisch zu Astra Control hinzugefügt.

> Der deklarative Kubernetes Astra Connector ist nur im Rahmen des Astra Control Early Adopter Program (EAP) verfügbar. Informationen zum Beitritt zum EAP erhalten Sie von Ihrem NetApp Ansprechpartner.

Installieren Sie die vorherige Version von Astra Connector

Astra Control Service verwendet die vorherige Version von Astra Connector, um die Kommunikation zwischen Astra Control Service und privaten Clustern zu ermöglichen, die über nicht-Kubernetes-native Workflows gemanagt werden. Sie müssen Astra Connector auf privaten Clustern installieren, die Sie mit nicht-Kubernetes-nativen Workflows managen möchten.

Die vorherige Version von Astra Connector unterstützt die folgenden Typen von privaten Clustern, die mit nicht-Kubernetes-nativen Workflows gemanagt werden:

- Amazon Elastic Kubernetes Service (EKS)
- Azure Kubernetes-Service (AKS)
- Google Kubernetes Engine (GKE)
- Red hat OpenShift Service auf AWS (ROSA)
- ROSA mit AWS PrivateLink
- Red hat OpenShift-Container-Plattform vor Ort

Über diese Aufgabe

- Wenn Sie diese Schritte ausführen, führen Sie diese Befehle für den privaten Cluster aus, den Sie mit Astra Control Service managen möchten.
- Wenn Sie einen Bastion-Host verwenden, geben Sie diese Befehle über die Befehlszeile des Bastion-Hosts aus.

Bevor Sie beginnen

- Sie benötigen Zugriff auf den privaten Cluster, den Sie mit Astra Control Service managen möchten.
- Sie benötigen Kubernetes-Administratorberechtigungen, um den Astra Connector Operator auf dem Cluster zu installieren.

Schritte

1. Installieren Sie den vorherigen Astra Connector Operator auf dem privaten Cluster, den Sie mit nicht-Kubernetes-nativen Workflows managen möchten. Wenn Sie diesen Befehl ausführen, wird der Namespace verwendet `astra-connector-operator` Wird erstellt und die Konfiguration wird auf den Namespace angewendet:

```
kubectl apply -f https://github.com/NetApp/astra-connector-operator/releases/download/23.07.0-202310251519/astraconnector_operator.yaml
```

2. Überprüfen Sie, ob der Bediener installiert und bereit ist:

```
kubectl get all -n astra-connector-operator
```

3. Holen Sie sich ein API-Token von Astra Control. Siehe "Dokumentation von Astra Automation" Weitere Anweisungen.
4. astra-Connector-Namespace erstellen:

```
kubectl create ns astra-connector
```

5. Erstellen Sie die Astra Connector CR-Datei und benennen Sie sie `astra-connector-cr.yaml`. Aktualisieren Sie die Werte in Klammern <>, um sie an die Astra Control-Umgebung und die Cluster-Konfiguration anzupassen:
   - **<ASTRA_CONTROL_SERVICE_URL>**: Die Web UI URL des Astra Control Service. Beispiel:

     ```
     https://astra.netapp.io
     ```

   - **<ASTRA_CONTROL_SERVICE_API_TOKEN>**: Das Astra Control API Token, das Sie im vorherigen Schritt erhalten haben.
   - **<PRIVATE_AKS_CLUSTER_NAME>**: (Nur AKS-Cluster) - der Cluster-Name des privaten Azure Kubernetes Service Clusters. Heben Sie die Kommentareingabe auf und füllen Sie diese Zeile nur dann aus, wenn Sie einen privaten AKS-Cluster hinzufügen.
   - **<ASTRA_CONTROL_ACCOUNT_ID>**: Erhalten von der Astra Control Web-Benutzeroberfläche. Wählen Sie das Symbol oben rechts auf der Seite aus und wählen Sie **API Access** aus.

```
apiVersion: netapp.astraconnector.com/v1
kind: AstraConnector
metadata:
  name: astra-connector
  namespace: astra-connector
spec:
  natssync-client:
    cloud-bridge-url: <ASTRA_CONTROL_SERVICE_URL>
  imageRegistry:
    name: theotw
    secret: ""
  astra:
    token: <ASTRA_CONTROL_SERVICE_API_TOKEN>
    #clusterName: <PRIVATE_AKS_CLUSTER_NAME>
    accountId: <ASTRA_CONTROL_ACCOUNT_ID>
    acceptEULA: yes
```

6. Nachdem Sie das ausgefüllt haben `astra-connector-cr.yaml` Datei mit den richtigen Werten, CR anwenden:

```
kubectl apply -f astra-connector-cr.yaml
```

7. Überprüfen Sie, ob der Astra Connector vollständig bereitgestellt ist:

```
kubectl get all -n astra-connector
```

8. Überprüfen Sie, ob das Cluster bei Astra Control registriert ist:

```
kubectl get astraconnector -n astra-connector
```

Sie sollten eine Ausgabe wie die folgende sehen:

```
NAME              REGISTERED   ASTRACONNECTORID
STATUS
astra-connector   true         be475ae5-1511-4eaa-9b9e-712f09b0d065
Registered with Astra
```

Notieren Sie sich die ASTRACONNECTORID, die Sie benötigen, wenn Sie den Cluster zu Astra Control hinzufügen.

Was kommt als Nächstes?

Nachdem Sie jetzt Astra Connector installiert haben, können Sie jetzt Ihrem privaten Cluster den Astra Control Service hinzufügen.

- "Fügen Sie dem Astra Control Service einen über privaten Provider gemanagten Cluster hinzu": Verwenden Sie diese Schritte, um einen Cluster hinzuzufügen, der eine private IP-Adresse hat und von einem Cloud-Provider verwaltet wird. Sie benötigen das Service Principal-Konto, das Service-Konto oder das Benutzerkonto für den Cloud-Provider.
- "Fügen Sie Astra Control Service einen privaten, selbst gemanagten Cluster hinzu": Verwenden Sie diese Schritte, um einen Cluster hinzuzufügen, der eine private IP-Adresse hat und von Ihrer Organisation verwaltet wird. Sie müssen eine kubeconfig-Datei für den Cluster erstellen, den Sie hinzufügen möchten.

Finden Sie weitere Informationen

- "Fügen Sie einen Cluster hinzu"

(Tech Preview) Installieren Sie den deklarativen Kubernetes Astra Connector

Cluster, die über deklarative Kubernetes-Workflows gemanagt werden, ermöglichen über Astra Connector die Kommunikation zwischen dem gemanagten Cluster und Astra Control. Sie müssen Astra Connector auf allen Clustern installieren, die Sie mit deklarativen Kubernetes-Workflows managen werden.

Sie installieren den deklarativen Kubernetes Astra Connector mithilfe von Kubernetes-Befehlen und CR-Dateien (Custom Resource).

Über diese Aufgabe

- Wenn Sie diese Schritte ausführen, führen Sie diese Befehle auf dem Cluster aus, den Sie mit Astra Control managen möchten.
- Wenn Sie einen Bastion-Host verwenden, geben Sie diese Befehle über die Befehlszeile des Bastion-Hosts aus.

Bevor Sie beginnen

- Sie benötigen Zugriff auf den Cluster, den Sie mit Astra Control managen möchten.
- Sie benötigen Kubernetes-Administratorberechtigungen, um den Astra Connector Operator auf dem Cluster zu installieren.

> ⓘ Wenn das Cluster mit der Durchsetzung der Pod-Sicherheitszulassung konfiguriert ist, was der Standard für Kubernetes-Cluster ab Version 1.25 ist, müssen Sie die PSA-Einschränkungen für die entsprechenden Namespaces aktivieren. Siehe "Bereiten Sie Ihre Umgebung mit Astra Control auf das Cluster-Management vor" Weitere Anweisungen.

Schritte

1. Installieren Sie den Astra Connector Operator auf dem Cluster, das Sie mit deklarativen Kubernetes-Workflows managen möchten. Wenn Sie diesen Befehl ausführen, wird der Namespace verwendet `astra-connector-operator` Wird erstellt und die Konfiguration wird auf den Namespace angewendet:

```
kubectl apply -f https://github.com/NetApp/astra-connector-
operator/releases/download/24.02.0-
202403151353/astraconnector_operator.yaml
```

2. Überprüfen Sie, ob der Bediener installiert und bereit ist:

```
kubectl get all -n astra-connector-operator
```

3. Holen Sie sich ein API-Token von Astra Control. Siehe "Dokumentation von Astra Automation" Weitere Anweisungen.
4. Erstellen Sie mithilfe des Tokens einen Schlüssel. Ersetzen Sie <API_TOKEN> durch das Token, das Sie von Astra Control erhalten haben:

```
kubectl create secret generic astra-token \
--from-literal=apiToken=<API_TOKEN> \
-n astra-connector
```

5. Erstellen Sie einen Docker-Schlüssel, um das Astra Connector-Image zu übertragen. Ersetzen Sie Werte in Klammern <> durch Informationen aus Ihrer Umgebung:

> Die <ASTRA_CONTROL_ACCOUNT_ID> finden Sie in der Web-UI von Astra Control. Wählen Sie in der Web-Benutzeroberfläche das Symbol oben rechts auf der Seite aus und wählen Sie **API Access**.

```
kubectl create secret docker-registry regcred \
--docker-username=<ASTRA_CONTROL_ACCOUNT_ID> \
--docker-password=<API_TOKEN> \
-n astra-connector \
--docker-server=cr.astra.netapp.io
```

6. Erstellen Sie die Astra Connector CR-Datei und benennen Sie sie `astra-connector-cr.yaml`. Aktualisieren Sie die Werte in Klammern <>, um sie an die Astra Control-Umgebung und die Cluster-Konfiguration anzupassen:
   - <ASTRA_CONTROL_ACCOUNT_ID>: Erhalten von der Astra Control Web-UI während des vorhergehenden Schritts.
   - <CLUSTER_NAME>: Der Name, dem dieser Cluster in Astra Control zugewiesen werden soll.
   - <ASTRA_CONTROL_URL>: Die Web UI URL von Astra Control. Beispiel:

```
https://astra.control.url
```

```
apiVersion: astra.netapp.io/v1
kind: AstraConnector
metadata:
  name: astra-connector
  namespace: astra-connector
spec:
  astra:
    accountId: <ASTRA_CONTROL_ACCOUNT_ID>
    clusterName: <CLUSTER_NAME>
    #Only set `skipTLSValidation` to `true` when using the default
self-signed
    #certificate in a proof-of-concept environment.
    skipTLSValidation: false #Should be set to false in production
environments
    tokenRef: astra-token
  natsSyncClient:
    cloudBridgeURL: <ASTRA_CONTROL_HOST_URL>
  imageRegistry:
    name: cr.astra.netapp.io
    secret: regcred
```

7. Nachdem Sie das ausgefüllt haben `astra-connector-cr.yaml` Datei mit den richtigen Werten, CR anwenden:

```
kubectl apply -n astra-connector -f astra-connector-cr.yaml
```

8. Überprüfen Sie, ob der Astra Connector vollständig bereitgestellt ist:

```
kubectl get all -n astra-connector
```

9. Überprüfen Sie, ob das Cluster bei Astra Control registriert ist:

```
kubectl get astraconnectors.astra.netapp.io -A
```

Sie sollten eine Ausgabe wie die folgende sehen:

```
NAMESPACE         NAME              REGISTERED   ASTRACONNECTORID
STATUS
astra-connector   astra-connector   true         00ac8-2cef-41ac-8777-
ed0583e   Registered with Astra
```

10. Überprüfen Sie, ob der Cluster in der Liste der verwalteten Cluster auf der Seite **Cluster** der Astra Control Web UI angezeigt wird.

