Inizia subito

Astra Control Center

NetApp
November 21, 2023

Sommario

Inizia subito 1
Requisiti di Astra Control Center 1
Avvio rapido per Astra Control Center 7
Panoramica dell’installazione 9
Configurare Astra Control Center 56
Domande frequenti per Astra Control Center 76

Inizia subito

Requisiti di Astra Control Center

Inizia verificando la preparazione del tuo ambiente operativo, dei cluster di applicazioni, delle applicazioni, delle licenze e del browser Web.

- Requisiti dell'ambiente operativo
- Backend di storage supportati
- Requisiti del cluster di applicazioni
- Requisiti di gestione delle applicazioni
- Prerequisiti per la replica
- Accesso a Internet
- Licenza
- Ingresso per cluster Kubernetes on-premise
- Requisiti di rete
- Browser Web supportati

Requisiti dell'ambiente operativo

Astra Control Center è stato validato per i seguenti tipi di ambienti operativi:

- Google anthos 1.10 o 1.11
- Kubernetes da 1.22 a 1.24
- Rancher Kubernetes Engine (RKE):
  - RKE 1.2.16 con Rancher 2.5.12 e RKE 1.3.3 con 2.6.3
  - RKE 2 (v1.23.6+rke2r2) con Rancher 2.6.3
- Red Hat OpenShift Container Platform 4.8, 4.9 o 4.10
- VMware Tanzu Kubernetes Grid 1.4 o 1.5
- VMware Tanzu Kubernetes Grid Integrated Edition 1.12.2 o 1.13

Assicurarsi che l'ambiente operativo scelto per ospitare Astra Control Center soddisfi i requisiti delle risorse di base descritti nella documentazione ufficiale dell'ambiente. Astra Control Center richiede le seguenti risorse oltre ai requisiti delle risorse dell'ambiente:

Componente	Requisito
Capacità di back-end dello storage	Almeno 500 GB disponibili
Nodi di lavoro	Almeno 3 nodi di lavoro in totale, con 4 core CPU e 12 GB di RAM ciascuno
Indirizzo FQDN	Un indirizzo FQDN per Astra Control Center

Componente	Requisito
Astra Trident	Astra Trident 21.10.1 o versione successiva installata e configurata Astra Trident 22.07 o versione successiva per la replica dell'applicazione basata su SnapMirror

> Questi requisiti presuppongono che Astra Control Center sia l'unica applicazione in esecuzione nell'ambiente operativo. Se nell'ambiente sono in esecuzione applicazioni aggiuntive, modificare di conseguenza questi requisiti minimi.

- **Registro immagini**: È necessario disporre di un registro di immagini Docker privato in cui è possibile trasferire le immagini di build di Astra Control Center. È necessario fornire l'URL del registro delle immagini in cui verranno caricate le immagini.
- **Astra Trident / ONTAP Configuration**: Astra Control Center richiede la creazione e l'impostazione di una classe di storage come classe di storage predefinita. Centro di controllo Astra supporta i seguenti driver ONTAP forniti da Astra Trident:
  - ontap-nas
  - ontap-san
  - ontap-san-economy

> Durante la clonazione delle applicazioni in ambienti OpenShift, Astra Control Center deve consentire a OpenShift di montare volumi e modificare la proprietà dei file. Per questo motivo, è necessario configurare un criterio di esportazione dei volumi ONTAP per consentire queste operazioni. Puoi farlo con i seguenti comandi:
>
> 1. `export-policy rule modify -vserver <storage virtual machine name> -policyname <policy name> -ruleindex 1 -superuser sys`
> 2. `export-policy rule modify -vserver <storage virtual machine name> -policyname <policy name> -ruleindex 1 -anon 65534`

> Se si prevede di aggiungere un secondo ambiente operativo OpenShift come risorsa di calcolo gestita, è necessario assicurarsi che la funzione Astra Trident Volume Snapshot sia attivata. Per abilitare e testare le snapshot dei volumi con Astra Trident, "Consulta le istruzioni ufficiali di Astra Trident".

Requisiti del cluster Grid VMware Tanzu Kubernetes

Quando si ospita Astra Control Center su un cluster VMware Tanzu Kubernetes Grid (TKG) o Tanzu Kubernetes Grid Integrated Edition (TKGi), tenere presente quanto segue.

- Disattivare l'applicazione della classe di storage predefinita TKG o TKGi su qualsiasi cluster di applicazioni che deve essere gestito da Astra Control. Per eseguire questa operazione, modificare il `TanzuKubernetesCluster` risorsa sul cluster dello spazio dei nomi.
- Nell'ambito dell'installazione di Astra Control Center, le seguenti risorse vengono create in un ambiente con restrizioni Pod Security Policy (PSP):
  - policy di sicurezza pod

    - Ruolo RBAC
    - RBAC RoleBinding il ruolo RBAC e le risorse RoleBinding vengono create in `netapp-acc` namespace.
- Quando si implementa Astra Control Center in un ambiente TKG o TKGi, è necessario conoscere i requisiti specifici di Astra Trident. Per ulteriori informazioni, consultare "Documentazione di Astra Trident".

> Il token del file di configurazione predefinito di VMware TKG e TKGi scade dieci ore dopo l'implementazione. Se si utilizzano prodotti del portfolio Tanzu, è necessario generare un file di configurazione del cluster Tanzu Kubernetes con un token non in scadenza per evitare problemi di connessione tra Astra Control Center e cluster di applicazioni gestiti. Per istruzioni, visitare il sito "Documentazione del prodotto VMware NSX-T Data Center."

Requisiti del cluster Google anthos

Quando si ospita Astra Control Center su un cluster Google anthos, Google anthos include per impostazione predefinita il bilanciamento del carico MetalLB e il servizio di gateway di ingresso Istio, consentendo di utilizzare semplicemente le funzionalità di ingresso generiche di Astra Control Center durante l'installazione. Vedere "Configurare Astra Control Center" per ulteriori informazioni.

Backend di storage supportati

Astra Control Center supporta i seguenti backend di storage.

- NetApp ONTAP 9.5 o sistemi AFF e FAS più recenti
- NetApp ONTAP 9.8 o sistemi AFF e FAS più recenti per la replica delle applicazioni basata su SnapMirror
- NetApp Cloud Volumes ONTAP

Per utilizzare il centro di controllo Astra, verificare di disporre delle seguenti licenze ONTAP, a seconda delle operazioni da eseguire:

- FlexClone
- SnapMirror: Opzionale. Necessario solo per la replica su sistemi remoti utilizzando la tecnologia SnapMirror. Fare riferimento a. "Informazioni sulla licenza SnapMirror".
- Licenza S3: Opzionale. Necessario solo per i bucket ONTAP S3

Si consiglia di verificare se il sistema ONTAP dispone delle licenze richieste. Fare riferimento a. "Gestire le licenze ONTAP".

Requisiti del cluster di applicazioni

Astra Control Center ha i seguenti requisiti per i cluster che si intende gestire da Astra Control Center. Questi requisiti si applicano anche se il cluster che si intende gestire è il cluster dell'ambiente operativo che ospita Astra Control Center.

- La versione più recente di Kubernetes "componente snapshot-controller" è installato
- Un tridente Astra "oggetto volumesnapshotclass" è stato definito da un amministratore
- Nel cluster esiste una classe di storage Kubernetes predefinita
- Almeno una classe di storage è configurata per utilizzare Astra Trident

Il cluster di applicazioni deve disporre di un `kubeconfig.yaml` file che definisce un solo elemento *context*. Visitare la documentazione Kubernetes per "informazioni sulla creazione di file kubeconfig".

Quando si gestiscono i cluster di applicazioni in un ambiente Rancher, modificare il contesto predefinito del cluster di applicazioni in `kubeconfig` File 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 gestione delle applicazioni

Astra Control ha i seguenti requisiti di gestione delle applicazioni:

- **Licensing**: Per gestire le applicazioni utilizzando Astra Control Center, è necessaria una licenza Astra Control Center.
- **Namespaces**: Astra Control richiede che un'applicazione non si estende più di un singolo namespace, ma uno spazio dei nomi può contenere più di un'applicazione.
- **StorageClass**: Se si installa un'applicazione con un StorageClass esplicitamente impostato ed è necessario clonare l'applicazione, il cluster di destinazione per l'operazione di clone deve avere la StorageClass specificata in origine. Il cloning di un'applicazione con un StorageClass esplicitamente impostato su un cluster che non ha lo stesso StorageClass avrà esito negativo.
- **Kubernetes resources**: Le applicazioni che utilizzano risorse Kubernetes non raccolte da Astra Control potrebbero non disporre di funzionalità complete di gestione dei dati delle applicazioni. Astra Control raccoglie le seguenti risorse Kubernetes:

ClusterRole	ClusterRoleBinding	ConfigMap
Lavoro di cassa	CustomResourceDefinition	CustomResource
DemonSet	DeploymentConfig	HorizontalPodAutoscaler
Ingresso	MutatingWebhook	NetworkPolicy
PersistentVolumeClaim	Pod	PodDisruptionBudget
PodTemplate	ReplicaSet	Ruolo
RoleBinding	Percorso	Segreto
Servizio	ServiceAccount	StatefulSet
ValidatingWebhook		

Prerequisiti per la replica

La replica dell'applicazione Astra Control richiede che i seguenti prerequisiti siano soddisfatti prima di iniziare:

- Per ottenere un disaster recovery perfetto, si consiglia di implementare Astra Control Center in un terzo dominio di errore o in un sito secondario.
- Il cluster Kubernetes host dell'applicazione e il cluster Kubernetes di destinazione devono essere disponibili e connessi a due cluster ONTAP, idealmente in diversi domini o siti di errore.
- I cluster ONTAP e la SVM host devono essere associati. Vedere "Panoramica del peering di cluster e SVM".

- La SVM remota associata deve essere disponibile per Trident sul cluster di destinazione.
- Trident versione 22.07 o superiore deve essere presente sia sul cluster ONTAP di origine che su quello di destinazione.
- Le licenze asincrone di ONTAP SnapMirror che utilizzano il bundle di protezione dati devono essere attivate sia sul cluster ONTAP di origine che su quello di destinazione. Vedere "Panoramica sulle licenze SnapMirror in ONTAP".
- Quando si aggiunge un backend di storage ONTAP al centro di controllo Astra, applicare le credenziali utente con il ruolo "admin", che dispone di metodi di accesso `http` e. `ontapi` Abilitato su entrambi i cluster ONTAP. Vedere "Gestire gli account utente" per ulteriori informazioni.
- I cluster Kubernetes di origine e destinazione e i cluster ONTAP devono essere gestiti da Astra Control.

> È possibile replicare contemporaneamente un'altra applicazione (in esecuzione sull'altro cluster o sito) nella direzione opposta. Ad esempio, è possibile replicare le applicazioni A, B, C dal Datacenter 1 al Datacenter 2 e le applicazioni X, Y, Z dal Datacenter 2 al Datacenter 1.

Scopri come "Replica delle applicazioni su un sistema remoto utilizzando la tecnologia SnapMirror".

Metodi di installazione delle applicazioni supportati

Astra Control supporta i seguenti metodi di installazione dell'applicazione:

- **Manifest file**: Astra Control supporta le applicazioni installate da un file manifest utilizzando kubectl. Ad esempio:

```
kubectl apply -f myapp.yaml
```

- **Helm 3**: Se utilizzi Helm per installare le app, Astra Control richiede Helm versione 3. La gestione e la clonazione delle applicazioni installate con Helm 3 (o aggiornate da Helm 2 a Helm 3) sono completamente supportate. La gestione delle applicazioni installate con Helm 2 non è supportata.
- **Applicazioni distribuite dall'operatore**: Astra Control supporta le applicazioni installate con operatori con ambito namespace. Di seguito sono riportate alcune applicazioni che sono state validate per questo modello di installazione:
  - "Apache K8ssandra"
  - "Ci Jenkins"
  - "Cluster XtraDB Percona"

> Un operatore e l'applicazione che installa devono utilizzare lo stesso namespace; potrebbe essere necessario modificare il file .yaml di implementazione per l'operatore per assicurarsi che questo sia il caso.

Accesso a Internet

È necessario determinare se si dispone di un accesso esterno a Internet. In caso contrario, alcune funzionalità potrebbero essere limitate, ad esempio la ricezione di dati di monitoraggio e metriche da NetApp Cloud Insights o l'invio di pacchetti di supporto a "Sito di supporto NetApp".

Licenza

Astra Control Center richiede una licenza Astra Control Center per una funzionalità completa. Ottenere una licenza di valutazione o una licenza completa da NetApp. Hai bisogno di una licenza per proteggere le tue applicazioni e i tuoi dati. Fare riferimento a. "Caratteristiche di Astra Control Center" per ulteriori informazioni.

Puoi provare Astra Control Center con una licenza di valutazione, che ti consente di utilizzare Astra Control Center per 90 giorni dalla data di download della licenza. Puoi iscriverti per una prova gratuita registrandoti "qui".

Per ulteriori informazioni sulle licenze necessarie per i backend di storage ONTAP, fare riferimento a. "Backend di storage supportati".

Per ulteriori informazioni sul funzionamento delle licenze, vedere "Licensing".

Ingresso per cluster Kubernetes on-premise

È possibile scegliere il tipo di ingresso di rete utilizzato da Astra Control Center. Per impostazione predefinita, Astra Control Center implementa il gateway Astra Control Center (servizio/traefik) come risorsa a livello di cluster. Astra Control Center supporta anche l'utilizzo di un servizio di bilanciamento del carico, se consentito nel tuo ambiente. Se si preferisce utilizzare un servizio di bilanciamento del carico e non ne si dispone già di uno configurato, è possibile utilizzare il bilanciamento del carico MetalLB per assegnare automaticamente un indirizzo IP esterno al servizio. Nella configurazione del server DNS interno, puntare il nome DNS scelto per Astra Control Center sull'indirizzo IP con bilanciamento del carico.

> Se si ospita Astra Control Center su un cluster Tanzu Kubernetes Grid, utilizzare `kubectl get nsxlbmonitors -A` per verificare se è già stato configurato un monitor dei servizi per accettare il traffico in entrata. Se ne esiste uno, non installare MetalLB, perché il monitor di servizio esistente sovrascriverà qualsiasi nuova configurazione del bilanciamento del carico.

Per ulteriori informazioni, vedere "Impostare l'ingresso per il bilanciamento del carico".

Requisiti di rete

L'ambiente operativo che ospita Astra Control Center comunica utilizzando le seguenti porte TCP. Assicurarsi che queste porte siano consentite attraverso qualsiasi firewall e configurare i firewall in modo da consentire qualsiasi traffico HTTPS in uscita dalla rete Astra. Alcune porte richiedono la connettività tra l'ambiente che ospita Astra Control Center e ciascun cluster gestito (annotato dove applicabile).

> Puoi implementare Astra Control Center in un cluster Kubernetes dual-stack, mentre Astra Control Center può gestire le applicazioni e i back-end di storage configurati per il funzionamento dual-stack. Per ulteriori informazioni sui requisiti del cluster dual-stack, vedere "Documentazione Kubernetes".

Origine	Destinazione	Porta	Protocollo	Scopo
PC client	Centro di controllo Astra	443	HTTPS	Accesso UI/API - assicurarsi che questa porta sia aperta in entrambi i modi tra il cluster che ospita Astra Control Center e ciascun cluster gestito
Metriche consumer	Nodo di lavoro Astra Control Center	9090	HTTPS	Comunicazione dei dati delle metriche - garantire che ciascun cluster gestito possa accedere a questa porta sul cluster che ospita Astra Control Center (è richiesta una comunicazione bidirezionale)
Centro di controllo Astra	Servizio Hosted Cloud Insights (https://cloudinsights.netapp.com)	443	HTTPS	Comunicazione Cloud Insights
Centro di controllo Astra	Provider di bucket di storage Amazon S3 (https://my-bucket.s3.us-west-2.amazonaws.com/)	443	HTTPS	Comunicazione con lo storage Amazon S3
Centro di controllo Astra	NetApp AutoSupport (https://support.netapp.com)	443	HTTPS	Comunicazioni NetApp AutoSupport

Browser Web supportati

Astra Control Center supporta versioni recenti di Firefox, Safari e Chrome con una risoluzione minima di 1280 x 720.

Cosa succederà

Visualizzare il "avvio rapido" panoramica.

Avvio rapido per Astra Control Center

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

Provalo! Se si desidera provare Astra Control Center, è possibile utilizzare una licenza di valutazione di 90

giorni. Vedere "informazioni sulle licenze" per ulteriori informazioni.

1 Esaminare i requisiti del cluster Kubernetes

- Astra funziona con i cluster Kubernetes con un backend di storage ONTAP configurato con Trident o un backend di storage Astra Data Store.
- I cluster devono essere in esecuzione in condizioni di salute, con almeno tre nodi di lavoro online.
- Il cluster deve eseguire Kubernetes.

Scopri di più su "Requisiti di Astra Control Center".

2 Scaricare e installare Astra Control Center

- Scaricare Astra Control Center dal "Sito di supporto NetApp pagina Download di Astra Control Center".
- Installare Astra Control Center nell'ambiente locale.

  Se lo si desidera, installare Astra Control Center utilizzando Red Hat OperatorHub.

  Se lo si desidera, installare il centro di controllo Astra con un backend di storage Cloud Volumes ONTAP.

Scopri di più "Installazione di Astra Control Center".

3 Completare alcune attività di configurazione iniziali

- Aggiunta di una licenza Astra Control e di eventuali licenze ONTAP di supporto.
- Aggiungi un cluster Kubernetes e Astra Control Center scopre i dettagli.
- Aggiungere un backend di storage ONTAP.
- Facoltativamente, Aggiungi un bucket di store di oggetti che memorizzerà i backup delle app.

Scopri di più su "processo di installazione iniziale".

4 Utilizzare Astra Control Center

Dopo aver completato la configurazione di Astra Control Center, ecco cosa fare:

- Gestire un'applicazione. Scopri di più su come "gestire le applicazioni".
- Proteggi le app configurando le policy di protezione per le app, replicando le app su sistemi remoti e clonando e migrando le app. Scopri di più su come "proteggi le app".
- Gestire gli account (inclusi utenti, ruoli, LDAP per l'autenticazione dell'utente, credenziali, connessioni al repository e altro ancora). Scopri di più su come "gestire gli utenti".
- Se lo si desidera, connettersi a NetApp Cloud Insights per visualizzare le metriche sullo stato di salute del sistema, sulla capacità e sul throughput all'interno dell'interfaccia utente di Astra Control Center. Scopri di più "Connessione a Cloud Insights".

5 Continuare da questa guida di avvio rapido

"Installare Astra Control Center".

Trova ulteriori informazioni

- "Utilizzare l'API di controllo Astra"

Panoramica dell'installazione

Scegliere e completare una delle seguenti procedure di installazione di Astra Control Center:

- "Installare Astra Control Center utilizzando il processo standard"
- "(Se utilizzi Red Hat OpenShift) Installa Astra Control Center usando OpenShift OperatorHub"
- "Installare il centro di controllo Astra con un backend di storage Cloud Volumes ONTAP"

Installare Astra Control Center utilizzando il processo standard

Per installare Astra Control Center, scaricare il pacchetto di installazione dal NetApp Support Site ed eseguire la procedura seguente per installare Astra Control Center Operator e Astra Control Center nel proprio ambiente. È possibile utilizzare questa procedura per installare Astra Control Center in ambienti connessi a Internet o con connessione ad aria.

