# Inizia subito

## Astra Control Service

NetApp
April 24, 2024

# Sommario

# Inizia subito

## Scopri di più su Astra Control

Astra Control è una soluzione per la gestione del ciclo di vita dei dati delle applicazioni Kubernetes che semplifica le operazioni per le applicazioni stateful. Proteggi, esegui il backup e migra facilmente i carichi di lavoro Kubernetes e crea istantaneamente cloni applicativi funzionanti.

### Caratteristiche

Astra Control offre funzionalità critiche per la gestione del ciclo di vita dei dati delle applicazioni Kubernetes:

- Gestire automaticamente lo storage persistente
- Creazione di snapshot e backup on-demand basati sulle applicazioni
- Automatizzare le operazioni di backup e snapshot basate su policy
- Migrare applicazioni e dati da un cluster Kubernetes a un altro
- Replica delle applicazioni su un sistema remoto utilizzando la tecnologia NetApp SnapMirror (Astra Control Center)
- Clonare le applicazioni dallo staging alla produzione
- Visualizzare lo stato di salute e protezione dell'applicazione
- Utilizzare un'interfaccia utente Web o un'API per implementare i flussi di lavoro di backup e migrazione

### Modelli di implementazione

Astra Control è disponibile in due modelli di implementazione:

- **Astra Control Service**: Un servizio gestito da NetApp che offre la gestione dei dati application-aware dei cluster Kubernetes in ambienti di cloud provider multipli, oltre ai cluster Kubernetes autogestiti.
- **Astra Control Center**: Software autogestito che fornisce la gestione dei dati applicativa dei cluster Kubernetes in esecuzione nel tuo ambiente on-premise. Il centro di controllo Astra può essere installato anche in ambienti di cloud provider multipli con un backend di storage NetApp Cloud Volumes ONTAP.

| | **Servizio di controllo Astra** | **Centro di controllo Astra** |
|---|---|---|
| **Come viene offerto?** | Come servizio cloud completamente gestito da NetApp | Come software che puoi scaricare, installare e gestire |
| **Dove è ospitato?** | Su un cloud pubblico scelto da NetApp | Sul tuo cluster Kubernetes |
| **Come viene aggiornato?** | Gestito da NetApp | Gli aggiornamenti vengono gestiti |

| | Servizio di controllo Astra | Centro di controllo Astra |
|---|---|---|
| **Quali sono le distribuzioni Kubernetes supportate?** | • **Cloud provider**<br>◦ Amazon Web Services<br>▪ Amazon Elastic Kubernetes Service (EKS)<br>◦ Google Cloud<br>▪ Google Kubernetes Engine (GKE)<br>◦ Microsoft Azure<br>▪ Servizio Azure Kubernetes (AKS)<br>• **Cluster autogestiti**<br>◦ Kubernetes (upstream)<br>◦ Rancher Kubernetes Engine (RKE)<br>◦ Red Hat OpenShift Container Platform<br>• **Cluster on-premise**<br>◦ Red Hat OpenShift Container Platform all'interno dell'hotel | • Azure Kubernetes Service su Azure Stack HCI<br>• Google anthos<br>• Kubernetes (upstream)<br>• Rancher Kubernetes Engine (RKE)<br>• Red Hat OpenShift Container Platform |

| | Servizio di controllo Astra | Centro di controllo Astra |
|---|---|---|
| **Quali sono i backend di storage supportati?** | • **Cloud provider**<br>◦ Amazon Web Services<br>▪ Amazon EBS<br>▪ Amazon FSX per NetApp ONTAP<br>▪ "Cloud Volumes ONTAP"<br>◦ Google Cloud<br>▪ Disco persistente di Google<br>▪ NetApp Cloud Volumes Service<br>▪ "Cloud Volumes ONTAP"<br>◦ Microsoft Azure<br>▪ Dischi gestiti Azure<br>▪ Azure NetApp Files<br>▪ "Cloud Volumes ONTAP"<br>• **Cluster autogestiti**<br>◦ Amazon EBS<br>◦ Dischi gestiti Azure<br>◦ Disco persistente di Google<br>◦ "Cloud Volumes ONTAP"<br>◦ NetApp MetroCluster<br>◦ "Longhorn"<br>• **Cluster on-premise**<br>◦ NetApp MetroCluster<br>◦ Sistemi NetApp ONTAP AFF e FAS<br>◦ NetApp ONTAP Select<br>◦ "Cloud Volumes ONTAP"<br>◦ "Longhorn" | • Sistemi NetApp ONTAP AFF e FAS<br>• NetApp ONTAP Select<br>• "Cloud Volumes ONTAP"<br>• "Longhorn" |

## Come funziona Astra Control Service

Astra Control Service è un servizio cloud gestito da NetApp sempre attivo e aggiornato con le funzionalità più recenti. Utilizza diversi componenti per consentire la gestione del ciclo di vita dei dati delle applicazioni.

Ad alto livello, Astra Control Service funziona come segue:

- Per iniziare a utilizzare Astra Control Service, devi configurare il tuo cloud provider e registrarti per un account Astra.

+ ** per i cluster GKE, Astra Control Service utilizza "NetApp Cloud Volumes Service per Google Cloud" O Google Persistent Disks come back-end di storage per i volumi persistenti.

+ ** per i cluster AKS, Astra Control Service utilizza "Azure NetApp Files" O Azure Managed Disks come back-end di storage per i volumi persistenti.

+ ** per i cluster Amazon EKS, Astra Control Service utilizza "Amazon Elastic Block Store" oppure "Amazon FSX per NetApp ONTAP" come back-end di storage per i volumi persistenti.

- Aggiungi il tuo primo calcolo Kubernetes ad Astra Control Service. Astra Control Service esegue le seguenti operazioni:
  - Crea un archivio di oggetti nel tuo account cloud provider, dove vengono memorizzate le copie di backup.

In Azure, Astra Control Service crea anche un gruppo di risorse, un account di storage e chiavi per il container Blob.

- Crea un nuovo ruolo di amministratore e un nuovo account del servizio Kubernetes sul cluster.
- Utilizza questo nuovo ruolo di amministratore per installare link../concepts/architecture#astra-control-components[Astra Control Protivioner] nel cluster e per creare una o più classi di storage.
- Se utilizzi un'offerta di cloud storage NetApp come back-end dello storage, Astra Control Service utilizza Astra Control Provivisioner per il provisioning dei volumi persistenti per le tue app. Se si utilizzano dischi gestiti Amazon EBS o Azure come back-end dello storage, è necessario installare un driver CSI specifico del provider. Le istruzioni di installazione sono fornite in "Configurare Amazon Web Services" e. "Configurare Microsoft Azure con dischi gestiti Azure".
  - A questo punto, è possibile definire le applicazioni dal cluster. Il provisioning dei volumi persistenti sul back-end dello storage viene eseguito attraverso la nuova classe di storage predefinita.
  - Quindi, utilizza Astra Control Service per gestire queste applicazioni e iniziare a creare snapshot, backup e cloni.

Il piano gratuito di Astra Control ti consente di gestire fino a 10 spazi dei nomi nel tuo account. Se si desidera gestire più di 10 spazi dei nomi, è necessario impostare la fatturazione eseguendo l'aggiornamento dal piano gratuito al piano Premium.

## Come funziona Astra Control Center

Astra Control Center viene eseguito localmente nel tuo cloud privato.

Astra Control Center supporta i cluster Kubernetes con una classe di storage configurata da Astra Control Provisioner con un backend di storage ONTAP.

Il centro di controllo Astra è completamente integrato nell'ecosistema AutoSupport e Active IQ per fornire agli utenti e al supporto NetApp informazioni sulla risoluzione dei problemi e sull'utilizzo.

Puoi provare Astra Control Center utilizzando una licenza di valutazione di 90 giorni. La versione di valutazione è supportata tramite e-mail e opzioni della community. Inoltre, puoi accedere agli articoli e alla documentazione della Knowledge base dalla dashboard di supporto all'interno del prodotto.

Per installare e utilizzare Astra Control Center, è necessario soddisfare determinati requisiti "requisiti".

Ad alto livello, Astra Control Center funziona come segue:

- Astra Control Center viene installato nel proprio ambiente locale. Scopri di più su come "Installare Astra

Control Center".

- È possibile completare alcune attività di configurazione, come ad esempio:
  - Impostare la licenza.
  - Aggiungere il primo cluster.
  - Aggiungere il backend di storage rilevato quando si aggiunge il cluster.
  - Aggiungi un bucket di store di oggetti che memorizzerà i backup delle tue app.

Scopri di più su come "Configurare Astra Control Center".

È possibile aggiungere applicazioni al cluster. In alternativa, se nel cluster gestito sono già presenti alcune applicazioni, è possibile utilizzare Astra Control Center per gestirle. Quindi, utilizza Astra Control Center per creare snapshot, backup, cloni e relazioni di replica.

### Per ulteriori informazioni

- "Documentazione della famiglia di prodotti NetApp Astra"
- "Documentazione di Astra Control Center"
- "Documentazione sull'API Astra Control"
- "Documentazione di Astra Trident"
- "Documentazione ONTAP"

# Implementazioni Kubernetes supportate

Astra Control Service può gestire le applicazioni in esecuzione su un cluster Kubernetes gestito in Amazon Elastic Kubernetes Service (EKS) e i cluster gestiti da soli.

Astra Control Service è in grado di gestire le applicazioni in esecuzione su un cluster Kubernetes gestito in Google Kubernetes Engine (GKE) e i cluster gestiti autonomamente.

Astra Control Service può gestire le applicazioni in esecuzione su un cluster Kubernetes gestito in Azure Kubernetes Service (AKS) e i cluster gestiti da soli.

- "Scopri come configurare Amazon Web Services per Astra Control Service".
- "Scopri come configurare Google Cloud per Astra Control Service".
- "Scopri come configurare Microsoft Azure con Azure NetApp Files per il servizio di controllo Astra".
- "Scopri come configurare Microsoft Azure con dischi gestiti Azure per Astra Control Service".
- "Scopri come preparare i cluster autogestiti prima di aggiungerli ad Astra Control Service".

# Avvio rapido per Astra Control Service

Questa pagina fornisce una panoramica generale dei passaggi da completare per iniziare a utilizzare Astra Control Service. I collegamenti all'interno di ogni passaggio consentono di accedere a una pagina che fornisce ulteriori dettagli.

## [Uno] Configura il tuo cloud provider

1. Google Cloud:
    - Esaminare i requisiti del cluster di Google Kubernetes Engine.
    - Acquista Cloud Volumes Service per Google Cloud da Google Cloud Marketplace.
    - Abilitare le API richieste.
    - Creare un account di servizio e una chiave dell'account di servizio.
    - Imposta il peering di rete dal tuo VPC a Cloud Volumes Service per Google Cloud.

      "Scopri di più sui requisiti di Google Cloud".

2. Servizi Web Amazon:
    - Esaminare i requisiti del cluster di Amazon Web Services.
    - Crea un account Amazon.
    - Installare la CLI di Amazon Web Services.
    - Creare un utente IAM.
    - Creare e allegare un criterio di autorizzazioni.
    - Salvare le credenziali per l'utente IAM.

      "Scopri di più sui requisiti di Amazon Web Services".

3. Microsoft Azure:

    - Esaminate i requisiti del cluster di Azure Kubernetes Service per il backend di storage che intendete utilizzare.

      "Scopri di più sui requisiti di Microsoft Azure e Azure NetApp Files".

   "Scopri di più sui requisiti dei dischi gestiti di Microsoft Azure e Azure".

Se stai gestendo il tuo cluster e non è ospitato da un cloud provider, esamina i requisiti per i cluster autogestiti. "Scopri di più sui requisiti del cluster a gestione automatica".

## [Due] Completare la registrazione di Astra Control

1. Creare un "NetApp BlueXP" account.
2. Specifica il tuo ID email NetApp BlueXP durante la creazione dell'account Astra Control "Dalla pagina del prodotto Astra Control".

"Scopri di più sul processo di registrazione".

## [Tre] Aggiungere cluster ad Astra Control

Dopo aver effettuato l'accesso, selezionare **Add cluster** (Aggiungi cluster) per iniziare a gestire il cluster con Astra Control.

"Scopri di più sull'aggiunta di cluster".

# Configura il tuo cloud provider

## Configurare Amazon Web Services

Sono necessari alcuni passaggi per preparare il tuo progetto Amazon Web Services prima di poter gestire i cluster Amazon Elastic Kubernetes Service (EKS) con Astra Control Service.

### Avvio rapido per la configurazione di Amazon Web Services

Inizia subito seguendo questi passaggi o scorri verso il basso fino alle restanti sezioni per ottenere informazioni dettagliate.

#### [Uno] Consulta i requisiti del servizio Astra Control per Amazon Web Services

Assicurarsi che i cluster siano integri e che eseguano una versione supportata di Kubernetes, che i nodi di lavoro siano online e che eseguano Linux o Windows e altro ancora. Scopri di più su questo passaggio.

#### [Due] Crea un account Amazon

Se non disponi già di un account Amazon, devi crearne uno per poter utilizzare EKS. Scopri di più su questo passaggio.

#### [Tre] Installare la CLI di Amazon Web Services

Installare la CLI AWS in modo da poter gestire AWS dalla riga di comando. Seguire le istruzioni dettagliate.

#### [Quattro] Facoltativo: Creare un utente IAM

Creare un utente Amazon Identity and Access Management (IAM). Puoi anche saltare questo passaggio e utilizzare un utente IAM esistente con Astra Control Service.

Leggi le istruzioni dettagliate.

#### [Cinque] Creare e allegare un criterio di autorizzazioni

Creare una policy con le autorizzazioni richieste per Astra Control Service per interagire con l'account AWS.

Leggi le istruzioni dettagliate.

#### [Sei] Salvare le credenziali per l'utente IAM

Salvare le credenziali per l'utente IAM in modo da poter importare le credenziali in Astra Control Service.

Leggi le istruzioni dettagliate.

### Requisiti del cluster EKS

Un cluster Kubernetes deve soddisfare i seguenti requisiti per consentirne il rilevamento e la gestione da Astra Control Service.

#### Versione di Kubernetes

Un cluster deve eseguire una versione di Kubernetes compresa tra 1,25 e 1,28.

**Tipo di immagine**

Il tipo di immagine per ciascun nodo di lavoro deve essere Linux.

**Stato del cluster**

I cluster devono essere in esecuzione in uno stato integro e avere almeno un nodo di lavoro online senza nodi di lavoro in uno stato di errore.

**Astra Control provisioner**

Astra Control Provivisioner e un controller delle snapshot esterno sono necessari per le operazioni con backend di storage. Per attivare queste operazioni, procedere come segue:

1. "Installare gli snapshot CRD e lo snapshot controller".
2. "Abilita Astra Control Provisioner".
3. "Creare una classe VolumeSnapshotClass".

**Driver CSI per Amazon Elastic Block Store (EBS)**

Se si utilizza il backend dello storage Amazon EBS, è necessario installare il driver CSI (Container Storage Interface) per EBS (non viene installato automaticamente).