Fügen Sie ein vom Anbieter verwaltetes Cluster hinzu

Fügen Sie Astra Control Service einen über einen öffentlichen Provider gemanagten Cluster hinzu

Nachdem Sie Ihre Cloud-Umgebung eingerichtet haben, sind Sie bereit, ein Kubernetes-Cluster zu erstellen und dieses dann zu Astra Control Service hinzuzufügen.

- Erstellen eines Kubernetes-Clusters
- Fügen Sie das Cluster zu Astra Control Service hinzu
- Ändern der Standard-Storage-Klasse

Erstellen eines Kubernetes-Clusters

Wenn Sie noch keinen Cluster haben, können Sie ein Cluster erstellen, das erfüllt "Astra Control Service-Anforderungen für Amazon Elastic Kubernetes Service (EKS)". Wenn Sie noch keinen Cluster haben, können Sie ein Cluster erstellen, das erfüllt "Astra Control Service-Anforderungen für Google Kubernetes Engine (GKE)". Wenn Sie noch keinen Cluster haben, können Sie ein Cluster erstellen, das erfüllt "Astra Control Service-Anforderungen für Azure Kubernetes Service (AKS) mit Azure NetApp Files" Oder "Astra Control Service-Anforderungen für Azure Kubernetes Service (AKS) mit von Azure gemanagten Festplatten".

> Astra Control Service unterstützt AKS-Cluster, die Azure Active Directory (Azure AD) zur Authentifizierung und Identitätsverwaltung nutzen. Wenn Sie das Cluster erstellen, befolgen Sie die Anweisungen im "Offizielle Dokumentation" Um den Cluster mit Azure AD zu konfigurieren. Stellen Sie sicher, dass Ihre Cluster die Anforderungen für die AKS-verwaltete Azure AD-Integration erfüllen.

Fügen Sie das Cluster zu Astra Control Service hinzu

Nachdem Sie sich beim Astra Control Service angemeldet haben, beginnen Sie zunächst mit dem Verwalten Ihrer Cluster. Bevor Sie Astra Control Service ein Cluster hinzufügen, müssen Sie bestimmte Aufgaben ausführen und sicherstellen, dass das Cluster bestimmte Anforderungen erfüllt.

Beachten Sie beim Management von Azure Kubernetes Service und Google Kubernetes Engine-Clustern, dass für die Installation von Astra Control und das Lifecycle Management zwei Optionen zur Verfügung stehen:

- Mit Astra Control Service können Sie den Lebenszyklus von Astra Control Provisioner automatisch managen. Vergewissern Sie sich dazu, dass Astra Trident nicht installiert ist und Astra Control Provisioner nicht auf dem Cluster aktiviert ist, den Sie mit Astra Control Service managen möchten. In diesem Fall aktiviert Astra Control Service automatisch die Astra Control-Bereitstellung, wenn Sie mit dem Cluster-Management beginnen. Upgrades für die Astra Control-Bereitstellung werden automatisch durchgeführt.
- Sie können den Lebenszyklus der Astra Control Provisionierung selbst managen. Aktivieren Sie hierfür die Astra Control-Provisionierung im Cluster, bevor Sie das Cluster mit Astra Control Service verwalten. In diesem Fall erkennt Astra Control Service, dass die Provisionierung von Astra Control bereits aktiviert ist. Es wird weder neu installiert noch Astra Control Provisioner-Upgrades gemanagt. Siehe "Astra Control Provisioner Aktivieren" Für die Schritte aktivieren Sie die Astra Control-Provisionierung.

Wenn Sie Amazon Web Services Cluster mit Astra Control Service managen, müssen Sie bei Bedarf Storage-Back-Ends, die nur mit dem Astra Control Provisioner verwendet werden können, die Astra Control Service manuell im Cluster aktivieren, bevor Sie die Bereitstellung mit Astra Control Service managen. Siehe "Astra

Control Provisioner Aktivieren" Enthält die Schritte zum Aktivieren der Astra Control-Bereitstellung.

Bevor Sie beginnen

Amazon Web Services

- Sie sollten die JSON-Datei mit den Anmeldedaten des IAM-Benutzers haben, der das Cluster erstellt hat. "Erfahren Sie, wie ein IAM-Benutzer erstellt wird".
- Astra Control Provisioner ist für Amazon FSX for NetApp ONTAP erforderlich. Wenn Sie Amazon FSX for NetApp ONTAP als Storage-Backend für Ihr EKS-Cluster verwenden möchten, finden Sie in den Informationen zur Astra Control-Bereitstellung im "EKS-Clusteranforderungen".
- (Optional) Wenn Sie angeben müssen `kubectl` Befehlszugriff für ein Cluster auf andere IAM-Benutzer, die nicht der Ersteller des Clusters sind, finden Sie in den Anweisungen unter "Wie erhalte ich Zugriff auf andere IAM-Benutzer und Rollen nach der Cluster-Erstellung in Amazon EKS?".
- Wenn Sie NetApp Cloud Volumes ONTAP als Storage-Backend verwenden möchten, müssen Sie Cloud Volumes ONTAP für die Nutzung mit Amazon Web Services konfigurieren. Weitere Informationen finden Sie im Cloud Volumes ONTAP "Setup-Dokumentation".

Microsoft Azure

- Sie sollten beim Erstellen des Service-Principal die JSON-Datei haben, die die Ausgabe aus der Azure CLI enthält. "Erfahren Sie, wie Sie einen Service-Principal einrichten".

  Außerdem benötigen Sie Ihre Azure Abonnement-ID, wenn Sie sie nicht zur JSON-Datei hinzugefügt haben.

- Wenn Sie NetApp Cloud Volumes ONTAP als Storage-Backend verwenden möchten, müssen Sie Cloud Volumes ONTAP für die Zusammenarbeit mit Microsoft Azure konfigurieren. Weitere Informationen finden Sie im Cloud Volumes ONTAP "Setup-Dokumentation".

Google Cloud

- Sie sollten die Servicekontoschlüsseldatei für ein Servicekonto haben, das über die erforderlichen Berechtigungen verfügt. "Erfahren Sie, wie Sie ein Service-Konto einrichten".
- Wenn Sie NetApp Cloud Volumes ONTAP als Storage-Backend verwenden möchten, müssen Sie Cloud Volumes ONTAP für die Zusammenarbeit mit Google Cloud konfigurieren. Weitere Informationen finden Sie im Cloud Volumes ONTAP "Setup-Dokumentation".

Schritte

1. (Optional) Wenn Sie einen Amazon EKS Cluster hinzufügen oder die Installation und Upgrades von Astra Control Provisioner selbst managen möchten, aktivieren Sie die Astra Control Provisioner-Funktion im Cluster. Siehe "Astra Control Provisioner Aktivieren" Für Enablement-Schritte.
2. Öffnen Sie die Web-UI des Astra Control Service in einem Browser.
3. Wählen Sie im Dashboard **Kubernetes Cluster managen** aus.

   Befolgen Sie die Aufforderungen zum Hinzufügen des Clusters.

4. **Provider**: Wählen Sie Ihren Cloud-Provider aus und geben Sie dann entweder die erforderlichen Anmeldedaten für die Erstellung einer neuen Cloud-Instanz an, oder wählen Sie eine vorhandene Cloud-Instanz aus.
5. **Amazon Web Services**: Geben Sie Details über Ihr Amazon Web Services IAM-Benutzerkonto an, indem Sie eine JSON-Datei hochladen oder den Inhalt dieser JSON-Datei aus Ihrer Zwischenablage einfügen.

Die JSON-Datei sollte die Anmeldeinformationen des IAM-Benutzers enthalten, der das Cluster erstellt hat.

6. **Microsoft Azure**: Geben Sie Details zu Ihrem Azure Service Principal an, indem Sie eine JSON-Datei hochladen oder den Inhalt dieser JSON-Datei aus Ihrer Zwischenablage einfügen.

   Die JSON-Datei sollte beim Erstellen des Service-Principal die Ausgabe aus der Azure CLI enthalten. Sie können auch Ihre Abonnement-ID angeben, damit sie automatisch in den Astra aufgenommen wird. Andernfalls müssen Sie die ID manuell eingeben, nachdem Sie den JSON bereitgestellt haben.

7. **Google Cloud Platform**: Stellen Sie die Service-Konto-Schlüsseldatei entweder durch das Hochladen der Datei oder durch Einfügen der Inhalte aus Ihrer Zwischenablage bereit.

   Astra Control Service nutzt das Service-Konto, um Cluster zu erkennen, die in der Google Kubernetes Engine ausgeführt werden.

8. **Andere**: Diese Registerkarte ist nur für die Verwendung mit selbst verwalteten Clustern vorgesehen.
   a. **Cloud-Instanzname**: Geben Sie einen Namen für die neue Cloud-Instanz an, die beim Hinzufügen dieses Clusters erstellt wird. Weitere Informationen zu "Cloud-Instanzen".
   b. Wählen Sie **Weiter**.

      Astra Control Service zeigt eine Liste von Clustern an, aus denen Sie auswählen können.

   c. **Cluster**: Wählen Sie einen Cluster aus der Liste aus, der zu Astra Control Service hinzugefügt werden soll.

      > Wenn Sie aus der Liste der Cluster auswählen, achten Sie auf die Spalte **Eligiblity**. Wenn ein Cluster „nicht berechtigt“ oder „teilweise berechtigt“ ist, bewegen Sie den Mauszeiger über den Status, um zu ermitteln, ob ein Problem im Cluster vorliegt. Beispielsweise kann sie erkennen, dass für das Cluster kein Worker Node vorhanden ist.

   d. Wählen Sie **Weiter**.
   e. (Optional) **Speicher**: Wählen Sie optional die Storage-Klasse aus, die Kubernetes-Anwendungen, die auf diesem Cluster bereitgestellt werden sollen, standardmäßig verwenden sollen.
9. Um eine neue Standard-Storage-Klasse für den Cluster auszuwählen, aktivieren Sie das Kontrollkästchen **Neue Standard-Storage-Klasse zuweisen**.
10. Wählen Sie eine neue Standard-Storage-Klasse aus der Liste aus.

Jeder Storage-Service eines Cloud-Providers enthält die folgenden Informationen zu Preis, Performance und Ausfallsicherheit:

- Cloud Volumes Service für Google Cloud: Informationen zu Preis, Performance und Ausfallsicherheit
- Google Persistent Disk: Keine Informationen über Preis, Performance oder Ausfallsicherheit verfügbar
- Azure NetApp Files: Informationen zu Performance und Ausfallsicherheit
- Azure Managed Disks: Es sind weder Preis-, Performance- oder Resilience-Informationen verfügbar
- Amazon Elastic Block Store: Keine Informationen zu Preis, Performance oder Ausfallsicherheit verfügbar
- Amazon FSX für NetApp ONTAP: Keine Informationen zu Preis, Performance und Ausfallsicherheit verfügbar
- NetApp Cloud Volumes ONTAP: Keine Informationen zu Preis, Performance oder Ausfallsicherheit verfügbar

Jede Storage-Klasse kann einen der folgenden Services nutzen:

- "Cloud Volumes Service für Google Cloud"
- "Google Persistent Disk"
  - "Azure NetApp Dateien"
  - "Von Azure gemanagte Festplatten"
  - "Amazon Elastic Block Store"
  - "Amazon FSX für NetApp ONTAP"
  - "NetApp Cloud Volumes ONTAP"

  Weitere Informationen zu "Storage-Klassen für Amazon Web Services Cluster". Weitere Informationen zu "Speicherklassen für AKS-Cluster". Weitere Informationen zu "Speicherklassen für GKE-Cluster".

  a. Wählen Sie **Weiter**.
  b. **Überprüfen und genehmigen**: Überprüfen Sie die Konfigurationsdetails.
  c. Wählen Sie **Add**, um den Cluster zu Astra Control Service hinzuzufügen.

