Los geht's

Astra Automation 22.04

NetApp
December 04, 2023

Inhalt

Los geht’s

Bevor Sie beginnen

Sie können sich schnell auf den Einstieg in die Astra Control REST API vorbereiten, indem Sie die unten aufgeführten Schritte überprüfen.

Astra-Konto besitzen Anmeldedaten

Sie benötigen Astra-Anmeldeinformationen, um sich bei der Astra Web-Benutzeroberfläche anzumelden und ein API-Token zu generieren. Mit Astra Control Center verwalten Sie diese Anmeldedaten lokal. Mit dem Astra Control Service können Sie über den **Auth0**-Dienst auf die Anmeldeinformationen zugreifen.

Lernen Sie die grundlegenden Kubernetes-Konzepte kennen

Sie sollten mit verschiedenen grundlegenden Kubernetes-Konzepten vertraut sein. Siehe "Grundlegende Kubernetes-Konzepte" Finden Sie weitere Informationen.

ÜBERPRÜFUNG DER REST-Konzepte und der Implementierung

Prüfen Sie die Daten "Core-REST-Implementierung" Für Informationen über REST-Konzepte und die Details zur Entwicklung der Astra Control REST API.

Weitere Informationen

Beachten Sie die zusätzlichen Informationsressourcen, wie in vorgeschlagen "Weitere Ressourcen".

Holen Sie sich ein API-Token

Sie müssen ein Astra API Token erhalten, um die Astra Control REST API zu verwenden.

Einführung

Ein API-Token identifiziert den Anrufer an Astra und muss bei jedem REST-API-Aufruf enthalten sein.

- Sie können über die Astra Web-Benutzeroberfläche ein API-Token generieren.
- Die mit dem Token getragene Benutzeridentität wird vom Benutzer bestimmt, der das Token erstellt.
- Das Token muss in das enthalten sein `Authorization` HTTP-Anfragekopf.
- Ein Token läuft nie ab, nachdem es erstellt wurde.
- Sie können ein Token über die Astra Web-Benutzeroberfläche widerrufen.

Verwandte Informationen

- "Ein API-Token widerrufen"

Erstellen Sie ein Astra API-Token

In den folgenden Schritten wird beschrieben, wie ein Astra API Token erstellt wird.

Bevor Sie beginnen

Sie benötigen die Zugangsdaten für ein Astra-Konto.

Über diese Aufgabe

Diese Aufgabe erzeugt ein API-Token in der Astra-Webschnittstelle. Sie sollten auch die Account-ID abrufen, die auch bei API-Aufrufen benötigt wird.

Schritte

1. Melden Sie sich mit Ihren Anmeldedaten im Astra an.

   Rufen Sie die folgende Website für den Astra Control Service auf: "https://astra.netapp.io"

2. Klicken Sie auf das Figurensymbol oben rechts auf der Seite und wählen Sie **API Access**.
3. Klicken Sie auf **API-Token generieren** auf der Seite und klicken Sie im Popup-Fenster auf **API-Token generieren**.
4. Klicken Sie auf das Symbol, um die Token-Zeichenfolge in die Zwischenablage zu kopieren und im Editor zu speichern.
5. Kopieren Sie die Konto-id, die auf derselben Seite verfügbar ist, und speichern Sie sie.

Nachdem Sie fertig sind

Wenn Sie über Curl oder eine Programmiersprache auf die Astra Control REST API zugreifen, müssen Sie das API-Träger-Token in das HTTP einbeziehen `Authorization` Kopfzeile der Anfrage.

Hallo Welt

Sie können einen einfachen Curl Befehl an Ihrer Workstation CLI ausgeben, um mit der Astra Control REST API zu beginnen und seine Verfügbarkeit zu bestätigen.

Bevor Sie beginnen

Das Dienstprogramm Curl muss auf Ihrer lokalen Workstation verfügbar sein. Sie müssen außerdem über ein API-Token und die zugehörige Account-ID verfügen. Siehe "Holen Sie sich ein API-Token" Finden Sie weitere Informationen.