Per istruzioni sull'installazione del driver CSI, fare riferimento alla procedura.

**Installare uno snap-shot esterno**

Se non l'hai già fatto, "Installare gli snapshot CRD e lo snapshot controller".

**Installare il driver CSI come add-on Amazon EKS**

1. Creare il ruolo IAM del driver CSI Amazon EBS per gli account del servizio. Seguire le istruzioni "Nella documentazione Amazon", Utilizzando i comandi CLI di AWS nelle istruzioni.
2. Aggiungere il componente aggiuntivo Amazon EBS CSI utilizzando il seguente comando AWS CLI, sostituendo le informazioni tra parentesi <> con valori specifici per il proprio ambiente. Sostituire <DRIVER_ROLE> con il nome del ruolo del driver EBS CSI creato nel passaggio precedente:

```
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>
```

**Configurare la classe di storage EBS**

1. Clonare il repository GitHub del driver CSI di Amazon EBS nel sistema.

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

2. Accedere alla directory di esempio del provisioning dinamico.

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

3. Implementare la classe di storage ebs-sc e l'attestazione di volume persistente ebs-claim dalla directory manifests.

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

4. Descrivere la classe di storage ebs-sc.

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

Viene visualizzato un output che descrive gli attributi della classe di storage.

### Crea un account Amazon

Se non disponi già di un account Amazon, devi crearne uno per attivare la fatturazione per Amazon EKS.

#### Fasi

1. Accedere alla "Pagina principale Amazon" , Selezionare **Accedi** in alto a destra e selezionare **inizia qui**.
2. Seguire le istruzioni per creare un account.

### Installare la CLI di Amazon Web Services

Installare la CLI AWS in modo da poter gestire le risorse AWS dalla riga di comando.

#### Fase

1. Passare a. "Introduzione a AWS CLI" E seguire le istruzioni per installare l'interfaccia CLI.

### Facoltativo: Creare un utente IAM

Creare un utente IAM in modo da poter utilizzare e gestire i servizi e le risorse AWS con maggiore sicurezza. È inoltre possibile saltare questo passaggio e utilizzare un utente IAM esistente con Astra Control Service.

#### Fase

1. Passare a. "Creazione di utenti IAM" E seguire le istruzioni per creare un utente IAM.

### Creare e allegare un criterio di autorizzazioni

Creare una policy con le autorizzazioni richieste per Astra Control Service per interagire con l'account AWS.

#### Fasi

1. Creare un nuovo file chiamato `policy.json`.
2. Copiare il seguente contenuto JSON nel file:

```
{
    "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. Creare la policy:

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

4. Allegare il criterio all'utente IAM. Sostituire `<IAM-USER-NAME>` Con il nome utente dell'utente IAM creato o con un utente IAM esistente:

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

#### Salvare le credenziali per l'utente IAM

Salvare le credenziali per l'utente IAM in modo da rendere Astra Control Service consapevole dell'utente.

#### Fasi

1. Scarica le credenziali. Sostituire `<IAM-USER-NAME>` Con il nome utente dell'utente IAM che si desidera utilizzare:

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

#### Risultato

Il `credential.json` Il file viene creato ed è possibile importare le credenziali in Astra Control Service.

## Configurare Google Cloud

Sono necessari alcuni passaggi per preparare il tuo progetto Google Cloud prima di poter gestire i cluster di Google Kubernetes Engine con Astra Control Service.

> Se non si inizia a utilizzare Google Cloud Volumes Service per Google Cloud come back-end di storage ma si prevede di utilizzarlo in un secondo momento, è necessario completare i passaggi necessari per configurare Google Cloud Volumes Service per Google Cloud ora. La creazione di un account di servizio in un secondo momento implica la perdita di accesso ai bucket di storage esistenti.

#### Avvio rapido per la configurazione di Google Cloud

Inizia subito seguendo questi passaggi o scorri verso il basso fino alle restanti sezioni per ottenere informazioni dettagliate.

##### [Uno] Consulta i requisiti del servizio Astra Control per Google Kubernetes Engine

Assicurarsi che i cluster siano integri e che eseguano una versione di Kubernetes supportata, che i nodi di lavoro siano online e che eseguano un tipo di immagine supportato e altro ancora. Scopri di più su questo passaggio.

##### [Due] (Facoltativo): Acquista Cloud Volumes Service per Google Cloud

Se si intende utilizzare Cloud Volumes Service per Google Cloud come back-end di storage, accedere alla pagina NetApp Cloud Volumes Service nel Google Cloud Marketplace e selezionare Acquista. Scopri di più su questo passaggio.

##### [Tre] Abilita le API nel tuo progetto Google Cloud

Abilitare le seguenti API di Google Cloud:

- Motore di Google Kubernetes
- Cloud Storage
- API JSON per lo storage cloud

- Utilizzo del servizio
- API Cloud Resource Manager
- NetApp Cloud Volumes Service
  - Richiesto per Cloud Volumes Service per Google Cloud
  - Opzionale (ma consigliato) per Google Persistent Disk
- API Service Consumer Management
- API di Service Networking
- API di gestione dei servizi

Seguire le istruzioni dettagliate.

#### [Quattro] Creare un account di servizio con le autorizzazioni richieste

Creare un account di servizio Google Cloud con le seguenti autorizzazioni:

- Amministratore del motore di Kubernetes
- NetApp Cloud Volumes Admin
  - Richiesto per Cloud Volumes Service per Google Cloud
  - Opzionale (ma consigliato) per Google Persistent Disk
- Amministratore dello storage
- Visualizzatore utilizzo servizio
- Visualizzatore di Compute Network

Leggi le istruzioni dettagliate.

#### [Cinque] Creare una chiave dell'account del servizio

Creare una chiave per l'account del servizio e salvare il file delle chiavi in una posizione sicura. Seguire le istruzioni dettagliate.

#### [Sei] (Facoltativo): Impostare il peering di rete per il VPC

Se intendi utilizzare Cloud Volumes Service per Google Cloud come back-end di storage, imposta il peering di rete dal tuo VPC a Cloud Volumes Service per Google Cloud. Seguire le istruzioni dettagliate.

### Requisiti del cluster GKE

Un cluster Kubernetes deve soddisfare i seguenti requisiti per consentirne il rilevamento e la gestione da Astra Control Service. Alcuni di questi requisiti si applicano solo se si prevede di utilizzare Cloud Volumes Service per Google Cloud come back-end di storage.

#### Versione di Kubernetes

Un cluster deve eseguire una versione di Kubernetes compresa tra 1,26 e 1,28.

#### Tipo di immagine

Il tipo di immagine per ciascun nodo di lavoro deve essere `COS_CONTAINERD`.

**Stato del cluster**

I cluster devono essere in esecuzione in uno stato integro e avere almeno un nodo di lavoro online senza nodi di lavoro in uno stato di errore.

**Regione di Google Cloud**

Se si prevede di utilizzare Cloud Volumes Service per Google Cloud come backend di storage, i cluster devono essere eseguiti in un "Area di Google Cloud in cui è supportato Cloud Volumes Service per Google Cloud." Si noti che Astra Control Service supporta entrambi i tipi di servizio: CVS e CVS-Performance. Come Best practice, devi scegliere una regione che supporti Cloud Volumes Service per Google Cloud, anche se non la utilizzi come back-end di storage. In questo modo sarà più semplice utilizzare Cloud Volumes Service per Google Cloud come back-end di storage in futuro, se i requisiti di performance cambiano.

**Networking**

Se si intende utilizzare Cloud Volumes Service per Google Cloud come backend di storage, il cluster deve risiedere in un VPC con Cloud Volumes Service per Google Cloud. Questo passaggio è descritto di seguito.

**Cluster privati**

Se il cluster è privato, il "reti autorizzate" Deve consentire l'indirizzo IP di Astra Control Service:

52.188.218.166/32

**Modalità operativa per un cluster GKE**

Si consiglia di utilizzare la modalità operativa standard. La modalità Autopilot non è stata testata al momento. "Scopri di più sulle modalità operative".

**Pool di storage**

Se si utilizza NetApp Cloud Volumes Service come backend di storage con il tipo di servizio CVS, è necessario configurare i pool di storage prima di poter eseguire il provisioning dei volumi. Fare riferimento a. "Tipo di servizio, classi di storage e dimensione PV per cluster GKE" per ulteriori informazioni.

### Opzionale: Acquista Cloud Volumes Service per Google Cloud

Il servizio di controllo Astra può utilizzare Cloud Volumes Service per Google Cloud come back-end di storage per i volumi persistenti. Se intendi utilizzare questo servizio, devi acquistare Cloud Volumes Service per Google Cloud da Google Cloud Marketplace per abilitare la fatturazione per volumi persistenti.

**Fase**

1. Accedere alla "Pagina Cloud Volumes Service di NetApp" In Google Cloud Marketplace, selezionare **Purchase** (Acquista) e seguire le istruzioni.

   "Seguire le istruzioni dettagliate nella documentazione di Google Cloud per acquistare e attivare il servizio".

### Abilitare le API nel progetto

Il progetto richiede autorizzazioni per accedere a specifiche API di Google Cloud. Le API vengono utilizzate per interagire con le risorse cloud di Google, come i cluster GKE e lo storage NetApp Cloud Volumes Service.

**Fase**

1. "Utilizzare la console Google Cloud o la CLI gcloud per abilitare le seguenti API":
   - Motore di Google Kubernetes

- Cloud Storage
- API JSON per lo storage cloud
- Utilizzo del servizio
- API Cloud Resource Manager
- NetApp Cloud Volumes Service (richiesto per Cloud Volumes Service per Google Cloud)
- API Service Consumer Management
- API di Service Networking
- API di gestione dei servizi

Il video seguente mostra come abilitare le API dalla console Google Cloud.

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

#### Creare un account di servizio

Astra Control Service utilizza un account di servizio Google Cloud per facilitare la gestione dei dati dell'applicazione Kubernetes per conto dell'utente.

**Fasi**

1. Accedere a Google Cloud e. "creare un account di servizio utilizzando la console, il comando gcloud o un altro metodo preferito".
2. Assegnare all'account del servizio i seguenti ruoli:
   - **Kubernetes Engine Admin** - utilizzato per elencare i cluster e creare l'accesso amministratore per gestire le applicazioni.
   - **NetApp Cloud Volumes Admin** - utilizzato per gestire lo storage persistente per le applicazioni.
   - **Storage Admin** - utilizzato per gestire bucket e oggetti per il backup delle applicazioni.
   - **Visualizzatore utilizzo servizio** - consente di verificare se le API Cloud Volumes Service per Google Cloud richieste sono attivate.
   - **Visualizzatore di rete di calcolo** - utilizzato per verificare se il VPC Kubernetes è autorizzato a raggiungere Cloud Volumes Service per Google Cloud.

Se si desidera utilizzare gcloud, è possibile seguire i passaggi dall'interfaccia Astra Control. Selezionare **account > credenziali > Aggiungi credenziali**, quindi selezionare **istruzioni**.

Se si desidera utilizzare la console Google Cloud, il video seguente mostra come creare l'account del servizio dalla console.

► https://docs.netapp.com/it-it/astra-control-service/media/get-started/video-create-gcp-service-account.mp4

*(video)*

#### Configurare l’account di servizio per un VPC condiviso

Per gestire i cluster GKE che risiedono in un progetto, ma utilizzano un VPC di un progetto diverso (un VPC condiviso), è necessario specificare l’account del servizio Astra come membro del progetto host con il ruolo **Compute Network Viewer**.

**Fasi**

1. Dalla console di Google Cloud, accedere a **IAM & Admin** e selezionare **Service Accounts**.
2. Individuare l’account di servizio Astra "le autorizzazioni richieste" quindi copiare l’indirizzo e-mail.
3. Accedere al progetto host e selezionare **IAM & Admin** > **IAM**.
4. Selezionare **Aggiungi** e aggiungere una voce per l’account del servizio.
   a. **Nuovi membri**: Inserire l’indirizzo e-mail dell’account del servizio.
   b. **Ruolo**: Selezionare **Compute Network Viewer**.
   c. Selezionare **Salva**.

**Risultato**

L’aggiunta di un cluster GKE utilizzando un VPC condiviso funziona perfettamente con Astra.

### Creare una chiave dell’account del servizio

Invece di fornire un nome utente e una password ad Astra Control Service, fornirai una chiave account del servizio quando Aggiungi il tuo primo cluster. Astra Control Service utilizza la chiave dell’account del servizio per stabilire l’identità dell’account del servizio appena configurato.

La chiave dell’account del servizio è in formato non crittografato e memorizzata nel formato JSON (JavaScript Object Notation). Contiene informazioni sulle risorse GCP a cui si dispone dei diritti di accesso.

È possibile visualizzare o scaricare il file JSON solo quando si crea la chiave. Tuttavia, è possibile creare una nuova chiave in qualsiasi momento.

**Fasi**

1. Accedere a Google Cloud e. "creare una chiave dell’account di servizio utilizzando la console, il comando gcloud o un altro metodo preferito".
2. Quando richiesto, salvare il file delle chiavi dell’account di servizio in una posizione sicura.

Il video seguente mostra come creare la chiave dell’account di servizio dalla console Google Cloud.

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

key.mp4 *(video)*

### Opzionale: Configurare il peering di rete per il VPC

Se intendi utilizzare Cloud Volumes Service per Google Cloud come servizio di back-end per lo storage, il passaggio finale è configurare il peering di rete dal tuo VPC a Cloud Volumes Service per Google Cloud.

Il modo più semplice per configurare il peering di rete è ottenere i comandi gcloud direttamente da Cloud Volumes Service. I comandi sono disponibili da Cloud Volumes Service quando si crea un nuovo file system.

**Fasi**

1. "Vai alle mappe delle regioni globali BlueXP di NetApp" E identificare il tipo di servizio che si utilizza nell'area di Google Cloud in cui risiede il cluster.

   Cloud Volumes Service offre due tipi di servizio: CVS e CVS-Performance. "Scopri di più su questi tipi di servizi".

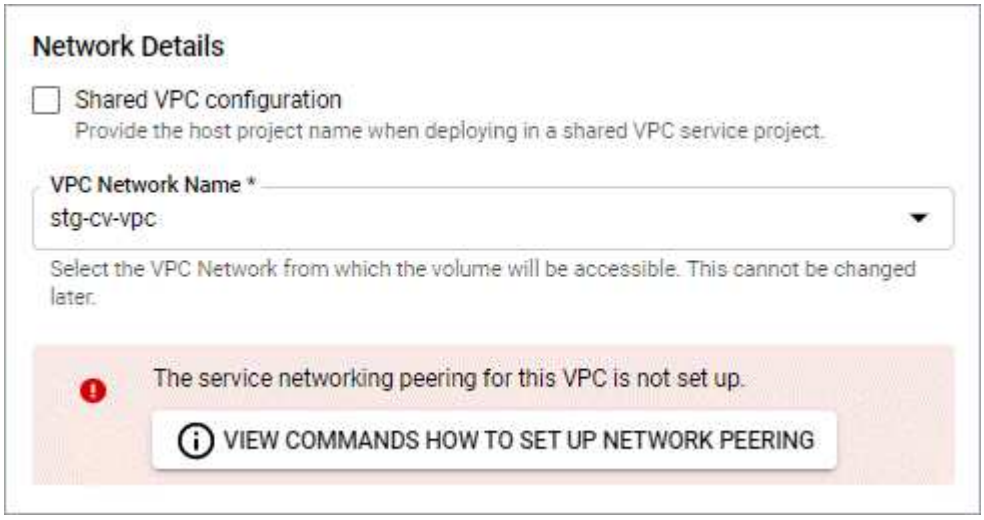
2. "Vai a Cloud Volumes in Google Cloud Platform".
3. Nella pagina **volumi**, selezionare **Crea**.
4. In **tipo di servizio**, selezionare **CVS** o **CVS-Performance**.

   Devi scegliere il tipo di servizio corretto per la tua area geografica Google Cloud. Questo è il tipo di servizio identificato al punto 1. Dopo aver selezionato un tipo di servizio, l'elenco delle regioni nella pagina viene aggiornato con le regioni in cui tale tipo di servizio è supportato.

   Dopo questa fase, è sufficiente inserire le informazioni di rete per ottenere i comandi.

5. In **Regione**, selezionare la propria regione e zona.
6. In **Dettagli rete**, selezionare il VPC.

   Se non hai configurato il peering di rete, verrà visualizzata la seguente notifica:




7. Selezionare il pulsante per visualizzare i comandi di configurazione del peering di rete.
8. Copiare i comandi ed eseguirli in Cloud Shell.

   Per ulteriori informazioni sull'utilizzo di questi comandi, fare riferimento a. "Guida rapida per Cloud Volumes Service per GCP".

"Scopri di più sulla configurazione dell'accesso ai servizi privati e sulla configurazione del peering di rete".

9. Al termine, selezionare Annulla nella pagina **Crea file system**.

   Abbiamo iniziato a creare questo volume solo per ottenere i comandi per il peering di rete.

## Configurare Microsoft Azure con Azure NetApp Files

Sono necessari alcuni passaggi per preparare l'abbonamento a Microsoft Azure prima di poter gestire i cluster di servizi Azure Kubernetes con Astra Control Service. Seguire queste istruzioni se si intende utilizzare Azure NetApp Files come backend di storage.

### Avvio rapido per la configurazione di Azure

Inizia subito seguendo questi passaggi o scorri verso il basso fino alle restanti sezioni per ottenere informazioni dettagliate.

#### [Uno] Esaminare i requisiti del servizio di controllo Astra per Azure Kubernetes Service

Assicurarsi che i cluster siano integri e che eseguano una versione supportata di Kubernetes, che i pool di nodi siano online e che eseguano Linux e altro ancora. Scopri di più su questo passaggio.

#### [Due] Iscriviti a Microsoft Azure

Creare un account Microsoft Azure. Scopri di più su questo passaggio.

#### [Tre] Registrati a Azure NetApp Files

Registrare il NetApp Resource Provider. Scopri di più su questo passaggio.

#### [Quattro] Creare un account NetApp

Accedere a Azure NetApp Files nel portale Azure e creare un account NetApp. Scopri di più su questo passaggio.

#### [Cinque] Configurare i pool di capacità

Configurare uno o più pool di capacità per i volumi persistenti. Scopri di più su questo passaggio.

#### [Sei] Delegare una subnet a Azure NetApp Files

Delegare una subnet a Azure NetApp Files in modo che il servizio di controllo Astra possa creare volumi persistenti in tale subnet. Scopri di più su questo passaggio.

#### [Sette] Creare un'entità del servizio Azure

Creare un'entità del servizio Azure con il ruolo di collaboratore. Scopri di più su questo passaggio.

#### [Otto] Opzionale: Configurare la ridondanza per i bucket di backup di Azure

Per impostazione predefinita, i bucket Astra Control Service utilizzati per memorizzare i backup di Azure Kubernetes Service utilizzano l'opzione di ridondanza LRS (Redundant Storage) locale. Come passaggio opzionale, è possibile configurare un livello di ridondanza più duraturo per i bucket Azure. Scopri di più su questo passaggio.

### Azure Kubernetes Service Cluster Requirements

Un cluster Kubernetes deve soddisfare i seguenti requisiti per consentirne il rilevamento e la gestione da Astra Control Service.

**Versione di Kubernetes**

I cluster devono eseguire Kubernetes versione 1,26 - 1,28.

**Tipo di immagine**

Il tipo di immagine per tutti i pool di nodi deve essere Linux.

**Stato del cluster**

I cluster devono essere in esecuzione in uno stato integro e avere almeno un nodo di lavoro online senza nodi di lavoro in uno stato di errore.

**Regione di Azure**

I cluster devono risiedere in una regione in cui è disponibile Azure NetApp Files. "Visualizza i prodotti Azure per regione".

**Iscrizione**

I cluster devono risiedere in un abbonamento in cui Azure NetApp Files è attivato. Scegli un abbonamento quando lo desideri Registrati a Azure NetApp Files.

**VNET**

Considerare i seguenti requisiti VNET:

- I cluster devono risiedere in una rete virtuale con accesso diretto a una subnet delegata da Azure NetApp Files. Scopri come configurare una subnet delegata.
- Se i cluster Kubernetes si trovano in un VNET collegato alla subnet delegata Azure NetApp Files di un altro VNET, entrambi i lati della connessione di peering devono essere in linea.
- Tenere presente che il limite predefinito per il numero di IP utilizzati in una rete virtuale (inclusi i VNet con peering immediato) con Azure NetApp Files è 1,000. "Visualizza i limiti delle risorse Azure NetApp Files".

  Se sei vicino al limite, hai due opzioni:

  - È possibile "inviare una richiesta di aumento del limite". Per assistenza, contatta il tuo rappresentante NetApp.
  - Quando si crea un nuovo cluster Amazon Kubernetes Service (AKS), specificare una nuova rete per il cluster. Una volta creata la nuova rete, eseguire il provisioning di una nuova subnet e delegare la subnet a Azure NetApp Files.

### Iscriviti a Microsoft Azure

Se non disponi di un account Microsoft Azure, inizia iscrivendoti a Microsoft Azure.

**Fasi**

1. Accedere alla "Pagina di iscrizione Azure" Per iscriversi al servizio Azure.
2. Selezionare un piano e seguire le istruzioni per completare l'abbonamento.

### Registrati a Azure NetApp Files

Ottieni l'accesso a Azure NetApp Files registrando il provider di risorse NetApp.

#### Fasi

1. Accedere al portale Azure.
2. "Seguire la documentazione di Azure NetApp Files per registrare il provider di risorse NetApp".

### Creare un account NetApp

Creare un account NetApp in Azure NetApp Files.

#### Fase

1. "Seguire la documentazione di Azure NetApp Files per creare un account NetApp dal portale Azure".

### Impostare un pool di capacità

Sono necessari uno o più pool di capacità per consentire ad Astra Control Service di eseguire il provisioning di volumi persistenti in un pool di capacità. Astra Control Service non crea pool di capacità per te.

Durante la configurazione dei pool di capacità per le applicazioni Kubernetes, prendere in considerazione quanto segue:

- I pool di capacità devono essere creati nella stessa regione di Azure in cui i cluster AKS saranno gestiti con Astra Control Service.
- Un pool di capacità può avere un livello di servizio Ultra, Premium o Standard. Ciascuno di questi livelli di servizio è progettato per soddisfare diverse esigenze di performance. Astra Control Service supporta tutti e tre.

  È necessario impostare un pool di capacità per ciascun livello di servizio che si desidera utilizzare con i cluster Kubernetes.

  "Scopri di più sui livelli di servizio per Azure NetApp Files".

- Prima di creare un pool di capacità per le applicazioni che si intende proteggere con Astra Control Service, scegliere le prestazioni e la capacità richieste per tali applicazioni.

  Il provisioning della giusta quantità di capacità garantisce agli utenti la possibilità di creare volumi persistenti in base alle esigenze. Se la capacità non è disponibile, non è possibile eseguire il provisioning dei volumi persistenti.

- Un pool di capacità Azure NetApp Files può utilizzare il tipo di QoS manuale o automatico. Astra Control Service supporta i pool di capacità QoS automatici. I pool di capacità QoS manuali non sono supportati.

#### Fase

1. "Seguire la documentazione di Azure NetApp Files per impostare un pool di capacità QoS automatico".

### Delegare una subnet a Azure NetApp Files

È necessario delegare una subnet a Azure NetApp Files in modo che il servizio di controllo Astra possa creare volumi persistenti in tale subnet. Tenere presente che Azure NetApp Files consente di avere una sola subnet delegata in una rete virtuale.

Se si utilizzano reti virtuali peering, entrambi i lati della connessione di peering devono essere online: La rete

virtuale in cui risiedono i cluster Kubernetes e la rete virtuale con la subnet delegata Azure NetApp Files.

**Fase**

1. "Seguire la documentazione di Azure NetApp Files per delegare una subnet a Azure NetApp Files".

**Al termine**

Attendere circa 10 minuti prima di rilevare il cluster in esecuzione nella subnet delegata.

### Creare un'entità del servizio Azure

Astra Control Service richiede un'entità del servizio Azure a cui viene assegnato il ruolo di collaboratore. Astra Control Service utilizza questo principio del servizio per facilitare la gestione dei dati delle applicazioni Kubernetes per conto dell'utente.

Un service principal è un'identità creata appositamente per l'utilizzo con applicazioni, servizi e strumenti. L'assegnazione di un ruolo all'entità del servizio limita l'accesso a risorse Azure specifiche.

Seguire la procedura riportata di seguito per creare un'entità servizio utilizzando la CLI di Azure. Sarà necessario salvare l'output in un file JSON e fornirlo successivamente ad Astra Control Service. "Fare riferimento alla documentazione di Azure per ulteriori dettagli sull'utilizzo della CLI".

I seguenti passaggi presuppongono che si disponga dell'autorizzazione per creare un'entità servizio e che Microsoft Azure SDK (comando az) sia installato sul computer.

**Requisiti**

- L'entità del servizio deve utilizzare l'autenticazione regolare. I certificati non sono supportati.
- All'entità del servizio deve essere concesso l'accesso al tuo abbonamento Azure da parte di Contributor o Owner.
- L'abbonamento o il gruppo di risorse scelto per l'ambito deve contenere i cluster AKS e l'account Azure NetApp Files.

**Fasi**

1. Identificare l'abbonamento e l'ID tenant in cui risiedono i cluster AKS (si tratta dei cluster che si desidera gestire in Astra Control Service).

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

2. Eseguire una delle seguenti operazioni, a seconda che si utilizzi un'intera sottoscrizione o un gruppo di risorse:
   - Creare l'entità del servizio, assegnare il ruolo di collaboratore e specificare l'ambito dell'intera sottoscrizione in cui risiedono i cluster.

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

   - Creare l'entità del servizio, assegnare il ruolo di collaboratore e specificare il gruppo di risorse in cui risiedono i cluster.

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

3. Memorizzare l'output della CLI Azure risultante come file JSON.

   Dovrai fornire questo file in modo che Astra Control Service possa rilevare i tuoi cluster AKS e gestire le operazioni di gestione dei dati Kubernetes. "Scopri di più sulla gestione delle credenziali in Astra Control Service".

4. Facoltativo: Aggiungere l'ID di abbonamento al file JSON in modo che Astra Control Service compili automaticamente l'ID quando si seleziona il file.

   In caso contrario, dovrai inserire l'ID dell'abbonamento in Astra Control Service quando richiesto.

   **Esempio**