Ergebnis

Wenn dies der erste Cluster ist, den Sie für diesen Cloud-Provider hinzugefügt haben, erstellt Astra Control Service einen Objektspeicher für den Cloud-Provider für Backups von Anwendungen, die auf geeigneten Clustern ausgeführt werden. (Wenn Sie nachfolgende Cluster für diesen Cloud-Provider hinzufügen, werden keine weiteren Objektspeicher erstellt.) Wenn Sie eine Standard-Storage-Klasse angegeben haben, setzt Astra Control Service die von Ihnen angegebene Standard-Storage-Klasse ein. Für Cluster, die in Amazon Web Services oder Google Cloud Platform gemanagt werden, erstellt Astra Control Service auch ein Administratorkonto auf dem Cluster. Diese Vorgänge können mehrere Minuten dauern.

Ändern der Standard-Storage-Klasse

Sie können die Standard-Storage-Klasse für ein Cluster ändern.

Ändern Sie die Standard-Storage-Klasse mit Astra Control

Sie können die Standard-Storage-Klasse für ein Cluster aus Astra Control ändern. Wenn Ihr Cluster einen zuvor installierten Speicher-Backend-Service verwendet, können Sie diese Methode möglicherweise nicht verwenden, um die Standard-Speicherklasse zu ändern (die Aktion **default** ist nicht wählbar). In diesem Fall können Sie Ändern Sie die Standard-Storage-Klasse über die Befehlszeile.

Schritte

1. Wählen Sie in der Astra Control Service-UI **Cluster** aus.
2. Wählen Sie auf der Seite **Cluster** den Cluster aus, den Sie ändern möchten.
3. Wählen Sie die Registerkarte **Storage** aus.
4. Wählen Sie die Kategorie **Speicherklassen** aus.
5. Wählen Sie das Menü **Aktionen** für die Speicherklasse aus, die Sie als Standard festlegen möchten.
6. Wählen Sie **als Standard**.

Ändern Sie die Standard-Storage-Klasse über die Befehlszeile

Sie können die Standard-Storage-Klasse für ein Cluster mit Kubernetes-Befehlen ändern. Diese Methode funktioniert unabhängig von der Konfiguration Ihres Clusters.

Schritte

1. Melden Sie sich bei Ihrem Kubernetes Cluster an.
2. Listen Sie die Storage-Klassen in Ihrem Cluster auf:

```
kubectl get storageclass
```

3. Entfernen Sie die Standardbezeichnung aus der Standardspeicherklasse. Ersetzen Sie <SC_NAME> durch den Namen der Speicherklasse:

```
kubectl patch storageclass <SC_NAME> -p '{"metadata":
{"annotations":{"storageclass.kubernetes.io/is-default-
class":"false"}}}'
```

4. Markieren Sie standardmäßig eine andere Storage-Klasse. Ersetzen Sie <SC_NAME> durch den Namen der Speicherklasse:

```
kubectl patch storageclass <SC_NAME> -p '{"metadata":
{"annotations":{"storageclass.kubernetes.io/is-default-class":"true"}}}'
```

5. Bestätigen Sie die neue Standard-Speicherklasse:

```
kubectl get storageclass
```

Fügen Sie dem Astra Control Service einen über privaten Provider gemanagten Cluster hinzu

Mit Astra Control Service können Sie private GKE-Cluster (Google Kubernetes Engine) managen. In diesen Anweisungen wird davon ausgegangen, dass Sie bereits einen privaten AKS- oder OpenShift-Cluster erstellt und eine sichere Methode für den Remote-Zugriff darauf vorbereitet haben. Weitere Informationen zum Erstellen und Zugreifen auf private AKS- oder OpenShift-Cluster finden Sie in der folgenden Dokumentation:

- "Azure-Dokumentation für private AKS-Cluster"
- "Azure-Dokumentation für private OpenShift-Cluster"

Mit Astra Control Service können Sie private Azure Kubernetes Service (AKS)-Cluster sowie private Red hat OpenShift-Cluster in AKS managen. In diesen Anweisungen wird davon ausgegangen, dass Sie bereits einen privaten AKS- oder OpenShift-Cluster erstellt und eine sichere Methode für den Remote-Zugriff darauf vorbereitet haben. Weitere Informationen zum Erstellen und Zugreifen auf private AKS- oder OpenShift-Cluster finden Sie in der folgenden Dokumentation:

- "Azure-Dokumentation für private AKS-Cluster"
- "Azure-Dokumentation für private OpenShift-Cluster"

Mit Astra Control Service können Sie private EKS-Cluster (Amazon Elastic Kubernetes Service) managen. In diesen Anweisungen wird davon ausgegangen, dass Sie bereits einen privaten EKS-Cluster erstellt und eine sichere Methode für den Remote-Zugriff darauf vorbereitet haben. Weitere Informationen zum Erstellen und Zugreifen auf private EKS-Cluster finden Sie im "Amazon EKS-Dokumentation".

Führen Sie die folgenden Aufgaben aus, um Ihren privaten Cluster zum Astra Control Service hinzuzufügen:

1. Astra Connector Installieren
2. Einrichtung von persistentem Storage
3. Fügen Sie den über den privaten Provider gemanagten Cluster zu Astra Control Service hinzu

Astra Connector Installieren

Bevor Sie einen privaten Cluster hinzufügen, müssen Sie Astra Connector im Cluster installieren, damit Astra Control damit kommunizieren kann. Siehe "Installieren Sie die vorherige Version von Astra Connector für private Cluster, die mit nicht-Kubernetes-nativen Workflows gemanagt werden" Weitere Anweisungen.

Einrichtung von persistentem Storage

Konfigurieren Sie persistenten Storage für das Cluster. In der Dokumentation „erste Schritte“ finden Sie weitere Informationen zum Konfigurieren von persistentem Storage:

- "Microsoft Azure mit Azure NetApp Files einrichten"
- "Richten Sie Microsoft Azure mit von Azure gemanagten Festplatten ein"
- "Einrichten von Amazon Web Services"
- "Google Cloud einrichten"

Fügen Sie den über den privaten Provider gemanagten Cluster zu Astra Control Service hinzu

Sie können den privaten Cluster jetzt dem Astra Control Service hinzufügen.

Beachten Sie beim Management von Azure Kubernetes Service und Google Kubernetes Engine-Clustern, dass für die Installation von Astra Control und das Lifecycle Management zwei Optionen zur Verfügung stehen:

- Mit Astra Control Service können Sie den Lebenszyklus von Astra Control Provisioner automatisch managen. Vergewissern Sie sich dazu, dass Astra Trident nicht installiert ist und Astra Control Provisioner nicht auf dem Cluster aktiviert ist, den Sie mit Astra Control Service managen möchten. In diesem Fall aktiviert Astra Control Service automatisch die Astra Control-Bereitstellung, wenn Sie mit dem Cluster-Management beginnen. Upgrades für die Astra Control-Bereitstellung werden automatisch durchgeführt.
- Sie können den Lebenszyklus der Astra Control Provisionierung selbst managen. Aktivieren Sie hierfür die Astra Control-Provisionierung im Cluster, bevor Sie das Cluster mit Astra Control Service verwalten. In diesem Fall erkennt Astra Control Service, dass die Provisionierung von Astra Control bereits aktiviert ist. Es wird weder neu installiert noch Astra Control Provisioner-Upgrades gemanagt. Siehe "Astra Control Provisioner Aktivieren" Für die Schritte aktivieren Sie die Astra Control-Provisionierung.

Wenn Sie Amazon Web Services Cluster mit Astra Control Service managen, müssen Sie bei Bedarf Storage-Back-Ends, die nur mit dem Astra Control Provisioner verwendet werden können, die Astra Control Service manuell im Cluster aktivieren, bevor Sie die Bereitstellung mit Astra Control Service managen. Siehe "Astra Control Provisioner Aktivieren" Enthält die Schritte zum Aktivieren der Astra Control-Bereitstellung.

Bevor Sie beginnen

Amazon Web Services

- Sie sollten die JSON-Datei mit den Anmeldedaten des IAM-Benutzers haben, der das Cluster erstellt hat. "Erfahren Sie, wie ein IAM-Benutzer erstellt wird".
- Astra Control Provisioner ist für Amazon FSX for NetApp ONTAP erforderlich. Wenn Sie Amazon FSX for NetApp ONTAP als Storage-Backend für Ihr EKS-Cluster verwenden möchten, finden Sie in den Informationen zur Astra Control-Bereitstellung im "EKS-Clusteranforderungen".
- (Optional) Wenn Sie angeben müssen `kubectl` Befehlszugriff für ein Cluster auf andere IAM-Benutzer, die nicht der Ersteller des Clusters sind, finden Sie in den Anweisungen unter "Wie erhalte ich Zugriff auf andere IAM-Benutzer und Rollen nach der Cluster-Erstellung in Amazon EKS?".
- Wenn Sie NetApp Cloud Volumes ONTAP als Storage-Backend verwenden möchten, müssen Sie Cloud Volumes ONTAP für die Nutzung mit Amazon Web Services konfigurieren. Weitere Informationen finden Sie im Cloud Volumes ONTAP "Setup-Dokumentation".

Microsoft Azure

- Sie sollten beim Erstellen des Service-Principal die JSON-Datei haben, die die Ausgabe aus der Azure CLI enthält. "Erfahren Sie, wie Sie einen Service-Principal einrichten".

  Außerdem benötigen Sie Ihre Azure Abonnement-ID, wenn Sie sie nicht zur JSON-Datei hinzugefügt haben.

- Wenn Sie NetApp Cloud Volumes ONTAP als Storage-Backend verwenden möchten, müssen Sie Cloud Volumes ONTAP für die Zusammenarbeit mit Microsoft Azure konfigurieren. Weitere Informationen finden Sie im Cloud Volumes ONTAP "Setup-Dokumentation".

Google Cloud

- Sie sollten die Servicekontoschlüsseldatei für ein Servicekonto haben, das über die erforderlichen Berechtigungen verfügt. "Erfahren Sie, wie Sie ein Service-Konto einrichten".
- Wenn das Cluster privat ist, gilt das "Autorisierte Netzwerke" Die Astra Control Service-IP-Adresse muss zugelassen werden:

  52.188.218.166/32

- Wenn Sie NetApp Cloud Volumes ONTAP als Storage-Backend verwenden möchten, müssen Sie Cloud Volumes ONTAP für die Zusammenarbeit mit Google Cloud konfigurieren. Weitere Informationen finden Sie im Cloud Volumes ONTAP "Setup-Dokumentation".

Schritte

1. (Optional) Wenn Sie einen Amazon EKS Cluster hinzufügen oder die Installation und Upgrades von Astra Control Provisioner selbst managen möchten, aktivieren Sie die Astra Control Provisioner-Funktion im Cluster. Siehe "Astra Control Provisioner Aktivieren" Für Enablement-Schritte.
2. Öffnen Sie die Web-UI des Astra Control Service in einem Browser.
3. Wählen Sie im Dashboard **Kubernetes Cluster managen** aus.

   Befolgen Sie die Aufforderungen zum Hinzufügen des Clusters.

4. **Provider**: Wählen Sie Ihren Cloud-Provider aus und geben Sie dann entweder die erforderlichen Anmeldedaten für die Erstellung einer neuen Cloud-Instanz an, oder wählen Sie eine vorhandene Cloud-Instanz aus.

5. **Amazon Web Services**: Geben Sie Details über Ihr Amazon Web Services IAM-Benutzerkonto an, indem Sie eine JSON-Datei hochladen oder den Inhalt dieser JSON-Datei aus Ihrer Zwischenablage einfügen.

