Replica dei volumi con SnapMirror

Astra Trident

NetApp
July 18, 2025

This PDF was generated from https://docs.netapp.com/it-it/trident-2402/trident-use/enable-acp.html on July 18, 2025. Always check docs.netapp.com for the latest.

Sommario

Replica dei volumi con SnapMirror . . . 1
- Prerequisiti per la replica . . . 1
- Creare un PVC specchiato . . . 1
- Stati di replica dei volumi . . . 4
- Promozione del PVC secondario durante un failover non pianificato . . . 5
- Promozione del PVC secondario durante un failover pianificato . . . 5
- Ripristinare una relazione di mirroring dopo un failover . . . 5
- Operazioni supplementari . . . 6
  - Replicare il PVC primario in un nuovo PVC secondario . . . 6
  - Ridimensionare un PVC specchiato, primario o secondario . . . 6
  - Rimuovere la replica da un PVC . . . 6
  - Eliminazione di un PVC (precedentemente specchiato) . . . 6
  - Eliminare una TMR . . . 6
- Aggiorna relazioni mirror quando ONTAP è online . . . 6
- Aggiorna relazioni di mirroring quando ONTAP non è in linea . . . 7
- Abilita Astra Control Provisioner . . . 7

Replica dei volumi con SnapMirror

Con Astra Control Provivisioner, puoi creare relazioni di mirroring tra un volume di origine su un cluster e il volume di destinazione sul cluster in peering per replicare i dati per il disaster recovery. È possibile utilizzare una definizione di risorsa personalizzata (CRD) con nome per eseguire le seguenti operazioni:

- Creare relazioni di mirroring tra volumi (PVC)
- Rimuovere le relazioni di mirroring tra volumi
- Interrompere le relazioni di mirroring
- Promozione del volume secondario in condizioni di disastro (failover)
- Eseguire la transizione senza perdita di dati delle applicazioni da cluster a cluster (durante failover o migrazioni pianificati)

Prerequisiti per la replica

Prima di iniziare, verificare che siano soddisfatti i seguenti prerequisiti:

Cluster ONTAP

- **Astra Control Provisioner**: Astra Control Provisioner versione 23,10 o successiva deve esistere sia sui cluster Kubernetes di origine che di destinazione che utilizzano ONTAP come backend.
- **Licenze**: 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. Per ulteriori informazioni, fare riferimento "Panoramica sulle licenze SnapMirror in ONTAP" a.

Peering

- **Cluster e SVM**: I backend dello storage ONTAP devono essere peering. Per ulteriori informazioni, fare riferimento "Panoramica del peering di cluster e SVM" a.

  > Assicurati che i nomi delle SVM utilizzati nella relazione di replica tra due cluster ONTAP siano univoci.

- **Astra Control Provisioner e SVM**: Le SVM remote in cui è stato eseguito il peering devono essere disponibili per Astra Control Provisioner nel cluster di destinazione.

Driver supportati

- La replica di un volume è supportata per i driver ontap-nas e ontap-san.

Creare un PVC specchiato

Seguire questi passaggi e utilizzare gli esempi CRD per creare una relazione di mirroring tra volumi primari e secondari.

Fasi

1. Eseguire i seguenti passaggi sul cluster Kubernetes primario:
   a. Creare un oggetto StorageClass con il `trident.netapp.io/replication: true` parametro.

Esempio

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: csi-nas
provisioner: csi.trident.netapp.io
parameters:
  backendType: "ontap-nas"
  fsType: "nfs"
  trident.netapp.io/replication: "true"
```

b. Crea un PVC con StorageClass creato in precedenza.

Esempio

```
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: csi-nas
spec:
  accessModes:
  - ReadWriteMany
  resources:
    requests:
      storage: 1Gi
  storageClassName: csi-nas
```

c. Creare una CR MirrorRelationship con informazioni locali.

Esempio

```
kind: TridentMirrorRelationship
apiVersion: trident.netapp.io/v1
metadata:
  name: csi-nas
spec:
  state: promoted
  volumeMappings:
  - localPVCName: csi-nas
```

Astra Control Provivisioner recupera le informazioni interne del volume e dello stato attuale di data Protection (DP) del volume, quindi popola il campo di stato di MirrorRelationship.

d. Procurarsi il TridentMirrorRelationship CR per ottenere il nome interno e la SVM del PVC.