Per gli ambienti Red Hat OpenShift, è possibile utilizzare un "procedura alternativa" Per installare Astra Control Center utilizzando OpenShift OperatorHub.

Di cosa hai bisogno

- "Prima di iniziare l'installazione, preparare l'ambiente per l'implementazione di Astra Control Center".
- Se hai configurato o vuoi configurare le policy di sicurezza dei pod nel tuo ambiente, familiarizza con le policy di sicurezza dei pod e con il modo in cui influiscono sull'installazione di Astra Control Center. Vedere "Comprendere le restrizioni delle policy di sicurezza del pod".
- Assicurarsi che tutti gli operatori del cluster siano in buono stato e disponibili.

  ```
  kubectl get clusteroperators
  ```

- Assicurarsi che tutti i servizi API siano in buono stato e disponibili:

  ```
  kubectl get apiservices
  ```

- Assicurarsi che l'FQDN Astra che si intende utilizzare sia instradabile a questo cluster. Ciò significa che si dispone di una voce DNS nel server DNS interno o si sta utilizzando un percorso URL principale già registrato.
- Se nel cluster esiste già un cert-manager, è necessario eseguirne alcune "fasi preliminari" In modo che Astra Control Center non installi il proprio cert-manager.

A proposito di questa attività

Il processo di installazione di Astra Control Center esegue le seguenti operazioni:

- Installa i componenti Astra in `netapp-acc` namespace (o personalizzato).

- Crea un account predefinito.
- Stabilisce un indirizzo e-mail amministrativo predefinito per l'utente e una password monouso predefinita. A questo utente viene assegnato il ruolo Owner (Proprietario) nel sistema necessario per il primo accesso all'interfaccia utente.
- Consente di determinare se tutti i pod Astra Control Center sono in esecuzione.
- Installa l'interfaccia utente Astra.

> (Valido solo per la release Astra Data Store Early Access Program (EAP)). Se si intende gestire Astra Data Store utilizzando Astra Control Center e abilitare i flussi di lavoro VMware, implementare Astra Control Center solo su `pcloud` namespace e non su `netapp-acc` namespace o uno spazio dei nomi personalizzato descritto nei passaggi di questa procedura.

> Non eseguire il seguente comando durante l'intero processo di installazione per evitare di eliminare tutti i pod di Astra Control Center: `kubectl delete -f astra_control_center_operator_deploy.yaml`

> Se si utilizza Podman di Red Hat invece di Docker Engine, è possibile utilizzare i comandi Podman al posto dei comandi Docker.

Fasi

Per installare Astra Control Center, procedere come segue:

- Scarica e disimballa il bundle Astra Control Center
- Installare il plug-in NetApp Astra kubectl
- Aggiungere le immagini al registro locale
- Impostare namespace e secret per i registri con requisiti di autenticazione
- Installare l'operatore del centro di controllo Astra
- Configurare Astra Control Center
- Completare l'installazione dell'Astra Control Center e dell'operatore
- Verificare lo stato del sistema
- Impostare l'ingresso per il bilanciamento del carico
- Accedere all'interfaccia utente di Astra Control Center

Scarica e disimballa il bundle Astra Control Center

1. Scarica il bundle Astra Control Center (`astra-control-center-[version].tar.gz`) da "Sito di supporto NetApp".
2. Scarica la zip dei certificati e delle chiavi di Astra Control Center dal "Sito di supporto NetApp".
3. (Facoltativo) utilizzare il seguente comando per verificare la firma del bundle:

```
openssl dgst -sha256 -verify AstraControlCenter-public.pub -signature
astra-control-center-[version].tar.gz.sig astra-control-center-
[version].tar.gz
```

4. Estrarre le immagini:

```
tar -vxzf astra-control-center-[version].tar.gz
```

Installare il plug-in NetApp Astra kubectl

NetApp Astra `kubectl` Il plug-in della riga di comando consente di risparmiare tempo durante l'esecuzione di attività comuni associate all'implementazione e all'aggiornamento di Astra Control Center.

Di cosa hai bisogno

NetApp fornisce binari per il plug-in per diverse architetture CPU e sistemi operativi. Prima di eseguire questa attività, è necessario conoscere la CPU e il sistema operativo in uso. Sui sistemi operativi Linux e Mac, è possibile utilizzare `uname -a` per raccogliere queste informazioni.

Fasi

1. Elencare NetApp Astra disponibile `kubectl` Binari del plug-in e annotare il nome del file necessario per il sistema operativo e l'architettura della CPU:

```
ls kubectl-astra/
```

2. Copiare il file nella stessa posizione dello standard `kubectl` utility. In questo esempio, il `kubectl` l'utility si trova in `/usr/local/bin` directory. Sostituire `<binary-name>` con il nome del file desiderato:

```
cp kubectl-astra/<binary-name> /usr/local/bin/kubectl-astra
```

Aggiungere le immagini al registro locale

1. Completare la sequenza di passaggi appropriata per il motore dei container:

Docker

1. Passare alla directory Astra:

```
cd acc
```

2. inserire le immagini del pacchetto nella directory delle immagini di Astra Control Center nel registro locale. Eseguire le seguenti sostituzioni prima di eseguire il comando:
   - Sostituire BUNDLE_FILE con il nome del file bundle Astra Control (ad esempio, `acc.manifest.yaml`).
   - Sostituire MY_REGISTRY con l'URL del repository Docker.
   - Sostituire MY_REGISTRY_USER con il nome utente.
   - Sostituire MY_REGISTRY_TOKEN con un token autorizzato per il Registro di sistema.

```
kubectl astra packages push-images -m BUNDLE_FILE -r MY_REGISTRY
-u MY_REGISTRY_USER -p MY_REGISTRY_TOKEN
```

Podman

1. Accedere al Registro di sistema:

```
podman login [your_registry_path]
```

2. Eseguire il seguente script, eseguendo la sostituzione <YOUR_REGISTRY> come indicato nei commenti:

```
# You need to be at the root of the tarball.
# You should see these files to confirm correct location:
#   acc.manifest.yaml
#   acc/

# Replace <YOUR_REGISTRY> with your own registry (e.g
registry.customer.com or registry.customer.com/testing, etc..)
export REGISTRY=<YOUR_REGISTRY>
export PACKAGENAME=acc
export PACKAGEVERSION=22.08.1-26
export DIRECTORYNAME=acc
for astraImageFile in $(ls ${DIRECTORYNAME}/images/*.tar) ; do
  # Load to local cache
  astraImage=$(podman load --input ${astraImageFile} | sed 's/Loaded
image(s): //')

  # Remove path and keep imageName.
  astraImageNoPath=$(echo ${astraImage} | sed 's:.*/::')

  # Tag with local image repo.
  podman tag ${astraImage} ${REGISTRY}/netapp/astra/${PACKAGENAME}
/${PACKAGEVERSION}/${astraImageNoPath}

  # Push to the local repo.
  podman push ${REGISTRY}/netapp/astra/${PACKAGENAME}/
${PACKAGEVERSION}/${astraImageNoPath}
done
```

Impostare namespace e secret per i registri con requisiti di autenticazione

1. Esportare il KUBECONFIG per il cluster host Astra Control Center:

```
export KUBECONFIG=[file path]
```

2. Se si utilizza un registro che richiede l'autenticazione, è necessario effettuare le seguenti operazioni:

   a. Creare il `netapp-acc-operator` spazio dei nomi:

```
kubectl create ns netapp-acc-operator
```

   Risposta:

```
namespace/netapp-acc-operator created
```

b. Creare un segreto per `netapp-acc-operator` namespace. Aggiungere informazioni su Docker ed eseguire il seguente comando:

> Il segnaposto `your_registry_path` deve corrispondere alla posizione delle immagini caricate in precedenza (ad esempio, `[Registry_URL]/netapp/astra/astracc/22.08.1-26`).

```
kubectl create secret docker-registry astra-registry-cred -n netapp-
acc-operator --docker-server=[your_registry_path] --docker-username
=[username] --docker-password=[token]
```

Esempio di risposta:

```
secret/astra-registry-cred created
```

> Se si elimina lo spazio dei nomi dopo la generazione del segreto, è necessario rigenerare il segreto per lo spazio dei nomi dopo la ricostruzione dello spazio dei nomi.

c. Creare il `netapp-acc` namespace (o personalizzato).

```
kubectl create ns [netapp-acc or custom namespace]
```

Esempio di risposta:

```
namespace/netapp-acc created
```

d. Creare un segreto per `netapp-acc` namespace (o personalizzato). Aggiungere informazioni su Docker ed eseguire il seguente comando:

```
kubectl create secret docker-registry astra-registry-cred -n [netapp-
acc or custom namespace] --docker-server=[your_registry_path]
--docker-username=[username] --docker-password=[token]
```

Risposta

```
secret/astra-registry-cred created
```

a. (opzionale) se si desidera che il cluster venga gestito automaticamente da Astra Control Center

dopo l'installazione, assicurarsi di fornire il kubeconfig come segreto all'interno dello spazio dei nomi di Astra Control Center in cui si intende eseguire la distribuzione utilizzando questo comando:

```
kubectl create secret generic [acc-kubeconfig-cred or custom secret name] --from-file=<path-to-your-kubeconfig> -n [netapp-acc or custom namespace]
```

Installare l'operatore del centro di controllo Astra

1. Modificare la directory:

```
cd manifests
```

2. Modificare l'YAML di implementazione dell'operatore di Astra Control Center (`astra_control_center_operator_deploy.yaml`) per fare riferimento al registro locale e al segreto.

```
vim astra_control_center_operator_deploy.yaml
```

> Un YAML di esempio annotato segue questi passaggi.

a. Se si utilizza un registro che richiede l'autenticazione, sostituire la riga predefinita di `imagePullSecrets: []` con i seguenti elementi:

```
imagePullSecrets:
- name: <astra-registry-cred>
```

b. Cambiare `[your_registry_path]` per `kube-rbac-proxy` al percorso del registro in cui sono state inviate le immagini in a. passaggio precedente.

c. Cambiare `[your_registry_path]` per `acc-operator-controller-manager` al percorso del registro in cui sono state inviate le immagini in a. passaggio precedente.

d. (Per le installazioni che utilizzano l'anteprima di Astra Data Store) vedere questo problema noto relativo a. "Provisioning delle classi di storage e modifiche aggiuntive da apportare al programma YAML".

```
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    control-plane: controller-manager
  name: acc-operator-controller-manager
  namespace: netapp-acc-operator
spec:
  replicas: 1
  selector:
    matchLabels:
      control-plane: controller-manager
  template:
    metadata:
      labels:
        control-plane: controller-manager
    spec:
      containers:
      - args:
        - --secure-listen-address=0.0.0.0:8443
        - --upstream=http://127.0.0.1:8080/
        - --logtostderr=true
        - --v=10
        image: [your_registry_path]/kube-rbac-proxy:v4.8.0
        name: kube-rbac-proxy
        ports:
        - containerPort: 8443
          name: https
      - args:
        - --health-probe-bind-address=:8081
        - --metrics-bind-address=127.0.0.1:8080
        - --leader-elect
        command:
        - /manager
        env:
        - name: ACCOP_LOG_LEVEL
          value: "2"
        image: [your_registry_path]/acc-operator:[version x.y.z]
        imagePullPolicy: IfNotPresent
      imagePullSecrets: []
```

3. Installare l’operatore del centro di controllo Astra:

```
kubectl apply -f astra_control_center_operator_deploy.yaml
```

Esempio di risposta:

```
namespace/netapp-acc-operator created
customresourcedefinition.apiextensions.k8s.io/astracontrolcenters.astra.
netapp.io created
role.rbac.authorization.k8s.io/acc-operator-leader-election-role created
clusterrole.rbac.authorization.k8s.io/acc-operator-manager-role created
clusterrole.rbac.authorization.k8s.io/acc-operator-metrics-reader
created
clusterrole.rbac.authorization.k8s.io/acc-operator-proxy-role created
rolebinding.rbac.authorization.k8s.io/acc-operator-leader-election-
rolebinding created
clusterrolebinding.rbac.authorization.k8s.io/acc-operator-manager-
rolebinding created
clusterrolebinding.rbac.authorization.k8s.io/acc-operator-proxy-
rolebinding created
configmap/acc-operator-manager-config created
service/acc-operator-controller-manager-metrics-service created
deployment.apps/acc-operator-controller-manager created
```

4. Verificare che i pod siano in esecuzione:

```
kubectl get pods -n netapp-acc-operator
```

Configurare Astra Control Center

1. Modificare il file delle risorse personalizzate (CR) di Astra Control Center (`astra_control_center_min.yaml`) Per creare account, AutoSupport, Registro di sistema e altre configurazioni necessarie:

> `astra_control_center_min.yaml` È il CR predefinito ed è adatto per la maggior parte delle installazioni. Familiarizzare con tutti "Opzioni CR e relativi valori potenziali" Per garantire la corretta implementazione di Astra Control Center per il proprio ambiente. Se sono necessarie personalizzazioni aggiuntive per il proprio ambiente, è possibile utilizzare `astra_control_center.yaml` Come CR alternativa.

```
vim astra_control_center_min.yaml
```

> Se si utilizza un registro che non richiede autorizzazione, è necessario eliminare `secret` linea entro `imageRegistry` in caso negativo, l'installazione non riesce.

a. Cambiare `[your_registry_path]` al percorso del registro di sistema in cui sono state inviate le immagini nel passaggio precedente.

b. Modificare il `accountName` stringa al nome che si desidera associare all'account.

c. Modificare il `astraAddress` Stringa all'FQDN che si desidera utilizzare nel browser per accedere ad Astra. Non utilizzare `http://` oppure `https://` nell'indirizzo. Copiare questo FQDN per utilizzarlo in un passo successivo.

d. Modificare il `email` stringa all'indirizzo iniziale predefinito dell'amministratore. Copiare questo indirizzo e-mail per utilizzarlo in passo successivo.

e. Cambiare `enrolled` Per AutoSupport a. `false` per i siti senza connettività internet o senza retain `true` per i siti connessi.

f. Se si utilizza un cert-manager esterno, aggiungere le seguenti righe a. `spec`:

```
spec:
  crds:
    externalCertManager: true
```

g. (Facoltativo) aggiungere un nome `firstName` e cognome `lastName` dell'utente associato all'account. È possibile eseguire questo passaggio ora o in un secondo momento all'interno dell'interfaccia utente.

h. (Facoltativo) modificare `storageClass` Valore per un'altra risorsa Trident storageClass, se richiesto dall'installazione.

i. (Facoltativo) se si desidera che il cluster venga gestito automaticamente da Astra Control Center dopo l'installazione e si è già provveduto creato il segreto contenente il kubeconfig per questo cluster, Fornire il nome del segreto aggiungendo un nuovo campo a questo file YAML chiamato `astraKubeConfigSecret: "acc-kubeconfig-cred or custom secret name"`

j. Completare una delle seguenti operazioni:

- **Other ingress controller (ingressType:Generic)**: Questa è l'azione predefinita con Astra Control Center. Dopo l'implementazione di Astra Control Center, è necessario configurare il controller di ingresso per esporre Astra Control Center con un URL.

  L'installazione predefinita di Astra Control Center imposta il gateway (`service/traefik`) per essere del tipo `ClusterIP`. Questa installazione predefinita richiede l'impostazione di Kubernetes IngressController/Ingress per instradare il traffico verso di essa. Se si desidera utilizzare un ingresso, vedere "Impostare l'ingresso per il bilanciamento del carico".

- **Service load balancer (ingressType:AccTraefik)**: Se non si desidera installare un IngressController o creare una risorsa Ingress, impostare `ingressType` a. `AccTraefik`.

  In questo modo viene implementato l'Astra Control Center `traefik` Gateway come servizio di tipo Kubernetes LoadBalancer.

  Astra Control Center utilizza un servizio del tipo "LoadBalancer" (`svc/traefik` Nello spazio dei nomi di Astra Control Center) e richiede l'assegnazione di un indirizzo IP esterno accessibile. Se nel proprio ambiente sono consentiti i bilanciatori di carico e non ne è già configurato uno, è possibile utilizzare MetalLB o un altro servizio di bilanciamento del carico esterno per assegnare un indirizzo IP esterno al servizio. Nella configurazione del server DNS interno, puntare il nome DNS scelto per Astra Control Center sull'indirizzo IP con bilanciamento del carico.

Per ulteriori informazioni sul tipo di servizio "LoadBalancer" e sull'ingresso, vedere "Requisiti".

```
apiVersion: astra.netapp.io/v1
kind: AstraControlCenter
metadata:
  name: astra
spec:
  accountName: "Example"
  astraVersion: "ASTRA_VERSION"
  astraAddress: "astra.example.com"
  astraKubeConfigSecret: "acc-kubeconfig-cred or custom secret name"
  ingressType: "Generic"
  autoSupport:
    enrolled: true
  email: "[admin@example.com]"
  firstName: "SRE"
  lastName: "Admin"
  imageRegistry:
    name: "[your_registry_path]"
    secret: "astra-registry-cred"
  storageClass: "ontap-gold"
```

Completare l'installazione dell'Astra Control Center e dell'operatore

1. Se non lo si è già fatto in un passaggio precedente, creare il `netapp-acc` namespace (o personalizzato):

   ```
   kubectl create ns [netapp-acc or custom namespace]
   ```

   Esempio di risposta:

   ```
   namespace/netapp-acc created
   ```

2. Installare Astra Control Center in `netapp-acc` spazio dei nomi (o personalizzato):

   ```
   kubectl apply -f astra_control_center_min.yaml -n [netapp-acc or custom
   namespace]
   ```

   Esempio di risposta:

   ```
   astracontrolcenter.astra.netapp.io/astra created
   ```

Verificare lo stato del sistema

Se preferisci utilizzare OpenShift, puoi utilizzare comandi oc paragonabili per le fasi di verifica.

1. Verificare che tutti i componenti del sistema siano installati correttamente.

```
kubectl get pods -n [netapp-acc or custom namespace]
```

   Ogni pod deve avere uno stato di `Running`. L'implementazione dei pod di sistema potrebbe richiedere alcuni minuti.