   Die JSON-Datei sollte die Anmeldeinformationen des IAM-Benutzers enthalten, der das Cluster erstellt hat.

6. **Microsoft Azure**: Geben Sie Details zu Ihrem Azure Service Principal an, indem Sie eine JSON-Datei hochladen oder den Inhalt dieser JSON-Datei aus Ihrer Zwischenablage einfügen.

   Die JSON-Datei sollte beim Erstellen des Service-Principal die Ausgabe aus der Azure CLI enthalten. Sie können auch Ihre Abonnement-ID angeben, damit sie automatisch in den Astra aufgenommen wird. Andernfalls müssen Sie die ID manuell eingeben, nachdem Sie den JSON bereitgestellt haben.

7. **Google Cloud Platform**: Stellen Sie die Service-Konto-Schlüsseldatei entweder durch das Hochladen der Datei oder durch Einfügen der Inhalte aus Ihrer Zwischenablage bereit.

   Astra Control Service nutzt das Service-Konto, um Cluster zu erkennen, die in der Google Kubernetes Engine ausgeführt werden.

8. **Andere**: Diese Registerkarte ist nur für die Verwendung mit selbst verwalteten Clustern vorgesehen.
   a. **Cloud-Instanzname**: Geben Sie einen Namen für die neue Cloud-Instanz an, die beim Hinzufügen dieses Clusters erstellt wird. Weitere Informationen zu "Cloud-Instanzen".
   b. Wählen Sie **Weiter**.

      Astra Control Service zeigt eine Liste von Clustern an, aus denen Sie auswählen können.

   c. **Cluster**: Wählen Sie einen Cluster aus der Liste aus, der zu Astra Control Service hinzugefügt werden soll.

      Wenn Sie aus der Liste der Cluster auswählen, achten Sie auf die Spalte **Eligiblity**. Wenn ein Cluster „nicht berechtigt“ oder „teilweise berechtigt“ ist, bewegen Sie den Mauszeiger über den Status, um zu ermitteln, ob ein Problem im Cluster vorliegt. Beispielsweise kann sie erkennen, dass für das Cluster kein Worker Node vorhanden ist.

9. Wählen Sie **Weiter**.
10. (Optional) **Speicher**: Wählen Sie optional die Storage-Klasse aus, die Kubernetes-Anwendungen, die auf diesem Cluster bereitgestellt werden sollen, standardmäßig verwenden sollen.
    a. Um eine neue Standard-Storage-Klasse für den Cluster auszuwählen, aktivieren Sie das Kontrollkästchen **Neue Standard-Storage-Klasse zuweisen**.
    b. Wählen Sie eine neue Standard-Storage-Klasse aus der Liste aus.

Jeder Storage-Service eines Cloud-Providers enthält die folgenden Informationen zu Preis, Performance und Ausfallsicherheit:

- Cloud Volumes Service für Google Cloud: Informationen zu Preis, Performance und Ausfallsicherheit
- Google Persistent Disk: Keine Informationen über Preis, Performance oder Ausfallsicherheit verfügbar
- Azure NetApp Files: Informationen zu Performance und Ausfallsicherheit
- Azure Managed Disks: Es sind weder Preis-, Performance- oder Resilience-Informationen verfügbar
- Amazon Elastic Block Store: Keine Informationen zu Preis, Performance oder Ausfallsicherheit verfügbar
- Amazon FSX für NetApp ONTAP: Keine Informationen zu Preis, Performance und Ausfallsicherheit verfügbar
- NetApp Cloud Volumes ONTAP: Keine Informationen zu Preis, Performance oder Ausfallsicherheit verfügbar

Jede Storage-Klasse kann einen der folgenden Services nutzen:

- "Cloud Volumes Service für Google Cloud"
- "Google Persistent Disk"
- "Azure NetApp Dateien"
- "Von Azure gemanagte Festplatten"
- "Amazon Elastic Block Store"
- "Amazon FSX für NetApp ONTAP"
- "NetApp Cloud Volumes ONTAP"

Weitere Informationen zu "Storage-Klassen für Amazon Web Services Cluster". Weitere Informationen zu "Speicherklassen für AKS-Cluster". Weitere Informationen zu "Speicherklassen für GKE-Cluster".

c. Wählen Sie **Weiter**.
d. **Überprüfen und genehmigen**: Überprüfen Sie die Konfigurationsdetails.
e. Wählen Sie **Add**, um den Cluster zu Astra Control Service hinzuzufügen.

Ergebnis

Wenn dies der erste Cluster ist, den Sie für diesen Cloud-Provider hinzugefügt haben, erstellt Astra Control Service einen Objektspeicher für den Cloud-Provider für Backups von Anwendungen, die auf geeigneten Clustern ausgeführt werden. (Wenn Sie nachfolgende Cluster für diesen Cloud-Provider hinzufügen, werden keine weiteren Objektspeicher erstellt.) Wenn Sie eine Standard-Storage-Klasse angegeben haben, setzt Astra Control Service die von Ihnen angegebene Standard-Storage-Klasse ein. Für Cluster, die in Amazon Web Services oder Google Cloud Platform gemanagt werden, erstellt Astra Control Service auch ein Administratorkonto auf dem Cluster. Diese Vorgänge können mehrere Minuten dauern.

Ändern der Standard-Storage-Klasse

Sie können die Standard-Storage-Klasse für ein Cluster ändern.

Ändern Sie die Standard-Storage-Klasse mit Astra Control

Sie können die Standard-Storage-Klasse für ein Cluster aus Astra Control ändern. Wenn Ihr Cluster einen zuvor installierten Speicher-Backend-Service verwendet, können Sie diese Methode möglicherweise nicht verwenden, um die Standard-Speicherklasse zu ändern (die Aktion **default** ist nicht wählbar). In diesem Fall können Sie Ändern Sie die Standard-Storage-Klasse über die Befehlszeile.

Schritte

1. Wählen Sie in der Astra Control Service-UI **Cluster** aus.
2. Wählen Sie auf der Seite **Cluster** den Cluster aus, den Sie ändern möchten.
3. Wählen Sie die Registerkarte **Storage** aus.
4. Wählen Sie die Kategorie **Speicherklassen** aus.
5. Wählen Sie das Menü **Aktionen** für die Speicherklasse aus, die Sie als Standard festlegen möchten.
6. Wählen Sie **als Standard**.

Ändern Sie die Standard-Storage-Klasse über die Befehlszeile

Sie können die Standard-Storage-Klasse für ein Cluster mit Kubernetes-Befehlen ändern. Diese Methode funktioniert unabhängig von der Konfiguration Ihres Clusters.

Schritte

1. Melden Sie sich bei Ihrem Kubernetes Cluster an.
2. Listen Sie die Storage-Klassen in Ihrem Cluster auf:

```
kubectl get storageclass
```

3. Entfernen Sie die Standardbezeichnung aus der Standardspeicherklasse. Ersetzen Sie <SC_NAME> durch den Namen der Speicherklasse:

```
kubectl patch storageclass <SC_NAME> -p '{"metadata":
{"annotations":{"storageclass.kubernetes.io/is-default-
class":"false"}}}'
```

4. Markieren Sie standardmäßig eine andere Storage-Klasse. Ersetzen Sie <SC_NAME> durch den Namen der Speicherklasse:

```
kubectl patch storageclass <SC_NAME> -p '{"metadata":
{"annotations":{"storageclass.kubernetes.io/is-default-class":"true"}}}'
```

5. Bestätigen Sie die neue Standard-Speicherklasse:

```
kubectl get storageclass
```

Hinzufügen eines selbstverwalteten Clusters

Fügen Sie Astra Control Service einen öffentlichen, selbst gemanagten Cluster hinzu

Nach der Einrichtung der Umgebung erstellen Sie sofort einen Kubernetes Cluster und fügen ihn dann dem Astra Control Service hinzu.

Ein selbstverwalteter Cluster ist ein Cluster, den Sie direkt bereitstellen und managen können. Astra Control Service unterstützt selbst gemanagte Cluster, die in einer Public-Cloud-Umgebung ausgeführt werden. Sie können einen selbstverwalteten Cluster zum Astra Control Service hinzufügen, indem Sie ein hochladen `kubeconfig.yaml` Datei: Sie müssen sicherstellen, dass das Cluster die hier aufgeführten Anforderungen erfüllt.

Unterstützte Kubernetes-Distributionen

Mit Astra Control Service können Sie folgende Arten von öffentlichen, selbst gemanagten Clustern managen:

Kubernetes-Distribution	Unterstützte Versionen
Kubernetes (Vorgelagert)	1.27 bis 1.29
Rancher Kubernetes Engine (RKE)	RKE 1: Versionen 1.24.17, 1.25.13, 1.26.8 mit Rancher Manager 2.7.9 RKE 2: Versionen 1.23.16 und 1.24.13 mit Rancher Manager 2.6.13 RKE 2: Versionen 1.24.17, 1.25.14, 1.26.9 mit Rancher Manager 2.7.9
Red hat OpenShift Container Platform	4.12 bis 4.14

Bei diesen Anweisungen wird davon ausgegangen, dass Sie bereits einen selbstverwalteten Cluster erstellt haben.

- Fügen Sie das Cluster zu Astra Control Service hinzu
- Ändern der Standard-Storage-Klasse

Fügen Sie das Cluster zu Astra Control Service hinzu

Nachdem Sie sich beim Astra Control Service angemeldet haben, beginnen Sie zunächst mit dem Verwalten Ihrer Cluster. Bevor Sie Astra Control Service ein Cluster hinzufügen, müssen Sie bestimmte Aufgaben ausführen und sicherstellen, dass das Cluster bestimmte Anforderungen erfüllt.

Bevor Sie beginnen

Ein selbstverwalteter Cluster ist ein Cluster, den Sie direkt bereitstellen und managen können. Astra Control Service unterstützt selbst gemanagte Cluster, die in einer Public-Cloud-Umgebung ausgeführt werden. Die selbstverwalteten Cluster können über Astra Control Provisioner eine Schnittstelle zu NetApp Storage-Services aufbauen. Alternativ können sie über CSI-Treiber (Container Storage Interface) eine Schnittstelle zu Amazon Elastic Block Store (EBS), Azure Managed Disks und Google Persistent Disk erstellen.

Astra Control Service unterstützt selbst gemanagte Cluster, die die folgenden Kubernetes-Distributionen verwenden:

- Red hat OpenShift Container Platform
- Rancher Kubernetes Engine
- Vorgelagerte Kubernetes-Systeme

Ihr Self-Managed-Cluster muss folgende Anforderungen erfüllen:

- Der Cluster muss über das Internet zugänglich sein.
- Wenn Sie Speicher mit CSI-Treibern verwenden oder planen, diese zu verwenden, müssen auf dem Cluster die entsprechenden CSI-Treiber installiert sein. Weitere Informationen zur Verwendung von CSI-Treibern zur Integration von Speicher finden Sie in der Dokumentation Ihres Speicherservices.
- Sie haben Zugriff auf die Cluster-Datei kubeconfig, die nur ein Kontextelement enthält. Folgen "Diese Anweisungen" Um eine kubeconfig-Datei zu erzeugen.
- Wenn Sie den Cluster mit einer kubeconfig-Datei hinzufügen, die auf eine private Zertifizierungsstelle verweist, fügen Sie der folgende Zeile hinzu `cluster` Abschnitt der Datei kubeconfig. So kann Astra Control das Cluster hinzufügen:

  ```
  insecure-skip-tls-verify: true
  ```

- **Rancher only**: Ändern Sie beim Verwalten von Anwendungsclustern in einer Rancher-Umgebung den Standardkontext des Anwendungsclusters in der von Rancher bereitgestellten kubeconfig-Datei, um einen Steuerebenen-Kontext anstelle des Rancher API-Serverkontexts zu verwenden. So wird die Last auf dem Rancher API Server reduziert und die Performance verbessert.
- **Anforderungen für die Astra Control-Bereitstellung**: Sie sollten einen ordnungsgemäß konfigurierten Astra Control Provisioner einschließlich der Astra Trident-Komponenten verwenden, um Cluster zu managen.
  - **Umgebungsanforderungen für Astra Trident prüfen**: Lesen Sie vor der Installation oder dem Upgrade von Astra Control Provisioner die "Unterstützte Frontends, Back-Ends und Host-Konfigurationen".
  - **Astra Control-Provisioner aktivieren**: Es wird dringend empfohlen, Astra Trident 23.10 oder höher zu installieren und zu aktivieren "Astra Control bietet erweiterte Storage-Funktionen zur Bereitstellung". In den kommenden Versionen unterstützt Astra Control nicht Astra Trident, wenn der Astra Control Provisioner nicht ebenfalls aktiviert ist.
  - **Konfiguration eines Speicher-Backends**: Mindestens ein Speicher-Backend muss sein "In Astra Trident konfiguriert" Auf dem Cluster.
  - **Konfiguration einer Storage-Klasse**: Mindestens eine Storage-Klasse muss sein "In Astra Trident konfiguriert" Auf dem Cluster. Wenn eine Standardspeicherklasse konfiguriert ist, stellen

Sie sicher, dass sie die **einzige** Speicherklasse ist, die die Standardanmerkung hat.