```
kubectl get tmr csi-nas
```

```
kind: TridentMirrorRelationship
apiVersion: trident.netapp.io/v1
metadata:
  name: csi-nas
  generation: 1
spec:
  state: promoted
  volumeMappings:
  - localPVCName: csi-nas
status:
  conditions:
  - state: promoted
    localVolumeHandle:
"datavserver:trident_pvc_3bedd23c_46a8_4384_b12b_3c38b313c1e1"
    localPVCName: csi-nas
    observedGeneration: 1
```

2. Eseguire i seguenti passaggi sul cluster Kubernetes secondario:
   a. Creare una classe StorageClass con il parametro trident.netapp.io/replication: true.

      Esempio

      ```
      apiVersion: storage.k8s.io/v1
      kind: StorageClass
      metadata:
        name: csi-nas
      provisioner: csi.trident.netapp.io
      parameters:
        trident.netapp.io/replication: true
      ```

   b. Creare una CR MirrorRelationship con informazioni sulla destinazione e sulla sorgente.

Esempio

```
kind: TridentMirrorRelationship
apiVersion: trident.netapp.io/v1
metadata:
  name: csi-nas
spec:
  state: established
  volumeMappings:
  - localPVCName: csi-nas
    remoteVolumeHandle:
"datavserver:trident_pvc_3bedd23c_46a8_4384_b12b_3c38b313c1e1"
```

Astra Control Provisioner creerà una relazione SnapMirror con il nome della policy di relazione configurata (o default per ONTAP) e la inizializzerà.

c. Crea un PVC con StorageClass creato in precedenza per agire come secondario (destinazione SnapMirror).

Esempio