```
{
  "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. Facoltativo: Verificare l'entità del servizio. Scegliere tra i seguenti comandi di esempio a seconda dell'ambito utilizzato dall'entità del servizio.

   **Scopo dell'abbonamento**

```
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
```

   **Ambito del gruppo di risorse**

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

#### Opzionale: Configurare la ridondanza per i bucket di backup di Azure

È possibile configurare un livello di ridondanza più duraturo per i bucket di backup di Azure. Per impostazione predefinita, i bucket Astra Control Service utilizzati per memorizzare i backup di Azure Kubernetes Service utilizzano l'opzione di ridondanza LRS (Redundant Storage) locale. Per utilizzare un'opzione di ridondanza più durevole per i bucket Azure, è necessario eseguire le seguenti operazioni:

**Fasi**

1. Creare un account di storage Azure che utilizzi il livello di ridondanza necessario "queste istruzioni".
2. Creare un container Azure nel nuovo account storage utilizzando "queste istruzioni".
3. Aggiungere il container come bucket ad Astra Control Service. Fare riferimento a. "Aggiungere un bucket aggiuntivo".
4. (Facoltativo) per utilizzare il bucket appena creato come bucket predefinito per i backup di Azure, impostarlo come bucket predefinito per Azure. Fare riferimento a. "Modificare il bucket predefinito".

## Configurare Microsoft Azure con dischi gestiti Azure

Sono necessari alcuni passaggi per preparare l'abbonamento a Microsoft Azure prima di poter gestire i cluster di servizi Azure Kubernetes con Astra Control Service. Seguire queste istruzioni se si intende utilizzare i dischi gestiti da Azure come backend di storage.

#### Avvio rapido per la configurazione di Azure

Inizia subito seguendo questi passaggi o scorri verso il basso fino alle restanti sezioni per ottenere informazioni dettagliate.

##### [Uno] Esaminare i requisiti del servizio di controllo Astra per Azure Kubernetes Service

Assicurarsi che i cluster siano integri e che eseguano una versione supportata di Kubernetes, che i pool di nodi siano online e che eseguano Linux e altro ancora. Scopri di più su questo passaggio.

##### [Due] Iscriviti a Microsoft Azure

Creare un account Microsoft Azure. Scopri di più su questo passaggio.

##### [Tre] Creare un'entità del servizio Azure

Creare un'entità del servizio Azure con il ruolo di collaboratore. Scopri di più su questo passaggio.

##### [Quattro] Configurare i dettagli del driver CSI (Container Storage Interface)

È necessario configurare l'abbonamento Azure e il cluster per il funzionamento con i driver CSI. Scopri di più su questo passaggio.

##### [Cinque] Opzionale: Configurare la ridondanza per i bucket di backup di Azure

Per impostazione predefinita, i bucket Astra Control Service utilizzati per memorizzare i backup di Azure Kubernetes Service utilizzano l'opzione di ridondanza LRS (Redundant Storage) locale. Come passaggio opzionale, è possibile configurare un livello di ridondanza più duraturo per i bucket Azure. Scopri di più su questo passaggio.

### Azure Kubernetes Service Cluster Requirements

Un cluster Kubernetes deve soddisfare i seguenti requisiti per consentirne il rilevamento e la gestione da Astra Control Service.

**Versione di Kubernetes**

I cluster devono eseguire Kubernetes versione 1,26 - 1,28.

**Tipo di immagine**

Il tipo di immagine per tutti i pool di nodi deve essere Linux.

**Stato del cluster**

I cluster devono essere in esecuzione in uno stato integro e avere almeno un nodo di lavoro online senza nodi di lavoro in uno stato di errore.