- **Konfigurieren Sie einen Volume-Snapshot-Controller und installieren Sie eine Volume-Snapshot-Klasse**: "Installieren Sie einen Volume-Snapshot-Controller" Damit Snapshots in Astra Control erstellt werden können. "Erstellen" Mindestens eine `VolumeSnapshotClass` Einsatz von Astra Trident:

Schritte

1. Wählen Sie im Dashboard **Kubernetes Cluster managen** aus.

   Befolgen Sie die Aufforderungen zum Hinzufügen des Clusters.

2. **Provider**: Wählen Sie den Reiter **andere**, um Details zu Ihrem selbst verwalteten Cluster hinzuzufügen.
   a. **Other**: Geben Sie Details über Ihren selbstverwalteten Cluster durch das Hochladen eines `kubeconfig.yaml` Datei oder durch Einfügen des Inhalts des `kubeconfig.yaml` Datei aus der Zwischenablage.

   > Wenn Sie Ihre eigenen erstellen `kubeconfig` Datei, Sie sollten nur ein **ein**-Kontext -Element darin definieren. Siehe "Kubernetes-Dokumentation" Weitere Informationen zum Erstellen `kubeconfig` Dateien:

3. **Credential Name**: Geben Sie einen Namen für die selbstverwalteten Cluster-Zugangsdaten ein, die Sie auf Astra Control hochladen. Standardmäßig wird der Name der Anmeldeinformationen automatisch als Name des Clusters ausgefüllt.
4. **Private Route Identifier**: Dieses Feld ist nur für private Cluster bestimmt.
5. Wählen Sie **Weiter**.
6. (Optional) **Speicher**: Wählen Sie optional die Storage-Klasse aus, die Kubernetes-Anwendungen, die auf diesem Cluster bereitgestellt werden sollen, standardmäßig verwenden sollen.
   a. Um eine neue Standard-Storage-Klasse für den Cluster auszuwählen, aktivieren Sie das Kontrollkästchen **Neue Standard-Storage-Klasse zuweisen**.
   b. Wählen Sie eine neue Standard-Storage-Klasse aus der Liste aus.

Jeder Storage-Service eines Cloud-Providers enthält die folgenden Informationen zu Preis, Performance und Ausfallsicherheit:

- Cloud Volumes Service für Google Cloud: Informationen zu Preis, Performance und Ausfallsicherheit
- Google Persistent Disk: Keine Informationen über Preis, Performance oder Ausfallsicherheit verfügbar
- Azure NetApp Files: Informationen zu Performance und Ausfallsicherheit
- Azure Managed Disks: Es sind weder Preis-, Performance- oder Resilience-Informationen verfügbar
- Amazon Elastic Block Store: Keine Informationen zu Preis, Performance oder Ausfallsicherheit verfügbar
- Amazon FSX für NetApp ONTAP: Keine Informationen zu Preis, Performance und Ausfallsicherheit verfügbar
- NetApp Cloud Volumes ONTAP: Keine Informationen zu Preis, Performance oder Ausfallsicherheit verfügbar

Jede Storage-Klasse kann einen der folgenden Services nutzen:

- "Cloud Volumes Service für Google Cloud"
- "Google Persistent Disk"
  - "Azure NetApp Dateien"
  - "Von Azure gemanagte Festplatten"
  - "Amazon Elastic Block Store"
  - "Amazon FSX für NetApp ONTAP"
  - "NetApp Cloud Volumes ONTAP"

  Weitere Informationen zu "Storage-Klassen für Amazon Web Services Cluster". Weitere Informationen zu "Speicherklassen für AKS-Cluster". Weitere Informationen zu "Speicherklassen für GKE-Cluster".

c. Wählen Sie **Weiter**.
d. **Überprüfen und genehmigen**: Überprüfen Sie die Konfigurationsdetails.
e. Wählen Sie **Add**, um den Cluster zu Astra Control Service hinzuzufügen.

Ändern der Standard-Storage-Klasse

Sie können die Standard-Storage-Klasse für ein Cluster ändern.

Ändern Sie die Standard-Storage-Klasse mit Astra Control

Sie können die Standard-Storage-Klasse für ein Cluster aus Astra Control ändern. Wenn Ihr Cluster einen zuvor installierten Speicher-Backend-Service verwendet, können Sie diese Methode möglicherweise nicht verwenden, um die Standard-Speicherklasse zu ändern (die Aktion **default** ist nicht wählbar). In diesem Fall können Sie Ändern Sie die Standard-Storage-Klasse über die Befehlszeile.

Schritte

1. Wählen Sie in der Astra Control Service-UI **Cluster** aus.
2. Wählen Sie auf der Seite **Cluster** den Cluster aus, den Sie ändern möchten.
3. Wählen Sie die Registerkarte **Storage** aus.
4. Wählen Sie die Kategorie **Speicherklassen** aus.
5. Wählen Sie das Menü **Aktionen** für die Speicherklasse aus, die Sie als Standard festlegen möchten.
6. Wählen Sie **als Standard**.

Ändern Sie die Standard-Storage-Klasse über die Befehlszeile

Sie können die Standard-Storage-Klasse für ein Cluster mit Kubernetes-Befehlen ändern. Diese Methode funktioniert unabhängig von der Konfiguration Ihres Clusters.

Schritte

1. Melden Sie sich bei Ihrem Kubernetes Cluster an.
2. Listen Sie die Storage-Klassen in Ihrem Cluster auf:

```
kubectl get storageclass
```

3. Entfernen Sie die Standardbezeichnung aus der Standardspeicherklasse. Ersetzen Sie <SC_NAME> durch den Namen der Speicherklasse:

```
kubectl patch storageclass <SC_NAME> -p '{"metadata":
{"annotations":{"storageclass.kubernetes.io/is-default-
class":"false"}}}'
```

4. Markieren Sie standardmäßig eine andere Storage-Klasse. Ersetzen Sie <SC_NAME> durch den Namen der Speicherklasse:

```
kubectl patch storageclass <SC_NAME> -p '{"metadata":
{"annotations":{"storageclass.kubernetes.io/is-default-class":"true"}}}'
```

5. Bestätigen Sie die neue Standard-Speicherklasse:

```
kubectl get storageclass
```

Fügen Sie Astra Control Service einen privaten, selbst gemanagten Cluster hinzu

Nach der Einrichtung der Umgebung erstellen Sie sofort einen Kubernetes Cluster und fügen ihn dann dem Astra Control Service hinzu.

Ein selbstverwalteter Cluster ist ein Cluster, den Sie direkt bereitstellen und managen können. Astra Control Service unterstützt selbst gemanagte Cluster, die in einer Public-Cloud-Umgebung ausgeführt werden. Sie können einen selbstverwalteten Cluster zum Astra Control Service hinzufügen, indem Sie ein hochladen

`kubeconfig.yaml` Datei: Sie müssen sicherstellen, dass das Cluster die hier aufgeführten Anforderungen erfüllt.

Unterstützte Kubernetes-Distributionen

Mit Astra Control Service können Sie folgende Arten von privaten, selbst gemanagten Clustern managen:

Kubernetes-Distribution	Unterstützte Versionen
Kubernetes (Vorgelagert)	1.27 bis 1.29
Rancher Kubernetes Engine (RKE)	RKE 1: Versionen 1.24.17, 1.25.13, 1.26.8 mit Rancher Manager 2.7.9 RKE 2: Versionen 1.23.16 und 1.24.13 mit Rancher Manager 2.6.13 RKE 2: Versionen 1.24.17, 1.25.14, 1.26.9 mit Rancher Manager 2.7.9
Red hat OpenShift Container Platform	4.12 bis 4.14

In diesen Anweisungen wird davon ausgegangen, dass Sie bereits einen privaten Cluster erstellt und eine sichere Methode für den Remote-Zugriff darauf vorbereitet haben.

Führen Sie die folgenden Aufgaben aus, um Ihren privaten Cluster zum Astra Control Service hinzuzufügen:

1. Astra Connector Installieren
2. Einrichtung von persistentem Storage
3. selbst gemanagten Cluster zum Astra Control Service hinzu

Astra Connector Installieren

Bevor Sie einen privaten Cluster hinzufügen, müssen Sie Astra Connector im Cluster installieren, damit Astra Control damit kommunizieren kann. Siehe "Installieren Sie die vorherige Version von Astra Connector für private Cluster, die mit nicht-Kubernetes-nativen Workflows gemanagt werden" Weitere Anweisungen.

Einrichtung von persistentem Storage

Konfigurieren Sie persistenten Storage für das Cluster. In der Dokumentation „erste Schritte“ finden Sie weitere Informationen zum Konfigurieren von persistentem Storage:

- "Microsoft Azure mit Azure NetApp Files einrichten"
- "Richten Sie Microsoft Azure mit von Azure gemanagten Festplatten ein"
- "Einrichten von Amazon Web Services"
- "Google Cloud einrichten"

Fügen Sie den privaten, selbst gemanagten Cluster zum Astra Control Service hinzu

Sie können den privaten Cluster jetzt dem Astra Control Service hinzufügen.

Bevor Sie beginnen

Ein selbstverwalteter Cluster ist ein Cluster, den Sie direkt bereitstellen und managen können. Astra Control Service unterstützt selbst gemanagte Cluster, die in einer Public-Cloud-Umgebung ausgeführt werden. Die selbstverwalteten Cluster können über Astra Control Provisioner eine Schnittstelle zu NetApp Storage-Services aufbauen. Alternativ können sie über CSI-Treiber (Container Storage Interface) eine Schnittstelle zu Amazon Elastic Block Store (EBS), Azure Managed Disks und Google Persistent Disk erstellen.