Beispiel für die Wellung

Der folgende Curl Befehl ruft eine Liste der Astra-Benutzer ab. Geben Sie die entsprechenden <ACCOUNT_ID> und <API_TOKEN> wie angegeben an.

```
curl --location --request GET
'https://astra.netapp.io/accounts/<ACCOUNT_ID>/core/v1/users' --header
'Content-Type: application/json' --header 'Authorization: Bearer
<API_TOKEN>'
```

Beispiel für eine JSON-Ausgabe

```
{
  "items": [
    [
    "David",
    "Peterson",
    "844ec6234-11e0-49ea-8434-a992a6270ec1"
    ],
    [
      "Scott",
      "Morris",
      "2a3e227c-fda7-4145-a86c-ed9aa0183a6c"
    ]
  ],
  "metadata": {}
}
```

Die Nutzung der Workflows wird vorbereitet

Vor der Live-Implementierung sollten Sie sich mit der Organisation und dem Format der Astra-Workflows vertraut machen.

Einführung

Ein *Workflow* ist eine Sequenz aus einem oder mehreren Schritten, die zum Erreichen einer bestimmten administrativen Aufgabe oder eines bestimmten Ziels erforderlich sind. Jeder Schritt in einem Astra Control Workflow ist einer der folgenden:

- REST-API-Aufruf (mit Details wie Curl- und JSON-Beispiele)
- Aufruf eines weiteren Astra-Workflows
- Sonstige verwandte Aufgaben (z. B. die Entscheidung für eine erforderliche Designentscheidung)

Workflows umfassen die wichtigsten Schritte und Parameter, die zur Durchführung jeder Aufgabe erforderlich sind. Sie bieten als Ausgangspunkt für die Anpassung Ihrer Automatisierungsumgebung.

Allgemeine Eingabeparameter

Die unten beschriebenen Eingabeparameter sind für alle Curl-Proben, die zur Veranschaulichung eines REST-API-Aufrufs verwendet werden, üblich.

> Da diese Eingabeparameter universell erforderlich sind, werden sie in den einzelnen Arbeitsabläufen nicht weiter beschrieben. Wenn für ein bestimmtes Curl-Beispiel zusätzliche Eingabeparameter verwendet werden, werden sie im Abschnitt **zusätzliche Eingabeparameter** beschrieben.

Pfadparameter

Der bei jedem REST-API-Aufruf verwendete Endpunkt-Pfad umfasst die folgenden Parameter. Siehe auch "URL-Format" Finden Sie weitere Informationen.

Konto-ID

Dies ist der UUIDv4-Wert, der das Astra-Konto identifiziert, auf dem der API-Vorgang ausgeführt wird. Siehe "Holen Sie sich ein API-Token" Weitere Informationen zur Suche nach Ihrer Konto-ID.

Anfragekopfzeilen

Je nach REST-API-Aufruf müssen Sie möglicherweise mehrere Anforderungsheader einbeziehen.

Autorisierung

Für alle API-Aufrufe in den Workflows wird ein API-Token zur Identifizierung des Benutzers benötigt. Sie müssen das Token in das aufnehmen `Authorization` Kopfzeile der Anfrage. Siehe "Holen Sie sich ein API-Token" Weitere Informationen zum Generieren eines API-Tokens finden Sie unter.

Content-Typ

Mit dem HTTP-POST und PUT-Anfragen, bei denen JSON im Anforderungstext enthalten ist, sollten Sie den Medientyp basierend auf der Astra-Ressource deklarieren. Beispielsweise können Sie die Kopfzeile einschließen `Content-Type: application/astra-appSnap+json` Beim Erstellen eines Snapshots für eine verwaltete Anwendung.

Akzeptieren

Sie können den spezifischen Medientyp des Inhalts, den Sie in der Antwort erwarten, basierend auf der Astra-Ressource erklären. Beispielsweise können Sie die Kopfzeile einschließen `Accept: application/astra-appBackup+json` Wenn Sie die Backups für eine gemanagte Applikation auflisten. Zur Vereinfachung akzeptieren die Wellproben in den Workflows jedoch alle Medientypen.