Esempio di risposta

```
NAME                                        READY   STATUS    RESTARTS
AGE
acc-helm-repo-6b44d68d94-d8m55              1/1     Running   0
13m
activity-78f99ddf8-hltct                    1/1     Running   0
10m
api-token-authentication-457nl              1/1     Running   0
9m28s
api-token-authentication-dgwsz              1/1     Running   0
9m28s
api-token-authentication-hmqqc              1/1     Running   0
9m28s
asup-75fd554dc6-m6qzh                       1/1     Running   0
9m38s
authentication-6779b4c85d-92gds             1/1     Running   0
8m11s
bucketservice-7cc767f8f8-lqwr8              1/1     Running   0
9m31s
certificates-549fd5d6cb-5kmd6               1/1     Running   0
9m56s
certificates-549fd5d6cb-bkjh9               1/1     Running   0
9m56s
cloud-extension-7bcb7948b-hn8h2             1/1     Running   0
10m
cloud-insights-service-56ccf86647-fgg69     1/1     Running   0
9m46s
composite-compute-677685b9bb-7vgsf          1/1     Running   0
10m
composite-volume-657d6c5585-dnq79           1/1     Running   0
9m49s
credentials-755fd867c8-vrlmt                1/1     Running   0
11m
entitlement-86495cdf5b-nwhh2                1/1     Running   2
10m
features-5684fb8b56-8d6s8                   1/1     Running   0
10m
fluent-bit-ds-rhx7v                         1/1     Running   0
7m48s
fluent-bit-ds-rjms4                         1/1     Running   0
7m48s
fluent-bit-ds-zf5ph                         1/1     Running   0
7m48s
graphql-server-66d895f544-w6hjd             1/1     Running   0
3m29s
```

```
identity-744df448d5-rlcmm                1/1     Running   0
10m
influxdb2-0                              1/1     Running   0
13m
keycloak-operator-75c965cc54-z7csw       1/1     Running   0
8m16s
krakend-798d6df96f-9z2sk                 1/1     Running   0
3m26s
license-5fb7d75765-f8mjg                 1/1     Running   0
9m50s
login-ui-7d5b7df85d-l2s7s                1/1     Running   0
3m20s
loki-0                                   1/1     Running   0
13m
metrics-facade-599b9d7fcc-gtmgl          1/1     Running   0
9m40s
monitoring-operator-67cc74f844-cdplp     2/2     Running   0
8m11s
nats-0                                   1/1     Running   0
13m
nats-1                                   1/1     Running   0
13m
nats-2                                   1/1     Running   0
12m
nautilus-769f5b74cd-k5jxm                1/1     Running   0
9m42s
nautilus-769f5b74cd-kd9gd                1/1     Running   0
8m59s
openapi-84f6ccd8ff-76kvp                 1/1     Running   0
9m34s
packages-6f59fc67dc-4g2f5                1/1     Running   0
9m52s
polaris-consul-consul-server-0           1/1     Running   0
13m
polaris-consul-consul-server-1           1/1     Running   0
13m
polaris-consul-consul-server-2           1/1     Running   0
13m
polaris-keycloak-0                       1/1     Running   0
8m7s
polaris-keycloak-1                       1/1     Running   0
5m49s
polaris-keycloak-2                       1/1     Running   0
5m15s
polaris-keycloak-db-0                    1/1     Running   0
8m6s
```

```
polaris-keycloak-db-1                     1/1     Running   0
5m49s
polaris-keycloak-db-2                     1/1     Running   0
4m57s
polaris-mongodb-0                         2/2     Running   0
13m
polaris-mongodb-1                         2/2     Running   0
12m
polaris-mongodb-2                         2/2     Running   0
12m
polaris-ui-565f56bf7b-zwr8b               1/1     Running   0
3m19s
polaris-vault-0                           1/1     Running   0
13m
polaris-vault-1                           1/1     Running   0
13m
polaris-vault-2                           1/1     Running   0
13m
public-metrics-6d86d66444-2wbzl           1/1     Running   0
9m30s
storage-backend-metrics-77c5d98dcd-dbhg5 1/1     Running   0
9m44s
storage-provider-78c885f57c-6zcv4         1/1     Running   0
9m36s
telegraf-ds-2l2m9                         1/1     Running   0
7m48s
telegraf-ds-qfzgh                         1/1     Running   0
7m48s
telegraf-ds-shrms                         1/1     Running   0
7m48s
telegraf-rs-bjpkt                         1/1     Running   0
7m48s
telemetry-service-6684696c64-qzfdf        1/1     Running   0
10m
tenancy-6596b6c54d-vmpsm                  1/1     Running   0
10m
traefik-7489dc59f9-6mnst                  1/1     Running   0
3m19s
traefik-7489dc59f9-xrkgg                  1/1     Running   0
3m4s
trident-svc-6c8dc458f5-jswcl              1/1     Running   0
10m
vault-controller-6b954f9b76-gz9nm         1/1     Running   0
11m
```

2. (Facoltativo) per assicurarsi che l'installazione sia completata, è possibile guardare `acc-operator` registra usando il seguente comando.

```
kubectl logs deploy/acc-operator-controller-manager -n netapp-acc-
operator -c manager -f
```

> `accHost` la registrazione del cluster è una delle ultime operazioni e, in caso di errore, la distribuzione non avrà esito negativo. In caso di errore di registrazione del cluster indicato nei registri, è possibile tentare di nuovo la registrazione attraverso il flusso di lavoro add cluster "Nell'interfaccia utente" O API.

3. Una volta eseguiti tutti i pod, verificare che l'installazione sia stata eseguita correttamente (`READY` è `True`) E ottenere la password monouso da utilizzare per l'accesso ad Astra Control Center:

```
kubectl get AstraControlCenter -n netapp-acc
```

Risposta:

```
NAME    UUID                                      VERSION     ADDRESS
READY
astra   ACC-9aa5fdae-4214-4cb7-9976-5d8b4c0ce27f  22.08.1-26
10.111.111.111  True
```

> Copiare il valore UUID. La password è `ACC-` Seguito dal valore UUID (`ACC-[UUID]` oppure, in questo esempio, `ACC-9aa5fdae-4214-4cb7-9976-5d8b4c0ce27f`).

Impostare l'ingresso per il bilanciamento del carico

È possibile configurare un controller di ingresso Kubernetes che gestisce l'accesso esterno ai servizi, come il bilanciamento del carico in un cluster.

Questa procedura spiega come configurare un controller di ingresso (`ingressType:Generic`). Questa è l'azione predefinita con Astra Control Center. Dopo l'implementazione di Astra Control Center, è necessario configurare il controller di ingresso per esporre Astra Control Center con un URL.

> Se non si desidera configurare un controller di ingresso, è possibile impostarlo `ingressType:AccTraefik`). Astra Control Center utilizza un servizio del tipo "LoadBalancer" (`svc/traefik` Nello spazio dei nomi di Astra Control Center) e richiede l'assegnazione di un indirizzo IP esterno accessibile. Se nel proprio ambiente sono consentiti i bilanciatori di carico e non ne è già configurato uno, è possibile utilizzare MetalLB o un altro servizio di bilanciamento del carico esterno per assegnare un indirizzo IP esterno al servizio. Nella configurazione del server DNS interno, puntare il nome DNS scelto per Astra Control Center sull'indirizzo IP con bilanciamento del carico. Per ulteriori informazioni sul tipo di servizio "LoadBalancer" e sull'ingresso, vedere "Requisiti".

I passaggi variano a seconda del tipo di controller di ingresso utilizzato:

- Ingresso Istio
- Controller di ingresso nginx
- Controller di ingresso OpenShift

Di cosa hai bisogno

- Il necessario "controller di ingresso" dovrebbe essere già implementato.
- Il "classe di ingresso" corrispondente al controller di ingresso dovrebbe già essere creato.
- Si stanno utilizzando versioni di Kubernetes comprese tra v1.19 e v1.22.

Passaggi per l'ingresso di Istio

1. Configurare l'ingresso Istio.

   > Questa procedura presuppone che Istio venga distribuito utilizzando il profilo di configurazione "predefinito".

2. Raccogliere o creare il certificato e il file della chiave privata desiderati per Ingress Gateway.

   È possibile utilizzare un certificato CA o autofirmato. Il nome comune deve essere l'indirizzo Astra (FQDN).

   Esempio di comando:

   ```
   openssl req -x509 -nodes -days 365 -newkey rsa:2048
   -keyout tls.key -out tls.crt
   ```

3. Crea un segreto `tls secret name` di tipo `kubernetes.io/tls` Per una chiave privata TLS e un certificato in `istio-system namespace` Come descritto in TLS secrets (segreti TLS).

   Esempio di comando:

   ```
   kubectl create secret tls [tls secret name]
   --key="tls.key"
   --cert="tls.crt" -n istio-system
   ```

   > Il nome del segreto deve corrispondere a. `spec.tls.secretName` fornito in `istio-ingress.yaml` file.

4. Implementare una risorsa income in `netapp-acc` (O con nome personalizzato) che utilizza lo spazio dei nomi v1beta1 (deprecato in Kubernetes versione inferiore a o 1.22) o il tipo di risorsa v1 per uno schema obsoleto o per uno schema nuovo:

   Uscita:

```
apiVersion: networking.k8s.io/v1beta1
kind: IngressClass
metadata:
  name: istio
spec:
  controller: istio.io/ingress-controller
---
apiVersion: networking.k8s.io/v1beta1
kind: Ingress
metadata:
  name: ingress
  namespace: istio-system
spec:
  ingressClassName: istio
  tls:
  - hosts:
    - <ACC addess>
    secretName: [tls secret name]
  rules:
  - host: [ACC addess]
    http:
      paths:
      - path: /
        pathType: Prefix
        backend:
          serviceName: traefik
          servicePort: 80
```

Per il nuovo schema v1, seguire questo esempio:

```
kubectl apply -f istio-Ingress.yaml
```

Uscita:

```
apiVersion: networking.k8s.io/v1
kind: IngressClass
metadata:
  name: istio
spec:
  controller: istio.io/ingress-controller
---
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: ingress
  namespace: istio-system
spec:
  ingressClassName: istio
  tls:
  - hosts:
    - <ACC addess>
    secretName: [tls secret name]
  rules:
  - host: [ACC addess]
    http:
      paths:
      - path: /
        pathType: Prefix
        backend:
          service:
            name: traefik
            port:
              number: 80
```

5. Implementare Astra Control Center come di consueto.
6. Controllare lo stato dell'ingresso:

```
kubectl get ingress -n netapp-acc
```

Risposta:

```
NAME    CLASS HOSTS             ADDRESS         PORTS   AGE
ingress istio astra.example.com 172.16.103.248  80, 443 1h
```

Procedura per il controller di ingresso Nginx

1. Creare un segreto di tipo[`kubernetes.io/tls`] Per una chiave privata TLS e un certificato in `netapp-acc` (o con nome personalizzato) come descritto in "Segreti TLS".

2. Implementare una risorsa income in `netapp-acc` (o con nome personalizzato) namespace utilizzando `v1beta1` (Obsoleto in Kubernetes versione inferiore a o 1.22) o. `v1` tipo di risorsa per uno schema obsoleto o nuovo:

   a. Per a. `v1beta1` schema obsoleto, seguire questo esempio:

```
apiVersion: extensions/v1beta1
Kind: IngressClass
metadata:
  name: ingress-acc
  namespace: [netapp-acc or custom namespace]
  annotations:
    kubernetes.io/ingress.class: [class name for nginx controller]
spec:
  tls:
  - hosts:
    - <ACC address>
    secretName: [tls secret name]
  rules:
  - host: [ACC address]
    http:
      paths:
      - backend:
        serviceName: traefik
        servicePort: 80
        pathType: ImplementationSpecific
```

   b. Per `v1` nuovo schema, seguire questo esempio:

```yaml
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: netapp-acc-ingress
  namespace: [netapp-acc or custom namespace]
spec:
  ingressClassName: [class name for nginx controller]
  tls:
  - hosts:
    - <ACC address>
    secretName: [tls secret name]
  rules:
  - host: <ACC addess>
    http:
      paths:
        - path:
          backend:
            service:
              name: traefik
              port:
                number: 80
          pathType: ImplementationSpecific
```

Procedura per il controller di ingresso OpenShift

1. Procurarsi il certificato e ottenere la chiave, il certificato e i file CA pronti per l'uso con il percorso OpenShift.
2. Creare il percorso OpenShift:

```
oc create route edge --service=traefik
--port=web -n [netapp-acc or custom namespace]
--insecure-policy=Redirect --hostname=<ACC address>
--cert=cert.pem --key=key.pem
```

Accedere all'interfaccia utente di Astra Control Center

Dopo aver installato Astra Control Center, si modifica la password dell'amministratore predefinito e si accede alla dashboard dell'interfaccia utente di Astra Control Center.

Fasi

1. In un browser, immettere l'FQDN utilizzato in `astraAddress` in `astra_control_center_min.yaml` CR quando Astra Control Center è stato installato.
2. Accettare i certificati autofirmati quando richiesto.

> È possibile creare un certificato personalizzato dopo l'accesso.

3. Nella pagina di accesso di Astra Control Center, inserire il valore utilizzato per `email` poll `astra_control_center_min.yaml` CR quando Astra Control Center è stato installato, seguito dalla password monouso (`ACC-[UUID]`).

> Se si immette una password errata per tre volte, l'account admin viene bloccato per 15 minuti.

4. Selezionare **Login**.
5. Modificare la password quando richiesto.

> Se si tratta del primo accesso e si dimentica la password e non sono ancora stati creati altri account utente amministrativi, contattare il supporto NetApp per assistenza per il recupero della password.

6. (Facoltativo) rimuovere il certificato TLS autofirmato esistente e sostituirlo con un "Certificato TLS personalizzato firmato da un'autorità di certificazione (CA)".

Risolvere i problemi di installazione

Se uno dei servizi è in `Error` stato, è possibile esaminare i registri. Cercare i codici di risposta API nell'intervallo da 400 a 500. Questi indicano il luogo in cui si è verificato un guasto.

Fasi

1. Per esaminare i registri dell'operatore di Astra Control Center, immettere quanto segue:

```
kubectl logs --follow -n netapp-acc-operator $(kubectl get pods -n
netapp-acc-operator -o name) -c manager
```

Cosa succederà

Completare l'implementazione eseguendo "attività di installazione".

=
:allow-uri-read:

Comprendere le restrizioni delle policy di sicurezza del pod

Astra Control Center supporta la limitazione dei privilegi tramite PSP (Pod Security policy). Le policy di sicurezza Pod consentono di limitare gli utenti o i gruppi in grado di eseguire i container e i privilegi che questi possono avere.

Alcune distribuzioni di Kubernetes, come RKE2, dispongono di un criterio di protezione pod predefinito troppo restrittivo e causano problemi durante l'installazione di Astra Control Center.

È possibile utilizzare le informazioni e gli esempi inclusi qui per comprendere le policy di sicurezza dei pod create da Astra Control Center e configurare le policy di sicurezza dei pod che forniscono la protezione necessaria senza interferire con le funzioni di Astra Control Center.

PSP installati da Astra Control Center

Astra Control Center crea diverse policy di sicurezza del pod durante l'installazione. Alcune di queste sono permanenti, alcune vengono create durante determinate operazioni e vengono rimosse una volta completata l'operazione.

PSP creati durante l'installazione

Durante l'installazione di Astra Control Center, l'operatore di Astra Control Center installa un criterio di protezione pod personalizzato, un oggetto ruolo e un oggetto RoleBinding per supportare la distribuzione dei servizi Astra Control Center nello spazio dei nomi Astra Control Center.

I nuovi criteri e oggetti hanno i seguenti attributi:

```
kubectl get psp

NAME                            PRIV    CAPS          SELINUX    RUNASUSER
FSGROUP     SUPGROUP    READONLYROOTFS   VOLUMES
avp-psp                         false                 RunAsAny   RunAsAny
RunAsAny    RunAsAny    false            *
netapp-astra-deployment-psp     false                 RunAsAny   RunAsAny
RunAsAny    RunAsAny    false            *

kubectl get role

NAME                                     CREATED AT
netapp-astra-deployment-role             2022-06-27T19:34:58Z

kubectl get rolebinding

NAME                                     ROLE
AGE
netapp-astra-deployment-rb               Role/netapp-astra-deployment-role
32m
```

PSP creati durante le operazioni di backup

Durante le operazioni di backup, Astra Control Center crea un criterio di protezione Pod dinamico, un oggetto ClusterRole e un oggetto RoleBinding. Questi supportano il processo di backup, che avviene in uno spazio dei nomi separato.

I nuovi criteri e oggetti hanno i seguenti attributi:

```
kubectl get psp

NAME                            PRIV    CAPS
SELINUX    RUNASUSER          FSGROUP     SUPGROUP    READONLYROOTFS
VOLUMES
netapp-astra-backup             false   DAC_READ_SEARCH
RunAsAny   RunAsAny           RunAsAny    RunAsAny    false            *

kubectl get role

NAME                  CREATED AT
netapp-astra-backup   2022-07-21T00:00:00Z

kubectl get rolebinding

NAME                  ROLE                        AGE
netapp-astra-backup   Role/netapp-astra-backup   62s
```

PSP creati durante la gestione del cluster

Quando gestisci un cluster, Astra Control Center installa l'operatore di monitoraggio netapp nel cluster gestito. Questo operatore crea un criterio di protezione pod, un oggetto ClusterRole e un oggetto RoleBinding per implementare i servizi di telemetria nello spazio dei nomi Astra Control Center.

I nuovi criteri e oggetti hanno i seguenti attributi:

```
kubectl get psp

NAME                            PRIV    CAPS
SELINUX    RUNASUSER          FSGROUP     SUPGROUP    READONLYROOTFS
VOLUMES
netapp-monitoring-psp-nkmo      true    AUDIT_WRITE,NET_ADMIN,NET_RAW
RunAsAny   RunAsAny           RunAsAny    RunAsAny    false            *

kubectl get role

NAME                                            CREATED AT
netapp-monitoring-role-privileged               2022-07-21T00:00:00Z

kubectl get rolebinding

NAME                                                    ROLE
AGE
netapp-monitoring-role-binding-privileged               Role/netapp-
monitoring-role-privileged              2m5s
```

Abilitare la comunicazione di rete tra spazi dei nomi

Alcuni ambienti utilizzano costrutti NetworkPolicy per limitare il traffico tra gli spazi dei nomi. L'operatore di Astra Control Center, Astra Control Center e Astra Plugin per VMware vSphere sono tutti in spazi dei nomi diversi. I servizi in questi diversi spazi dei nomi devono essere in grado di comunicare tra loro. Per attivare questa comunicazione, attenersi alla seguente procedura.

Fasi

1. Eliminare le risorse NetworkPolicy presenti nello spazio dei nomi di Astra Control Center:

```
kubectl get networkpolicy -n netapp-acc
```

2. Per ogni oggetto NetworkPolicy restituito dal comando precedente, utilizzare il seguente comando per eliminarlo. Sostituire <OBJECT_NAME> con il nome dell'oggetto restituito:

```
kubectl delete networkpolicy <OBJECT_NAME> -n netapp-acc
```

3. Applicare il seguente file di risorse per configurare l'oggetto acc-avp-network-policy per consentire ai servizi Astra Plugin per VMware vSphere di effettuare richieste ai servizi Astra Control Center. Sostituire le informazioni tra parentesi <> con quelle dell'ambiente:

```
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
  name: acc-avp-network-policy
  namespace: <ACC_NAMESPACE_NAME> # REPLACE THIS WITH THE ASTRA CONTROL
CENTER NAMESPACE NAME
spec:
  podSelector: {}
  policyTypes:
    - Ingress
  ingress:
    - from:
        - namespaceSelector:
            matchLabels:
              kubernetes.io/metadata.name: <PLUGIN_NAMESPACE_NAME> #
REPLACE THIS WITH THE ASTRA PLUGIN FOR VMWARE VSPHERE NAMESPACE NAME
```

4. Applicare il seguente file di risorse per configurare l'oggetto acc-operator-network-policy per consentire all'operatore Astra Control Center di comunicare con i servizi Astra Control Center. Sostituire le informazioni tra parentesi <> con quelle dell'ambiente:

```
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
  name: acc-operator-network-policy
  namespace: <ACC_NAMESPACE_NAME> # REPLACE THIS WITH THE ASTRA CONTROL
CENTER NAMESPACE NAME
spec:
  podSelector: {}
  policyTypes:
    - Ingress
  ingress:
    - from:
        - namespaceSelector:
            matchLabels:
              kubernetes.io/metadata.name: <NETAPP-ACC-OPERATOR> #
REPLACE THIS WITH THE OPERATOR NAMESPACE NAME
```

Rimuovere le limitazioni delle risorse

Alcuni ambienti utilizzano gli oggetti ResourceQuotas e LimitRanges per impedire alle risorse di uno spazio dei nomi di consumare tutta la CPU e la memoria disponibili nel cluster. Astra Control Center non imposta limiti massimi, pertanto non sarà conforme a tali risorse. È necessario rimuoverli dagli spazi dei nomi in cui si intende installare Astra Control Center.