Astra Control Service unterstützt selbst gemanagte Cluster, die die folgenden Kubernetes-Distributionen verwenden:

- Red hat OpenShift Container Platform
- Rancher Kubernetes Engine
- Vorgelagerte Kubernetes-Systeme

Ihr Self-Managed-Cluster muss folgende Anforderungen erfüllen:

- Der Cluster muss über das Internet zugänglich sein.
- Wenn Sie Speicher mit CSI-Treibern verwenden oder planen, diese zu verwenden, müssen auf dem Cluster die entsprechenden CSI-Treiber installiert sein. Weitere Informationen zur Verwendung von CSI-Treibern zur Integration von Speicher finden Sie in der Dokumentation Ihres Speicherservices.
- Sie haben Zugriff auf die Cluster-Datei kubeconfig, die nur ein Kontextelement enthält. Folgen "Diese Anweisungen" Um eine kubeconfig-Datei zu erzeugen.
- Wenn Sie den Cluster mit einer kubeconfig-Datei hinzufügen, die auf eine private Zertifizierungsstelle verweist, fügen Sie der folgende Zeile hinzu `cluster` Abschnitt der Datei kubeconfig. So kann Astra Control das Cluster hinzufügen:

  ```
  insecure-skip-tls-verify: true
  ```

- **Rancher only**: Ändern Sie beim Verwalten von Anwendungsclustern in einer Rancher-Umgebung den Standardkontext des Anwendungsclusters in der von Rancher bereitgestellten kubeconfig-Datei, um einen Steuerebenen-Kontext anstelle des Rancher API-Serverkontexts zu verwenden. So wird die Last auf dem Rancher API Server reduziert und die Performance verbessert.
- **Anforderungen für die Astra Control-Bereitstellung**: Sie sollten einen ordnungsgemäß konfigurierten Astra Control Provisioner einschließlich der Astra Trident-Komponenten verwenden, um Cluster zu managen.
  - **Umgebungsanforderungen für Astra Trident prüfen**: Lesen Sie vor der Installation oder dem Upgrade von Astra Control Provisioner die "Unterstützte Frontends, Back-Ends und Host-Konfigurationen".
  - **Astra Control-Provisioner aktivieren**: Es wird dringend empfohlen, Astra Trident 23.10 oder höher zu installieren und zu aktivieren "Astra Control bietet erweiterte Storage-Funktionen zur Bereitstellung". In den kommenden Versionen unterstützt Astra Control nicht Astra Trident, wenn der Astra Control Provisioner nicht ebenfalls aktiviert ist.
  - **Konfiguration eines Speicher-Backends**: Mindestens ein Speicher-Backend muss sein "In Astra Trident konfiguriert" Auf dem Cluster.
  - **Konfiguration einer Storage-Klasse**: Mindestens eine Storage-Klasse muss sein "In Astra Trident konfiguriert" Auf dem Cluster. Wenn eine Standardspeicherklasse konfiguriert ist, stellen

Sie sicher, dass sie die **einzige** Speicherklasse ist, die die Standardanmerkung hat.

- **Konfigurieren Sie einen Volume-Snapshot-Controller und installieren Sie eine Volume-Snapshot-Klasse**: "Installieren Sie einen Volume-Snapshot-Controller" Damit Snapshots in Astra Control erstellt werden können. "Erstellen" Mindestens eine `VolumeSnapshotClass` Einsatz von Astra Trident:

Schritte

1. Wählen Sie im Dashboard **Kubernetes Cluster managen** aus.

   Befolgen Sie die Aufforderungen zum Hinzufügen des Clusters.

2. **Provider**: Wählen Sie den Reiter **andere**, um Details zu Ihrem selbst verwalteten Cluster hinzuzufügen.
3. **Other**: Geben Sie Details über Ihren selbstverwalteten Cluster durch das Hochladen eines `kubeconfig.yaml` Datei oder durch Einfügen des Inhalts des `kubeconfig.yaml` Datei aus der Zwischenablage.

   Wenn Sie Ihre eigenen erstellen `kubeconfig` Datei, Sie sollten nur ein **ein**-Kontext -Element darin definieren. Siehe "Diese Anweisungen" Weitere Informationen zum Erstellen `kubeconfig` Dateien:

4. **Credential Name**: Geben Sie einen Namen für die selbstverwalteten Cluster-Zugangsdaten ein, die Sie auf Astra Control hochladen. Standardmäßig wird der Name der Anmeldeinformationen automatisch als Name des Clusters ausgefüllt.
5. **Private Route Identifier**: Geben Sie die private Route Identifier ein, die Sie vom Astra Connector erhalten können. Wenn Sie den Astra Connector über die abfragen `kubectl get astraconnector -n astra-connector` Die Kennung der privaten Route wird als bezeichnet `ASTRACONNECTORID`.

   Die Private-Route-ID ist der Name, der dem Astra Connector zugeordnet ist. Damit kann ein privates Kubernetes-Cluster von Astra gemanagt werden. In diesem Kontext ist ein privates Cluster ein Kubernetes-Cluster, das seinen API-Server nicht zum Internet bereitstellt.

6. Wählen Sie **Weiter**.
7. (Optional) **Speicher**: Wählen Sie optional die Storage-Klasse aus, die Kubernetes-Anwendungen, die auf diesem Cluster bereitgestellt werden sollen, standardmäßig verwenden sollen.
   a. Um eine neue Standard-Storage-Klasse für den Cluster auszuwählen, aktivieren Sie das Kontrollkästchen **Neue Standard-Storage-Klasse zuweisen**.
   b. Wählen Sie eine neue Standard-Storage-Klasse aus der Liste aus.

Jeder Storage-Service eines Cloud-Providers enthält die folgenden Informationen zu Preis, Performance und Ausfallsicherheit:

- Cloud Volumes Service für Google Cloud: Informationen zu Preis, Performance und Ausfallsicherheit
- Google Persistent Disk: Keine Informationen über Preis, Performance oder Ausfallsicherheit verfügbar
- Azure NetApp Files: Informationen zu Performance und Ausfallsicherheit
- Azure Managed Disks: Es sind weder Preis-, Performance- oder Resilience-Informationen verfügbar
- Amazon Elastic Block Store: Keine Informationen zu Preis, Performance oder Ausfallsicherheit verfügbar
- Amazon FSX für NetApp ONTAP: Keine Informationen zu Preis, Performance und Ausfallsicherheit verfügbar
- NetApp Cloud Volumes ONTAP: Keine Informationen zu Preis, Performance oder Ausfallsicherheit verfügbar

Jede Storage-Klasse kann einen der folgenden Services nutzen:

- "Cloud Volumes Service für Google Cloud"
- "Google Persistent Disk"
- "Azure NetApp Dateien"
- "Von Azure gemanagte Festplatten"
- "Amazon Elastic Block Store"
- "Amazon FSX für NetApp ONTAP"
- "NetApp Cloud Volumes ONTAP"

Weitere Informationen zu "Storage-Klassen für Amazon Web Services Cluster". Weitere Informationen zu "Speicherklassen für AKS-Cluster". Weitere Informationen zu "Speicherklassen für GKE-Cluster".

c. Wählen Sie **Weiter**.
d. **Überprüfen und genehmigen**: Überprüfen Sie die Konfigurationsdetails.
e. Wählen Sie **Add**, um den Cluster zu Astra Control Service hinzuzufügen.

Ändern der Standard-Storage-Klasse

Sie können die Standard-Storage-Klasse für ein Cluster ändern.

Ändern Sie die Standard-Storage-Klasse mit Astra Control

Sie können die Standard-Storage-Klasse für ein Cluster aus Astra Control ändern. Wenn Ihr Cluster einen zuvor installierten Speicher-Backend-Service verwendet, können Sie diese Methode möglicherweise nicht verwenden, um die Standard-Speicherklasse zu ändern (die Aktion **default** ist nicht wählbar). In diesem Fall können Sie Ändern Sie die Standard-Storage-Klasse über die Befehlszeile.

Schritte

1. Wählen Sie in der Astra Control Service-UI **Cluster** aus.
2. Wählen Sie auf der Seite **Cluster** den Cluster aus, den Sie ändern möchten.
3. Wählen Sie die Registerkarte **Storage** aus.
4. Wählen Sie die Kategorie **Speicherklassen** aus.
5. Wählen Sie das Menü **Aktionen** für die Speicherklasse aus, die Sie als Standard festlegen möchten.
6. Wählen Sie **als Standard**.

Ändern Sie die Standard-Storage-Klasse über die Befehlszeile

Sie können die Standard-Storage-Klasse für ein Cluster mit Kubernetes-Befehlen ändern. Diese Methode funktioniert unabhängig von der Konfiguration Ihres Clusters.

Schritte

1. Melden Sie sich bei Ihrem Kubernetes Cluster an.
2. Listen Sie die Storage-Klassen in Ihrem Cluster auf:

```
kubectl get storageclass
```

3. Entfernen Sie die Standardbezeichnung aus der Standardspeicherklasse. Ersetzen Sie <SC_NAME> durch den Namen der Speicherklasse:

```
kubectl patch storageclass <SC_NAME> -p '{"metadata":
{"annotations":{"storageclass.kubernetes.io/is-default-
class":"false"}}}'
```

4. Markieren Sie standardmäßig eine andere Storage-Klasse. Ersetzen Sie <SC_NAME> durch den Namen der Speicherklasse:

```
kubectl patch storageclass <SC_NAME> -p '{"metadata":
{"annotations":{"storageclass.kubernetes.io/is-default-class":"true"}}}'
```

5. Bestätigen Sie die neue Standard-Speicherklasse:

```
kubectl get storageclass
```

Prüfen Sie die Astra Trident Version

Wenn Sie einen selbst gemanagten Cluster hinzufügen möchten, der Astra Control Provisioner oder Astra Trident für Storage-Services verwendet, müssen Sie sicherstellen, dass die installierte Version von Astra Trident 23.10 oder aktuell ist.

Schritte

1. Bestimmen Sie die Astra Trident-Version, die Sie ausführen:

```
kubectl get tridentversions -n trident
```

   Wenn Astra Trident installiert ist, wird die Ausgabe wie folgt ausgegeben:

```
NAME      VERSION
trident   24.02.0
```

   Wenn Astra Trident nicht installiert ist, wird die Ausgabe wie folgt angezeigt:

```
error: the server doesn't have a resource type "tridentversions"
```

2. Führen Sie einen der folgenden Schritte aus:
   - Wenn Sie Astra Trident 23.01 oder eine frühere Version verwenden, verwenden Sie diese "Anweisungen" Das Upgrade auf eine neuere Version von Astra Trident erfolgt vor dem Upgrade auf Astra Control Provisioner. Das können Sie "Führen Sie ein direktes Upgrade durch" Astra Control Provisioner 24.02, wenn Ihr Astra Trident in einem Fenster mit vier Versionen von Version 24.02 angezeigt wird. Sie können beispielsweise direkt von Astra Trident 23.04 auf Astra Control Provisioner 24.02 aktualisieren.
   - Wenn Sie Astra Trident 23.10 oder höher verwenden, stellen Sie sicher, dass es für Astra Control Provisioner verwendet wurde "Aktiviert". Astra Control Provisioner kann nicht mit Versionen von Astra Control Center vor 23.10 verwendet werden. "Upgrade für die Astra Control Provisioner" Da es nun dieselbe Version wie das Astra Control Center hat, stellen Sie ein Upgrade auf die neuesten Funktionen bereit.
3. Stellen Sie sicher, dass die Pods ausgeführt werden:

```
kubectl get pods -n trident
```

4. Prüfen Sie, ob die Storage-Klassen die unterstützten Astra Trident Treiber verwenden. Der bereitstellungsname sollte lauten `csi.trident.netapp.io`. Im folgenden Beispiel finden Sie weitere Informationen:

```
kubectl get sc
```

   Beispielantwort:

```
NAME                     PROVISIONER                    RECLAIMPOLICY
VOLUMEBINDINGMODE    ALLOWVOLUMEEXPANSION    AGE
ontap-gold (default)    csi.trident.netapp.io          Delete
Immediate            true                    5d23h
```

Erstellen Sie eine kubeconfig-Datei

Sie können dem Astra Control Service ein Cluster mithilfe einer kubeconfig-Datei hinzufügen. Je nach dem Typ des Clusters, den Sie hinzufügen möchten, müssen Sie möglicherweise manuell eine kubeconfig-Datei für Ihr Cluster mithilfe bestimmter Schritte erstellen.