Darstellung von Token und Identifikatoren

Das API-Token und andere ID-Werte, die mit den Curl-Beispielen verwendet werden, sind undurchsichtig und haben keine erkennbare Bedeutung. Um die Lesbarkeit der Proben zu verbessern, werden die tatsächlichen Token- und ID-Werte nicht verwendet. Stattdessen werden kleinere reservierte Schlüsselwörter verwendet, die mehrere Vorteile haben:

- Die Curl- und JSON-Proben sind klarer und leichter zu verstehen.
- Da alle Schlüsselwörter dasselbe Format mit Klammern und Großbuchstaben verwenden, können Sie schnell den Ort und den Inhalt identifizieren, der eingefügt oder extrahiert werden soll.
- Kein Wert geht verloren, da die ursprünglichen Parameter nicht kopiert und bei einer tatsächlichen Implementierung verwendet werden können.

Hier sind einige der in den Curl-Beispielen verwendeten allgemein reservierten Schlüsselwörter. Diese Liste ist nicht vollständig und weitere Schlüsselwörter werden bei Bedarf verwendet. Ihre Bedeutung sollte auf der Grundlage des Kontexts offensichtlich sein.

Stichwort	Typ	Beschreibung
<ACCOUNT_ID>	Pfad	Der UUIDv4-Wert identifiziert das Konto, auf dem der API-Vorgang ausgeführt wird.
<API_TOKEN>	Kopfzeile	Das Inhaberzeichen, das den Anrufer identifiziert und autorisiert.

Stichwort	Typ	Beschreibung
<MANAGED_APP_ID>	Pfad	Der UUIDv4-Wert, der die verwaltete Anwendung für den API-Aufruf identifiziert.

Workflow-Kategorien

Je nach Ihrem Implementierungsmodell stehen Ihnen zwei weit gefassten Kategorien von Astra-Workflows zur Verfügung. Wenn Sie Astra Control Center nutzen, sollten Sie mit den Infrastruktur-Workflows beginnen und anschließend mit den Management-Workflows fortfahren. Mit dem Astra Control Service können Sie in der Regel direkt zu den Management-Workflows wechseln.

> Die Curl-Beispiele in den Workflows verwenden die URL für den Astra Control Service. Sie müssen die URL ändern, wenn Sie das On-Premise Astra Control Center entsprechend Ihrer Umgebung verwenden.

Infrastruktur-Workflows

Diese Workflows befassen sich mit der Astra-Infrastruktur, einschließlich Referenzen, Buckets und Storage-Back-Ends. Sie werden mit dem Astra Control Center benötigt, können aber in den meisten Fällen auch mit dem Astra Control Service verwendet werden. Die Workflows konzentrieren sich auf die Aufgaben, die für die Einrichtung und Wartung eines von Astra gemanagten Clusters erforderlich sind.

Management-Workflows

Diese Workflows können nach einem verwalteten Cluster verwendet werden. Die Workflows konzentrieren sich auf die Applikationssicherung und unterstützen Abläufe wie das Backup, die Wiederherstellung und das Klonen einer gemanagten Applikation.

Grundlegende Kubernetes-Konzepte

Es gibt verschiedene Kubernetes-Konzepte, die für die Verwendung der Astra REST API relevant sind.

Objekte

Die in einer Kubernetes-Umgebung gepflegten Objekte sind persistente Einheiten, die die Konfiguration des Clusters repräsentieren. Diese Objekte beschreiben zusammen den Status des Systems einschließlich des Cluster-Workloads.

Namespaces

Namespaces bieten eine Technik zum Isolation von Ressourcen in einem einzigen Cluster. Diese Organisationsstruktur ist nützlich, wenn die Arten von Arbeit, Nutzer und Ressourcen aufgeteilt werden. Objekte mit einem Umfang „*Namespace“* müssen innerhalb des Namespace eindeutig sein, während Objekte mit einem „*Cluster Scope*“ im gesamten Cluster eindeutig sein müssen.

Etiketten

Labels können den Kubernetes-Objekten zugeordnet werden. Sie beschreiben Attribute mit Schlüsselwert-Paaren und können eine beliebige Organisation auf dem Cluster durchsetzen. Diese können sich für ein Unternehmen nützlich sein, liegen aber nicht in der zentralen Handhabung von Kubernetes.