



Astra **Connector**をインストールしてクラスタを管理 Astra Control Service

NetApp
April 24, 2024

目次

Astra Connectorをインストールしてクラスタを管理	1
以前のバージョンのAstra Connectorをインストール	1
(テクニカルプレビュー) 宣言型Kubernetes Astra Connectorのインストール	4

Astra Connectorをインストールしてクラスタを管理

Astra Connectorは、管理対象クラスタに実装されるソフトウェアで、管理対象クラスタとAstra Controlの間の通信を促進します。Astra Control Serviceを使用して管理されるクラスタには、次の2つのバージョンのAstra Connectorがあります。

- *以前のバージョンのAstra Connector* : "[以前のバージョンのAstra Connectorをインストール](#)"
Kubernetesネイティブでないワークフローでクラスタを管理する場合は、クラスタ。
- [技術プレビュー]*宣言型Kubernetes Astra Connector* : "[Kubernetesの宣言ワークフローで管理されるクラスタ用にAstra Connectorをインストール](#)"
宣言型Kubernetesワークフローを使用してクラスタを管理する場合は、クラスタ上。クラスタにAstra Connectorをインストールすると、クラスタが自動的にAstra Controlに追加されます。



宣言型Kubernetes Astra Connectorは、Astra Control Early Adopter Program (EAP ; Astra Control早期導入プログラム) の一部としてのみ利用できます。EAPへの参加については、NetAppの営業担当者にお問い合わせください。

以前のバージョンのAstra Connectorをインストール

Astra Control Serviceでは、以前のバージョンのAstra Connectorを使用して、Astra Control Serviceと非Kubernetesネイティブのワークフローで管理されるプライベートクラスタ間の通信を可能にします。Kubernetesネイティブでないワークフローで管理するプライベートクラスタにAstra Connectorをインストールする必要があります。

以前のバージョンのAstra Connectorでは、非Kubernetesネイティブのワークフローで管理される次のタイプのプライベートクラスタがサポートされていました。

- Amazon Elastic Kubernetes Service (EKS)
- Azure Kubernetes Service (AKS)
- Google Kubernetes Engine (GKE)
- Red Hat OpenShift Service on AWS (ROSA)
- RosaとAWS PrivateLink
- Red Hat OpenShift Container Platform (オンプレミス)

このタスクについて

- これらの手順を実行する場合は、Astra Control Serviceで管理するプライベートクラスタに対してこれらのコマンドを実行します。
- 要塞ホストを使用している場合は、要塞ホストのコマンドラインからこれらのコマンドを問題 で実行します。

作業を開始する前に

- Astra Control Serviceで管理するプライベートクラスタにアクセスする必要があります。
- クラスタにAstra Connectorオペレータをインストールするには、Kubernetes管理者の権限が必要です。

手順

1. Kubernetesネイティブでないワークフローで管理するプライベートクラスタに、以前のAstra Connectorオペレータをインストールします。このコマンドを実行すると、名前空間 `astra-connector-operator` が作成され、設定が名前空間に適用されます。

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

2. オペレータが設置され、準備ができていることを確認します。

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

3. Astra ControlからAPIトークンを取得を参照してください ["Astra Automationのドキュメント"](#) 手順については、を参照し
4. Astra Connector名前空間を作成します。

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

5. Astra Connector CRファイルを作成してという名前を付ける `astra-connector-cr.yaml`。カッコ内の値を、Astra Controlの環境とクラスタの構成に合わせて更新します。

- **<ASTRA_CONTROL_SERVICE_URL>** : Astra ControlサービスのWeb UI URL。例 :