- Erstellen Sie eine kubeconfig-Datei für Amazon EKS-Cluster
- Erstellen Sie eine kubeconfig-Datei für Red hat OpenShift Service on AWS (ROSA) Cluster
- Erstellen Sie eine kubeconfig-Datei für andere Cluster-Typen

Erstellen Sie eine kubeconfig-Datei für Amazon EKS-Cluster

Befolgen Sie diese Anweisungen, um eine kubeconfig-Datei und ein permanentes Token-Geheimnis für Amazon EKS-Cluster zu erstellen. Für Cluster, die in EKS gehostet werden, ist ein permanenter Tokenschlüssel erforderlich.

Schritte

1. Befolgen Sie die Anweisungen in der Amazon-Dokumentation, um eine kubeconfig-Datei zu erstellen:

   "Erstellen oder Aktualisieren einer kubeconfig-Datei für einen Amazon EKS-Cluster"

2. Erstellen Sie ein Service-Konto wie folgt:

   a. Erstellen Sie eine Dienstkontendatei mit dem Namen `astracontrol-service-account.yaml`.

      Passen Sie den Namen des Servicekontos nach Bedarf an. Der Namespace `kube-system` Ist für diese Schritte erforderlich. Wenn Sie hier den Namen des Servicekontos ändern, sollten Sie die gleichen Änderungen in den folgenden Schritten anwenden.

```
<strong>astracontrol-service-account.yaml</strong>
```

   +

```
apiVersion: v1
kind: ServiceAccount
metadata:
  name: astra-admin-account
  namespace: kube-system
```

3. Wenden Sie das Servicekonto an:

```
kubectl apply -f astracontrol-service-account.yaml
```

4. Erstellen Sie ein `ClusterRoleBinding` Datei aufgerufen `astracontrol-clusterrolebinding.yaml`.

```
<strong>astracontrol-clusterrolebinding.yaml</strong>
```

```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
  name: astra-admin-binding
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: ClusterRole
  name: cluster-admin
subjects:
- kind: ServiceAccount
  name: astra-admin-account
  namespace: kube-system
```

5. Wenden Sie die Bindung der Cluster-Rolle an:

```
kubectl apply -f astracontrol-clusterrolebinding.yaml
```

6. Erstellen Sie eine Geheimdatei für das Dienstkonto-Token mit dem Namen `astracontrol-secret.yaml`.

```
<strong>astracontrol-secret.yaml</strong>
```

```
apiVersion: v1
kind: Secret
metadata:
  annotations:
    kubernetes.io/service-account.name: astra-admin-account
  name: astra-admin-account
  namespace: kube-system
type: kubernetes.io/service-account-token
```

7. Wenden Sie den Token-Schlüssel an:

```
kubectl apply -f astracontrol-secret.yaml
```

8. Rufen Sie den Token-Schlüssel ab:

```
kubectl get secret astra-admin-account -n kube-system -o
jsonpath='{.data.token}' | base64 -d
```

9. Ersetzen Sie den `user` Abschnitt der AWS EKS kubeconfig-Datei mit dem Token, wie im folgenden Beispiel gezeigt:

```
user:
    token: k8s-aws-
v1.aHR0cHM6Ly9zdHMudXMtd2VzdC0yLmFtYXpvbmF3cy5jb20vP0FjdGlvbj1HZXRDYWxsZ
XJJZGVudGl0eSZWZXJzaW9uPTIwMTEtMDYtMTUmWC1BbXotQWxnb3JpdGhtPUFXUzQtSE1BQ
y1TSEEyNTYmWC1BbXotQ3JlZGVudGlhbD1BS0lBM1JEWDdKU0haWU9LSEQ2SyUyRjIwMjMwN
DAzJTJGdXMtd2VzdC0yJTJGc3RzJTJGYXdzNF9yZXF1ZXN0JlgtQW16LURhdGU9MjAyMzA0M
DNUMjA0MzQwWiZYLUFtei1FeHBpcmVzPTYwJlgtQW16LVNpZ25lZEhlYWRlcnM9aG9zdCUzQ
ngtazhzLWF3cy1pZCZYLUFtei1TaWduYXR1cmU9YjU4ZWM0NzdiM2NkZGYxNGRhNzU4MGI2Z
WQ2zY2NzI2YWIwM2UyNThjMjRhNTJjNmVhNjc4MTRlNjJkOTg2Mg
```

Erstellen Sie eine kubeconfig-Datei für Red hat OpenShift Service on AWS (ROSA) Cluster

Befolgen Sie diese Anweisungen, um eine kubeconfig-Datei für Red hat OpenShift Service on AWS (ROSA)-Cluster zu erstellen.

Schritte

1. Melden Sie sich beim ROSA-Cluster an.
2. Service-Konto erstellen:

```
oc create sa astracontrol-service-account
```

3. Cluster-Rolle hinzufügen:

```
oc adm policy add-cluster-role-to-user cluster-admin -z astracontrol-
service-account
```

4. Erstellen Sie mithilfe des folgenden Beispiels eine geheime Konfigurationsdatei für das Dienstkonto:

```
<strong>secret-astra-sa.yaml</strong>
```

```
apiVersion: v1
kind: Secret
metadata:
  name: secret-astracontrol-service-account
  annotations:
    kubernetes.io/service-account.name: "astracontrol-service-account"
type: kubernetes.io/service-account-token
```

5. Erstellen Sie das Geheimnis:

```
oc create -f secret-astra-sa.yaml
```

6. Bearbeiten Sie das von Ihnen erstellte Dienstkonto, und fügen Sie dem den geheimen Namen des Astra Control-Dienstkontos hinzu `secrets` Abschnitt:

```
oc edit sa astracontrol-service-account
```

```
apiVersion: v1
imagePullSecrets:
- name: astracontrol-service-account-dockercfg-dvfcd
kind: ServiceAccount
metadata:
  creationTimestamp: "2023-08-04T04:18:30Z"
  name: astracontrol-service-account
  namespace: default
  resourceVersion: "169770"
  uid: 965fa151-923f-4fbd-9289-30cad15998ac
secrets:
- name: astracontrol-service-account-dockercfg-dvfcd
- name: secret-astracontrol-service-account ####ADD THIS ONLY####
```

7. Listen Sie die Geheimnisse des Dienstkontos auf, ersetzen Sie `<CONTEXT>` Mit dem richtigen Kontext für Ihre Installation:

```
kubectl get serviceaccount astracontrol-service-account --context
<CONTEXT> --namespace default -o json
```

Das Ende der Ausgabe sollte wie folgt aussehen:

```
"secrets": [
{ "name": "astracontrol-service-account-dockercfg-dvfcd"},
{ "name": "secret-astracontrol-service-account"}
]
```

Die Indizes für jedes Element im `secrets` Array beginnt mit 0. Im obigen Beispiel der Index für `astracontrol-service-account-dockercfg-dvfcd` Wäre 0 und der Index für `secret-astracontrol-service-account` Sind es 1. Notieren Sie sich in Ihrer Ausgabe die Indexnummer für den Geheimschlüssel des Dienstkontos. Diese Indexnummer benötigen Sie im nächsten Schritt.

8. Erzeugen Sie den kubeconfig wie folgt:
   a. Erstellen Sie ein `create-kubeconfig.sh` Datei: Austausch `TOKEN_INDEX` Am Anfang des folgenden Skripts mit dem korrekten Wert.

```
<strong>create-kubeconfig.sh</strong>
```

```
# Update these to match your environment.
# Replace TOKEN_INDEX with the correct value
# from the output in the previous step. If you
# didn't change anything else above, don't change
# anything else here.

SERVICE_ACCOUNT_NAME=astracontrol-service-account
NAMESPACE=default
NEW_CONTEXT=astracontrol
KUBECONFIG_FILE='kubeconfig-sa'

CONTEXT=$(kubectl config current-context)

SECRET_NAME=$(kubectl get serviceaccount ${SERVICE_ACCOUNT_NAME} \
  --context ${CONTEXT} \
  --namespace ${NAMESPACE} \
  -o jsonpath='{.secrets[TOKEN_INDEX].name}')
TOKEN_DATA=$(kubectl get secret ${SECRET_NAME} \
  --context ${CONTEXT} \
  --namespace ${NAMESPACE} \
  -o jsonpath='{.data.token}')

TOKEN=$(echo ${TOKEN_DATA} | base64 -d)

# Create dedicated kubeconfig
# Create a full copy
kubectl config view --raw > ${KUBECONFIG_FILE}.full.tmp
```

```
# Switch working context to correct context
kubectl --kubeconfig ${KUBECONFIG_FILE}.full.tmp config use-context
${CONTEXT}

# Minify
kubectl --kubeconfig ${KUBECONFIG_FILE}.full.tmp \
  config view --flatten --minify > ${KUBECONFIG_FILE}.tmp

# Rename context
kubectl config --kubeconfig ${KUBECONFIG_FILE}.tmp \
  rename-context ${CONTEXT} ${NEW_CONTEXT}

# Create token user
kubectl config --kubeconfig ${KUBECONFIG_FILE}.tmp \
  set-credentials ${CONTEXT}-${NAMESPACE}-token-user \
  --token ${TOKEN}

# Set context to use token user
kubectl config --kubeconfig ${KUBECONFIG_FILE}.tmp \
  set-context ${NEW_CONTEXT} --user ${CONTEXT}-${NAMESPACE}-token
-user

# Set context to correct namespace
kubectl config --kubeconfig ${KUBECONFIG_FILE}.tmp \
  set-context ${NEW_CONTEXT} --namespace ${NAMESPACE}

# Flatten/minify kubeconfig
kubectl config --kubeconfig ${KUBECONFIG_FILE}.tmp \
  view --flatten --minify > ${KUBECONFIG_FILE}

# Remove tmp
rm ${KUBECONFIG_FILE}.full.tmp
rm ${KUBECONFIG_FILE}.tmp
```

b. Geben Sie die Befehle an, um sie auf Ihren Kubernetes-Cluster anzuwenden.

```
source create-kubeconfig.sh
```

9. (Optional) Umbenennen Sie die kubeconfig auf einen aussagekräftigen Namen für Ihr Cluster.

```
mv kubeconfig-sa YOUR_CLUSTER_NAME_kubeconfig
```

Erstellen Sie eine kubeconfig-Datei für andere Cluster-Typen

Befolgen Sie diese Anweisungen, um eine begrenzte oder erweiterte Kubeconfig-Datei für Rancher-, Upstream-Kubernetes- und Red hat OpenShift-Cluster zu erstellen.

Für Cluster, die mit kubeconfig gemanagt werden, können Sie optional eine Administratorrolle mit eingeschränkter Berechtigung oder erweiterten Berechtigungen für Astra Control Service erstellen.

Dieses Verfahren hilft Ihnen, ein separates kubeconfig zu erstellen, wenn eines der folgenden Szenarien auf Ihre Umgebung zutrifft:

- Sie möchten die Astra Control-Berechtigungen auf die Cluster beschränken, die sie verwaltet
- Sie verwenden mehrere Kontexte und können nicht den Standard Astra Control kubeconfig verwenden, der während der Installation konfiguriert wurde, oder eine eingeschränkte Rolle mit einem einzelnen Kontext funktioniert nicht in Ihrer Umgebung

Bevor Sie beginnen

Stellen Sie sicher, dass Sie für den Cluster, den Sie verwalten möchten, vor dem Ausführen der Schritte des Verfahrens Folgendes haben:

- A "Unterstützte Version" Von kubectl ist installiert.
- Kubectl Zugriff auf den Cluster, den Sie mit Astra Control Service hinzufügen und managen möchten

> Für dieses Verfahren benötigen Sie keinen kubectl-Zugriff auf den Cluster, auf dem Astra Control Service ausgeführt wird.