**Regione di Azure**

Come Best practice, è necessario scegliere una regione che supporti Azure NetApp Files, anche se non viene utilizzata come back-end di storage. In questo modo sarà più semplice utilizzare Azure NetApp Files come back-end di storage in futuro se i requisiti di performance cambiano. "Visualizza i prodotti Azure per regione".

**Driver CSI**

I cluster devono avere installati i driver CSI appropriati.

### Iscriviti a Microsoft Azure

Se non disponi di un account Microsoft Azure, inizia iscrivendoti a Microsoft Azure.

**Fasi**

1. Accedere alla "Pagina di iscrizione Azure" Per iscriversi al servizio Azure.
2. Selezionare un piano e seguire le istruzioni per completare l'abbonamento.

### Creare un'entità del servizio Azure

Astra Control Service richiede un'entità del servizio Azure a cui viene assegnato il ruolo di collaboratore. Astra Control Service utilizza questo principio del servizio per facilitare la gestione dei dati delle applicazioni Kubernetes per conto dell'utente.

Un service principal è un'identità creata appositamente per l'utilizzo con applicazioni, servizi e strumenti. L'assegnazione di un ruolo all'entità del servizio limita l'accesso a risorse Azure specifiche.

Seguire la procedura riportata di seguito per creare un'entità servizio utilizzando la CLI di Azure. Sarà necessario salvare l'output in un file JSON e fornirlo successivamente ad Astra Control Service. "Fare riferimento alla documentazione di Azure per ulteriori dettagli sull'utilizzo della CLI".

I seguenti passaggi presuppongono che si disponga dell'autorizzazione per creare un'entità servizio e che Microsoft Azure SDK (comando az) sia installato sul computer.

**Requisiti**

- L'entità del servizio deve utilizzare l'autenticazione regolare. I certificati non sono supportati.
- All'entità del servizio deve essere concesso l'accesso al tuo abbonamento Azure da parte di Contributor o Owner.

- L'abbonamento o il gruppo di risorse scelto per l'ambito deve contenere i cluster AKS e l'account Azure NetApp Files.

**Fasi**

1. Identificare l'abbonamento e l'ID tenant in cui risiedono i cluster AKS (si tratta dei cluster che si desidera gestire in Astra Control Service).

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

2. Eseguire una delle seguenti operazioni, a seconda che si utilizzi un'intera sottoscrizione o un gruppo di risorse:
   - Creare l'entità del servizio, assegnare il ruolo di collaboratore e specificare l'ambito dell'intera sottoscrizione in cui risiedono i cluster.

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

   - Creare l'entità del servizio, assegnare il ruolo di collaboratore e specificare il gruppo di risorse in cui risiedono i cluster.

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

3. Memorizzare l'output della CLI Azure risultante come file JSON.

   Dovrai fornire questo file in modo che Astra Control Service possa rilevare i tuoi cluster AKS e gestire le operazioni di gestione dei dati Kubernetes. "Scopri di più sulla gestione delle credenziali in Astra Control Service".

4. Facoltativo: Aggiungere l'ID di abbonamento al file JSON in modo che Astra Control Service compili automaticamente l'ID quando si seleziona il file.

   In caso contrario, dovrai inserire l'ID dell'abbonamento in Astra Control Service quando richiesto.

   **Esempio**

```
{
  "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. Facoltativo: Verificare l'entità del servizio. Scegliere tra i seguenti comandi di esempio a seconda dell'ambito utilizzato dall'entità del servizio.

   **Scopo dell'abbonamento**

   ```
   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
   ```

   **Ambito del gruppo di risorse**

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

### Configurare i dettagli del driver CSI (Container Storage Interface)

Per utilizzare i dischi gestiti Azure con Astra Control Service, è necessario installare i driver CSI richiesti.

#### Attivare la funzione del driver CSI nell'abbonamento Azure

Prima di installare i driver CSI, è necessario attivare la funzionalità del driver CSI nell'abbonamento Azure.

**Fasi**

1. Aprire l'interfaccia della riga di comando di Azure.
2. Eseguire il seguente comando per registrare il driver:

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

3. Eseguire il seguente comando per assicurarsi che la modifica venga propagata:

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

L'output dovrebbe essere simile a quanto segue:

```
{
"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"
}
```

#### Installare i driver CSI del disco gestito Azure nel cluster Azure Kubernetes Service

È possibile installare i driver di Azure CSI per completare la preparazione.

**Fase**

1. Passare a. "La documentazione del driver Microsoft CSI".
2. Seguire le istruzioni per installare i driver CSI richiesti.

#### Opzionale: Configurare la ridondanza per i bucket di backup di Azure

È possibile configurare un livello di ridondanza più duraturo per i bucket di backup di Azure. Per impostazione predefinita, i bucket Astra Control Service utilizzati per memorizzare i backup di Azure Kubernetes Service utilizzano l'opzione di ridondanza LRS (Redundant Storage) locale. Per utilizzare un'opzione di ridondanza più durevole per i bucket Azure, è necessario eseguire le seguenti operazioni:

**Fasi**

1. Creare un account di storage Azure che utilizzi il livello di ridondanza necessario "queste istruzioni".
2. Creare un container Azure nel nuovo account storage utilizzando "queste istruzioni".
3. Aggiungere il container come bucket ad Astra Control Service. Fare riferimento a. "Aggiungere un bucket aggiuntivo".
4. (Facoltativo) per utilizzare il bucket appena creato come bucket predefinito per i backup di Azure, impostarlo come bucket predefinito per Azure. Fare riferimento a. "Modificare il bucket predefinito".

## Registrati per un account Astra Control Service

Per utilizzare il servizio Astra Control, devi disporre di un account Astra Control Service associato al tuo account NetApp BlueXP. Completa il processo di registrazione ad Astra Control Service e, se non disponi già di un account BlueXP, registrati a BlueXP per accedere ad Astra Control Service.

## Registrati per un account Astra Control

Prima di accedere ad Astra Control Service, è necessario completare un processo di registrazione per ottenere un account Astra Control Service.

Quando utilizzi Astra Control Service, gestirai le tue applicazioni dall'interno di un account. Un account include gli utenti che possono visualizzare e gestire le applicazioni all'interno dell'account, oltre ai dati di fatturazione.

**Fasi**

1. "Visita la pagina Astra Control su BlueXP".
2. Selezionare **Iscriviti al piano gratuito**.
3. Fornire le informazioni richieste nel modulo.

   Durante la compilazione del modulo, è necessario prendere nota di alcuni elementi importanti:

   - Il nome e l'indirizzo della tua azienda devono essere precisi perché li verificheremo per soddisfare i requisiti della Global Trade Compliance.
   - Il nome dell'account * Astra è il nome dell'account Astra Control Service della tua azienda. Questo nome viene visualizzato nell'interfaccia utente di Astra Control Service. Nota: Se necessario, è possibile creare altri account (fino a 5).
   - Nel campo **Indirizzo e-mail aziendale**, se si dispone di un account NetApp BlueXP, immettere qui l'e-mail che si utilizza per tale account. Se non disponi ancora di un account NetApp BlueXP, utilizza l'indirizzo email che inserisci qui quando effettui l'iscrizione ad BlueXP.
4. Selezionare **Crea account**.

## Iscriviti a BlueXP

Il servizio di controllo Astra è integrato nel servizio di autenticazione di NetApp BlueXP. Puoi accedere a NetApp BlueXP usando le tue credenziali del sito di supporto BlueXP o NetApp. Se non disponi già di un account NetApp BlueXP o del sito di supporto NetApp, iscriviti a BlueXP per poter accedere al servizio Astra Control e agli altri servizi cloud di NetApp. Se disponi già di un account BlueXP o sul sito di supporto NetApp e hai completato la registrazione, puoi accedere "Servizio di controllo Astra" Utilizzando direttamente le credenziali del sito di supporto BlueXP o NetApp.

> Puoi anche utilizzare il single sign-on per accedere a BlueXP utilizzando le credenziali della directory aziendale (identità federata). Per ulteriori informazioni, visitare il sito "Centro assistenza" Quindi selezionare **Cloud Central sign-in options** (Opzioni di accesso Cloud Central).

**Fasi**

1. Passare a. "NetApp BlueXP".
2. In alto a destra, seleziona **inizia**.
3. Selezionare **Registrati**.
4. Compila il modulo.

   Assicurati che il numero di telefono e l'indirizzo e-mail inseriti in questo campo siano gli stessi utilizzati nel modulo di registrazione Astra Control precedente.

5. Selezionare **Registrati**.

L'indirizzo email inserito in questi moduli corrisponde al tuo ID utente NetApp BlueXP. Utilizza questo ID utente BlueXP quando effettui l'iscrizione a un nuovo account Astra Control o quando un amministratore Astra Control ti invita a un account Astra Control esistente.

6. Attende un'e-mail da NetApp BlueXP. L'e-mail proviene dall'indirizzo saas.support@netapp.com e potrebbe richiedere alcuni minuti per arrivare. Controllare la cartella spam.
7. All'arrivo del messaggio, selezionare il collegamento nell'e-mail per verificare l'indirizzo e-mail.

**Risultato**

Hai ora un accesso utente BlueXP attivo.

Ora che sei registrato, puoi accedere direttamente a Astra Control usando le tue credenziali BlueXP di https://astra.netapp.io.

# Aggiungere un cluster a Astra Control Service

Dopo aver configurato l'ambiente, è possibile creare un cluster Kubernetes e aggiungerlo ad Astra Control Service. Questo consente di utilizzare Astra Control Service per proteggere le applicazioni sul cluster.

A seconda del tipo di cluster da aggiungere ad Astra Control Service, è necessario utilizzare diversi passaggi per aggiungere il cluster.

- "Aggiungere un cluster gestito da un provider pubblico ad Astra Control Service": Utilizzare questa procedura per aggiungere un cluster con un indirizzo IP pubblico e gestito da un provider cloud. È necessario disporre dell'account Service Principal, dell'account del servizio o dell'account utente per il provider cloud.
- "Aggiungere un cluster gestito da provider privato ad Astra Control Service": Utilizzare questa procedura per aggiungere un cluster che dispone di un indirizzo IP privato ed è gestito da un provider cloud. È necessario disporre dell'account Service Principal, dell'account del servizio o dell'account utente per il provider cloud.
- "Aggiungere un cluster pubblico autogestato ad Astra Control Service": Utilizzare questa procedura per aggiungere un cluster con un indirizzo IP pubblico e gestito dall'organizzazione. Sarà necessario creare un file kubeconfig per il cluster che si desidera aggiungere.
- "Aggiungere un cluster privato autogestato ad Astra Control Service": Utilizzare questa procedura per aggiungere un cluster con un indirizzo IP privato e gestito dall'organizzazione. Sarà necessario creare un file kubeconfig per il cluster che si desidera aggiungere.

## Installa Astra Connector per gestire i cluster

Astra Connector è un software che risiede nei cluster gestiti e facilita la comunicazione tra il cluster gestito e Astra Control. Per i cluster gestiti mediante Astra Control Service, sono disponibili due versioni di Astra Connector:

- **Versione precedente del connettore Astra**: "Installare la versione precedente del connettore Astra" Sul tuo cluster, se intendi gestire il cluster con flussi di lavoro non nativi di Kubernetes.
- [Tech preview] * connettore dichiarativo di Kubernetes Astra*: "Installa Astra Connector per i cluster gestiti con flussi di lavoro Kubernetes dichiarativi" Sul cluster, se si intende gestire il cluster utilizzando flussi di

lavoro Kubernetes dichiarativi. Dopo aver installato Astra Connector sul cluster, il cluster viene aggiunto automaticamente ad Astra Control.

> Il connettore dichiarativo Kubernetes Astra è disponibile solo come parte del programma Astra Control Early Adopter Program (EAP). Per informazioni sulla partecipazione al programma EAP, contattare il rappresentante commerciale NetApp di zona.

#### Installare la versione precedente del connettore Astra

Astra Control Service utilizza la versione precedente di Astra Connector per consentire la comunicazione tra Astra Control Service e i cluster privati gestiti con flussi di lavoro non nativi per Kubernetes. Devi installare Astra Connector su cluster privati che vuoi gestire con flussi di lavoro non nativi di Kubernetes.

La versione precedente di Astra Connector supporta i seguenti tipi di cluster privati gestiti con flussi di lavoro non nativi di Kubernetes:

- Amazon Elastic Kubernetes Service (EKS)
- Servizio Azure Kubernetes (AKS)
- Google Kubernetes Engine (GKE)
- Red Hat OpenShift Service su AWS (ROSA)
- ROSA con AWS PrivateLink
- Piattaforma Red Hat OpenShift Container all'interno dell'hotel

**A proposito di questa attività**

- Per eseguire questi passaggi, esegui questi comandi sul cluster privato che desideri gestire con Astra Control Service.
- Se si utilizza un host Bastion, eseguire questi comandi dalla riga di comando dell'host Bastion.

**Prima di iniziare**

- Devi accedere al cluster privato da gestire con Astra Control Service.
- Sono necessarie autorizzazioni di amministratore di Kubernetes per installare l'operatore Astra Connector sul cluster.

**Fasi**

1. Installa l'operatore Astra Connector precedente sul cluster privato che desideri gestire con flussi di lavoro non nativi di Kubernetes. Quando si esegue questo comando, lo spazio dei nomi `astra-connector-operator` viene creato e la configurazione viene applicata allo spazio dei nomi:

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

2. Verificare che l'operatore sia installato e pronto:

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

3. Ottieni un token API da Astra Control. Fare riferimento a. "Documentazione di Astra Automation" per istruzioni.
4. Creare lo spazio dei nomi astra-Connector:

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

5. Creare il file Astra Connector CR e assegnargli un nome `astra-connector-cr.yaml`. Aggiorna i valori tra parentesi <> per farli corrispondere all'ambiente Astra Control e alla configurazione del cluster:
   - **<ASTRA_CONTROL_SERVICE_URL>**: L'URL dell'interfaccia utente web del servizio di controllo Astra. Ad esempio:

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

   - **<ASTRA_CONTROL_SERVICE_API_TOKEN>**: Il token dell'API di controllo Astra ottenuto nel passaggio precedente.
   - **<PRIVATE_AKS_CLUSTER_NAME>**: (Solo cluster AKS) - il nome del cluster del cluster privato Azure Kubernetes Service. Annullare il commento e popolare questa riga solo se si aggiunge un cluster AKS privato.
   - **<ASTRA_CONTROL_ACCOUNT_ID>**: Ottenuto dall'interfaccia utente web Astra Control. Selezionare l'icona a forma di figura in alto a destra nella pagina e selezionare **accesso API**.

     ```
     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. Dopo aver popolato il `astra-connector-cr.yaml` File con i valori corretti, applicare il CR:

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

7. Verificare che il connettore Astra sia completamente distribuito:

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

8. Verifica che il cluster sia registrato con Astra Control:

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

L'output dovrebbe essere simile a quanto segue:

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

Prendere nota di ASTRACONNECTORID; sarà necessario quando si aggiunge il cluster ad Astra Control.

**Quali sono le prossime novità?**

Una volta installato Astra Connector, puoi aggiungere il cluster privato ad Astra Control Service.

- "Aggiungere un cluster gestito da provider privato ad Astra Control Service": Utilizzare questa procedura per aggiungere un cluster che dispone di un indirizzo IP privato ed è gestito da un provider cloud. È necessario disporre dell'account Service Principal, dell'account del servizio o dell'account utente per il provider cloud.
- "Aggiungere un cluster privato autogestato ad Astra Control Service": Utilizzare questa procedura per aggiungere un cluster con un indirizzo IP privato e gestito dall'organizzazione. Sarà necessario creare un file kubeconfig per il cluster che si desidera aggiungere.

**Per ulteriori informazioni**

- "Aggiungere un cluster"

### (Anteprima tecnica) Installa il connettore dichiarativo di Kubernetes Astra

I cluster gestiti utilizzando flussi di lavoro Kubernetes dichiarativi utilizzano Astra Connector per consentire la comunicazione tra il cluster gestito e Astra Control. Devi installare Astra Connector su tutti i cluster che verranno gestiti con flussi di lavoro Kubernetes dichiarativi.

Viene installato il connettore Astra dichiarativo di Kubernetes utilizzando i comandi di Kubernetes e i file Custom Resource (CR).

**A proposito di questa attività**

- Quando esegui questi passaggi, esegui questi comandi sul cluster che desideri gestire con Astra Control.
- Se si utilizza un host Bastion, eseguire questi comandi dalla riga di comando dell'host Bastion.

**Prima di iniziare**

- Devi accedere al cluster da gestire con Astra Control.
- Sono necessarie autorizzazioni di amministratore di Kubernetes per installare l'operatore Astra Connector sul cluster.

> Se il cluster è configurato con l'imposizione dell'ammissione di sicurezza pod, che è l'impostazione predefinita per i cluster Kubernetes 1,25 e versioni successive, è necessario abilitare le restrizioni PSA sugli spazi dei nomi appropriati. Fare riferimento a. "Prepara il tuo ambiente per la gestione dei cluster utilizzando Astra Control" per istruzioni.

**Fasi**

1. Installare l'operatore Astra Connector sul cluster che si desidera gestire con flussi di lavoro Kubernetes dichiarativi. Quando si esegue questo comando, lo spazio dei nomi `astra-connector-operator` viene creato e la configurazione viene applicata allo spazio dei nomi:

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

2. Verificare che l'operatore sia installato e pronto:

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

3. Ottieni un token API da Astra Control. Fare riferimento a. "Documentazione di Astra Automation" per istruzioni.
4. Creare un segreto utilizzando il token. Sostituisci <API_TOKEN> con il token ricevuto da Astra Control:

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

5. Crea un Docker Secret da usare per estrarre l'immagine di Astra Connector. Sostituire i valori tra parentesi <> con le informazioni dell'ambiente:

> Puoi trovare il <ASTRA_CONTROL_ACCOUNT_ID> nell'interfaccia utente web di Astra Control. Nell'interfaccia utente Web, selezionare l'icona della figura in alto a destra nella pagina e selezionare **accesso API**.