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

- **<ASTRA_CONTROL_SERVICE_API_TOKEN>** : 前の手順で取得したAstra Control APIトークン。
- ****** : (AKSクラスタのみ) -プライベート<PRIVATE_AKS_CLUSTER_NAME> Kubernetes Serviceクラスタのクラスタ名。プライベートAKSクラスタを追加する場合にのみ、この行のコメントを解除して入力します。
- **<ASTRA_CONTROL_ACCOUNT_ID>** : Astra ControlのWeb UIから取得。ページ右上の図アイコンを選択し、* API access *を選択します。

```
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
```

- データを入力した後、astra-connector-cr.yaml 正しい値を持つファイルを作成し、CRを適用します。

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

- Astra Connectorが完全に導入されたことを確認します。

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

- クラスタがAstra Controlに登録されたことを確認します。

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

次のような出力が表示されます。

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



ASTRACONNECTORIDはメモしておきます。ASTRACONNECTORIDは、クラスタをAstra Controlに追加するときに必要になります。

次の手順

Astra Connectorのインストールが完了したら、Astra Control Serviceにプライベートクラスタを追加する準備が整いました。

- ["プロバイダが管理するプライベートクラスタをAstra Control Serviceに追加します"](#)：プライベートIPアドレスを持ち、クラウドプロバイダによって管理されるクラスタを追加するには、次の手順を実行します。クラウドプロバイダのサービスプリンシパルアカウント、サービスアカウント、またはユーザアカウントが必要です。
- ["Astra Control Serviceに自己管理型プライベートクラスタを追加"](#)：プライベートIPアドレスを持ち、組織で管理されているクラスタを追加するには、次の手順を実行します。追加するクラスタのkubeconfigファイルを作成する必要があります。

を参照してください。

- ["クラスタを追加"](#)

(テクニカルプレビュー) 宣言型Kubernetes Astra Connectorのインストール

宣言型Kubernetesワークフローで管理されるクラスタは、Astra Connectorを使用して、管理対象クラスタとAstra Controlの間の通信を有効にします。Kubernetesの宣言ワークフローで管理するすべてのクラスタにAstra Connectorをインストールする必要があります。

宣言型Kubernetes Astra Connectorは、KubernetesコマンドとCustom Resource (CR) ファイルを使用してインストールします。

このタスクについて

- これらの手順を実行する場合は、Astra Controlで管理するクラスタでこれらのコマンドを実行します。
- 要塞ホストを使用している場合は、要塞ホストのコマンドラインからこれらのコマンドを問題 で実行します。

作業を開始する前に

- Astra Controlで管理するクラスタへのアクセスが必要です。
- クラスタにAstra Connectorオペレータをインストールするには、Kubernetes管理者の権限が必要です。



Kubernetes 1.25以降のクラスタのデフォルトであるポッドセキュリティアドミッション適用がクラスタに設定されている場合は、適切な名前スペースに対してPSA制限を有効にする必要があります。を参照してください ["Astra Controlを使用して、クラスタ管理のための環境を準備する"](#) 手順については、を参照し

手順

1. Kubernetesの宣言ワークフローを使用して管理するクラスタにAstra Connectorオペレータをインストールします。このコマンドを実行すると、名前スペース `astra-connector-operator` が作成され、設定が名前スペースに適用されます。

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

2. オペレータが設置され、準備ができていることを確認します。


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

3. Astra ControlからAPIトークンを取得を参照してください "[Astra Automationのドキュメント](#)" 手順については、[こちら](#)を参照し
4. トークンを使用してシークレットを作成します。<API_TOKEN>を、Astra Controlから受け取ったトークンに置き換えます。

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

5. Astra Connectorイメージの取得に使用するDockerシークレットを作成します。括弧<>の値は、環境の情報で置き換えます。



<ASTRA_CONTROL_ACCOUNT_ID>はAstra Control Web UIで確認できます。Web UIで、ページの右上にあるアイコンを選択し、*[API access]*を選択します。

```
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. Astra Connector CRファイルを作成してという名前を付ける astra-connector-cr.yaml。カッコ内の値を、Astra Controlの環境とクラスタの構成に合わせて更新します。
 - <ASTRA_CONTROL_ACCOUNT_ID>：前の手順でAstra Control Web UIから取得。
 - <CLUSTER_NAME>：このクラスタをAstra Controlで割り当てる名前。
 - <ASTRA_CONTROL_URL>：Astra ControlのWeb UI URL。例：