Per recuperare e rimuovere le quote e i limiti, procedere come segue. In questi esempi, l'output del comando viene visualizzato immediatamente dopo il comando.

Fasi

1. Ottieni le quote delle risorse nello spazio dei nomi netapp-acc:

   ```
   kubectl get quota -n netapp-acc
   ```

   Risposta:

   ```
   NAME          AGE   REQUEST                                          LIMIT
   pods-high     16s   requests.cpu: 0/20, requests.memory: 0/100Gi
   limits.cpu: 0/200, limits.memory: 0/1000Gi
   pods-low      15s   requests.cpu: 0/1, requests.memory: 0/1Gi
   limits.cpu: 0/2, limits.memory: 0/2Gi
   pods-medium   16s   requests.cpu: 0/10, requests.memory: 0/20Gi
   limits.cpu: 0/20, limits.memory: 0/200Gi
   ```

2. Eliminare tutte le quote delle risorse in base al nome:

```
kubectl delete resourcequota  pods-high -n netapp-acc
```

```
kubectl delete resourcequota  pods-low -n netapp-acc
```

```
kubectl delete resourcequota  pods-medium -n netapp-acc
```

3. Ottieni gli intervalli limite nello spazio dei nomi netapp-acc:

```
kubectl get limits -n netapp-acc
```

Risposta:

```
NAME              CREATED AT
cpu-limit-range   2022-06-27T19:01:23Z
```

4. Eliminare gli intervalli di limiti in base al nome:

```
kubectl delete limitrange cpu-limit-range -n netapp-acc
```

=
:allow-uri-read:

Installare Astra Control Center utilizzando OpenShift OperatorHub

Se utilizzi Red Hat OpenShift, puoi installare Astra Control Center usando l'operatore certificato Red Hat. Seguire questa procedura per installare Astra Control Center da "Catalogo Red Hat Ecosystem" Oppure utilizzando Red Hat OpenShift Container Platform.

Una volta completata questa procedura, tornare alla procedura di installazione per completare la "fasi rimanenti" per verificare che l'installazione sia riuscita e accedere.

Di cosa hai bisogno

- "Prima di iniziare l'installazione, preparare l'ambiente per l'implementazione di Astra Control Center".
- Dal tuo cluster OpenShift, assicurati che tutti gli operatori del cluster siano in buono stato (`available` è `true`):

```
oc get clusteroperators
```

- Dal cluster OpenShift, assicurati che tutti i servizi API siano in buono stato (`available` è `true`):

```
oc get apiservices
```

- Creare un indirizzo FQDN per Astra Control Center nel data center.
- Ottenere le autorizzazioni necessarie e l'accesso alla piattaforma container Red Hat OpenShift per eseguire le fasi di installazione descritte.
- Se nel cluster esiste già un cert-manager, è necessario eseguirne alcune "fasi preliminari" In modo che Astra Control Center non installi il proprio cert-manager.

Fasi

- Scarica e disimballa il bundle Astra Control Center
- Installare il plug-in NetApp Astra kubectl
- Aggiungere le immagini al registro locale
- Individuare la pagina di installazione dell'operatore
- Installare l'operatore
- Installare Astra Control Center

Scarica e disimballa il bundle Astra Control Center

1. Scarica il bundle Astra Control Center (`astra-control-center-[version].tar.gz`) da "Sito di supporto NetApp".
2. Scarica la zip dei certificati e delle chiavi di Astra Control Center dal "Sito di supporto NetApp".
3. (Facoltativo) utilizzare il seguente comando per verificare la firma del bundle:

```
openssl dgst -sha256 -verify AstraControlCenter-public.pub -signature
astra-control-center-[version].tar.gz.sig astra-control-center-
[version].tar.gz
```

4. Estrarre le immagini:

```
tar -vxzf astra-control-center-[version].tar.gz
```

Installare il plug-in NetApp Astra kubectl

NetApp Astra `kubectl` Il plug-in della riga di comando consente di risparmiare tempo durante l'esecuzione di attività comuni associate all'implementazione e all'aggiornamento di Astra Control Center.

Di cosa hai bisogno

NetApp fornisce binari per il plug-in per diverse architetture CPU e sistemi operativi. Prima di eseguire questa attività, è necessario conoscere la CPU e il sistema operativo in uso. Sui sistemi operativi Linux e Mac, è possibile utilizzare `uname -a` per raccogliere queste informazioni.

Fasi

1. Elencare NetApp Astra disponibile `kubectl` Binari del plug-in e annotare il nome del file necessario per il

sistema operativo e l'architettura della CPU:

```
ls kubectl-astra/
```

2. Copiare il file nella stessa posizione dello standard `kubectl` utility. In questo esempio, il `kubectl` l'utility si trova in `/usr/local/bin` directory. Sostituire `<binary-name>` con il nome del file desiderato:

```
cp kubectl-astra/<binary-name> /usr/local/bin/kubectl-astra
```

Aggiungere le immagini al registro locale

1. Completare la sequenza di passaggi appropriata per il motore dei container:

Docker

1. Passare alla directory Astra:

```
cd acc
```

2. inserire le immagini del pacchetto nella directory delle immagini di Astra Control Center nel registro locale. Eseguire le seguenti sostituzioni prima di eseguire il comando:
   - Sostituire BUNDLE_FILE con il nome del file bundle Astra Control (ad esempio, `acc.manifest.yaml`).
   - Sostituire MY_REGISTRY con l'URL del repository Docker.
   - Sostituire MY_REGISTRY_USER con il nome utente.
   - Sostituire MY_REGISTRY_TOKEN con un token autorizzato per il Registro di sistema.

```
kubectl astra packages push-images -m BUNDLE_FILE -r MY_REGISTRY
-u MY_REGISTRY_USER -p MY_REGISTRY_TOKEN
```

Podman

1. Accedere al Registro di sistema:

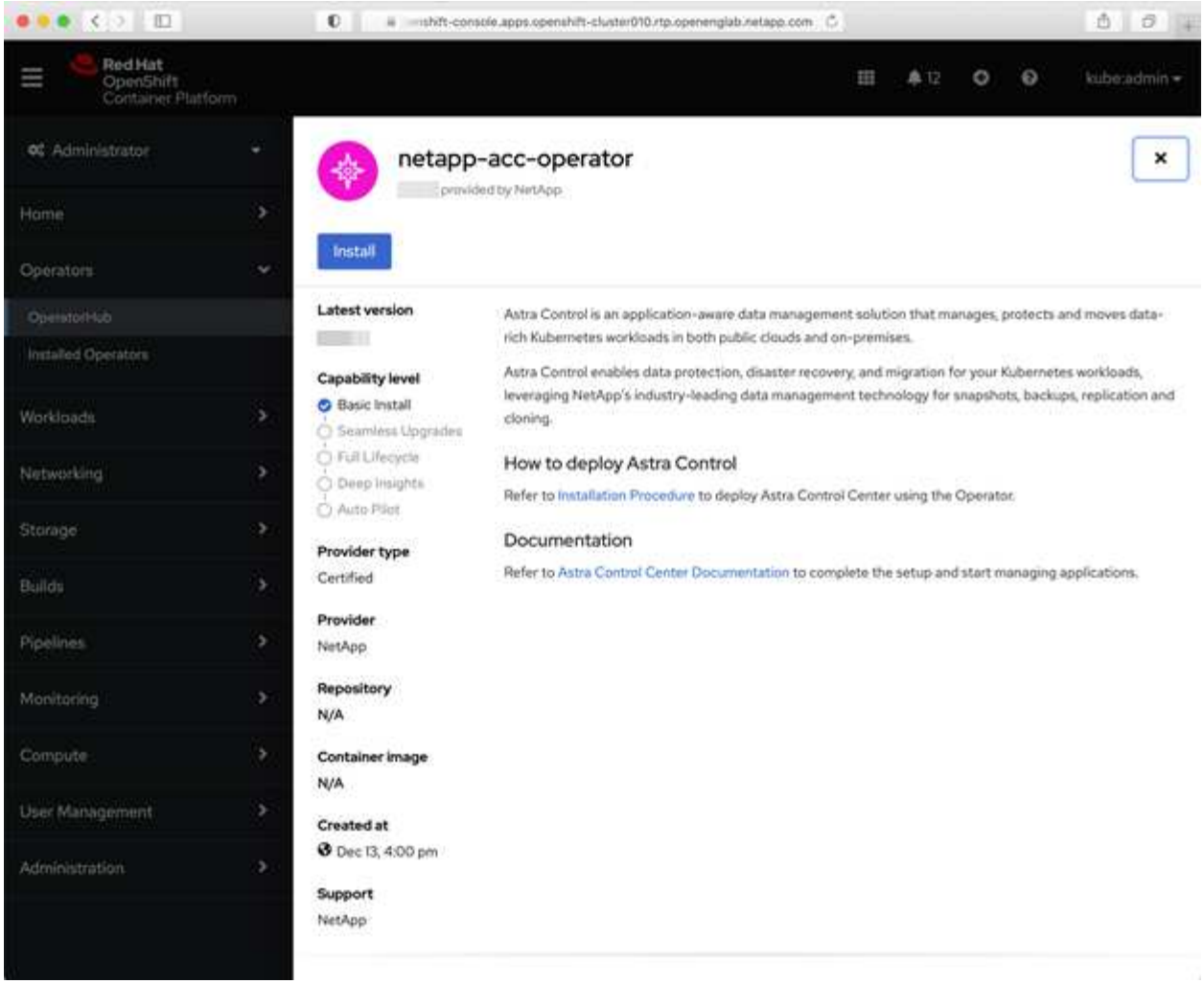
```
podman login [your_registry_path]
```

2. Eseguire il seguente script, eseguendo la sostituzione <YOUR_REGISTRY> come indicato nei commenti:

```
# You need to be at the root of the tarball.
# You should see these files to confirm correct location:
#   acc.manifest.yaml
#   acc/

# Replace <YOUR_REGISTRY> with your own registry (e.g
registry.customer.com or registry.customer.com/testing, etc..)
export REGISTRY=<YOUR_REGISTRY>
export PACKAGENAME=acc
export PACKAGEVERSION=22.08.1-26
export DIRECTORYNAME=acc
for astraImageFile in $(ls ${DIRECTORYNAME}/images/*.tar) ; do
  # Load to local cache
  astraImage=$(podman load --input ${astraImageFile} | sed 's/Loaded
image(s): //')

  # Remove path and keep imageName.
  astraImageNoPath=$(echo ${astraImage} | sed 's:.*/::')

  # Tag with local image repo.
  podman tag ${astraImage} ${REGISTRY}/netapp/astra/${PACKAGENAME}
/${PACKAGEVERSION}/${astraImageNoPath}

  # Push to the local repo.
  podman push ${REGISTRY}/netapp/astra/${PACKAGENAME}/
${PACKAGEVERSION}/${astraImageNoPath}
done
```

Individuare la pagina di installazione dell'operatore

1. Completare una delle seguenti procedure per accedere alla pagina di installazione dell'operatore:
   - Dalla console Web Red Hat OpenShift:



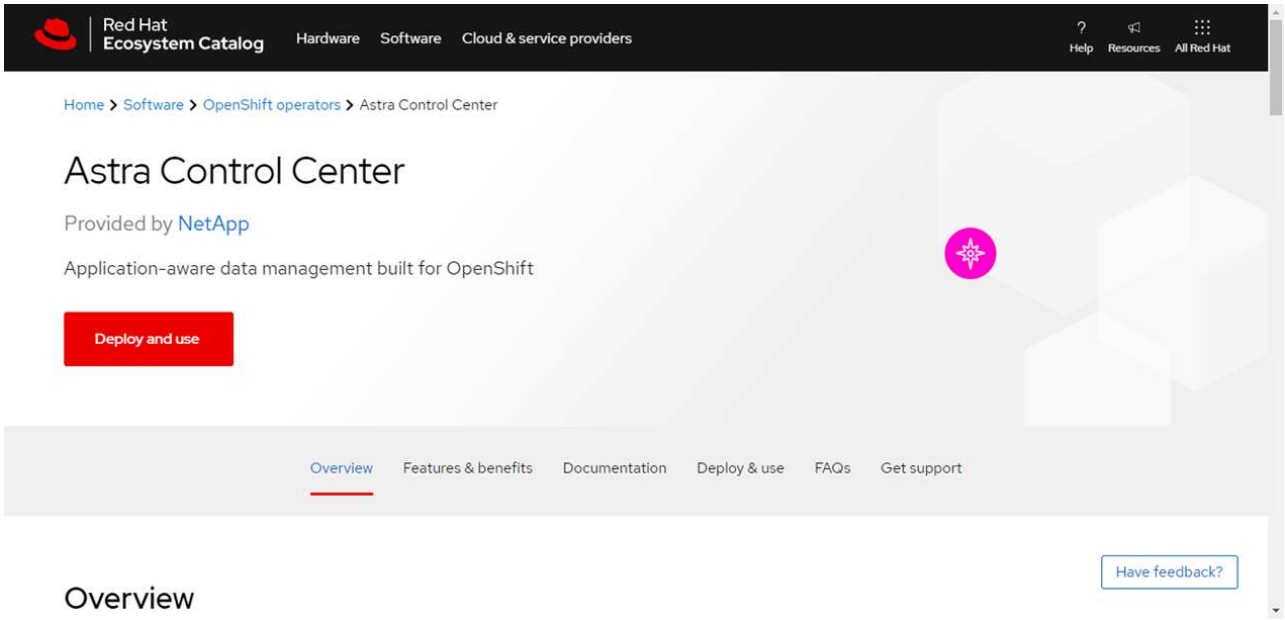
i. Accedere all'interfaccia utente di OpenShift Container Platform.

ii. Dal menu laterale, selezionare **Operator (operatori) > OperatorHub**.

iii. Selezionare l'operatore di NetApp Astra Control Center.

iv. Selezionare **Installa**.

- Dal Red Hat Ecosystem Catalog:



i. Selezionare NetApp Astra Control Center "operatore".

ii. Selezionare **Deploy and Use** (implementazione e utilizzo).

Installare l'operatore

1. Completare la pagina **Install Operator** (Installazione operatore) e installare l'operatore:

   L'operatore sarà disponibile in tutti gli spazi dei nomi dei cluster.

   a. Selezionare lo spazio dei nomi dell'operatore o. `netapp-acc-operator` lo spazio dei nomi verrà creato automaticamente come parte dell'installazione dell'operatore.

   b. Selezionare una strategia di approvazione manuale o automatica.

   Si consiglia l'approvazione manuale. Per ogni cluster dovrebbe essere in esecuzione una sola istanza dell'operatore.

   c. Selezionare **Installa**.

   Se è stata selezionata una strategia di approvazione manuale, verrà richiesto di approvare il piano di installazione manuale per questo operatore.

2. Dalla console, accedere al menu OperatorHub e verificare che l'installazione dell'operatore sia stata eseguita correttamente.

Installare Astra Control Center

1. Dalla console nella vista dettagli dell'operatore Astra Control Center, selezionare `Create instance` Nella sezione API fornite.

2. Completare il `Create AstraControlCenter` campo del modulo:

   a. Mantenere o regolare il nome di Astra Control Center.

   b. (Facoltativo) attivare o disattivare il supporto automatico. Si consiglia di mantenere la funzionalità di supporto automatico.

c. Inserire l'indirizzo di Astra Control Center. Non entrare `http://` oppure `https://` nell'indirizzo.
d. Inserire la versione di Astra Control Center, ad esempio 21.12.60.
e. Immettere un nome account, un indirizzo e-mail e un cognome amministratore.
f. Mantenere la policy di recupero del volume predefinita.
g. In **Image Registry**, immettere il percorso locale del Registro di sistema dell'immagine container. Non entrare `http://` oppure `https://` nell'indirizzo.
h. Se si utilizza un registro che richiede l'autenticazione, immettere il segreto.
i. Inserire il nome admin.
j. Configurare la scalabilità delle risorse.
k. Mantenere la classe di storage predefinita.
l. Definire le preferenze di gestione CRD.

3. Selezionare `Create`.

Cosa succederà

Verificare che Astra Control Center sia stato installato correttamente e completare il "fasi rimanenti" per accedere. Inoltre, completerai l'implementazione eseguendo anche questa operazione "attività di installazione".

Installare il centro di controllo Astra con un backend di storage Cloud Volumes ONTAP

Con Astra Control Center, puoi gestire le tue app in un ambiente di cloud ibrido con cluster Kubernetes e istanze di Cloud Volumes ONTAP autogestiti. Puoi implementare Astra Control Center nei tuoi cluster Kubernetes on-premise o in uno dei cluster Kubernetes autogestiti nell'ambiente cloud.

Con una di queste implementazioni, è possibile eseguire operazioni di gestione dei dati delle applicazioni utilizzando Cloud Volumes ONTAP come back-end dello storage. È inoltre possibile configurare un bucket S3 come destinazione del backup.

Per installare Astra Control Center in Amazon Web Services (AWS), Google Cloud Platform (GCP) e Microsoft Azure con un backend di storage Cloud Volumes ONTAP, eseguire i seguenti passaggi a seconda dell'ambiente cloud in uso.

- Implementare Astra Control Center in Amazon Web Services
- Implementare Astra Control Center nella piattaforma Google Cloud
- Implementare Astra Control Center in Microsoft Azure

Puoi gestire le tue applicazioni nelle distribuzioni con cluster Kubernetes autogestiti, come OpenShift Container Platform (OCP). Solo i cluster OCP autogestiti sono validati per l'implementazione di Astra Control Center.

Implementare Astra Control Center in Amazon Web Services

Puoi implementare Astra Control Center su un cluster Kubernetes autogestito ospitato su un cloud pubblico Amazon Web Services (AWS).

Ciò di cui hai bisogno per AWS

Prima di implementare Astra Control Center in AWS, sono necessari i seguenti elementi:

- Licenza Astra Control Center. Vedere "Requisiti di licenza di Astra Control Center".
- "Soddisfare i requisiti di Astra Control Center".
- Account NetApp Cloud Central
- Se si utilizza OCP, autorizzazioni Red Hat OpenShift Container Platform (OCP) (a livello di spazio dei nomi per creare i pod)
- Credenziali AWS, Access ID e Secret Key con autorizzazioni che consentono di creare bucket e connettori
- Accesso e login al Registro dei container elastici (ECR) dell'account AWS
- Per accedere all'interfaccia utente di Astra Control, è necessario immettere AWS Hosted zone e Route 53

Requisiti dell'ambiente operativo per AWS

Astra Control Center richiede il seguente ambiente operativo per AWS:

- Red Hat OpenShift Container Platform 4.8

> Assicurarsi che l'ambiente operativo scelto per ospitare Astra Control Center soddisfi i requisiti di base delle risorse descritti nella documentazione ufficiale dell'ambiente.