```
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. Creare il file Astra Connector CR e assegnargli un nome `astra-connector-cr.yaml`. Aggiorna i valori tra parentesi <> per farli corrispondere all'ambiente Astra Control e alla configurazione del cluster:
   - <ASTRA_CONTROL_ACCOUNT_ID>: Ottenuto dall'interfaccia utente web Astra Control durante la fase precedente.
   - <CLUSTER_NAME>: Il nome che il cluster deve essere assegnato in Astra Control.
   - <ASTRA_CONTROL_URL>: L'URL dell'interfaccia utente web di Astra Control. Ad esempio:

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

```yaml
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. Dopo aver popolato il `astra-connector-cr.yaml` File con i valori corretti, applicare il CR:

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

8. Verificare che il connettore Astra sia completamente distribuito:

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

9. Verifica che il cluster sia registrato con Astra Control:

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

L'output dovrebbe essere simile a quanto segue:

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

10. Verificare che il cluster compaia nell'elenco dei cluster gestiti nella pagina **cluster** dell'interfaccia utente Web Astra Control.

## Aggiungere un cluster gestito dal provider

### Aggiungere un cluster gestito da un provider pubblico ad Astra Control Service

Dopo aver configurato l'ambiente cloud, sei pronto per creare un cluster Kubernetes e aggiungerlo ad Astra Control Service.

- Creare un cluster Kubernetes
- Aggiungere il cluster ad Astra Control Service
- Modificare la classe di storage predefinita

#### Creare un cluster Kubernetes

Se non si dispone ancora di un cluster, è possibile crearne uno che soddisfi le esigenze "Requisiti del servizio Astra Control per Amazon Elastic Kubernetes Service (EKS)". Se non si dispone ancora di un cluster, è possibile crearne uno che soddisfi le esigenze "Requisiti del servizio Astra Control per Google Kubernetes Engine (GKE)". Se non si dispone ancora di un cluster, è possibile crearne uno che soddisfi le esigenze "Requisiti del servizio di controllo Astra per il servizio Azure Kubernetes (AKS) con Azure NetApp Files" oppure "Requisiti del servizio di controllo Astra per Azure Kubernetes Service (AKS) con dischi gestiti Azure".

> Astra Control Service supporta i cluster AKS che utilizzano Azure Active Directory (Azure ad) per l'autenticazione e la gestione delle identità. Quando si crea il cluster, seguire le istruzioni in "documentazione ufficiale" Per configurare il cluster per l'utilizzo di Azure ad. È necessario assicurarsi che i cluster soddisfino i requisiti per l'integrazione di Azure ad gestita da AKS.

#### Aggiungere il cluster ad Astra Control Service

Dopo aver effettuato l'accesso ad Astra Control Service, il primo passo è iniziare a gestire i cluster. Prima di aggiungere un cluster ad Astra Control Service, è necessario eseguire attività specifiche e assicurarsi che il cluster soddisfi determinati requisiti.

Quando gestisci i cluster Azure Kubernetes Service e Google Kubernetes Engine, tieni presente che hai due opzioni per l'installazione di Astra Control Provivisioner e la gestione del ciclo di vita:

- Puoi utilizzare Astra Control Service per gestire automaticamente il ciclo di vita di Astra Control Provisioner. Per fare questo, assicurati che Astra Trident non sia installato e Astra Control provisioner non sia abilitato nel cluster che vuoi gestire con Astra Control Service. In questo caso, Astra Control Service abilita automaticamente Astra Control Provisioner quando inizi a gestire il cluster e gli aggiornamenti di Astra Control Provisioner vengono gestiti automaticamente.
- Puoi gestire tu stesso il ciclo di vita di Astra Control Provisioner. A tale scopo, abilita Astra Control Provisioner sul cluster prima di gestire il cluster con Astra Control Service. In questo caso, Astra Control Service rileva che Astra Control Provisioner è già abilitato e non lo reinstalla né gestisce gli aggiornamenti di Astra Control Provisioner. Fare riferimento a. "Abilita Astra Control Provisioner" Per i passaggi, abilita Astra Control provisioner.

Se gestisci i cluster di Amazon Web Services con Astra Control Service, se hai bisogno di backend di storage che possono essere utilizzati solo con Astra Control Provivisioner, devi abilitare Astra Control Provivisioner manualmente sul cluster prima di gestirlo con Astra Control Service. Fare riferimento a. "Abilita Astra Control Provisioner" Per informazioni su come attivare Astra Control Provivioner.

**Prima di iniziare**

**Amazon Web Services**

- Il file JSON contiene le credenziali dell'utente IAM che ha creato il cluster. "Scopri come creare un utente IAM".
- Per Amazon FSX per NetApp ONTAP è necessario Astra Control Provisioner. Se intendi usare Amazon FSX per NetApp ONTAP come back-end dello storage per il tuo cluster EKS, fai riferimento alle informazioni Astra Control Provivisioner nel "Requisiti del cluster EKS".
- (Facoltativo) se è necessario fornire `kubectl` Accesso ai comandi per un cluster ad altri utenti IAM che non sono i creatori del cluster, fare riferimento alle istruzioni in "Come posso fornire l'accesso ad altri utenti e ruoli IAM dopo la creazione del cluster in Amazon EKS?".
- Se intendi utilizzare NetApp Cloud Volumes ONTAP come backend di storage, devi configurare Cloud Volumes ONTAP per l'utilizzo con Amazon Web Services. Fare riferimento alla Cloud Volumes ONTAP "documentazione di installazione".

**Microsoft Azure**

- Il file JSON che contiene l'output della CLI di Azure deve essere presente al momento della creazione dell'entità del servizio. "Scopri come configurare un service principal".

  Avrai inoltre bisogno del tuo ID di abbonamento Azure, se non lo hai aggiunto al file JSON.

- Se si intende utilizzare NetApp Cloud Volumes ONTAP come back-end per lo storage, è necessario configurare Cloud Volumes ONTAP per l'utilizzo con Microsoft Azure. Fare riferimento alla Cloud Volumes ONTAP "documentazione di installazione".

**Google Cloud**

- È necessario disporre del file della chiave dell'account di servizio per un account di servizio che dispone delle autorizzazioni necessarie. "Scopri come configurare un account di servizio".
- Se si intende utilizzare NetApp Cloud Volumes ONTAP come back-end per lo storage, è necessario configurare Cloud Volumes ONTAP per l'utilizzo con Google Cloud. Fare riferimento alla Cloud Volumes ONTAP "documentazione di installazione".

**Fasi**

1. (Facoltativo) se stai aggiungendo un cluster Amazon EKS o vuoi gestire da solo l'installazione e gli aggiornamenti di Astra Control Provivisioner, abilita Astra Control Provivisioner sul cluster. Fare riferimento a. "Abilita Astra Control Provisioner" per i passaggi di abilitazione.
2. Aprire l'interfaccia utente Web di Astra Control Service in un browser.
3. Nella dashboard, selezionare **Manage Kubernetes cluster** (Gestisci cluster Kubernetes).

   Seguire le istruzioni per aggiungere il cluster.

4. **Provider**: Seleziona il tuo cloud provider e fornisci le credenziali necessarie per creare una nuova istanza di cloud oppure seleziona un'istanza di cloud esistente da utilizzare.
5. **Amazon Web Services**: Fornisci i dettagli del tuo account utente IAM Amazon Web Services caricando un file JSON o incollando il contenuto del file JSON dagli Appunti.

   Il file JSON deve contenere le credenziali dell'utente IAM che ha creato il cluster.

6. **Microsoft Azure**: Fornisci dettagli sull'entità del servizio Azure caricando un file JSON o incollando il contenuto di tale file JSON dagli Appunti.

   Il file JSON deve contenere l'output dell'interfaccia CLI di Azure al momento della creazione dell'entità del servizio. Può anche includere il tuo ID di abbonamento per aggiungerlo automaticamente ad Astra. In caso contrario, è necessario inserire manualmente l'ID dopo aver fornito il codice JSON.

7. **Google Cloud Platform**: Fornire il file delle chiavi dell'account di servizio caricando il file o incollando il contenuto dagli Appunti.

   Astra Control Service utilizza l'account del servizio per rilevare i cluster in esecuzione in Google Kubernetes Engine.

8. **Altro**: Questa scheda è destinata solo ai cluster a gestione automatica.
   a. **Nome istanza cloud**: Fornire un nome per la nuova istanza cloud che verrà creata quando si aggiunge questo cluster. Scopri di più "istanze cloud".
   b. Selezionare **Avanti**.

      Astra Control Service visualizza un elenco di cluster tra i quali è possibile scegliere.

   c. **Cluster**: Selezionare un cluster dall'elenco da aggiungere ad Astra Control Service.

      > Durante la selezione dall'elenco dei cluster, prestare attenzione alla colonna **Eligiblity**. Se un cluster è "non idoneo" o "parzialmente idoneo", passare il mouse sullo stato per determinare se si è verificato un problema con il cluster. Ad esempio, potrebbe identificare che il cluster non dispone di un nodo di lavoro.

   d. Selezionare **Avanti**.
   e. (Facoltativo) **Storage**: Facoltativamente, selezionare la classe di storage che si desidera utilizzare per impostazione predefinita per le applicazioni Kubernetes distribuite in questo cluster.
9. Per selezionare una nuova classe di storage predefinita per il cluster, attivare la casella di controllo **Assegna una nuova classe di storage predefinita**.
10. Selezionare una nuova classe di storage predefinita dall'elenco.

Ogni servizio di storage del cloud provider visualizza le seguenti informazioni su prezzo, performance e resilienza:

- Cloud Volumes Service per Google Cloud: Informazioni su prezzi, performance e resilienza
- Google Persistent Disk: Non sono disponibili informazioni su prezzi, performance o resilienza
- Azure NetApp Files: Informazioni su performance e resilienza
- Dischi gestiti Azure: Non sono disponibili informazioni su prezzi, performance o resilienza
- Amazon Elastic Block Store: Nessuna informazione su prezzi, performance o resilienza disponibile
- Amazon FSX per NetApp ONTAP: Nessuna informazione su prezzi, performance o resilienza disponibile
- NetApp Cloud Volumes ONTAP: Non sono disponibili informazioni su prezzi, performance o resilienza

Ogni classe di storage può utilizzare uno dei seguenti servizi:

- "Cloud Volumes Service per Google Cloud"
- "Disco persistente di Google"
  - "Azure NetApp Files"
  - "Dischi gestiti da Azure"
  - "Amazon Elastic Block Store"
  - "Amazon FSX per NetApp ONTAP"
  - "NetApp Cloud Volumes ONTAP"

  Scopri di più "Classi di storage per cluster Amazon Web Services". Scopri di più "Classi di storage per cluster AKS". Scopri di più "Classi di storage per cluster GKE".

  a. Selezionare **Avanti**.
  b. **Review & Approve** (Rivedi e approva): Verifica dei dettagli della configurazione.
  c. Selezionare **Add** per aggiungere il cluster ad Astra Control Service.

**Risultato**

Se si tratta del primo cluster aggiunto per questo provider cloud, Astra Control Service crea un archivio di oggetti per il provider cloud per i backup delle applicazioni in esecuzione sui cluster idonei. (Quando si aggiungono cluster successivi per questo provider cloud, non vengono creati ulteriori archivi di oggetti). Se è stata specificata una classe di storage predefinita, Astra Control Service imposta la classe di storage predefinita specificata. Per i cluster gestiti in Amazon Web Services o Google Cloud Platform, Astra Control Service crea anche un account admin sul cluster. Queste operazioni possono richiedere alcuni minuti.

#### Modificare la classe di storage predefinita

È possibile modificare la classe di storage predefinita per un cluster.

### Modificare la classe di storage predefinita utilizzando Astra Control

È possibile modificare la classe di storage predefinita per un cluster da Astra Control. Se il cluster utilizza un servizio backend di storage precedentemente installato, potrebbe non essere possibile utilizzare questo metodo per modificare la classe di storage predefinita (l'azione **Set as default** non è selezionabile). In questo caso, è possibile Modificare la classe di storage predefinita utilizzando la riga di comando.

#### Fasi

1. Nell'interfaccia utente di Astra Control Service, selezionare **Clusters**.
2. Nella pagina **Clusters**, selezionare il cluster che si desidera modificare.
3. Selezionare la scheda **Storage**.
4. Selezionare la categoria **classi di storage**.
5. Selezionare il menu **azioni** per la classe di storage che si desidera impostare come predefinita.
6. Selezionare **Imposta come predefinito**.

### Modificare la classe di storage predefinita utilizzando la riga di comando

È possibile modificare la classe di storage predefinita per un cluster utilizzando i comandi Kubernetes. Questo metodo funziona indipendentemente dalla configurazione del cluster.

#### Fasi

1. Accedere al cluster Kubernetes.
2. Elencare le classi di storage nel cluster:

```
kubectl get storageclass
```

3. Rimuovere la designazione predefinita dalla classe di storage predefinita. Sostituire <SC_NAME> con il nome della classe di storage:

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

4. Contrassegna una classe di storage diversa come predefinita. Sostituire <SC_NAME> con il nome della classe di storage:

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

5. Confermare la nuova classe di storage predefinita:

```
kubectl get storageclass
```

#### Aggiungere un cluster gestito da provider privato ad Astra Control Service

Puoi utilizzare Astra Control Service per gestire cluster privati di Google Kubernetes Engine (GKE). Queste istruzioni presuppongono che sia già stato creato un cluster AKS o OpenShift privato e che sia stato preparato un metodo sicuro per accedervi in remoto; per ulteriori informazioni sulla creazione e l'accesso a cluster AKS o OpenShift privati, fare riferimento alla seguente documentazione:

- "Documentazione di Azure per cluster AKS privati"
- "Documentazione di Azure per cluster OpenShift privati"

Puoi utilizzare Astra Control Service per gestire cluster privati Azure Kubernetes Service (AKS) e cluster privati Red Hat OpenShift in AKS. Queste istruzioni presuppongono che sia già stato creato un cluster AKS o OpenShift privato e che sia stato preparato un metodo sicuro per accedervi in remoto; per ulteriori informazioni sulla creazione e l'accesso a cluster AKS o OpenShift privati, fare riferimento alla seguente documentazione:

- "Documentazione di Azure per cluster AKS privati"
- "Documentazione di Azure per cluster OpenShift privati"

Puoi utilizzare Astra Control Service per gestire cluster privati Amazon Elastic Kubernetes Service (EKS). Queste istruzioni presuppongono che sia già stato creato un cluster EKS privato e che sia stato preparato un metodo sicuro per accedervi in remoto; per ulteriori informazioni sulla creazione e l'accesso a cluster EKS privati, fare riferimento a. "Documentazione Amazon EKS".

Per aggiungere il cluster privato ad Astra Control Service, è necessario eseguire le seguenti operazioni:

1. Installare il connettore Astra
2. Configurare lo storage persistente
3. Aggiungere il cluster gestito dal provider privato ad Astra Control Service

##### Installare il connettore Astra

Prima di aggiungere un cluster privato, devi installare Astra Connector sul cluster in modo che Astra Control possa comunicare con esso. Fare riferimento a. "Installa la versione precedente di Astra Connector per cluster privati gestiti con flussi di lavoro non nativi di Kubernetes" per istruzioni.

##### Configurare lo storage persistente

Configurare lo storage persistente per il cluster. Fare riferimento alla documentazione introduttiva per ulteriori informazioni sulla configurazione dello storage persistente:

- "Configurare Microsoft Azure con Azure NetApp Files"
- "Configurare Microsoft Azure con dischi gestiti Azure"
- "Configurare Amazon Web Services"
- "Configurare Google Cloud"

##### Aggiungere il cluster gestito dal provider privato ad Astra Control Service

È ora possibile aggiungere il cluster privato ad Astra Control Service.

Quando gestisci i cluster Azure Kubernetes Service e Google Kubernetes Engine, tieni presente che hai due

opzioni per l'installazione di Astra Control Provivisioner e la gestione del ciclo di vita:

- Puoi utilizzare Astra Control Service per gestire automaticamente il ciclo di vita di Astra Control Provisioner. Per fare questo, assicurati che Astra Trident non sia installato e Astra Control provisioner non sia abilitato nel cluster che vuoi gestire con Astra Control Service. In questo caso, Astra Control Service abilita automaticamente Astra Control Provisioner quando inizi a gestire il cluster e gli aggiornamenti di Astra Control Provisioner vengono gestiti automaticamente.
- Puoi gestire tu stesso il ciclo di vita di Astra Control Provisioner. A tale scopo, abilita Astra Control Provisioner sul cluster prima di gestire il cluster con Astra Control Service. In questo caso, Astra Control Service rileva che Astra Control Provisioner è già abilitato e non lo reinstalla né gestisce gli aggiornamenti di Astra Control Provisioner. Fare riferimento a. "Abilita Astra Control Provisioner" Per i passaggi, abilita Astra Control provisioner.

Se gestisci i cluster di Amazon Web Services con Astra Control Service, se hai bisogno di backend di storage che possono essere utilizzati solo con Astra Control Provivisioner, devi abilitare Astra Control Provivisioner manualmente sul cluster prima di gestirlo con Astra Control Service. Fare riferimento a. "Abilita Astra Control Provisioner" Per informazioni su come attivare Astra Control Provivioner.

**Prima di iniziare**

**Amazon Web Services**

- Il file JSON contiene le credenziali dell'utente IAM che ha creato il cluster. "Scopri come creare un utente IAM".
- Per Amazon FSX per NetApp ONTAP è necessario Astra Control Provisioner. Se intendi usare Amazon FSX per NetApp ONTAP come back-end dello storage per il tuo cluster EKS, fai riferimento alle informazioni Astra Control Provivisioner nel "Requisiti del cluster EKS".
- (Facoltativo) se è necessario fornire `kubectl` Accesso ai comandi per un cluster ad altri utenti IAM che non sono i creatori del cluster, fare riferimento alle istruzioni in "Come posso fornire l'accesso ad altri utenti e ruoli IAM dopo la creazione del cluster in Amazon EKS?".
- Se intendi utilizzare NetApp Cloud Volumes ONTAP come backend di storage, devi configurare Cloud Volumes ONTAP per l'utilizzo con Amazon Web Services. Fare riferimento alla Cloud Volumes ONTAP "documentazione di installazione".

**Microsoft Azure**

- Il file JSON che contiene l'output della CLI di Azure deve essere presente al momento della creazione dell'entità del servizio. "Scopri come configurare un service principal".

  Avrai inoltre bisogno del tuo ID di abbonamento Azure, se non lo hai aggiunto al file JSON.

- Se si intende utilizzare NetApp Cloud Volumes ONTAP come back-end per lo storage, è necessario configurare Cloud Volumes ONTAP per l'utilizzo con Microsoft Azure. Fare riferimento alla Cloud Volumes ONTAP "documentazione di installazione".

**Google Cloud**

- È necessario disporre del file della chiave dell'account di servizio per un account di servizio che dispone delle autorizzazioni necessarie. "Scopri come configurare un account di servizio".
- Se il cluster è privato, il "reti autorizzate" Deve consentire l'indirizzo IP di Astra Control Service:

  52.188.218.166/32

- Se si intende utilizzare NetApp Cloud Volumes ONTAP come back-end per lo storage, è necessario configurare Cloud Volumes ONTAP per l'utilizzo con Google Cloud. Fare riferimento alla Cloud Volumes ONTAP "documentazione di installazione".

**Fasi**

1. (Facoltativo) se stai aggiungendo un cluster Amazon EKS o vuoi gestire da solo l'installazione e gli aggiornamenti di Astra Control Provivisioner, abilita Astra Control Provivisioner sul cluster. Fare riferimento a. "Abilita Astra Control Provisioner" per i passaggi di abilitazione.
2. Aprire l'interfaccia utente Web di Astra Control Service in un browser.
3. Nella dashboard, selezionare **Manage Kubernetes cluster** (Gestisci cluster Kubernetes).

   Seguire le istruzioni per aggiungere il cluster.

4. **Provider**: Seleziona il tuo cloud provider e fornisci le credenziali necessarie per creare una nuova istanza di cloud oppure seleziona un'istanza di cloud esistente da utilizzare.
5. **Amazon Web Services**: Fornisci i dettagli del tuo account utente IAM Amazon Web Services caricando un file JSON o incollando il contenuto del file JSON dagli Appunti.

Il file JSON deve contenere le credenziali dell'utente IAM che ha creato il cluster.

6. **Microsoft Azure**: Fornisci dettagli sull'entità del servizio Azure caricando un file JSON o incollando il contenuto di tale file JSON dagli Appunti.

   Il file JSON deve contenere l'output dell'interfaccia CLI di Azure al momento della creazione dell'entità del servizio. Può anche includere il tuo ID di abbonamento per aggiungerlo automaticamente ad Astra. In caso contrario, è necessario inserire manualmente l'ID dopo aver fornito il codice JSON.

7. **Google Cloud Platform**: Fornire il file delle chiavi dell'account di servizio caricando il file o incollando il contenuto dagli Appunti.

   Astra Control Service utilizza l'account del servizio per rilevare i cluster in esecuzione in Google Kubernetes Engine.

8. **Altro**: Questa scheda è destinata solo ai cluster a gestione automatica.
   a. **Nome istanza cloud**: Fornire un nome per la nuova istanza cloud che verrà creata quando si aggiunge questo cluster. Scopri di più "istanze cloud".
   b. Selezionare **Avanti**.

      Astra Control Service visualizza un elenco di cluster tra i quali è possibile scegliere.

   c. **Cluster**: Selezionare un cluster dall'elenco da aggiungere ad Astra Control Service.

      Durante la selezione dall'elenco dei cluster, prestare attenzione alla colonna **Eligiblity**. Se un cluster è "non idoneo" o "parzialmente idoneo", passare il mouse sullo stato per determinare se si è verificato un problema con il cluster. Ad esempio, potrebbe identificare che il cluster non dispone di un nodo di lavoro.

9. Selezionare **Avanti**.
10. (Facoltativo) **Storage**: Facoltativamente, selezionare la classe di storage che si desidera utilizzare per impostazione predefinita per le applicazioni Kubernetes distribuite in questo cluster.
    a. Per selezionare una nuova classe di storage predefinita per il cluster, attivare la casella di controllo **Assegna una nuova classe di storage predefinita**.
    b. Selezionare una nuova classe di storage predefinita dall'elenco.

Ogni servizio di storage del cloud provider visualizza le seguenti informazioni su prezzo, performance e resilienza:

- Cloud Volumes Service per Google Cloud: Informazioni su prezzi, performance e resilienza
- Google Persistent Disk: Non sono disponibili informazioni su prezzi, performance o resilienza
- Azure NetApp Files: Informazioni su performance e resilienza
- Dischi gestiti Azure: Non sono disponibili informazioni su prezzi, performance o resilienza
- Amazon Elastic Block Store: Nessuna informazione su prezzi, performance o resilienza disponibile
- Amazon FSX per NetApp ONTAP: Nessuna informazione su prezzi, performance o resilienza disponibile
- NetApp Cloud Volumes ONTAP: Non sono disponibili informazioni su prezzi, performance o resilienza

Ogni classe di storage può utilizzare uno dei seguenti servizi:

- "Cloud Volumes Service per Google Cloud"
- "Disco persistente di Google"
- "Azure NetApp Files"
- "Dischi gestiti da Azure"
- "Amazon Elastic Block Store"
- "Amazon FSX per NetApp ONTAP"
- "NetApp Cloud Volumes ONTAP"

Scopri di più "Classi di storage per cluster Amazon Web Services". Scopri di più "Classi di storage per cluster AKS". Scopri di più "Classi di storage per cluster GKE".

c. Selezionare **Avanti**.

d. **Review & Approve** (Rivedi e approva): Verifica dei dettagli della configurazione.

e. Selezionare **Add** per aggiungere il cluster ad Astra Control Service.

**Risultato**

Se si tratta del primo cluster aggiunto per questo provider cloud, Astra Control Service crea un archivio di oggetti per il provider cloud per i backup delle applicazioni in esecuzione sui cluster idonei. (Quando si aggiungono cluster successivi per questo provider cloud, non vengono creati ulteriori archivi di oggetti). Se è stata specificata una classe di storage predefinita, Astra Control Service imposta la classe di storage predefinita specificata. Per i cluster gestiti in Amazon Web Services o Google Cloud Platform, Astra Control Service crea anche un account admin sul cluster. Queste operazioni possono richiedere alcuni minuti.

#### Modificare la classe di storage predefinita

È possibile modificare la classe di storage predefinita per un cluster.

### Modificare la classe di storage predefinita utilizzando Astra Control

È possibile modificare la classe di storage predefinita per un cluster da Astra Control. Se il cluster utilizza un servizio backend di storage precedentemente installato, potrebbe non essere possibile utilizzare questo metodo per modificare la classe di storage predefinita (l'azione **Set as default** non è selezionabile). In questo caso, è possibile Modificare la classe di storage predefinita utilizzando la riga di comando.

#### Fasi

1. Nell'interfaccia utente di Astra Control Service, selezionare **Clusters**.
2. Nella pagina **Clusters**, selezionare il cluster che si desidera modificare.
3. Selezionare la scheda **Storage**.
4. Selezionare la categoria **classi di storage**.
5. Selezionare il menu **azioni** per la classe di storage che si desidera impostare come predefinita.
6. Selezionare **Imposta come predefinito**.

### Modificare la classe di storage predefinita utilizzando la riga di comando

È possibile modificare la classe di storage predefinita per un cluster utilizzando i comandi Kubernetes. Questo metodo funziona indipendentemente dalla configurazione del cluster.

#### Fasi

1. Accedere al cluster Kubernetes.
2. Elencare le classi di storage nel cluster:

```
kubectl get storageclass
```

3. Rimuovere la designazione predefinita dalla classe di storage predefinita. Sostituire <SC_NAME> con il nome della classe di storage:

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

4. Contrassegna una classe di storage diversa come predefinita. Sostituire <SC_NAME> con il nome della classe di storage:

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

5. Confermare la nuova classe di storage predefinita:

```
kubectl get storageclass
```

## Aggiungere un cluster a gestione automatica

### Aggiungere un cluster pubblico autogestato ad Astra Control Service

Dopo aver configurato l'ambiente, è possibile creare un cluster Kubernetes e aggiungerlo ad Astra Control Service.

Un cluster a gestione automatica è un cluster che viene direttamente provisioning e gestione. Astra Control Service supporta cluster autogestiti eseguiti in un ambiente di cloud pubblico. È possibile aggiungere un cluster a gestione automatica ad Astra Control Service caricando un `kubeconfig.yaml` file. È necessario assicurarsi che il cluster soddisfi i requisiti descritti di seguito.

#### Distribuzioni Kubernetes supportate

È possibile utilizzare Astra Control Service per gestire i seguenti tipi di cluster pubblici e autogestati:

| Distribuzione Kubernetes | Versioni supportate |
|---|---|
| Kubernetes (upstream) | da 1,27 a 1,29 |
| Rancher Kubernetes Engine (RKE) | RKE 1: Versioni 1.24.17, 1.25.13, 1.26.8 con Rancher Manager 2.7.9<br>RKE 2: Versioni 1.23.16 e 1.24.13 con Rancher Manager 2.6.13<br>RKE 2: Versioni 1.24.17, 1.25.14, 1.26.9 con Rancher Manager 2.7.9 |
| Red Hat OpenShift Container Platform | da 4,12 a 4,14 |

Queste istruzioni presuppongono che sia già stato creato un cluster a gestione automatica.

- Aggiungere il cluster ad Astra Control Service
- Modificare la classe di storage predefinita

#### Aggiungere il cluster ad Astra Control Service

Dopo aver effettuato l'accesso ad Astra Control Service, il primo passo è iniziare a gestire i cluster. Prima di aggiungere un cluster ad Astra Control Service, è necessario eseguire attività specifiche e assicurarsi che il cluster soddisfi determinati requisiti.

**Prima di iniziare**

Un cluster a gestione automatica è un cluster che viene direttamente provisioning e gestione. Astra Control Service supporta cluster autogestiti eseguiti in un ambiente di cloud pubblico. I tuoi cluster a gestione autonoma possono utilizzare Astra Control Provivisioner per interfacciarsi con i servizi storage NetApp, oppure possono utilizzare driver Container Storage Interface (CSI) per interfacciarsi con Amazon Elastic Block Store (EBS), Azure Managed Disks e Google Persistent Disk.

Astra Control Service supporta cluster autogestiti che utilizzano le seguenti distribuzioni Kubernetes:

- Red Hat OpenShift Container Platform
- Motore di rancher Kubernetes
- Kubernetes upstream

Il cluster a gestione automatica deve soddisfare i seguenti requisiti:

- Il cluster deve essere accessibile via Internet.
- Se si utilizza o si prevede di utilizzare lo storage abilitato con i driver CSI, i driver CSI appropriati devono essere installati sul cluster. Per ulteriori informazioni sull'utilizzo dei driver CSI per integrare lo storage, consultare la documentazione del servizio di storage.
- Si ha accesso al file cluster kubeconfig che include un solo elemento di contesto. Segui "queste istruzioni" per generare un file kubeconfig.
- Se si aggiunge il cluster utilizzando un file kubeconfig che fa riferimento a un'autorità di certificazione privata, aggiungere la riga seguente al `cluster` sezione del file kubeconfig. In questo modo si permette ad Astra Control di aggiungere il cluster:

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

- **Solo Rancher**: Quando si gestiscono i cluster di applicazioni in un ambiente Rancher, modificare il contesto predefinito del cluster di applicazioni nel file kubeconfig fornito da Rancher per utilizzare un contesto del piano di controllo invece del contesto del server API Rancher. In questo modo si riduce il carico sul server API Rancher e si migliorano le performance.
- **Requisiti di Astra Control Provisioner**: Dovresti avere un Astra Control Provisioner configurato correttamente, inclusi i suoi componenti Astra Trident, per gestire i cluster.
  - **Rivedi i requisiti dell'ambiente Astra Trident**: Prima di installare o aggiornare Astra Control provisioner, consulta "frontend, backend e configurazioni host supportati".
  - **Abilitare la funzionalità Astra Control Provisioner**: Si consiglia vivamente di installare Astra Trident 23,10 o versione successiva e di abilitare "Astra Control Provivisioner funzionalità di storage avanzate". Nelle prossime release, Astra Control non supporterà Astra Trident se anche Astra Control Provisioner non è abilitato.
  - **Configurare un backend di archiviazione**: Deve essere presente almeno un backend di archiviazione "Configurato in Astra Trident" sul cluster.
  - **Configurare una classe di archiviazione**: Deve essere presente almeno una classe di archiviazione "Configurato in Astra Trident" sul cluster. Se è configurata una classe di archiviazione predefinita, assicurarsi che sia la classe di archiviazione **only** con l'annotazione predefinita.
  - **Configurare un controller snapshot volume e installare una classe snapshot volume**: "Installare un controller per lo snapshot del volume" In modo che le snapshot possano essere

create in Astra Control. "Creare" almeno uno `VolumeSnapshotClass` Utilizzando Astra Trident.

**Fasi**

1. Nella dashboard, selezionare **Manage Kubernetes cluster** (Gestisci cluster Kubernetes).

   Seguire le istruzioni per aggiungere il cluster.

2. **Provider**: Selezionare la scheda **Other** per aggiungere dettagli sul cluster a gestione automatica.
   a. **Altro**: Fornisci dettagli sul tuo cluster autogestito caricando un `kubeconfig.yaml` o incollando il contenuto di `kubeconfig.yaml` file dagli appunti.

   > Se crei il tuo `kubeconfig` file, è necessario definire solo **un** elemento di contesto al suo interno. Fare riferimento a. "Documentazione Kubernetes" per informazioni sulla creazione `kubeconfig` file.

3. **Nome credenziale**: Fornire un nome per la credenziale del cluster a gestione automatica che si sta caricando su Astra Control. Per impostazione predefinita, il nome della credenziale viene compilato automaticamente come nome del cluster.
4. **Private route identifier**: Questo campo può essere utilizzato solo con cluster privati.
5. Selezionare **Avanti**.
6. (Facoltativo) **Storage**: Facoltativamente, selezionare la classe di storage che si desidera utilizzare per impostazione predefinita per le applicazioni Kubernetes distribuite in questo cluster.
   a. Per selezionare una nuova classe di storage predefinita per il cluster, attivare la casella di controllo **Assegna una nuova classe di storage predefinita**.
   b. Selezionare una nuova classe di storage predefinita dall'elenco.

   > Ogni servizio di storage del cloud provider visualizza le seguenti informazioni su prezzo, performance e resilienza:
   >
   > - Cloud Volumes Service per Google Cloud: Informazioni su prezzi, performance e resilienza
   > - Google Persistent Disk: Non sono disponibili informazioni su prezzi, performance o resilienza
   > - Azure NetApp Files: Informazioni su performance e resilienza
   > - Dischi gestiti Azure: Non sono disponibili informazioni su prezzi, performance o resilienza
   > - Amazon Elastic Block Store: Nessuna informazione su prezzi, performance o resilienza disponibile
   > - Amazon FSX per NetApp ONTAP: Nessuna informazione su prezzi, performance o resilienza disponibile
   > - NetApp Cloud Volumes ONTAP: Non sono disponibili informazioni su prezzi, performance o resilienza

   Ogni classe di storage può utilizzare uno dei seguenti servizi:

   - "Cloud Volumes Service per Google Cloud"

- "Disco persistente di Google"
  - "Azure NetApp Files"
  - "Dischi gestiti da Azure"
  - "Amazon Elastic Block Store"
  - "Amazon FSX per NetApp ONTAP"
  - "NetApp Cloud Volumes ONTAP"

    Scopri di più "Classi di storage per cluster Amazon Web Services". Scopri di più "Classi di storage per cluster AKS". Scopri di più "Classi di storage per cluster GKE".

c. Selezionare **Avanti**.

d. **Review & Approve** (Rivedi e approva): Verifica dei dettagli della configurazione.

e. Selezionare **Add** per aggiungere il cluster ad Astra Control Service.

#### Modificare la classe di storage predefinita

È possibile modificare la classe di storage predefinita per un cluster.

### Modificare la classe di storage predefinita utilizzando Astra Control

È possibile modificare la classe di storage predefinita per un cluster da Astra Control. Se il cluster utilizza un servizio backend di storage precedentemente installato, potrebbe non essere possibile utilizzare questo metodo per modificare la classe di storage predefinita (l'azione **Set as default** non è selezionabile). In questo caso, è possibile Modificare la classe di storage predefinita utilizzando la riga di comando.

#### Fasi

1. Nell'interfaccia utente di Astra Control Service, selezionare **Clusters**.
2. Nella pagina **Clusters**, selezionare il cluster che si desidera modificare.
3. Selezionare la scheda **Storage**.
4. Selezionare la categoria **classi di storage**.
5. Selezionare il menu **azioni** per la classe di storage che si desidera impostare come predefinita.
6. Selezionare **Imposta come predefinito**.

### Modificare la classe di storage predefinita utilizzando la riga di comando

È possibile modificare la classe di storage predefinita per un cluster utilizzando i comandi Kubernetes. Questo metodo funziona indipendentemente dalla configurazione del cluster.

#### Fasi

1. Accedere al cluster Kubernetes.
2. Elencare le classi di storage nel cluster:

```
kubectl get storageclass
```

3. Rimuovere la designazione predefinita dalla classe di storage predefinita. Sostituire <SC_NAME> con il nome della classe di storage:

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

4. Contrassegna una classe di storage diversa come predefinita. Sostituire <SC_NAME> con il nome della classe di storage:

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

5. Confermare la nuova classe di storage predefinita:

```
kubectl get storageclass
```

### Aggiungere un cluster privato autogestato ad Astra Control Service

Dopo aver configurato l'ambiente, è possibile creare un cluster Kubernetes e aggiungerlo ad Astra Control Service.

Un cluster a gestione automatica è un cluster che viene direttamente provisioning e gestione. Astra Control Service supporta cluster autogestiti eseguiti in un ambiente di cloud pubblico. È possibile aggiungere un cluster a gestione automatica ad Astra Control Service caricando un `kubeconfig.yaml` file. È necessario assicurarsi che il cluster soddisfi i requisiti descritti di seguito.

#### Distribuzioni Kubernetes supportate

È possibile utilizzare Astra Control Service per gestire i seguenti tipi di cluster privati a gestione automatica:

| Distribuzione Kubernetes | Versioni supportate |
|---|---|
| Kubernetes (upstream) | da 1,27 a 1,29 |
| Rancher Kubernetes Engine (RKE) | RKE 1: Versioni 1.24.17, 1.25.13, 1.26.8 con Rancher Manager 2.7.9<br>RKE 2: Versioni 1.23.16 e 1.24.13 con Rancher Manager 2.6.13<br>RKE 2: Versioni 1.24.17, 1.25.14, 1.26.9 con Rancher Manager 2.7.9 |
| Red Hat OpenShift Container Platform | da 4,12 a 4,14 |

Queste istruzioni presuppongono che sia già stato creato un cluster privato e che sia stato preparato un metodo sicuro per accedervi in remoto.

Per aggiungere il cluster privato ad Astra Control Service, è necessario eseguire le seguenti operazioni:

1. Installare il connettore Astra
2. Configurare lo storage persistente

3. Aggiungere il cluster privato autogestato ad Astra Control Service

#### Installare il connettore Astra

Prima di aggiungere un cluster privato, devi installare Astra Connector sul cluster in modo che Astra Control possa comunicare con esso. Fare riferimento a. "Installa la versione precedente di Astra Connector per cluster privati gestiti con flussi di lavoro non nativi di Kubernetes" per istruzioni.

#### Configurare lo storage persistente

Configurare lo storage persistente per il cluster. Fare riferimento alla documentazione introduttiva per ulteriori informazioni sulla configurazione dello storage persistente:

- "Configurare Microsoft Azure con Azure NetApp Files"
- "Configurare Microsoft Azure con dischi gestiti Azure"
- "Configurare Amazon Web Services"
- "Configurare Google Cloud"

#### Aggiungere il cluster privato autogestato ad Astra Control Service

È ora possibile aggiungere il cluster privato ad Astra Control Service.

**Prima di iniziare**

Un cluster a gestione automatica è un cluster che viene direttamente provisioning e gestione. Astra Control Service supporta cluster autogestiti eseguiti in un ambiente di cloud pubblico. I tuoi cluster a gestione autonoma possono utilizzare Astra Control Provivisioner per interfacciarsi con i servizi storage NetApp, oppure possono utilizzare driver Container Storage Interface (CSI) per interfacciarsi con Amazon Elastic Block Store (EBS), Azure Managed Disks e Google Persistent Disk.

Astra Control Service supporta cluster autogestiti che utilizzano le seguenti distribuzioni Kubernetes:

- Red Hat OpenShift Container Platform
- Motore di rancher Kubernetes
- Kubernetes upstream

Il cluster a gestione automatica deve soddisfare i seguenti requisiti:

- Il cluster deve essere accessibile via Internet.
- Se si utilizza o si prevede di utilizzare lo storage abilitato con i driver CSI, i driver CSI appropriati devono essere installati sul cluster. Per ulteriori informazioni sull'utilizzo dei driver CSI per integrare lo storage, consultare la documentazione del servizio di storage.
- Si ha accesso al file cluster kubeconfig che include un solo elemento di contesto. Segui "queste istruzioni" per generare un file kubeconfig.
- Se si aggiunge il cluster utilizzando un file kubeconfig che fa riferimento a un'autorità di certificazione privata, aggiungere la riga seguente al `cluster` sezione del file kubeconfig. In questo modo si permette ad Astra Control di aggiungere il cluster:

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

- **Solo Rancher**: Quando si gestiscono i cluster di applicazioni in un ambiente Rancher, modificare il contesto predefinito del cluster di applicazioni nel file kubeconfig fornito da Rancher per utilizzare un contesto del piano di controllo invece del contesto del server API Rancher. In questo modo si riduce il carico sul server API Rancher e si migliorano le performance.
- **Requisiti di Astra Control Provisioner**: Dovresti avere un Astra Control Provisioner configurato correttamente, inclusi i suoi componenti Astra Trident, per gestire i cluster.
  - **Rivedi i requisiti dell'ambiente Astra Trident**: Prima di installare o aggiornare Astra Control provisioner, consulta "frontend, backend e configurazioni host supportati".
  - **Abilitare la funzionalità Astra Control Provisioner**: Si consiglia vivamente di installare Astra Trident 23,10 o versione successiva e di abilitare "Astra Control Provivisioner funzionalità di storage avanzate". Nelle prossime release, Astra Control non supporterà Astra Trident se anche Astra Control Provisioner non è abilitato.
  - **Configurare un backend di archiviazione**: Deve essere presente almeno un backend di archiviazione "Configurato in Astra Trident" sul cluster.
  - **Configurare una classe di archiviazione**: Deve essere presente almeno una classe di archiviazione "Configurato in Astra Trident" sul cluster. Se è configurata una classe di archiviazione predefinita, assicurarsi che sia la classe di archiviazione **only** con l'annotazione predefinita.
  - **Configurare un controller snapshot volume e installare una classe snapshot volume**: "Installare un controller per lo snapshot del volume" In modo che le snapshot possano essere

create in Astra Control. "Creare" almeno uno `VolumeSnapshotClass` Utilizzando Astra Trident.

**Fasi**

1. Nella dashboard, selezionare **Manage Kubernetes cluster** (Gestisci cluster Kubernetes).

   Seguire le istruzioni per aggiungere il cluster.

2. **Provider**: Selezionare la scheda **Other** per aggiungere dettagli sul cluster a gestione automatica.
3. **Altro**: Fornisci dettagli sul tuo cluster autogestito caricando un `kubeconfig.yaml` o incollando il contenuto di `kubeconfig.yaml` file dagli appunti.

   > Se crei il tuo `kubeconfig` file, è necessario definire solo **un** elemento di contesto al suo interno. Fare riferimento a. "queste istruzioni" per informazioni sulla creazione `kubeconfig` file.

4. **Nome credenziale**: Fornire un nome per la credenziale del cluster a gestione automatica che si sta caricando su Astra Control. Per impostazione predefinita, il nome della credenziale viene compilato automaticamente come nome del cluster.
5. **Private route identifier**: Immettere l'identificativo di percorso privato, che è possibile ottenere da Astra Connector. Se si esegue una query su Astra Connector tramite `kubectl get astraconnector -n astra-connector` l'identificatore di route privato viene definito `ASTRACONNECTORID`.

   > L'identificatore di route privato è il nome associato al connettore Astra che consente la gestione di un cluster Kubernetes privato da parte di Astra. In questo contesto, un cluster privato è un cluster Kubernetes che non espone il proprio server API a Internet.

6. Selezionare **Avanti**.
7. (Facoltativo) **Storage**: Facoltativamente, selezionare la classe di storage che si desidera utilizzare per impostazione predefinita per le applicazioni Kubernetes distribuite in questo cluster.
   a. Per selezionare una nuova classe di storage predefinita per il cluster, attivare la casella di controllo **Assegna una nuova classe di storage predefinita**.
   b. Selezionare una nuova classe di storage predefinita dall'elenco.

Ogni servizio di storage del cloud provider visualizza le seguenti informazioni su prezzo, performance e resilienza:

- Cloud Volumes Service per Google Cloud: Informazioni su prezzi, performance e resilienza
- Google Persistent Disk: Non sono disponibili informazioni su prezzi, performance o resilienza
- Azure NetApp Files: Informazioni su performance e resilienza
- Dischi gestiti Azure: Non sono disponibili informazioni su prezzi, performance o resilienza
- Amazon Elastic Block Store: Nessuna informazione su prezzi, performance o resilienza disponibile
- Amazon FSX per NetApp ONTAP: Nessuna informazione su prezzi, performance o resilienza disponibile
- NetApp Cloud Volumes ONTAP: Non sono disponibili informazioni su prezzi, performance o resilienza

Ogni classe di storage può utilizzare uno dei seguenti servizi:

- "Cloud Volumes Service per Google Cloud"
- "Disco persistente di Google"
- "Azure NetApp Files"
- "Dischi gestiti da Azure"
- "Amazon Elastic Block Store"
- "Amazon FSX per NetApp ONTAP"
- "NetApp Cloud Volumes ONTAP"

Scopri di più "Classi di storage per cluster Amazon Web Services". Scopri di più "Classi di storage per cluster AKS". Scopri di più "Classi di storage per cluster GKE".

c. Selezionare **Avanti**.
d. **Review & Approve** (Rivedi e approva): Verifica dei dettagli della configurazione.
e. Selezionare **Add** per aggiungere il cluster ad Astra Control Service.

#### Modificare la classe di storage predefinita

È possibile modificare la classe di storage predefinita per un cluster.

### Modificare la classe di storage predefinita utilizzando Astra Control

È possibile modificare la classe di storage predefinita per un cluster da Astra Control. Se il cluster utilizza un servizio backend di storage precedentemente installato, potrebbe non essere possibile utilizzare questo metodo per modificare la classe di storage predefinita (l'azione **Set as default** non è selezionabile). In questo caso, è possibile Modificare la classe di storage predefinita utilizzando la riga di comando.

#### Fasi

1. Nell'interfaccia utente di Astra Control Service, selezionare **Clusters**.

2. Nella pagina **Clusters**, selezionare il cluster che si desidera modificare.
3. Selezionare la scheda **Storage**.
4. Selezionare la categoria **classi di storage**.
5. Selezionare il menu **azioni** per la classe di storage che si desidera impostare come predefinita.
6. Selezionare **Imposta come predefinito**.

#### Modificare la classe di storage predefinita utilizzando la riga di comando

È possibile modificare la classe di storage predefinita per un cluster utilizzando i comandi Kubernetes. Questo metodo funziona indipendentemente dalla configurazione del cluster.

**Fasi**

1. Accedere al cluster Kubernetes.
2. Elencare le classi di storage nel cluster:

```
kubectl get storageclass
```

3. Rimuovere la designazione predefinita dalla classe di storage predefinita. Sostituire <SC_NAME> con il nome della classe di storage:

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

4. Contrassegna una classe di storage diversa come predefinita. Sostituire <SC_NAME> con il nome della classe di storage:

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

5. Confermare la nuova classe di storage predefinita:

```
kubectl get storageclass
```

#### Controllare la versione di Astra Trident

Per aggiungere un cluster a gestione autonoma che utilizzi Astra Control Provivisioner o Astra Trident per i servizi di storage, assicurati che la versione installata di Astra Trident sia la 23,10 o più recente.

**Fasi**

1. Determina la versione di Astra Trident che stai utilizzando:

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

Se Astra Trident è installato, viene visualizzato un output simile a quanto segue:

```
NAME      VERSION
trident   24.02.0
```

Se Astra Trident non è installato, viene visualizzato un output simile a quanto segue:

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

2. Effettuare una delle seguenti operazioni:
   - Se utilizzi Astra Trident 23,01 o versione precedente, utilizza questi elementi "istruzioni" Per effettuare l'aggiornamento a una versione più recente di Astra Trident prima di effettuare l'aggiornamento a Astra Control Provivisioner. È possibile "eseguire un aggiornamento diretto" A Astra Control Provivisioner 24,02 se il tuo Astra Trident si trova all'interno di una finestra a quattro release della versione 24,02. Ad esempio, puoi eseguire l'upgrade direttamente da Astra Trident 23,04 a Astra Control Provisioner 24,02.
   - Se stai eseguendo Astra Trident 23,10 o versione successiva, verifica che Astra Control provisioner sia stato "attivato". Astra Control Provisioner non funzionerà con le versioni di Astra Control Center precedenti alla 23,10. "Aggiorna Astra Control provisioner" In modo che abbia la stessa versione di Astra Control Center che stai effettuando l'aggiornamento per accedere alle funzionalità più recenti.
3. Assicurarsi che i pod siano in funzione:

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

4. Controllare se le classi di storage utilizzano i driver Astra Trident supportati. Il nome del provider deve essere `csi.trident.netapp.io`. Fare riferimento al seguente esempio:

```
kubectl get sc
```

Esempio di risposta:

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

#### Creare un file kubeconfig

È possibile aggiungere un cluster ad Astra Control Service utilizzando un file kubeconfig.

A seconda del tipo di cluster che si desidera aggiungere, potrebbe essere necessario creare manualmente un file kubeconfig per il cluster utilizzando passaggi specifici.

- Creare un file kubeconfig per i cluster Amazon EKS
- Creare un file kubeconfig per Red Hat OpenShift Service su cluster AWS (ROSA)
- Creare un file kubeconfig per altri tipi di cluster

#### Creare un file kubeconfig per i cluster Amazon EKS

Segui queste istruzioni per creare un file kubeconfig e un token secret permanente per i cluster Amazon EKS. Per i cluster ospitati in EKS è necessario un token secret permanente.

**Fasi**

1. Seguire le istruzioni nella documentazione di Amazon per generare un file kubeconfig:

   "Creazione o aggiornamento di un file kubeconfig per un cluster Amazon EKS"

2. Creare un account di servizio come segue:

   a. Creare un file di account del servizio denominato `astracontrol-service-account.yaml`.

      Modificare il nome dell'account di servizio in base alle necessità. Lo spazio dei nomi `kube-system` è necessario per questi passaggi. Se si modifica il nome dell'account di servizio, è necessario apportare le stesse modifiche nei seguenti passaggi.

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

+

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

3. Applicare l'account del servizio:

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

4. Creare un `ClusterRoleBinding` file chiamato `astracontrol-clusterrolebinding.yaml`.

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

```yaml
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. Applicare l'associazione del ruolo del cluster:

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

6. Creare un file token secret dell'account di servizio chiamato `astracontrol-secret.yaml`.

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

```yaml
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. Applicare il token secret:

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

8. Recuperare il token secret:

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

9. Sostituire `user` Sezione del file kubeconfig AWS EKS con il token, come mostrato nell'esempio seguente:

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

#### Creare un file kubeconfig per Red Hat OpenShift Service su cluster AWS (ROSA)

Segui queste istruzioni per creare un file kubeconfig per Red Hat OpenShift Service su cluster AWS (ROSA).

**Fasi**

1. Accedere al cluster ROSA.
2. Creare un account di servizio:

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

3. Aggiungere un ruolo cluster:

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

4. Utilizzando l'esempio seguente, creare un file di configurazione segreto dell'account di servizio:

```
<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. Creare il segreto:

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

6. Modificare l'account di servizio creato e aggiungere il nome segreto dell'account del servizio Astra Control
   a. `secrets` sezione:

```
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. Elencare i segreti dell'account di servizio, sostituendo `<CONTEXT>` con il contesto corretto per l'installazione:

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

La fine dell'output dovrebbe essere simile a quanto segue:

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

Gli indici di ciascun elemento in `secrets` l'array inizia con 0. Nell'esempio precedente, l'indice per `astracontrol-service-account-dockercfg-dvfcd` sarebbe 0 e l'indice per `secret-astracontrol-service-account` sarebbe 1. Nell'output, annotare il numero dell'indice per il segreto dell'account del servizio. Questo numero di indice sarà necessario nella fase successiva.

8. Generare il kubeconfig come segue:
   a. Creare un `create-kubeconfig.sh` file. Sostituire `TOKEN_INDEX` all'inizio del seguente script con il valore corretto.

```
<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. Eseguire la sorgente dei comandi per applicarli al cluster Kubernetes.

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

9. (Facoltativo) rinominare il kubeconfig con un nome significativo per il cluster.

```
mv kubeconfig-sa YOUR_CLUSTER_NAME_kubeconfig
```

#### Creare un file kubeconfig per altri tipi di cluster

Segui queste istruzioni per creare un file kubeconfig con ruolo limitato o esteso per i cluster Rancher, Upstream Kubernetes e Red Hat OpenShift.

Per i cluster gestiti utilizzando kubeconfig, è possibile creare un'autorizzazione limitata o un ruolo di amministratore di autorizzazioni esteso per Astra Control Service.

Questa procedura consente di creare una configurazione separata se uno dei seguenti scenari si applica al proprio ambiente:

- Si desidera limitare le autorizzazioni di Astra Control sui cluster gestiti
- Si utilizzano più contesti e non è possibile utilizzare il kubeconfig di Astra Control predefinito configurato durante l'installazione oppure un ruolo limitato con un singolo contesto non funziona nell'ambiente

**Prima di iniziare**

Prima di completare la procedura, assicurarsi di disporre dei seguenti elementi per il cluster che si desidera gestire:

- R "versione supportata" di kubectl è installato.
- Kubectl accesso al cluster che si intende aggiungere e gestire con Astra Control Service

> Per questa procedura, non è necessario l'accesso kubectl al cluster che esegue Astra Control Service.

- Un kubeconfig attivo per il cluster che si intende gestire con i diritti di amministratore del cluster per il contesto attivo

**Fasi**

1. Creare un account di servizio:
   a. Creare un file di account del servizio denominato `astracontrol-service-account.yaml`.

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

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

   b. Applicare l'account del servizio:

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

2. Creare uno dei seguenti ruoli del cluster con autorizzazioni sufficienti per la gestione di un cluster da parte di Astra Control:

**Ruolo cluster limitato**

Questo ruolo contiene le autorizzazioni minime necessarie per gestire un cluster da Astra Control:

a. Creare un `ClusterRole` file chiamato, ad esempio, `astra-admin-account.yaml`.

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

```yaml
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. (Solo per i cluster OpenShift) aggiungere quanto segue alla fine di `astra-admin-account.yaml` file:

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

c. Applicare il ruolo del cluster:

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

**Ruolo cluster esteso**

Questo ruolo contiene autorizzazioni estese per un cluster da gestire con Astra Control. È possibile utilizzare questo ruolo se si utilizzano più contesti e non è possibile utilizzare il kubeconfig di Astra Control predefinito configurato durante l'installazione oppure se un ruolo limitato con un singolo contesto non funziona nell'ambiente:

> Quanto segue `ClusterRole` I passaggi sono un esempio generale di Kubernetes. Consultare la documentazione della distribuzione Kubernetes per istruzioni specifiche sull'ambiente in uso.

a. Creare un `ClusterRole` file chiamato, ad esempio, `astra-admin-account.yaml`.

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

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

b. Applicare il ruolo del cluster:

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

3. Creare l'associazione del ruolo del cluster all'account del servizio per il ruolo del cluster:

a. Creare un `ClusterRoleBinding` file chiamato `astracontrol-clusterrolebinding.yaml`.

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

```yaml
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. Applicare l'associazione del ruolo del cluster:

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

4. Creare e applicare il token secret:
   a. Creare un file token secret chiamato `secret-astracontrol-service-account.yaml`.

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

```yaml
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. Applicare il token secret:

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

5. Aggiungere il token secret all'account del servizio aggiungendo il nome a `secrets` array (l'ultima riga dell'esempio seguente):

```
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. Elencare i segreti dell'account di servizio, sostituendo `<context>` con il contesto corretto per l'installazione:

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

La fine dell'output dovrebbe essere simile a quanto segue:

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

Gli indici di ciascun elemento in `secrets` l'array inizia con 0. Nell'esempio precedente, l'indice per `astracontrol-service-account-dockercfg-48xhx` sarebbe 0 e l'indice per `secret-astracontrol-service-account` sarebbe 1. Nell'output, annotare il numero dell'indice per il segreto dell'account del servizio. Questo numero di indice è necessario nel passaggio successivo.

7. Generare il kubeconfig come segue:
   a. Creare un `create-kubeconfig.sh` file.
   b. Sostituire `TOKEN_INDEX` all'inizio del seguente script con il valore corretto.

```
<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. Eseguire la sorgente dei comandi per applicarli al cluster Kubernetes.

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

8. (Facoltativo) rinominare il kubeconfig con un nome significativo per il cluster.

```
mv kubeconfig-sa YOUR_CLUSTER_NAME_kubeconfig
```

## Quali sono le prossime novità?

Ora che hai effettuato l'accesso e aggiunto un cluster ad Astra Control, sei pronto per iniziare a utilizzare le funzionalità di gestione dei dati applicativi di Astra Control.

- "Inizia a gestire le app"
- "Proteggi le app"
- "Clonare le applicazioni"
- "Impostare la fatturazione"
- "Invitare e gestire gli utenti"
- "Gestire le credenziali del cloud provider"
- "Gestire le notifiche"
- "Implementa un'istanza autogestita di Astra Control"

## Video di Astra Control Service

Scopri NetApp TV per i contenuti video più recenti con Astra Control Service. NetApp TV include video che mostrano alcune funzionalità di Astra Control Service o mostrano come

completare determinate attività comuni.

"Video di Astra Control Service"