```
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. データを入力した後、astra-connector-cr.yaml 正しい値を持つファイルを作成し、CRを適用します。

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

8. Astra Connectorが完全に導入されたことを確認します。

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

9. クラスタがAstra Controlに登録されたことを確認します。

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

次のような出力が表示されます。

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

10. Astra Control Web UIの*[Clusters]*ページで、管理対象クラスタのリストにクラスタが表示されることを確認します。

著作権に関する情報

Copyright © 2024 NetApp, Inc. All Rights Reserved. Printed in the U.S.このドキュメントは著作権によって保護されています。著作権所有者の書面による事前承諾がある場合を除き、画像媒体、電子媒体、および写真複写、記録媒体、テープ媒体、電子検索システムへの組み込みを含む機械媒体など、いかなる形式および方法による複製も禁止します。

ネットアップの著作物から派生したソフトウェアは、次に示す使用許諾条項および免責条項の対象となります。

このソフトウェアは、ネットアップによって「現状のまま」提供されています。ネットアップは明示的な保証、または商品性および特定目的に対する適合性の暗示的保証を含み、かつこれに限定されないいかなる暗示的な保証も行いません。ネットアップは、代替品または代替サービスの調達、使用不能、データ損失、利益損失、業務中断を含み、かつこれに限定されない、このソフトウェアの使用により生じたすべての直接的損害、間接的損害、偶発的損害、特別損害、懲罰的損害、必然的損害の発生に対して、損失の発生の可能性が通知されていたとしても、その発生理由、根拠とする責任論、契約の有無、厳格責任、不法行為（過失またはそうでない場合を含む）にかかわらず、一切の責任を負いません。

ネットアップは、ここに記載されているすべての製品に対する変更を随時、予告なく行う権利を保有します。ネットアップによる明示的な書面による合意がある場合を除き、ここに記載されている製品の使用により生じる責任および義務に対して、ネットアップは責任を負いません。この製品の使用または購入は、ネットアップの特許権、商標権、または他の知的所有権に基づくライセンスの供与とはみなされません。

このマニュアルに記載されている製品は、1つ以上の米国特許、その他の国の特許、および出願中の特許によって保護されている場合があります。

権利の制限について：政府による使用、複製、開示は、DFARS 252.227-7013（2014年2月）およびFAR 5252.227-19（2007年12月）のRights in Technical Data -Noncommercial Items（技術データ - 非商用品目に関する諸権利）条項の(b)(3)項、に規定された制限が適用されます。

本書に含まれるデータは商用製品および/または商用サービス（FAR 2.101の定義に基づく）に関係し、データの所有権はNetApp, Inc.にあります。本契約に基づき提供されるすべてのネットアップの技術データおよびコンピュータソフトウェアは、商用目的であり、私費のみで開発されたものです。米国政府は本データに対し、非独占的かつ移転およびサブライセンス不可で、全世界を対象とする取り消し不能の制限付き使用权を有し、本データの提供の根拠となった米国政府契約に関連し、当該契約の裏付けとする場合にのみ本データを使用できます。前述の場合を除き、NetApp, Inc.の書面による許可を事前に得ることなく、本データを使用、開示、転載、改変するほか、上演または展示することはできません。国防総省にかかる米国政府のデータ使用权については、DFARS 252.227-7015(b)項（2014年2月）で定められた権利のみが認められます。

商標に関する情報

NetApp、NetAppのロゴ、<http://www.netapp.com/TM>に記載されているマークは、NetApp, Inc.の商標です。その他の会社名と製品名は、それを所有する各社の商標である場合があります。