Astra Control Center richiede le seguenti risorse oltre ai requisiti delle risorse dell'ambiente:

Componente	Requisito
Capacità di storage NetApp Cloud Volumes ONTAP di back-end	Almeno 300 GB disponibili
Nodi di lavoro (requisito AWS EC2)	Almeno 3 nodi di lavoro in totale, con 4 core vCPU e 12 GB di RAM ciascuno
Bilanciamento del carico	Tipo di servizio "LoadBalancer" disponibile per il traffico in ingresso da inviare ai servizi nel cluster dell'ambiente operativo
FQDN	Metodo per indirizzare l'FQDN di Astra Control Center all'indirizzo IP con bilanciamento del carico
Astra Trident (installato come parte del rilevamento dei cluster Kubernetes in NetApp Cloud Manager)	Astra Trident 21.04 o versione successiva installata e configurata e NetApp ONTAP versione 9.5 o successiva come backend di storage

Componente	Requisito
Registro delle immagini	È necessario disporre di un registro privato esistente, ad esempio AWS Elastic Container Registry, in cui è possibile trasferire le immagini di build di Astra Control Center. È necessario fornire l'URL del registro delle immagini in cui verranno caricate le immagini. Il cluster ospitato da Astra Control Center e il cluster gestito devono avere accesso alla stessa immagine di registro per poter eseguire il backup e il ripristino delle applicazioni utilizzando l'immagine basata su Restic.
Configurazione di Astra Trident/ONTAP	Astra Control Center richiede la creazione e l'impostazione di una classe di storage come classe di storage predefinita. Il centro di controllo Astra supporta le seguenti classi di storage Kubernetes create quando si importa il cluster Kubernetes in ONTAP. Questi sono forniti da Astra Trident: • `vsaworkingenvironment-<>-ha-nas csi.trident.netapp.io` • `vsaworkingenvironment-<>-ha-san csi.trident.netapp.io` • `vsaworkingenvironment-<>-single-nas csi.trident.netapp.io` • `vsaworkingenvironment-<>-single-san csi.trident.netapp.io`

Questi requisiti presuppongono che Astra Control Center sia l'unica applicazione in esecuzione nell'ambiente operativo. Se nell'ambiente sono in esecuzione applicazioni aggiuntive, modificare di conseguenza questi requisiti minimi.

Il token del Registro di sistema AWS scade tra 12 ore, dopodiché sarà necessario rinnovare il segreto del Registro di sistema dell'immagine Docker.

Panoramica dell'implementazione per AWS

Di seguito viene fornita una panoramica del processo di installazione di Astra Control Center per AWS con Cloud Volumes ONTAP come backend di storage.

Ciascuna di queste fasi viene illustrata più dettagliatamente di seguito.

1. Assicurarsi di disporre di autorizzazioni IAM sufficienti.
2. Installare un cluster RedHat OpenShift su AWS.
3. Configurare AWS.
4. Configurare NetApp Cloud Manager.
5. Installare Astra Control Center.

Assicurarsi di disporre di autorizzazioni IAM sufficienti

Assicurarsi di disporre di ruoli e autorizzazioni IAM sufficienti per installare un cluster RedHat OpenShift e un connettore NetApp Cloud Manager.

Vedere "Credenziali AWS iniziali".

Installare un cluster RedHat OpenShift su AWS

Installare un cluster RedHat OpenShift Container Platform su AWS.

Per istruzioni sull'installazione, vedere "Installazione di un cluster su AWS in OpenShift Container Platform".

Configurare AWS

Quindi, configurare AWS per creare una rete virtuale, configurare istanze di calcolo EC2, creare un bucket AWS S3, creare un Elastic Container Register (ECR) per ospitare le immagini di Astra Control Center e inviare le immagini a questo registro.

Seguire la documentazione di AWS per completare i seguenti passaggi. Vedere "Documentazione di installazione di AWS".

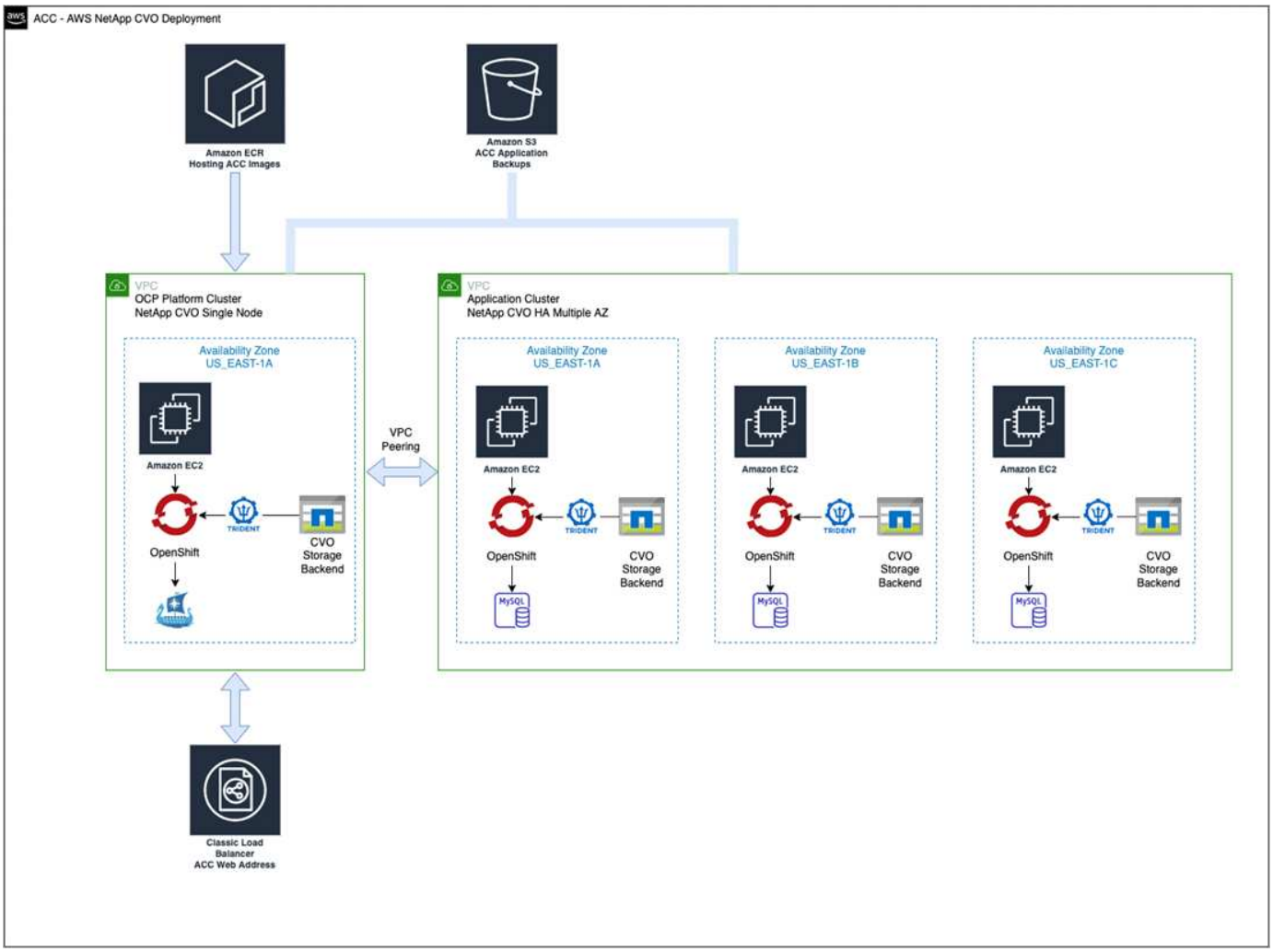
1. Creare una rete virtuale AWS.
2. Esaminare le istanze di calcolo EC2. Può trattarsi di un server bare metal o di macchine virtuali in AWS.
3. Se il tipo di istanza non corrisponde già ai requisiti minimi di risorsa Astra per i nodi master e worker, modificare il tipo di istanza in AWS per soddisfare i requisiti Astra. Vedere "Requisiti di Astra Control Center".
4. Creare almeno un bucket AWS S3 per memorizzare i backup.
5. Creare un AWS Elastic Container Registry (ECR) per ospitare tutte le immagini ACC.

> Se non si crea ECR, il centro di controllo Astra non può accedere ai dati di monitoraggio da un cluster contenente Cloud Volumes ONTAP con un backend AWS. Il problema si verifica quando il cluster che si tenta di rilevare e gestire utilizzando Astra Control Center non dispone dell'accesso ad AWS ECR.

6. Trasferire le immagini ACC nel registro definito.

> Il token AWS Elastic Container Registry (ECR) scade dopo 12 ore e causa il fallimento delle operazioni di cloni tra cluster. Questo problema si verifica quando si gestisce un backend di storage da Cloud Volumes ONTAP configurato per AWS. Per correggere questo problema, autenticare nuovamente con ECR e generare un nuovo segreto per la ripresa delle operazioni di clonazione.

Ecco un esempio di implementazione di AWS:



Configurare NetApp Cloud Manager

Utilizzando Cloud Manager, creare un'area di lavoro, aggiungere un connettore ad AWS, creare un ambiente di lavoro e importare il cluster.

Seguire la documentazione di Cloud Manager per completare i seguenti passaggi. Vedere quanto segue:

- "Introduzione a Cloud Volumes ONTAP in AWS".
- "Creare un connettore in AWS utilizzando Cloud Manager"

Fasi

1. Aggiungi le tue credenziali a Cloud Manager.
2. Creare un'area di lavoro.
3. Aggiungere un connettore per AWS. Scegliere AWS come provider.
4. Crea un ambiente di lavoro per il tuo ambiente cloud.
   a. Location: "Amazon Web Services (AWS)"
   b. Tipo: "Cloud Volumes ONTAP ha"
5. Importare il cluster OpenShift. Il cluster si connetterà all'ambiente di lavoro appena creato.
   a. Per visualizzare i dettagli del cluster NetApp, selezionare **K8s** > **elenco cluster** > **Dettagli cluster**.

b. Nell'angolo in alto a destra, prendere nota della versione di Trident.

c. Si noti che le classi di storage cluster Cloud Volumes ONTAP mostrano NetApp come provider.

   In questo modo, il cluster Red Hat OpenShift viene importato e viene assegnata una classe di storage predefinita. Selezionare la classe di storage. Trident viene installato automaticamente come parte del processo di importazione e rilevamento.

6. Tenere presenti tutti i volumi e i volumi persistenti in questa implementazione di Cloud Volumes ONTAP.

> Cloud Volumes ONTAP può funzionare come nodo singolo o in alta disponibilità. Se ha è attivato, annotare lo stato ha e lo stato di implementazione del nodo in esecuzione in AWS.

Installare Astra Control Center

Seguire lo standard "Istruzioni di installazione di Astra Control Center".

> AWS utilizza il tipo di bucket S3 generico.

Implementare Astra Control Center nella piattaforma Google Cloud

Puoi implementare Astra Control Center su un cluster Kubernetes autogestito ospitato su un cloud pubblico Google Cloud Platform (GCP).

Cosa ti serve per GCP

Prima di implementare Astra Control Center in GCP, sono necessari i seguenti elementi:

- Licenza Astra Control Center. Vedere "Requisiti di licenza di Astra Control Center".
- "Soddisfare i requisiti di Astra Control Center".
- Account NetApp Cloud Central
- Se si utilizza OCP, Red Hat OpenShift Container Platform (OCP) 4.10
- Se si utilizza OCP, autorizzazioni Red Hat OpenShift Container Platform (OCP) (a livello di spazio dei nomi per creare i pod)
- GCP Service account con autorizzazioni che consentono di creare bucket e connettori

Requisiti dell'ambiente operativo per GCP

> Assicurarsi che l'ambiente operativo scelto per ospitare Astra Control Center soddisfi i requisiti di base delle risorse descritti nella documentazione ufficiale dell'ambiente.

Astra Control Center richiede le seguenti risorse oltre ai requisiti delle risorse dell'ambiente:

Componente	Requisito
Capacità di storage NetApp Cloud Volumes ONTAP di back-end	Almeno 300 GB disponibili
Nodi di lavoro (requisito di calcolo GCP)	Almeno 3 nodi di lavoro in totale, con 4 core vCPU e 12 GB di RAM ciascuno

Componente	Requisito
Bilanciamento del carico	Tipo di servizio "LoadBalancer" disponibile per il traffico in ingresso da inviare ai servizi nel cluster dell'ambiente operativo
FQDN (GCP DNS ZONE)	Metodo per indirizzare l'FQDN di Astra Control Center all'indirizzo IP con bilanciamento del carico
Astra Trident (installato come parte del rilevamento dei cluster Kubernetes in NetApp Cloud Manager)	Astra Trident 21.04 o versione successiva installata e configurata e NetApp ONTAP versione 9.5 o successiva come backend di storage
Registro delle immagini	È necessario disporre di un registro privato esistente, ad esempio Google Container Registry, in cui è possibile trasferire le immagini di build di Astra Control Center. È necessario fornire l'URL del registro delle immagini in cui verranno caricate le immagini. È necessario abilitare l'accesso anonimo per estrarre le immagini Restic per i backup.
Configurazione di Astra Trident/ONTAP	Astra Control Center richiede la creazione e l'impostazione di una classe di storage come classe di storage predefinita. Il centro di controllo Astra supporta le seguenti classi di storage Kubernetes create quando si importa il cluster Kubernetes in ONTAP. Questi sono forniti da Astra Trident: • `vsaworkingenvironment-<>-ha-nas csi.trident.netapp.io` • `vsaworkingenvironment-<>-ha-san csi.trident.netapp.io` • `vsaworkingenvironment-<>-single-nas csi.trident.netapp.io` • `vsaworkingenvironment-<>-single-san csi.trident.netapp.io`

Questi requisiti presuppongono che Astra Control Center sia l'unica applicazione in esecuzione nell'ambiente operativo. Se nell'ambiente sono in esecuzione applicazioni aggiuntive, modificare di conseguenza questi requisiti minimi.

Panoramica dell'implementazione per GCP

Di seguito viene fornita una panoramica del processo di installazione di Astra Control Center su un cluster OCP autogestiti in GCP con Cloud Volumes ONTAP come backend di storage.

Ciascuna di queste fasi viene illustrata più dettagliatamente di seguito.

1. Installare un cluster RedHat OpenShift su GCP.
2. Crea un progetto GCP e un cloud privato virtuale.

3. Assicurarsi di disporre di autorizzazioni IAM sufficienti.
4. Configurare GCP.
5. Configurare NetApp Cloud Manager.
6. Installare e configurare Astra Control Center.

Installare un cluster RedHat OpenShift su GCP

Il primo passo consiste nell'installare un cluster RedHat OpenShift su GCP.

Per istruzioni sull'installazione, consultare quanto segue:

- "Installazione di un cluster OpenShift in GCP"
- "Creazione di un account di servizio GCP"

Crea un progetto GCP e un cloud privato virtuale

Creare almeno un progetto GCP e Virtual Private Cloud (VPC).

> OpenShift potrebbe creare i propri gruppi di risorse. Inoltre, è necessario definire un VPC GCP. Fare riferimento alla documentazione di OpenShift.

È possibile creare un gruppo di risorse del cluster di piattaforme e un gruppo di risorse del cluster OpenShift dell'applicazione di destinazione.

Assicurarsi di disporre di autorizzazioni IAM sufficienti

Assicurarsi di disporre di ruoli e autorizzazioni IAM sufficienti per installare un cluster RedHat OpenShift e un connettore NetApp Cloud Manager.

Vedere "Credenziali e permessi GCP iniziali".

Configurare GCP

Quindi, configurare GCP per creare un VPC, configurare istanze di calcolo, creare un Google Cloud Object Storage, creare un Google Container Register per ospitare le immagini di Astra Control Center e inviare le immagini a questo registro.

Seguire la documentazione GCP per completare i seguenti passaggi. Vedere Installazione del cluster OpenShift in GCP.

1. Creare un progetto GCP e un VPC nel GCP che si intende utilizzare per il cluster OCP con backend CVO.
2. Esaminare le istanze di calcolo. Questo può essere un server bare metal o VM in GCP.
3. Se il tipo di istanza non corrisponde già ai requisiti minimi di risorsa Astra per i nodi master e worker, modificare il tipo di istanza in GCP per soddisfare i requisiti Astra. Vedere "Requisiti di Astra Control Center".
4. Crea almeno un bucket di storage cloud GCP per memorizzare i tuoi backup.
5. Creare un segreto, necessario per l'accesso al bucket.
6. Creare un Google Container Registry per ospitare tutte le immagini di Astra Control Center.
7. Impostare l'accesso al Google Container Registry per il push/pull di Docker per tutte le immagini di Astra Control Center.

Esempio: Le immagini ACC possono essere inviate a questo registro inserendo il seguente script:

```
gcloud auth activate-service-account <service account email address>
--key-file=<GCP Service Account JSON file>
```

Questo script richiede un file manifesto di Astra Control Center e la posizione del Google Image Registry.

Esempio:

```
manifestfile=astra-control-center-<version>.manifest
GCP_CR_REGISTRY=<target image repository>
ASTRA_REGISTRY=<source ACC image repository>

while IFS= read -r image; do
    echo "image: $ASTRA_REGISTRY/$image $GCP_CR_REGISTRY/$image"
    root_image=${image%:*}
    echo $root_image
    docker pull $ASTRA_REGISTRY/$image
    docker tag $ASTRA_REGISTRY/$image $GCP_CR_REGISTRY/$image
    docker push $GCP_CR_REGISTRY/$image
done < astra-control-center-22.04.41.manifest
```

8. Impostare le zone DNS.

Configurare NetApp Cloud Manager

Utilizzando Cloud Manager, creare un'area di lavoro, aggiungere un connettore a GCP, creare un ambiente di lavoro e importare il cluster.

Seguire la documentazione di Cloud Manager per completare i seguenti passaggi. Vedere "Introduzione a Cloud Volumes ONTAP in GCP".

Di cosa hai bisogno

- Accesso all'account di servizio GCP con i ruoli e le autorizzazioni IAM richiesti

Fasi

1. Aggiungi le tue credenziali a Cloud Manager. Vedere "Aggiunta di account GCP".
2. Aggiungere un connettore per GCP.
   a. Scegliere "GCP" come provider.
   b. Immettere le credenziali GCP. Vedere "Creazione di un connettore in GCP da Cloud Manager".
   c. Assicurarsi che il connettore sia in funzione e passare a tale connettore.
3. Crea un ambiente di lavoro per il tuo ambiente cloud.
   a. Location: Italy
   b. Tipo: "Cloud Volumes ONTAP ha"
4. Importare il cluster OpenShift. Il cluster si connetterà all'ambiente di lavoro appena creato.

a. Per visualizzare i dettagli del cluster NetApp, selezionare **K8s** > **elenco cluster** > **Dettagli cluster**.
b. Nell'angolo in alto a destra, prendere nota della versione di Trident.
c. Si noti che le classi di storage del cluster Cloud Volumes ONTAP mostrano "NetApp" come provider.