```
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: csi-nas
  annotations:
    trident.netapp.io/mirrorRelationship: csi-nas
spec:
  accessModes:
  - ReadWriteMany
resources:
  requests:
    storage: 1Gi
storageClassName: csi-nas
```

Astra Control Provisioner controllerà la CRD TridentMirrorRelationship e non creerà il volume se la relazione non esiste. Se esiste una relazione, Astra Control Provisioner garantirà che il nuovo volume FlexVol venga inserito in una SVM a cui viene inviata la SVM remota definita in MirrorRelationship.

Stati di replica dei volumi

Una relazione mirror Trident (TMR) è un CRD che rappresenta un'estremità di una relazione di replica tra PVC. Il TMR di destinazione ha uno stato, che indica ad Astra Control Provisioner lo stato desiderato. Il TMR di destinazione ha i seguenti stati:

- **Stabilito**: Il PVC locale è il volume di destinazione di una relazione speculare, e questa è una nuova relazione.

- **Promosso**: Il PVC locale è ReadWrite e montabile, senza alcuna relazione speculare attualmente in vigore.
- **Ristabilito**: Il PVC locale è il volume di destinazione di una relazione speculare ed era anche precedentemente in quella relazione speculare.
  - Lo stato ristabilito deve essere utilizzato se il volume di destinazione era in una relazione con il volume di origine perché sovrascrive il contenuto del volume di destinazione.
  - Se il volume non era precedentemente in relazione con l'origine, lo stato ristabilito non riuscirà.

Promozione del PVC secondario durante un failover non pianificato

Eseguire il seguente passaggio sul cluster Kubernetes secondario:

- Aggiornare il campo *spec.state* di TridentMirrorRelationship a `promoted`.

Promozione del PVC secondario durante un failover pianificato

Durante un failover pianificato (migrazione), eseguire le seguenti operazioni per promuovere il PVC secondario:

Fasi

1. Sul cluster Kubernetes primario, creare una snapshot del PVC e attendere la creazione dello snapshot.
2. Sul cluster Kubernetes primario, creare SnapshotInfo CR per ottenere dettagli interni.

   Esempio

   ```
   kind: SnapshotInfo
   apiVersion: trident.netapp.io/v1
   metadata:
     name: csi-nas
   spec:
     snapshot-name: csi-nas-snapshot
   ```

3. Nel cluster Kubernetes secondario, aggiornare il campo *spec.state* del *TridentMirrorRelationship* CR a *Promoted* e *spec.promotedSnapshotHandle* come nome interno dello snapshot.
4. Sul cluster Kubernetes secondario, confermare lo stato (campo status.state) di TridentMirrorRelationship a promosso.

Ripristinare una relazione di mirroring dopo un failover

Prima di ripristinare una relazione di specchiatura, scegliere il lato che si desidera creare come nuovo primario.

Fasi

1. Nel cluster Kubernetes secondario, verificare che i valori per il campo *spec.remoteVolumeHandle* in TridentMirrorRelationship siano aggiornati.

2. Sul cluster Kubernetes secondario, aggiornare il campo *spec.mirror* di TridentMirrorRelationship a `reestablished`.

Operazioni supplementari

Astra Control Provivisioner supporta le seguenti operazioni sui volumi primario e secondario:

Replicare il PVC primario in un nuovo PVC secondario

Assicurarsi di disporre già di un PVC primario e di un PVC secondario.

Fasi

1. Eliminare i CRD PersistentVolumeClaim e TridentMirrorRelationship dal cluster (destinazione) secondario stabilito.
2. Eliminare il CRD TridentMirrorRelationship dal cluster primario (origine).
3. Creare un nuovo CRD TridentMirrorRelationship nel cluster primario (di origine) per il nuovo PVC secondario (di destinazione) che si desidera stabilire.

Ridimensionare un PVC specchiato, primario o secondario

Il PVC può essere ridimensionato normalmente, ONTAP espanderà automaticamente qualsiasi flevxols di destinazione se la quantità di dati supera le dimensioni correnti.

Rimuovere la replica da un PVC

Per rimuovere la replica, eseguire una delle seguenti operazioni sul volume secondario corrente:

- Eliminare MirrorRelationship sul PVC secondario. Questo interrompe la relazione di replica.
- In alternativa, aggiornare il campo spec.state a *Promoted*.

Eliminazione di un PVC (precedentemente specchiato)

Astra Control Provivisioner verifica la presenza di PVC replicati e rilascia la relazione di replica prima di tentare di eliminare il volume.

Eliminare una TMR

L'eliminazione di una TMR su un lato di una relazione specchiata fa sì che la TMR rimanente passi allo stato *promosso* prima che Astra Control Provivisioner completi l'eliminazione. Se il TMR selezionato per l'eliminazione è già nello stato *promosso*, non esiste alcuna relazione di mirror esistente e il TMR verrà rimosso e Astra Control Provisioner promuoverà il PVC locale in *ReadWrite*. Questa eliminazione rilascia i metadati SnapMirror per il volume locale in ONTAP. Se in futuro questo volume viene utilizzato in una relazione di mirroring, deve utilizzare un nuovo TMR con uno stato di replica del volume *stabilito* quando si crea la nuova relazione di mirroring.

Aggiorna relazioni mirror quando ONTAP è online

Le relazioni speculari possono essere aggiornate in qualsiasi momento dopo che sono state stabilite. È possibile utilizzare i `state: promoted` campi o `state: reestablished` per aggiornare le relazioni. Quando si trasferisce un volume di destinazione a un volume ReadWrite regolare, è possibile utilizzare

PromotedSnapshotHandle per specificare uno snapshot specifico su cui ripristinare il volume corrente.

Aggiorna relazioni di mirroring quando ONTAP non è in linea

Puoi utilizzare un CRD per eseguire un update di SnapMirror senza che Astra Control disponga di connettività diretta al cluster ONTAP. Fare riferimento al seguente formato di esempio di TridentActionMirrorUpdate:

Esempio

