Identität und Zugriff

Astra Automation

NetApp
December 01, 2023

Inhalt

Identität und Zugriff

Listen Sie die Benutzer auf

Sie können die Benutzer auflisten, die für ein bestimmtes Astra-Konto definiert sind.

1. Listen Sie die Benutzer auf

Führen Sie den folgenden REST-API-Aufruf aus.

HTTP-Methode	Pfad
GET	/Accounts/{Account_id}/Core/v1/users

Zusätzliche Eingabeparameter

Zusätzlich zu den Parametern, die bei allen REST-API-Aufrufen üblich sind, werden die folgenden Parameter auch in den Curl-Beispielen für diesen Schritt verwendet.

Parameter	Typ	Erforderlich	Beschreibung
Einschließlich	Abfrage	Nein	Wählen Sie optional die Werte aus, die in der Antwort zurückgegeben werden sollen.

Curl-Beispiel: Alle Daten für alle Benutzer zurückgeben

```
curl --location -i --request GET
'https://astra.netapp.io/accounts/<ACCOUNT_ID>/core/v1/users' --header
'Accept: */*' --header 'Authorization: Bearer <API_TOKEN>'
```

Curl Beispiel: Gibt den vor-, Nachnamen und die id für alle Benutzer zurück

```
curl --location -i --request GET
'https://astra.netapp.io/accounts/<ACCOUNT_ID>/core/v1/users?include=first
Name,lastName,id' --header 'Accept: */*' --header 'Authorization: Bearer
<API_TOKEN>'
```

Beispiel für eine JSON-Ausgabe

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

Erstellen Sie einen Benutzer

Sie können einen Benutzer mit spezifischen Anmeldedaten und einer vordefinierten Rolle erstellen. Sie können optional auch den Zugriff des Benutzers auf bestimmte Namespaces beschränken.

1. Wählen Sie einen Benutzernamen

Führen Sie den Workflow aus "Benutzer auflisten" Und wählen Sie einen verfügbaren Namen aus, der derzeit nicht verwendet wird.

2. Erstellen Sie den Benutzer

Führen Sie den folgenden REST-API-Aufruf aus, um einen Benutzer zu erstellen. Nach erfolgreichem Abschluss des Anrufs ist der neue Benutzer noch nicht nutzbar.

HTTP-Methode	Pfad
POST	/Accounts/{Account_id}/Core/v1/users

JSON-Eingabebeispiel

```
{
  "type" :  "application/astra-user",
  "version" : "1.1",
  "firstName" : "John",
  "lastName" : "West",
  "email" : "jwest@example.com"
}
```

Beispiel für die Wellung

```
curl --location -i --request POST
'https://astra.netapp.io/accounts/<ACCOUNT_ID>/core/v1/users' --header
'Accept: */*' --header 'Authorization: Bearer <API_TOKEN>' --data
@JSONinput
```

Beispiel für eine JSON-Ausgabe

```
{
    "metadata": {
        "creationTimestamp": "2022-11-20T17:23:15Z",
        "modificationTimestamp": "2022-11-20T17:23:15Z",
        "createdBy": "a20e91f3-2c49-443b-b240-615d940ec5f3",
        "labels": []
    },
    "type": "application/astra-user",
    "version": "1.2",
    "id": "d07dac0a-a328-4840-a216-12de16bbd484",
    "authProvider": "local",
    "authID": "jwest@example.com",
    "firstName": "John",
    "lastName": "West",
    "companyName": "",
    "email": "jwest@example.com",
    "postalAddress": {
        "addressCountry": "",
        "addressLocality": "",
        "addressRegion": "",
        "streetAddress1": "",
        "streetAddress2": "",
        "postalCode": ""
    },
    "state": "active",
    "sendWelcomeEmail": "false",
    "isEnabled": "true",
    "isInviteAccepted": "true",
    "enableTimestamp": "2022-11-20T17:23:15Z",
    "lastActTimestamp": ""
}
```

3. Wählen Sie optional die zulässigen Namespaces

Führen Sie den Workflow aus "Listen Sie die Namespaces auf" Und wählen Sie die Namespaces aus, auf die Sie den Zugriff beschränken möchten.

4. Binden Sie den Benutzer an eine Rolle

Führen Sie den folgenden REST-API-Aufruf durch, um den Benutzer an eine Rolle zu binden. Das Beispiel unten beschränkt den Namespace-Zugriff nicht. Siehe "Erweiterte RBAC mit Namespace-Granularität" Finden Sie weitere Informationen.

HTTP-Methode	Pfad
POST	/Accounts/{Account_id}/Core/v1/roleBindungen

JSON-Eingabebeispiel

```
{
  "type" :  "application/astra-roleBinding",
  "version" : "1.1",
  "userID" : "d07dac0a-a328-4840-a216-12de16bbd484",
  "accountID" : "29e1f39f-2bf4-44ba-a191-5b84ef414c95",
  "role" : "viewer",
  "roleConstraints": [ "*" ]
}
```

Beispiel für die Wellung

```
curl --location -i --request POST
'https://astra.netapp.io/accounts/<ACCOUNT_ID>/core/v1/roleBindings'
--header 'Accept: */*' --header 'Authorization: Bearer <API_TOKEN>' --data
@JSONinput
```

5. Erstellen Sie eine Anmeldeinformation

Führen Sie den folgenden REST-API-Aufruf durch, um eine Berechtigung zu erstellen und sie dem Benutzer zuzuordnen. Dieses Beispiel verwendet ein Passwort, das als base64-Wert angegeben wird. Der `name` Eigenschaft sollte die ID des im vorherigen Schritt zurückgegebenen Benutzers enthalten. Die Eingabeeigenschaft `change` Muss auch in base64 kodiert werden und bestimmt, ob der Benutzer bei der ersten Anmeldung sein Passwort ändern muss (`true` Oder `false`).

> Dieser Schritt ist nur bei Astra Control Center-Implementierungen mit lokaler Authentifizierung erforderlich. Bei Implementierungen des Astra Control Center mit LDAP oder mit Astra Control Service sind dies nicht erforderlich.

HTTP-Methode	Pfad
POST	/Accounts/{Account_id}/Core/v1/Credentials

JSON-Eingabebeispiel

```
{
  "type" :  "application/astra-credential",
  "version" : "1.1",
  "name" : "d07dac0a-a328-4840-a216-12de16bbd484",
  "keyType" : "passwordHash",
  "keyStore" : {
      "cleartext" : "TmV0QXBwMTIz",
      "change" : "ZmFsc2U="
  },
  "valid" : "true"
}
```

Beispiel für die Wellung

```
curl --location -i --request POST
'https://astra.netapp.io/accounts/<ACCOUNT_ID>/core/v1/credentials'
--header 'Accept: */*' --header 'Authorization: Bearer <API_TOKEN>' --data
@JSONinput
```