   In questo modo, il cluster Red Hat OpenShift viene importato e viene assegnata una classe di storage predefinita. Selezionare la classe di storage. Trident viene installato automaticamente come parte del processo di importazione e rilevamento.

5. Tenere presenti tutti i volumi e i volumi persistenti in questa implementazione di Cloud Volumes ONTAP.

> Cloud Volumes ONTAP può operare come un singolo nodo o in alta disponibilità (ha). Se ha è attivato, annotare lo stato ha e lo stato di implementazione del nodo in esecuzione in GCP.

Installare Astra Control Center

Seguire lo standard "Istruzioni di installazione di Astra Control Center".

> GCP utilizza il tipo di bucket S3 generico.

1. Generare il Docker Secret per estrarre le immagini per l'installazione di Astra Control Center:

```
kubectl create secret docker-registry <secret name>
--docker-server=<Registry location>
--docker-username=_json_key
--docker-password="$(cat <GCP Service Account JSON file>)"
--namespace=pcloud
```

Implementare Astra Control Center in Microsoft Azure

Puoi implementare Astra Control Center su un cluster Kubernetes autogestito ospitato su un cloud pubblico Microsoft Azure.

Ciò di cui hai bisogno per Azure

Prima di implementare Astra Control Center in Azure, sono necessari i seguenti elementi:

- Licenza Astra Control Center. Vedere "Requisiti di licenza di Astra Control Center".
- "Soddisfare i requisiti di Astra Control Center".
- Account NetApp Cloud Central
- Se si utilizza OCP, Red Hat OpenShift Container Platform (OCP) 4.8
- Se si utilizza OCP, autorizzazioni Red Hat OpenShift Container Platform (OCP) (a livello di spazio dei nomi per creare i pod)
- Credenziali Azure con autorizzazioni che consentono di creare bucket e connettori

Requisiti dell'ambiente operativo per Azure

Assicurarsi che l'ambiente operativo scelto per ospitare Astra Control Center soddisfi i requisiti di base delle risorse descritti nella documentazione ufficiale dell'ambiente.

Astra Control Center richiede le seguenti risorse oltre ai requisiti delle risorse dell'ambiente:

Vedere "Requisiti dell'ambiente operativo di Astra Control Center".

Componente	Requisito
Capacità di storage NetApp Cloud Volumes ONTAP di back-end	Almeno 300 GB disponibili
Nodi di lavoro (requisito di calcolo di Azure)	Almeno 3 nodi di lavoro in totale, con 4 core vCPU e 12 GB di RAM ciascuno
Bilanciamento del carico	Tipo di servizio "LoadBalancer" disponibile per il traffico in ingresso da inviare ai servizi nel cluster dell'ambiente operativo
FQDN (Azure DNS zone)	Metodo per indirizzare l'FQDN di Astra Control Center all'indirizzo IP con bilanciamento del carico
Astra Trident (installato come parte del rilevamento dei cluster Kubernetes in NetApp Cloud Manager)	Astra Trident 21.04 o versione successiva installata e configurata e NetApp ONTAP versione 9.5 o successiva verrà utilizzato come backend di storage
Registro delle immagini	È necessario disporre di un registro privato esistente, ad esempio Azure Container Registry (ACR), in cui è possibile trasferire le immagini di build di Astra Control Center. È necessario fornire l'URL del registro delle immagini in cui verranno caricate le immagini. È necessario abilitare l'accesso anonimo per estrarre le immagini Restic per i backup.
Configurazione di Astra Trident/ONTAP	Astra Control Center richiede la creazione e l'impostazione di una classe di storage come classe di storage predefinita. Il centro di controllo Astra supporta le seguenti classi di storage Kubernetes create quando si importa il cluster Kubernetes in ONTAP. Questi sono forniti da Astra Trident: • `vsaworkingenvironment-<>-ha-nas csi.trident.netapp.io` • `vsaworkingenvironment-<>-ha-san csi.trident.netapp.io` • `vsaworkingenvironment-<>-single-nas csi.trident.netapp.io` • `vsaworkingenvironment-<>-single-san csi.trident.netapp.io`

Questi requisiti presuppongono che Astra Control Center sia l'unica applicazione in esecuzione nell'ambiente operativo. Se nell'ambiente sono in esecuzione applicazioni aggiuntive, modificare di conseguenza questi requisiti minimi.

Panoramica dell'implementazione di Azure

Ecco una panoramica del processo di installazione di Astra Control Center per Azure.

Ciascuna di queste fasi viene illustrata più dettagliatamente di seguito.

1. Installare un cluster RedHat OpenShift su Azure.
2. Creare gruppi di risorse Azure.
3. Assicurarsi di disporre di autorizzazioni IAM sufficienti.
4. Configurare Azure.
5. Configurare NetApp Cloud Manager.
6. Installare e configurare Astra Control Center.

Installare un cluster RedHat OpenShift su Azure

Il primo passo consiste nell'installare un cluster RedHat OpenShift su Azure.

Per istruzioni sull'installazione, consulta la documentazione di RedHat all'indirizzo "Installazione del cluster OpenShift su Azure" e. "Installazione di un account Azure".

Creare gruppi di risorse Azure

Creare almeno un gruppo di risorse Azure.

> OpenShift potrebbe creare i propri gruppi di risorse. Oltre a questi, è necessario definire anche i gruppi di risorse di Azure. Fare riferimento alla documentazione di OpenShift.

È possibile creare un gruppo di risorse del cluster di piattaforme e un gruppo di risorse del cluster OpenShift dell'applicazione di destinazione.

Assicurarsi di disporre di autorizzazioni IAM sufficienti

Assicurarsi di disporre di ruoli e autorizzazioni IAM sufficienti per installare un cluster RedHat OpenShift e un connettore NetApp Cloud Manager.

Vedere "Credenziali e permessi di Azure".

Configurare Azure

Quindi, configurare Azure per creare una rete virtuale, configurare istanze di calcolo, creare un container Azure Blob, creare un Azure Container Register (ACR) per ospitare le immagini di Astra Control Center e inviare le immagini a questo registro.

Seguire la documentazione di Azure per completare i seguenti passaggi. Vedere "Installazione del cluster OpenShift su Azure".

1. Creare una rete virtuale Azure.
2. Esaminare le istanze di calcolo. Si tratta di un server bare metal o di macchine virtuali in Azure.
3. Se il tipo di istanza non corrisponde già ai requisiti minimi di risorsa Astra per i nodi master e worker, modificare il tipo di istanza in Azure per soddisfare i requisiti Astra. Vedere "Requisiti di Astra Control Center".

4. Creare almeno un container Azure Blob per memorizzare i backup.
5. Creare un account storage. Per creare un container da utilizzare come bucket in Astra Control Center è necessario un account storage.
6. Creare un segreto, necessario per l'accesso al bucket.
7. Creare un Azure Container Registry (ACR) per ospitare tutte le immagini di Astra Control Center.
8. Impostare l'accesso ACR per il push/pull di tutte le immagini di Astra Control Center di Docker.
9. Inviare le immagini ACC a questo registro inserendo il seguente script:

```
az acr login -n <AZ ACR URL/Location>
This script requires ACC manifest file and your Azure ACR location.
```

Esempio:

```
manifestfile=astra-control-center-<version>.manifest
AZ_ACR_REGISTRY=<target image repository>
ASTRA_REGISTRY=<source ACC image repository>

while IFS= read -r image; do
    echo "image: $ASTRA_REGISTRY/$image $AZ_ACR_REGISTRY/$image"
    root_image=${image%:*}
    echo $root_image
    docker pull $ASTRA_REGISTRY/$image
    docker tag $ASTRA_REGISTRY/$image $AZ_ACR_REGISTRYY/$image
    docker push $AZ_ACR_REGISTRY/$image
done < astra-control-center-22.04.41.manifest
```

10. Impostare le zone DNS.

Configurare NetApp Cloud Manager

Utilizzando Cloud Manager, creare un'area di lavoro, aggiungere un connettore ad Azure, creare un ambiente di lavoro e importare il cluster.

Seguire la documentazione di Cloud Manager per completare i seguenti passaggi. Vedere "Introduzione a Cloud Manager in Azure".

Di cosa hai bisogno

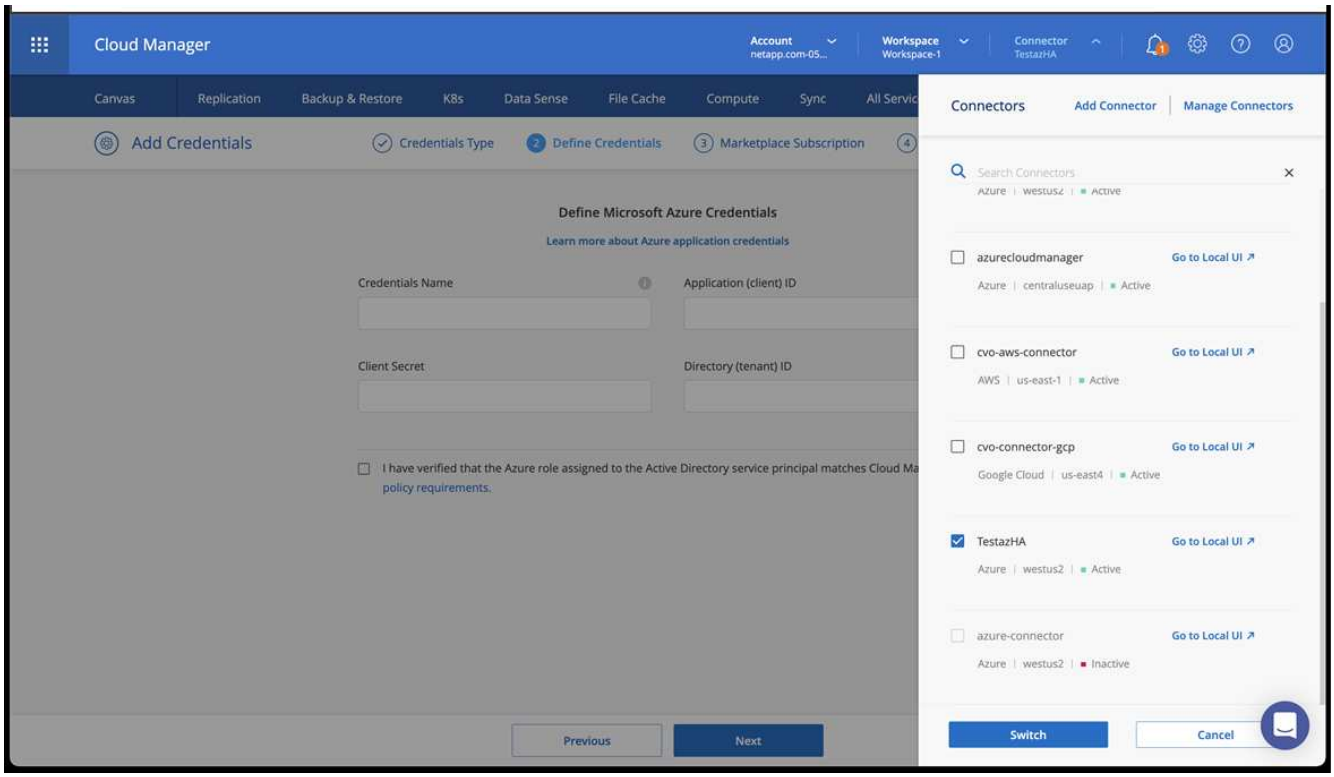
Accesso all'account Azure con le autorizzazioni e i ruoli IAM richiesti

Fasi

1. Aggiungi le tue credenziali a Cloud Manager.
2. Aggiungere un connettore per Azure. Vedere "Policy di Cloud Manager".
   a. Scegliere **Azure** come provider.
   b. Immettere le credenziali Azure, inclusi ID applicazione, segreto client e ID directory (tenant).

Vedere "Creazione di un connettore in Azure da Cloud Manager".

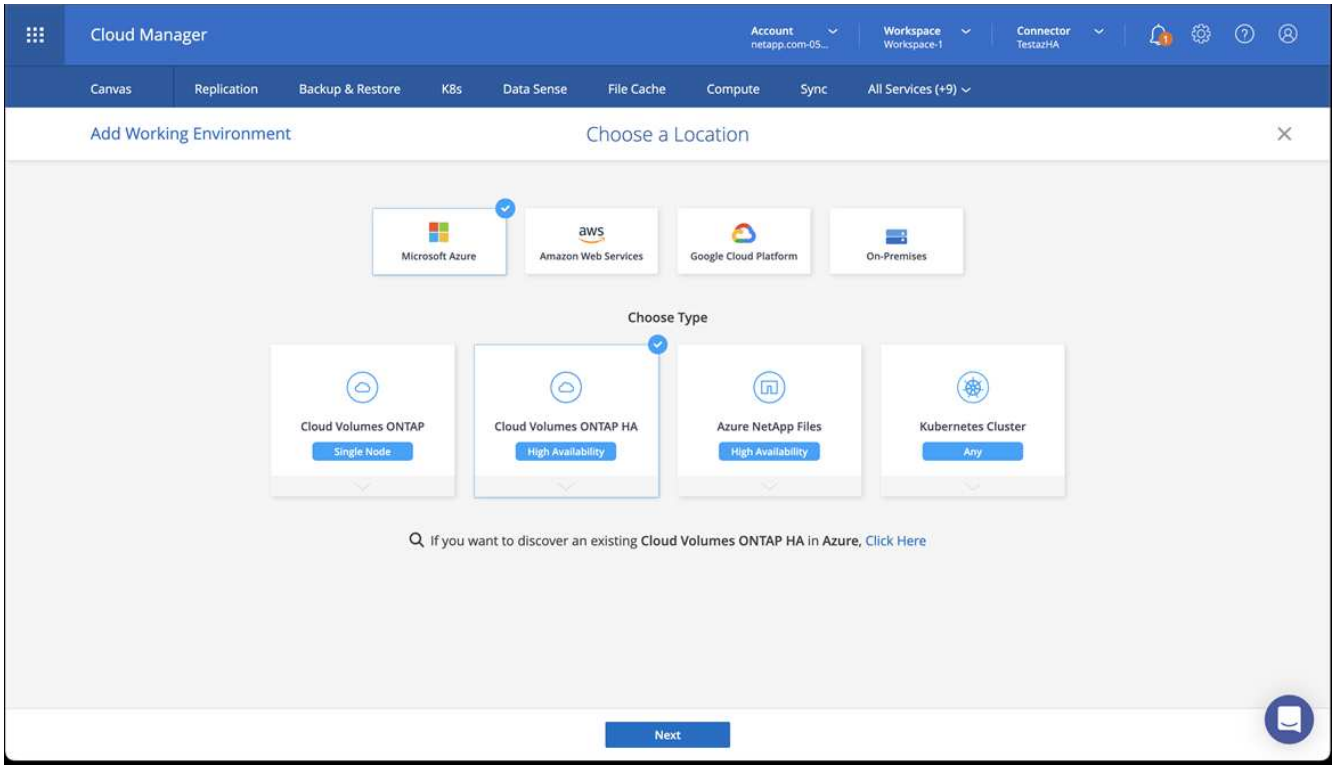
3. Assicurarsi che il connettore sia in funzione e passare a tale connettore.



4. Crea un ambiente di lavoro per il tuo ambiente cloud.

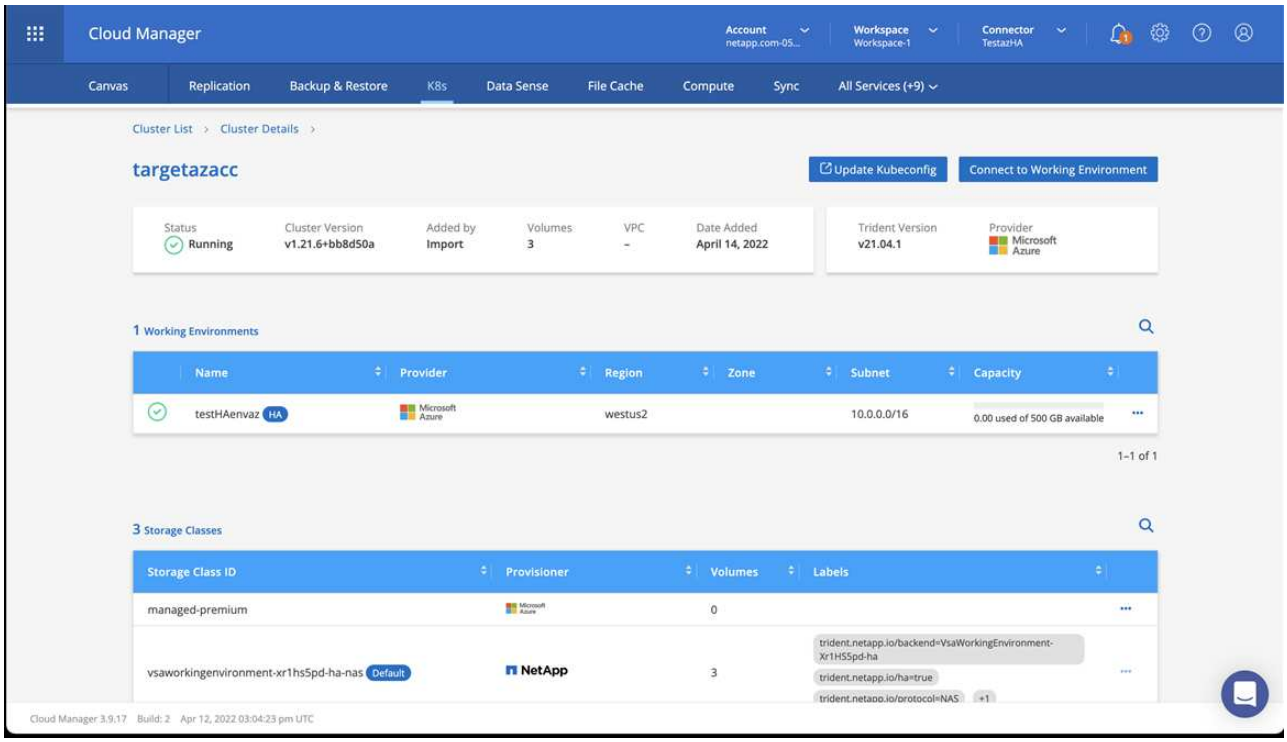
   a. Percorso: "Microsoft Azure".

   b. Tipo: "Cloud Volumes ONTAP ha".



5. Importare il cluster OpenShift. Il cluster si connetterà all'ambiente di lavoro appena creato.

   a. Per visualizzare i dettagli del cluster NetApp, selezionare **K8s** > **elenco cluster** > **Dettagli cluster**.



   b. Nell'angolo in alto a destra, prendere nota della versione di Trident.

   c. Si noti che le classi di storage cluster Cloud Volumes ONTAP mostrano NetApp come provider.

   In questo modo viene importato il cluster Red Hat OpenShift e viene assegnata una classe di storage predefinita. Selezionare la classe di storage. Trident viene installato automaticamente come parte del processo di importazione e rilevamento.

6. Tenere presenti tutti i volumi e i volumi persistenti in questa implementazione di Cloud Volumes ONTAP.
7. Cloud Volumes ONTAP può funzionare come nodo singolo o in alta disponibilità. Se ha è attivato, annotare lo stato ha e lo stato di implementazione del nodo in esecuzione in Azure.