- Ein aktiver kubeconfig für den Cluster, den Sie mit Clusteradministratorrechten für den aktiven Kontext verwalten möchten

Schritte

1. Service-Konto erstellen:
   a. Erstellen Sie eine Dienstkontendatei mit dem Namen `astracontrol-service-account.yaml`.

```
<strong>astracontrol-service-account.yaml</strong>
```

```
apiVersion: v1
kind: ServiceAccount
metadata:
  name: astracontrol-service-account
  namespace: default
```

   b. Wenden Sie das Servicekonto an:

```
kubectl apply -f astracontrol-service-account.yaml
```

2. Erstellen Sie eine der folgenden Clusterrollen mit ausreichenden Berechtigungen für ein Cluster, das von Astra Control gemanagt werden kann:

Eingeschränkte Cluster-Rolle

Diese Rolle enthält die Mindestberechtigungen, die für das Management eines Clusters durch Astra Control erforderlich sind:

a. Erstellen Sie ein `ClusterRole` Datei mit dem Namen, z. B. `astra-admin-account.yaml`.

```
<strong>astra-admin-account.yaml</strong>
```

```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  name: astra-admin-account
rules:

# Get, List, Create, and Update all resources
# Necessary to backup and restore all resources in an app
- apiGroups:
  - '*'
  resources:
  - '*'
  verbs:
  - get
  - list
  - create
  - patch

# Delete Resources
# Necessary for in-place restore and AppMirror failover
- apiGroups:
  - ""
  - apps
  - autoscaling
  - batch
  - crd.projectcalico.org
  - extensions
  - networking.k8s.io
  - policy
  - rbac.authorization.k8s.io
  - snapshot.storage.k8s.io
  - trident.netapp.io
  resources:
  - configmaps
  - cronjobs
  - daemonsets
```

```
    - deployments
    - horizontalpodautoscalers
    - ingresses
    - jobs
    - namespaces
    - networkpolicies
    - persistentvolumeclaims
    - poddisruptionbudgets
    - pods
    - podtemplates
    - replicasets
    - replicationcontrollers
    - replicationcontrollers/scale
    - rolebindings
    - roles
    - secrets
    - serviceaccounts
    - services
    - statefulsets
    - tridentmirrorrelationships
    - tridentsnapshotinfos
    - volumesnapshots
    - volumesnapshotcontents
    verbs:
    - delete

  # Watch resources
  # Necessary to monitor progress
  - apiGroups:
    - ""
    resources:
    - pods
    - replicationcontrollers
    - replicationcontrollers/scale
    verbs:
    - watch

  # Update resources
  - apiGroups:
    - ""
    - build.openshift.io
    - image.openshift.io
    resources:
    - builds/details
    - replicationcontrollers
    - replicationcontrollers/scale
```

```
  - imagestreams/layers
  - imagestreamtags
  - imagetags
  verbs:
  - update
```

b. (Nur für OpenShift-Cluster) Anhängen Sie am Ende des an `astra-admin-account.yaml` Datei:

```
# OpenShift security
- apiGroups:
  - security.openshift.io
  resources:
  - securitycontextconstraints
  verbs:
  - use
  - update
```

c. Wenden Sie die Cluster-Rolle an:

```
kubectl apply -f astra-admin-account.yaml
```

Erweiterte Cluster-Rolle

Diese Rolle enthält erweiterte Berechtigungen für ein Cluster, das von Astra Control gemanagt werden kann. Sie können diese Rolle verwenden, wenn Sie mehrere Kontexte verwenden und nicht den während der Installation konfigurierten Astra Control kubeconfig verwenden können oder eine eingeschränkte Rolle mit einem einzelnen Kontext in Ihrer Umgebung nicht funktioniert:

Im Folgenden `ClusterRole` Schritte sind ein allgemeines Kubernetes-Beispiel. Anweisungen zu Ihrer spezifischen Umgebung finden Sie in der Dokumentation zur Kubernetes-Distribution.

a. Erstellen Sie ein `ClusterRole` Datei mit dem Namen, z. B. `astra-admin-account.yaml`.

```
<strong>astra-admin-account.yaml</strong>
```

```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  name: astra-admin-account
rules:
- apiGroups:
  - '*'
  resources:
  - '*'
  verbs:
  - '*'
- nonResourceURLs:
  - '*'
  verbs:
  - '*'
```

b. Wenden Sie die Cluster-Rolle an:

```
kubectl apply -f astra-admin-account.yaml
```

3. Erstellen Sie die Cluster-Rolle, die für die Cluster-Rolle an das Service-Konto gebunden ist:

a. Erstellen Sie ein `ClusterRoleBinding` Datei aufgerufen `astracontrol-clusterrolebinding.yaml`.

```
<strong>astracontrol-clusterrolebinding.yaml</strong>
```

```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
  name: astracontrol-admin
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: ClusterRole
  name: astra-admin-account
subjects:
- kind: ServiceAccount
  name: astracontrol-service-account
  namespace: default
```

b. Wenden Sie die Bindung der Cluster-Rolle an:

```
kubectl apply -f astracontrol-clusterrolebinding.yaml
```

4. Erstellen und Anwenden des Token-Geheimnisses:
   a. Erstellen Sie eine Geheimdatei mit dem Namen Token `secret-astracontrol-service-account.yaml`.

```
<strong>secret-astracontrol-service-account.yaml</strong>
```

```
apiVersion: v1
kind: Secret
metadata:
  name: secret-astracontrol-service-account
  namespace: default
  annotations:
    kubernetes.io/service-account.name: "astracontrol-service-
account"
type: kubernetes.io/service-account-token
```

   b. Wenden Sie den Token-Schlüssel an:

```
kubectl apply -f secret-astracontrol-service-account.yaml
```

5. Fügen Sie dem Dienstkonto den Token-Schlüssel hinzu, indem Sie den Namen dem hinzufügen `secrets` Array (die letzte Zeile im folgenden Beispiel):

```
kubectl edit sa astracontrol-service-account
```

```
apiVersion: v1
imagePullSecrets:
- name: astracontrol-service-account-dockercfg-48xhx
kind: ServiceAccount
metadata:
  annotations:
    kubectl.kubernetes.io/last-applied-configuration: |

{"apiVersion":"v1","kind":"ServiceAccount","metadata":{"annotations":{},
"name":"astracontrol-service-account","namespace":"default"}}
  creationTimestamp: "2023-06-14T15:25:45Z"
  name: astracontrol-service-account
  namespace: default
  resourceVersion: "2767069"
  uid: 2ce068c4-810e-4a96-ada3-49cbf9ec3f89
secrets:
- name: astracontrol-service-account-dockercfg-48xhx
<strong>- name: secret-astracontrol-service-account</strong>
```

6. Listen Sie die Geheimnisse des Dienstkontos auf, ersetzen Sie `<context>` Mit dem richtigen Kontext für Ihre Installation:

```
kubectl get serviceaccount astracontrol-service-account --context
<context> --namespace default -o json
```

Das Ende der Ausgabe sollte wie folgt aussehen:

```
"secrets": [
{ "name": "astracontrol-service-account-dockercfg-48xhx"},
{ "name": "secret-astracontrol-service-account"}
]
```

Die Indizes für jedes Element im `secrets` Array beginnt mit 0. Im obigen Beispiel der Index für `astracontrol-service-account-dockercfg-48xhx` Wäre 0 und der Index für `secret-astracontrol-service-account` Sind es 1. Notieren Sie sich in Ihrer Ausgabe die Indexnummer für den Geheimschlüssel des Dienstkontos. Im nächsten Schritt benötigen Sie diese Indexnummer.

7. Erzeugen Sie den kubeconfig wie folgt:
   a. Erstellen Sie ein `create-kubeconfig.sh` Datei:
   b. Austausch `TOKEN_INDEX` Am Anfang des folgenden Skripts mit dem korrekten Wert.

```
<strong>create-kubeconfig.sh</strong>
```

```
# Update these to match your environment.
# Replace TOKEN_INDEX with the correct value
# from the output in the previous step. If you
# didn't change anything else above, don't change
# anything else here.

SERVICE_ACCOUNT_NAME=astracontrol-service-account
NAMESPACE=default
NEW_CONTEXT=astracontrol
KUBECONFIG_FILE='kubeconfig-sa'

CONTEXT=$(kubectl config current-context)

SECRET_NAME=$(kubectl get serviceaccount ${SERVICE_ACCOUNT_NAME} \
  --context ${CONTEXT} \
  --namespace ${NAMESPACE} \
  -o jsonpath='{.secrets[TOKEN_INDEX].name}')
TOKEN_DATA=$(kubectl get secret ${SECRET_NAME} \
  --context ${CONTEXT} \
  --namespace ${NAMESPACE} \
  -o jsonpath='{.data.token}')

TOKEN=$(echo ${TOKEN_DATA} | base64 -d)

# Create dedicated kubeconfig
# Create a full copy
kubectl config view --raw > ${KUBECONFIG_FILE}.full.tmp

# Switch working context to correct context
kubectl --kubeconfig ${KUBECONFIG_FILE}.full.tmp config use-context
${CONTEXT}

# Minify
kubectl --kubeconfig ${KUBECONFIG_FILE}.full.tmp \
  config view --flatten --minify > ${KUBECONFIG_FILE}.tmp

# Rename context
kubectl config --kubeconfig ${KUBECONFIG_FILE}.tmp \
  rename-context ${CONTEXT} ${NEW_CONTEXT}

# Create token user
kubectl config --kubeconfig ${KUBECONFIG_FILE}.tmp \
  set-credentials ${CONTEXT}-${NAMESPACE}-token-user \
  --token ${TOKEN}

# Set context to use token user
```

```
kubectl config --kubeconfig ${KUBECONFIG_FILE}.tmp \
  set-context ${NEW_CONTEXT} --user ${CONTEXT}-${NAMESPACE}-token-
user

# Set context to correct namespace
kubectl config --kubeconfig ${KUBECONFIG_FILE}.tmp \
  set-context ${NEW_CONTEXT} --namespace ${NAMESPACE}

# Flatten/minify kubeconfig
kubectl config --kubeconfig ${KUBECONFIG_FILE}.tmp \
  view --flatten --minify > ${KUBECONFIG_FILE}

# Remove tmp
rm ${KUBECONFIG_FILE}.full.tmp
rm ${KUBECONFIG_FILE}.tmp
```

c. Geben Sie die Befehle an, um sie auf Ihren Kubernetes-Cluster anzuwenden.

```
source create-kubeconfig.sh
```

8. (Optional) Umbenennen Sie die kubeconfig auf einen aussagekräftigen Namen für Ihr Cluster.

```
mv kubeconfig-sa YOUR_CLUSTER_NAME_kubeconfig
```

Was kommt als Nächstes?

Nachdem Sie sich angemeldet und Astra Control ein Cluster hinzugefügt haben, können Sie jetzt die Funktionen für das Applikations-Datenmanagement von Astra Control nutzen.

- "Starten Sie das Anwendungsmanagement"
- "Schützen von Applikationen"
- "Applikationen klonen"
- "Abrechnung einrichten"
- "Benutzer einladen und managen"
- "Management der Anmeldedaten von Cloud-Providern"
- "Benachrichtigungen verwalten"
- "Selbstverwaltete Instanz von Astra Control implementieren"

Videos des Astra Control Service

In NetApp TV finden Sie die neuesten Videoinhalte rund um den Astra Control Service. NetApp TV enthält Videos, in denen bestimmte Funktionen von Astra Control Service gezeigt oder dem Erledigung bestimmter allgemeiner Aufgaben gezeigt werden.

"Videos des Astra Control Service"