Installare e configurare Astra Control Center

Installare Astra Control Center con lo standard "istruzioni per l'installazione".

Utilizzando Astra Control Center, aggiungere un bucket Azure. Vedere "Configurare Astra Control Center e aggiungere i bucket".

Configurare Astra Control Center

Il centro di controllo Astra supporta e monitora l'archivio dati ONTAP e Astra come back-end dello storage. Dopo aver installato Astra Control Center, aver effettuato l'accesso all'interfaccia utente e aver modificato la password, sarà necessario impostare una licenza, aggiungere cluster, gestire lo storage e aggiungere bucket.

Attività

- Aggiungere una licenza per Astra Control Center

- Aggiungere il cluster
- Aggiungere un backend di storage
- Aggiungi un bucket

Aggiungere una licenza per Astra Control Center

È possibile aggiungere una nuova licenza utilizzando l'interfaccia utente o. "API" Per ottenere la funzionalità completa di Astra Control Center. Senza una licenza, l'utilizzo di Astra Control Center è limitato alla gestione degli utenti e all'aggiunta di nuovi cluster.

Per ulteriori informazioni sul calcolo delle licenze, vedere "Licensing".

> Per aggiornare una licenza di valutazione o una licenza completa, vedere "Aggiornare una licenza esistente".

Le licenze di Astra Control Center misurano le risorse della CPU utilizzando le unità CPU di Kubernetes. La licenza deve tenere conto delle risorse CPU assegnate ai nodi di lavoro di tutti i cluster Kubernetes gestiti. Prima di aggiungere una licenza, è necessario ottenere il file di licenza (NLF) da "Sito di supporto NetApp".

Puoi anche provare Astra Control Center con una licenza di valutazione, che ti consente di utilizzare Astra Control Center per 90 giorni dalla data di download della licenza. Puoi iscriverti per una prova gratuita registrandoti "qui".

> Se l'installazione supera il numero concesso in licenza di unità CPU, Astra Control Center impedisce la gestione di nuove applicazioni. Quando viene superata la capacità, viene visualizzato un avviso.

Di cosa hai bisogno

Quando si scarica Astra Control Center da "Sito di supporto NetApp", Inoltre, è stato scaricato il file di licenza NetApp (NLF). Assicurarsi di avere accesso a questo file di licenza.

Fasi

1. Accedere all'interfaccia utente di Astra Control Center.
2. Selezionare **account** > **licenza**.
3. Selezionare **Aggiungi licenza**.
4. Individuare il file di licenza (NLF) scaricato.
5. Selezionare **Aggiungi licenza**.

La pagina **account** > **licenza** visualizza le informazioni sulla licenza, la data di scadenza, il numero di serie della licenza, l'ID account e le unità CPU utilizzate.

> Se si dispone di una licenza di valutazione, assicurarsi di memorizzare l'ID account per evitare la perdita di dati in caso di guasto di Astra Control Center se non si inviano ASUP.

Aggiungere il cluster

Per iniziare a gestire le tue applicazioni, Aggiungi un cluster Kubernetes e gestilo come risorsa di calcolo. Devi aggiungere un cluster per Astra Control Center per scoprire le tue applicazioni Kubernetes. Per Astra Data Store, si desidera aggiungere il cluster di applicazioni Kubernetes che contiene applicazioni che utilizzano

volumi forniti da Astra Data Store.

> Si consiglia ad Astra Control Center di gestire il cluster su cui viene implementato prima di aggiungere altri cluster ad Astra Control Center da gestire. La gestione del cluster iniziale è necessaria per inviare i dati Kublemetrics e i dati associati al cluster per metriche e troubleshooting. È possibile utilizzare la funzione **Add Cluster** per gestire un cluster con Astra Control Center.

> Quando Astra Control gestisce un cluster, tiene traccia della classe di storage predefinita del cluster. Se si modifica la classe di storage utilizzando `kubectl` Comandi, Astra Control ripristina la modifica. Per modificare la classe di storage predefinita in un cluster gestito da Astra Control, utilizzare uno dei seguenti metodi:
>
> - Utilizzare l'API di controllo Astra `PUT /managedClusters` e assegnare una classe di storage predefinita diversa con `DefaultStorageClass` parametro.
> - Utilizzare l'interfaccia utente Web di Astra Control per assegnare una classe di storage predefinita diversa. Vedere Modificare la classe di storage predefinita.

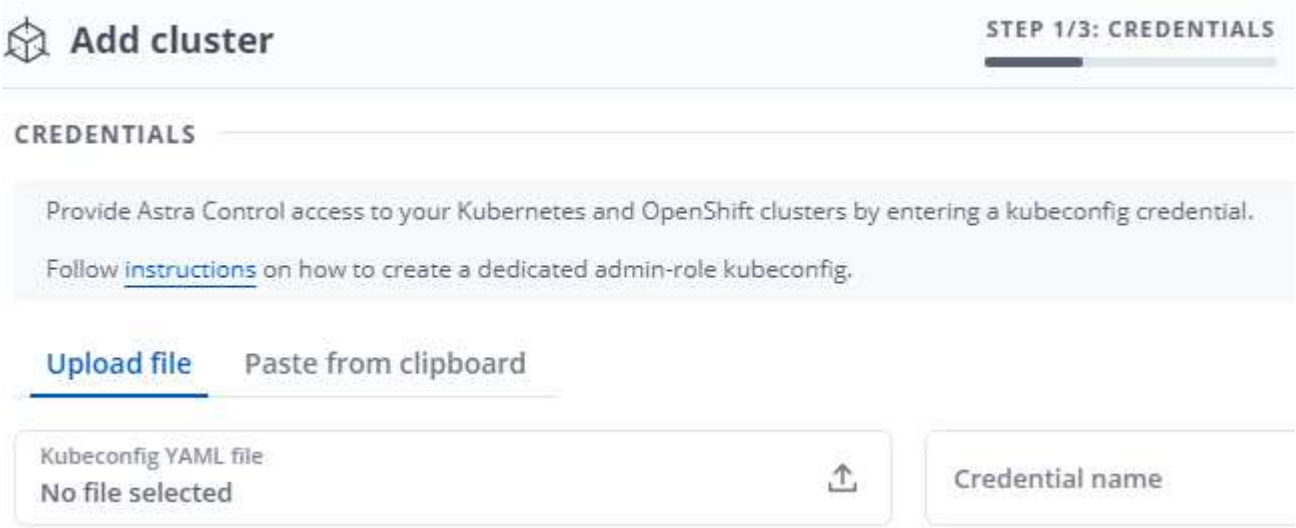
Di cosa hai bisogno

- Prima di aggiungere un cluster, esaminare ed eseguire le operazioni necessarie "attività prerequisite".

Fasi

1. Dal pannello **Dashboard** dell'interfaccia utente di Astra Control Center, selezionare **Add** (Aggiungi) nella sezione Clusters (Clusters).
2. Nella finestra **Add Cluster** che si apre, caricare un `kubeconfig.yaml` archiviare o incollare il contenuto di a. `kubeconfig.yaml` file.

   > Il `kubeconfig.yaml` il file deve includere **solo le credenziali del cluster per un cluster**.



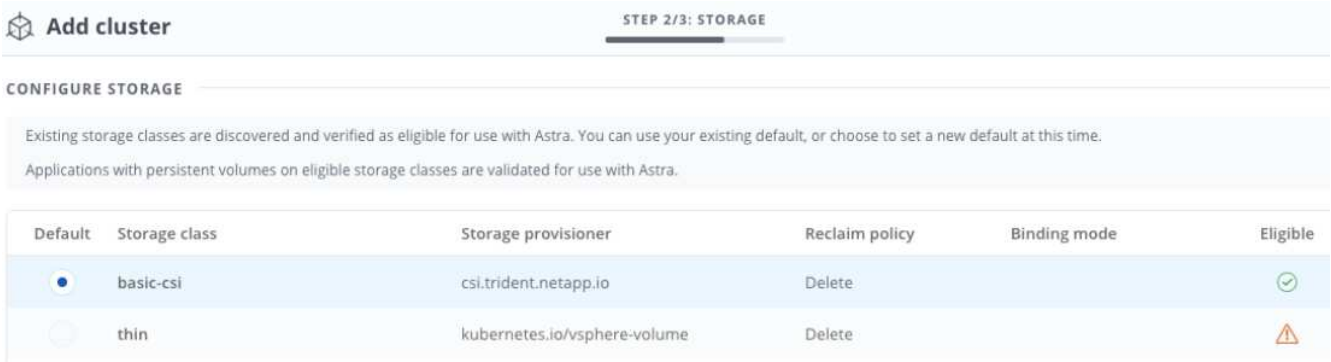
   > Se crei il tuo `kubeconfig` file, è necessario definire solo **un** elemento di contesto al suo interno. Vedere "Documentazione Kubernetes" per informazioni sulla creazione `kubeconfig` file.

3. Fornire un nome di credenziale. Per impostazione predefinita, il nome della credenziale viene compilato

automaticamente come nome del cluster.

4. Selezionare **Configura storage**.
5. Selezionare la classe di storage da utilizzare per questo cluster Kubernetes e selezionare **Review**.

> È necessario selezionare una classe di storage Trident supportata dallo storage ONTAP o dall'archivio dati Astra.



6. Esaminare le informazioni e, se l'aspetto è soddisfacente, selezionare **Aggiungi cluster**.

Risultato

Il cluster passa allo stato **rilevamento**, quindi passa a **in esecuzione**. Hai aggiunto un cluster Kubernetes e lo stai gestendo in Astra Control Center.

> Dopo aver aggiunto un cluster da gestire in Astra Control Center, l'implementazione dell'operatore di monitoraggio potrebbe richiedere alcuni minuti. Fino a quel momento, l'icona di notifica diventa rossa e registra un evento **Monitoring Agent Status Check Failed** (controllo stato agente non riuscito). È possibile ignorarlo, perché il problema si risolve quando Astra Control Center ottiene lo stato corretto. Se il problema non si risolve in pochi minuti, accedere al cluster ed eseguire `oc get pods -n netapp-monitoring` come punto di partenza. Per eseguire il debug del problema, consultare i log dell'operatore di monitoraggio.

Aggiungere un backend di storage

È possibile aggiungere un backend di storage in modo che Astra Control possa gestire le proprie risorse. È possibile implementare un backend di storage su un cluster gestito o utilizzare un backend di storage esistente.

La gestione dei cluster di storage in Astra Control come back-end dello storage consente di ottenere collegamenti tra volumi persistenti (PVS) e il back-end dello storage, oltre a metriche di storage aggiuntive.

Ciò di cui hai bisogno per le implementazioni di Astra Data Store esistenti

- Hai aggiunto il cluster di applicazioni Kubernetes e il cluster di calcolo sottostante.

> Dopo aver aggiunto il cluster di applicazioni Kubernetes per Astra Data Store ed essere gestito da Astra Control, il cluster viene visualizzato come `unmanaged` nell'elenco dei backend rilevati. È quindi necessario aggiungere il cluster di calcolo che contiene Astra Data Store e che si trova sotto il cluster di applicazioni Kubernetes. È possibile eseguire questa operazione da **Backend** nell'interfaccia utente. Selezionare il menu Actions (azioni) per il cluster, quindi scegliere `Manage`, e. "aggiungere il cluster". Dopo lo stato del cluster di `unmanaged` Modifiche al nome del cluster Kubernetes, è possibile procedere con l'aggiunta di un backend.

Ciò di cui hai bisogno per le nuove implementazioni di Astra Data Store

- Lo hai fatto "ha caricato la versione del bundle di installazione che si intende implementare" In una posizione accessibile da Astra Control.
- È stato aggiunto il cluster Kubernetes che si intende utilizzare per la distribuzione.
- Hai caricato Licenza Astra Data Store Per l'implementazione in una posizione accessibile ad Astra Control.

Opzioni

- Implementare le risorse di storage
- Utilizzare un backend di storage esistente

Implementare le risorse di storage

È possibile implementare un nuovo archivio dati Astra e gestire il backend dello storage associato.

Fasi

1. Spostarsi dal menu Dashboard o Backend:
   - Da **Dashboard**: Dal riepilogo delle risorse, selezionare un collegamento dal riquadro Storage Backends e selezionare **Add** dalla sezione Backend.
   - Da **backend**:
     i. Nell'area di navigazione a sinistra, selezionare **Backend**.
     ii. Selezionare **Aggiungi**.
2. Selezionare l'opzione di implementazione **Astra Data Store** nella scheda **Deploy**.
3. Selezionare il pacchetto Astra Data Store da implementare:
   a. Immettere un nome per l'applicazione Astra Data Store.
   b. Scegli la versione di Astra Data Store che desideri implementare.

   > Se non è stata ancora caricata la versione che si intende distribuire, è possibile utilizzare l'opzione **Add package** (Aggiungi pacchetto) o uscire dalla procedura guidata e utilizzarla "gestione dei pacchetti" per caricare il bundle di installazione.

4. Selezionare una licenza Astra Data Store precedentemente caricata oppure utilizzare l'opzione **Add License** (Aggiungi licenza) per caricare una licenza da utilizzare con l'applicazione.

> Le licenze di Astra Data Store con autorizzazioni complete sono associate al cluster Kubernetes e i cluster associati dovrebbero essere visualizzati automaticamente. Se non è presente alcun cluster gestito, è possibile selezionare l'opzione **Aggiungi un cluster** per aggiungerne uno alla gestione di Astra Control. Per le licenze Astra Data Store, se non è stata effettuata alcuna associazione tra la licenza e il cluster, è possibile definire questa associazione nella pagina successiva della procedura guidata.

5. Se non hai aggiunto un cluster Kubernetes alla gestione di Astra Control, devi farlo dalla pagina **Kubernetes cluster**. Selezionare un cluster esistente dall'elenco o selezionare **add the underlying cluster** (Aggiungi cluster sottostante) per aggiungere un cluster alla gestione di Astra Control.
6. Selezionare una dimensione del modello per il cluster Kubernetes che fornirà le risorse per Astra Data Store. È possibile scegliere una delle seguenti opzioni:
   - Se si sceglie `Recommended Kubernetes worker node requirements`, selezionare un modello da grande a piccolo in base a quanto consentito dalla licenza.
   - Se si sceglie `Custom Kubernetes worker node requirements`, selezionare il numero di core e la memoria totale desiderati per ciascun nodo del cluster. È inoltre possibile visualizzare il numero idoneo di nodi nel cluster che soddisfano i criteri di selezione per core e memoria.

> Quando si sceglie un modello, selezionare nodi più grandi con più memoria e core per carichi di lavoro più grandi o un numero maggiore di nodi per carichi di lavoro più piccoli. Selezionare un modello in base a quanto consentito dalla licenza. Ogni opzione di modello consigliata suggerisce il numero di nodi idonei che soddisfano il modello di modello per memoria, core e capacità per ciascun nodo.

7. Configurare i nodi:
   a. Aggiungere un'etichetta di nodo per identificare il pool di nodi di lavoro che supporta questo cluster Astra Data Store.

> L'etichetta deve essere aggiunta a ogni singolo nodo del cluster che verrà utilizzato per l'implementazione di Astra Data Store prima dell'inizio dell'implementazione, altrimenti l'implementazione non avrà esito positivo.

   b. Configurare manualmente la capacità (GiB) per nodo o selezionare la capacità massima consentita per nodo.
   c. Configurare un numero massimo di nodi consentiti nel cluster o consentire il numero massimo di nodi nel cluster.
8. (Solo per le licenze complete di Astra Data Store) inserire la chiave dell'etichetta che si desidera utilizzare per i domini di protezione.

> Creare almeno tre etichette univoche per la chiave per ciascun nodo. Ad esempio, se la chiave è `astra.datastore.protection.domain`, è possibile creare le seguenti etichette: `astra.datastore.protection.domain=domain1` ,`astra.datastore.protection.domain=domain2`, e. `astra.datastore.protection.domain=domain3`.

9. Configurare la rete di gestione:
   a. Inserire un indirizzo IP di gestione per la gestione interna di Astra Data Store che si trova sulla stessa sottorete degli indirizzi IP del nodo di lavoro.

b. Scegliere di utilizzare la stessa scheda NIC per reti di gestione e dati o configurarle separatamente.

c. Inserire il pool di indirizzi IP della rete dati, la subnet mask e il gateway per l'accesso allo storage.

10. Esaminare la configurazione e selezionare **Deploy** per iniziare l'installazione.

Risultato

Una volta completata l'installazione, il backend viene visualizzato in `available` indicare nell'elenco backend insieme alle informazioni sulle performance attive.

> Potrebbe essere necessario aggiornare la pagina per visualizzare il backend.

Utilizzare un backend di storage esistente

Puoi portare un backend di storage ONTAP o Astra Data Store scoperto nella gestione del centro di controllo Astra.

Fasi

1. Spostarsi dal menu Dashboard o Backend:
   - Da **Dashboard**: Dal riepilogo delle risorse, selezionare un collegamento dal riquadro Storage Backends e selezionare **Add** dalla sezione Backend.
   - Da **backend**:
     i. Nell'area di navigazione a sinistra, selezionare **Backend**.
     ii. Selezionare **Gestisci** su un backend rilevato dal cluster gestito oppure selezionare **Aggiungi** per gestire un backend esistente aggiuntivo.
2. Selezionare la scheda **Usa esistente**.
3. Eseguire una delle seguenti operazioni in base al tipo di backend:
   - **Archivio dati Astra**:
     i. Selezionare **Astra Data Store**.
     ii. Selezionare il cluster di calcolo gestito e selezionare **Avanti**.
     iii. Confermare i dettagli del back-end e selezionare **Add storage backend**.
   - **ONTAP**:
     i. Selezionare **ONTAP** e selezionare **Avanti**.
     ii. Inserire l'indirizzo IP di gestione del cluster ONTAP e le credenziali di amministratore.

     > L'utente di cui si inseriscono le credenziali deve disporre di `ontapi` Metodo di accesso all'accesso dell'utente abilitato in Gestione di sistema di ONTAP sul cluster ONTAP. Se si intende utilizzare la replica SnapMirror, attivare i metodi di accesso `ontapi` e. `http` Per l'utente su entrambi i cluster ONTAP. Vedere "Gestire gli account utente" per ulteriori informazioni.

     iii. Selezionare **Revisione**.
     iv. Confermare i dettagli del back-end e selezionare **Add storage backend**.

Risultato

Il backend viene visualizzato in `available` indicare nell'elenco le informazioni di riepilogo.

Potrebbe essere necessario aggiornare la pagina per visualizzare il backend.

Aggiungi un bucket

L'aggiunta di provider di bucket di archivi di oggetti è essenziale se si desidera eseguire il backup delle applicazioni e dello storage persistente o se si desidera clonare le applicazioni tra cluster. Astra Control memorizza i backup o i cloni nei bucket dell'archivio di oggetti definiti dall'utente.

Quando si aggiunge un bucket, Astra Control contrassegna un bucket come indicatore di bucket predefinito. Il primo bucket creato diventa quello predefinito.

Non è necessario un bucket se si clonano la configurazione dell'applicazione e lo storage persistente sullo stesso cluster.

Utilizzare uno dei seguenti tipi di bucket:

- NetApp ONTAP S3
- NetApp StorageGRID S3
- Generico S3

  Amazon Web Services (AWS) e Google Cloud Platform (GCP) utilizzano il tipo di bucket S3 generico.

- Microsoft Azure

  Sebbene Astra Control Center supporti Amazon S3 come provider di bucket S3 generico, Astra Control Center potrebbe non supportare tutti i vendor di archivi di oggetti che sostengono il supporto S3 di Amazon.

- Microsoft Azure

Per istruzioni su come aggiungere bucket utilizzando l'API Astra Control, vedere "Astra Automation e informazioni API".

Fasi

1. Nell'area di navigazione a sinistra, selezionare **Bucket**.
   a. Selezionare **Aggiungi**.
   b. Selezionare il tipo di bucket.

      Quando si aggiunge un bucket, selezionare il bucket provider corretto e fornire le credenziali corrette per tale provider. Ad esempio, l'interfaccia utente accetta come tipo NetApp ONTAP S3 e accetta le credenziali StorageGRID; tuttavia, questo causerà l'errore di tutti i backup e ripristini futuri dell'applicazione che utilizzano questo bucket.

   c. Creare un nuovo nome di bucket o inserire un nome di bucket esistente e una descrizione opzionale.

      Il nome e la descrizione del bucket vengono visualizzati come percorso di backup che è possibile scegliere in seguito quando si crea un backup. Il nome viene visualizzato anche durante la configurazione del criterio di protezione.

d. Inserire il nome o l'indirizzo IP dell'endpoint S3.

e. Se si desidera che questo bucket sia il bucket predefinito per tutti i backup, selezionare `Make this bucket the default bucket for this private cloud` opzione.

> Questa opzione non viene visualizzata per il primo bucket creato.

f. Continuare aggiungendo informazioni sulle credenziali.

Aggiungere le credenziali di accesso S3

Aggiungi credenziali di accesso S3 in qualsiasi momento.

Fasi

1. Dalla finestra di dialogo bucket, selezionare la scheda **Add** (Aggiungi) o **Use existing** (Usa esistente).
   a. Immettere un nome per la credenziale che la distingue dalle altre credenziali in Astra Control.
   b. Inserire l'ID di accesso e la chiave segreta incollando il contenuto dagli Appunti.

Modificare la classe di storage predefinita

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

Fasi

1. Nell'interfaccia utente Web di Astra Control Center, 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**.

Quali sono le prossime novità?

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

- "Gestire gli utenti"
- "Inizia a gestire le app"
- "Proteggi le app"
- "Clonare le applicazioni"
- "Gestire le notifiche"
- "Connettersi a Cloud Insights"
- "Aggiungere un certificato TLS personalizzato"

Trova ulteriori informazioni

- "Utilizzare l'API di controllo Astra"

- "Problemi noti"

Prerequisiti per l'aggiunta di un cluster

Prima di aggiungere un cluster, assicurarsi che le condizioni preliminari siano soddisfatte. È inoltre necessario eseguire i controlli di idoneità per assicurarsi che il cluster sia pronto per essere aggiunto ad Astra Control Center.

Cosa serve prima di aggiungere un cluster

Assicurarsi che il cluster soddisfi i requisiti descritti nella "Requisiti del cluster di applicazioni".

> Se si intende aggiungere un secondo cluster OpenShift 4.6, 4.7 o 4.8 come risorsa di calcolo gestita, assicurarsi che la funzione Astra Trident Volume Snapshot sia attivata. Vedi l'Astra Trident ufficiale "istruzioni" Per attivare e testare le istantanee dei volumi con Astra Trident.

- Astra Trident StorageClasses configurato con un "back-end di storage supportato" (richiesto per qualsiasi tipo di cluster)
- Il superuser e l'ID utente impostati sul sistema ONTAP di backup per eseguire il backup e il ripristino delle applicazioni con Centro di controllo Astra. Eseguire il seguente comando nella riga di comando di ONTAP: `export-policy rule modify -vserver <storage virtual machine name> -policyname <policy name> -ruleindex 1 -superuser sysm --anon 65534`
- Un tridente Astra `volumesnapshotclass` oggetto definito da un amministratore. Vedi Astra Trident "istruzioni" Per attivare e testare le istantanee dei volumi con Astra Trident.
- Assicurarsi di avere definito solo una singola classe di storage predefinita per il cluster Kubernetes.

Eseguire i controlli di idoneità

Eseguire i seguenti controlli di idoneità per assicurarsi che il cluster sia pronto per essere aggiunto ad Astra Control Center.

Fasi

1. Controllare la versione di Trident.

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

   Se Trident esiste, l'output è simile a quanto segue:

   ```
   NAME      VERSION
   trident   21.04.0
   ```

   Se Trident non esiste, viene visualizzato un output simile a quanto segue:

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

Se Trident non è installato o se la versione installata non è la più recente, è necessario installare la versione più recente di Trident prima di procedere. Vedere "Documentazione di Trident" per istruzioni.

2. Controllare se le classi di storage utilizzano i driver Trident supportati. Il nome del provider deve essere `csi.trident.netapp.io`. Vedere il seguente esempio:

```
kubectl get sc
NAME                    PROVISIONER                    RECLAIMPOLICY
VOLUMEBINDINGMODE   ALLOWVOLUMEEXPANSION   AGE
ontap-gold (default)    csi.trident.netapp.io          Delete
Immediate           true                   5d23h
thin                    kubernetes.io/vsphere-volume   Delete
Immediate           false                  6d
```

Creare un kubeconfig con ruolo di amministratore

Prima di eseguire la procedura, assicurarsi di disporre dei seguenti elementi sul computer:

- `kubectl` v1.19 o versione successiva installata
- Un kubeconfig attivo con diritti di amministratore del cluster per il contesto attivo

Fasi

1. Creare un account di servizio come segue:

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

      Regolare il nome e lo spazio dei nomi in base alle esigenze. Se le modifiche vengono apportate qui, è necessario applicare le stesse modifiche nei passaggi seguenti.

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

+

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

   a. Applicare l'account del servizio:

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

2. (Facoltativo) se il cluster utilizza una policy di sicurezza del pod restrittiva che non consente la creazione di pod privilegiati o consente l'esecuzione di processi all'interno dei container del pod come utente root, creare una policy di sicurezza del pod personalizzata per il cluster che consenta ad Astra Control di creare e gestire i pod. Per istruzioni, vedere "Creare una policy di sicurezza pod personalizzata".
3. Concedere le autorizzazioni di amministratore del cluster come segue:

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

      Modificare i nomi e gli spazi dei nomi modificati quando si crea l'account del servizio, in base alle necessità.

```
<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: cluster-admin
subjects:
- kind: ServiceAccount
  name: astracontrol-service-account
  namespace: default
```

   a. Applicare l'associazione del ruolo del cluster:

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

4. 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-vhz87"},
{ "name": "astracontrol-service-account-token-r59kr"}
]
```

Gli indici di ciascun elemento in `secrets` l'array inizia con 0. Nell'esempio precedente, l'indice per `astracontrol-service-account-dockercfg-vhz87` sarebbe 0 e l'indice per `astracontrol-service-account-token-r59kr` sarebbe 1. Nell'output, annotare l'indice del nome dell'account del servizio che contiene la parola "token".

5. 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
```

6. (**opzionale**) rinominare il kubeconfig con un nome significativo per il cluster. Proteggi la tua credenziale del cluster.

```
chmod 700 create-kubeconfig.sh
mv kubeconfig-sa.txt YOUR_CLUSTER_NAME_kubeconfig
```

Quali sono le prossime novità?

Ora che hai verificato che i prerequisiti sono stati soddisfatti, sei pronto "aggiungere un cluster".

Trova ulteriori informazioni

- "Documentazione di Trident"
- "Utilizzare l'API di controllo Astra"

Aggiungere un certificato TLS personalizzato

È possibile rimuovere il certificato TLS autofirmato esistente e sostituirlo con un certificato TLS firmato da un'autorità di certificazione (CA).

Di cosa hai bisogno

- Kubernetes cluster con Astra Control Center installato
- Accesso amministrativo a una shell dei comandi sul cluster da eseguire `kubectl` comandi
- Chiave privata e file di certificato dalla CA

Rimuovere il certificato autofirmato

Rimuovere il certificato TLS autofirmato esistente.

1. Utilizzando SSH, accedere al cluster Kubernetes che ospita Astra Control Center come utente amministrativo.
2. Individuare il segreto TLS associato al certificato corrente utilizzando il seguente comando, sostituendo `<ACC-deployment-namespace>` Con lo spazio dei nomi di implementazione di Astra Control Center:

```
kubectl get certificate -n <ACC-deployment-namespace>
```

3. Eliminare il certificato e il segreto attualmente installati utilizzando i seguenti comandi:

```
kubectl delete cert cert-manager-certificates -n <ACC-deployment-
namespace>
kubectl delete secret secure-testing-cert -n <ACC-deployment-namespace>
```

Aggiungere un nuovo certificato

Aggiungere un nuovo certificato TLS firmato da una CA.

1. Utilizzare il seguente comando per creare il nuovo segreto TLS con la chiave privata e i file di certificato della CA, sostituendo gli argomenti tra parentesi <> con le informazioni appropriate:

```
kubectl create secret tls <secret-name> --key <private-key-filename>
--cert <certificate-filename> -n <ACC-deployment-namespace>
```

2. Utilizzare il seguente comando e l'esempio per modificare il file CRD (Custom Resource Definition) del cluster e modificare `spec.selfSigned` valore a. `spec.ca.secretName` Per fare riferimento al segreto TLS creato in precedenza:

```
kubectl edit clusterissuers.cert-manager.io/cert-manager-certificates -n
<ACC-deployment-namespace>
....

#spec:
#  selfSigned: {}

spec:
  ca:
    secretName: <secret-name>
```

3. Utilizzare il seguente comando e l'output di esempio per confermare che le modifiche sono corrette e che il cluster è pronto per validare i certificati, sostituendo `<ACC-deployment-namespace>` Con lo spazio dei nomi di implementazione di Astra Control Center:

```
kubectl describe clusterissuers.cert-manager.io/cert-manager-
certificates -n <ACC-deployment-namespace>
....

Status:
  Conditions:
    Last Transition Time:  2021-07-01T23:50:27Z
    Message:               Signing CA verified
    Reason:                KeyPairVerified
    Status:                True
    Type:                  Ready
Events:                    <none>
```

4. Creare il `certificate.yaml` file utilizzando il seguente esempio, sostituendo i valori segnaposto tra parentesi <> con le informazioni appropriate:

```yaml
apiVersion: cert-manager.io/v1
kind: Certificate
metadata:
  name: <certificate-name>
  namespace: <ACC-deployment-namespace>
spec:
  secretName: <certificate-secret-name>
  duration: 2160h # 90d
  renewBefore: 360h # 15d
  dnsNames:
  - <astra.dnsname.example.com> #Replace with the correct Astra Control
Center DNS address
  issuerRef:
    kind: ClusterIssuer
    name: cert-manager-certificates
```

5. Creare il certificato utilizzando il seguente comando:

```
kubectl apply -f certificate.yaml
```

6. Utilizzando il seguente comando e l'output di esempio, verificare che il certificato sia stato creato correttamente e con gli argomenti specificati durante la creazione (ad esempio nome, durata, scadenza di rinnovo e nomi DNS).

```
kubectl describe certificate -n <ACC-deployment-namespace>
....

Spec:
  Dns Names:
    astra.example.com
  Duration:  125h0m0s
  Issuer Ref:
    Kind:        ClusterIssuer
    Name:        cert-manager-certificates
  Renew Before:  61h0m0s
  Secret Name:   <certificate-secret-name>
Status:
  Conditions:
    Last Transition Time:  2021-07-02T00:45:41Z
    Message:               Certificate is up to date and has not expired
    Reason:                Ready
    Status:                True
    Type:                  Ready
  Not After:               2021-07-07T05:45:41Z
  Not Before:              2021-07-02T00:45:41Z
  Renewal Time:            2021-07-04T16:45:41Z
  Revision:                1
Events:                    <none>
```

7. Modificare l’opzione TLS CRD di ingresso per indicare il nuovo segreto del certificato utilizzando il seguente comando ed esempio, sostituendo i valori segnaposto tra parentesi <> con le informazioni appropriate:

```
kubectl edit ingressroutes.traefik.containo.us -n <ACC-deployment-
namespace>
....

# tls:
#    options:
#      name: default
#    secretName: secure-testing-cert
#    store:
#      name: default

 tls:
    options:
      name: default
    secretName: <certificate-secret-name>
    store:
      name: default
```

8. Utilizzando un browser Web, accedere all'indirizzo IP di implementazione di Astra Control Center.
9. Verificare che i dettagli del certificato corrispondano ai dettagli del certificato installato.
10. Esportare il certificato e importare il risultato nel gestore dei certificati nel browser Web.

Creare una policy di sicurezza pod personalizzata

Astra Control deve creare e gestire i pod Kubernetes sui cluster gestiti. Se il cluster utilizza una policy di sicurezza del pod restrittiva che non consente la creazione di pod con privilegi o l'esecuzione di processi all'interno dei container del pod come utente root, è necessario creare una policy di sicurezza del pod meno restrittiva per consentire ad Astra Control di creare e gestire questi pod.

Fasi

1. Creare un criterio di protezione pod per il cluster meno restrittivo di quello predefinito e salvarlo in un file. Ad esempio:

```yaml
apiVersion: policy/v1beta1
kind: PodSecurityPolicy
metadata:
  name: astracontrol
  annotations:
    seccomp.security.alpha.kubernetes.io/allowedProfileNames: '*'
spec:
  privileged: true
  allowPrivilegeEscalation: true
  allowedCapabilities:
  - '*'
  volumes:
  - '*'
  hostNetwork: true
  hostPorts:
  - min: 0
    max: 65535
  hostIPC: true
  hostPID: true
  runAsUser:
    rule: 'RunAsAny'
  seLinux:
    rule: 'RunAsAny'
  supplementalGroups:
    rule: 'RunAsAny'
  fsGroup:
    rule: 'RunAsAny'
```

2. Creare un nuovo ruolo per la policy di sicurezza del pod.

```
kubectl-admin create role psp:astracontrol \
    --verb=use \
    --resource=podsecuritypolicy \
    --resource-name=astracontrol
```

3. Associare il nuovo ruolo all'account del servizio.

```
kubectl-admin create rolebinding default:psp:astracontrol \
    --role=psp:astracontrol \
    --serviceaccount=astracontrol-service-account:default
```

Domande frequenti per Astra Control Center

Queste FAQ possono essere utili se stai cercando una risposta rapida a una domanda.

Panoramica

Le sezioni seguenti forniscono risposte ad alcune domande aggiuntive che potrebbero essere presentate durante l'utilizzo di Astra Control Center. Per ulteriori chiarimenti, contatta il sito astra.feedback@netapp.com

Accesso al centro di controllo Astra

Cos'è l'URL di Astra Control?

Astra Control Center utilizza l'autenticazione locale e un URL specifico per ciascun ambiente.

Per l'URL, in un browser, immettere il nome di dominio completo (FQDN) impostato nel campo spec.astraAddress nel file Astra_Control_Center_min.yaml custom resource Definition (CRD) al momento dell'installazione di Astra Control Center. Il messaggio di posta elettronica è il valore impostato nel campo spec.email nel CRD Astra_Control_Center_min.yaml.

Licensing

Utilizzo la licenza di valutazione. Come si passa alla licenza completa?

È possibile passare facilmente a una licenza completa ottenendo il file di licenza NetApp (NLF).

Fasi

- Dalla barra di navigazione a sinistra, selezionare **account** > **licenza**.
- Selezionare **Aggiungi licenza**.
- Individuare il file di licenza scaricato e selezionare **Aggiungi**.

Utilizzo la licenza di valutazione. Posso comunque gestire le applicazioni?

Sì, puoi testare la funzionalità di gestione delle app con la licenza Evaluation.

Registrazione dei cluster Kubernetes

Devo aggiungere nodi di lavoro al cluster Kubernetes dopo l'aggiunta ad Astra Control. Cosa devo fare?

È possibile aggiungere nuovi nodi di lavoro ai pool esistenti. Questi verranno rilevati automaticamente da Astra Control. Se i nuovi nodi non sono visibili in Astra Control, controllare se i nuovi nodi di lavoro eseguono il tipo di immagine supportato. È inoltre possibile verificare lo stato dei nuovi nodi di lavoro utilizzando `kubectl get nodes` comando.

Come si annulla la gestione corretta di un cluster?

1. "Annulla la gestione delle applicazioni da Astra Control".
2. "Annullare la gestione del cluster da Astra Control".

Cosa succede alle mie applicazioni e ai miei dati dopo aver rimosso il cluster Kubernetes da Astra

Control?

La rimozione di un cluster da Astra Control non apporta alcuna modifica alla configurazione del cluster (applicazioni e storage persistente). Eventuali snapshot di Astra Control o backup delle applicazioni su quel cluster non saranno disponibili per il ripristino. I backup persistenti dello storage creati da Astra Control rimangono all'interno di Astra Control, ma non sono disponibili per il ripristino.

> Rimuovere sempre un cluster da Astra Control prima di eliminarlo con altri metodi. L'eliminazione di un cluster utilizzando un altro tool mentre viene ancora gestito da Astra Control può causare problemi all'account Astra Control.

NetApp Trident viene disinstallato automaticamente da un cluster quando viene disgestito? quando si disgestisce un cluster da Astra Control Center, Trident non viene disinstallato automaticamente dal cluster. Per disinstallare Trident, è necessario "Seguire questa procedura nella documentazione di Trident".

Gestione delle applicazioni

Astra Control può implementare un'applicazione?

Astra Control non implementa le applicazioni. Le applicazioni devono essere implementate all'esterno di Astra Control.

Cosa succede alle applicazioni dopo che non li gestisco da Astra Control?

Eventuali backup o snapshot esistenti verranno eliminati. Le applicazioni e i dati rimangono disponibili. Le operazioni di gestione dei dati non saranno disponibili per le applicazioni non gestite o per eventuali backup o snapshot ad esse appartenenti.

- Astra Control può gestire un'applicazione su storage non NetApp?*

No Mentre Astra Control è in grado di rilevare applicazioni che utilizzano storage non NetApp, non può gestire un'applicazione che utilizza storage non NetApp.

Dovrei gestire Astra Control da solo? No, non dovresti gestire Astra Control perché è un'applicazione di sistema.

I pod malsani influiscono sulla gestione delle applicazioni? se un'applicazione gestita ha i pod in uno stato non integro, Astra Control non può creare nuovi backup e cloni.

Operazioni di gestione dei dati

Nel mio account sono presenti snapshot che non ho creato. Da dove sono venuti?

In alcune situazioni, Astra Control crea automaticamente uno snapshot come parte di un processo di backup, clonazione o ripristino.

La mia applicazione utilizza diversi PVS. Astra Control eseguirà snapshot e backup di tutti questi PVC?

Sì. Un'operazione snapshot su un'applicazione di Astra Control include l'istantanea di tutti i PVS associati ai PVC dell'applicazione.

È possibile gestire le snapshot acquisite da Astra Control direttamente attraverso un'interfaccia o un'archiviazione a oggetti diversa?

No Le snapshot e i backup eseguiti da Astra Control possono essere gestiti solo con Astra Control.