```
apiVersion: trident.netapp.io/v1
kind: TridentActionMirrorUpdate
metadata:
  name: update-mirror-b
spec:
  snapshotHandle: "pvc-1234/snapshot-1234"
  tridentMirrorRelationshipName: mirror-b
```

`status.state` Riflette lo stato del CRD TridentActionMirrorUpdate. Può assumere un valore da *riuscito*, *in corso* o *non riuscito*.

Abilita Astra Control Provisioner

Le versioni 23,10 e successive di Trident includono la possibilità di utilizzare Astra Control Provisioner, che consente agli utenti dotati di licenza Astra Control di accedere a funzionalità avanzate di provisioning dello storage. Astra Control Provisioner fornisce questa funzionalità estesa oltre alle funzionalità standard basate su CSI Astra Trident. È possibile utilizzare questa procedura per abilitare e installare Astra Control Provisioner.

L'abbonamento a Astra Control Service include automaticamente la licenza per l'utilizzo di Astra Control Provisioner.

In arrivo gli update di Astra Control, Astra Control Provivisioner sostituirà Astra Trident come provisioner di storage e orchestrator e sarà obbligatorio per l'utilizzo di Astra Control. Per questo motivo, si consiglia vivamente agli utenti di Astra Control di attivare Astra Control Provisioner. Astra Trident continuerà a rimanere open source e ad essere rilasciato, mantenuto, supportato e aggiornato con le nuove CSI e altre funzionalità di NetApp.

Come faccio a sapere se devo abilitare Astra Control Provivisioner?

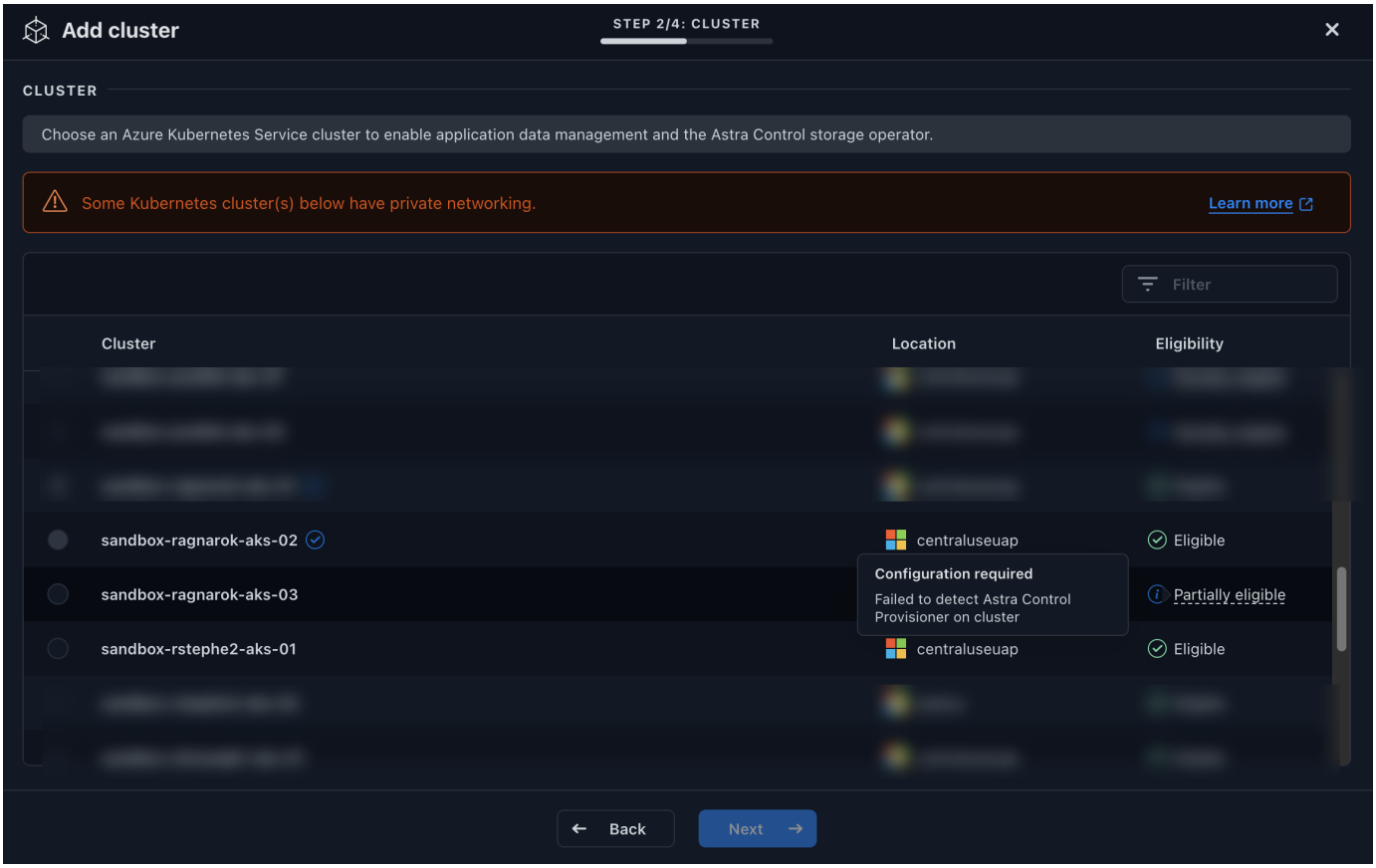
Se si aggiunge un cluster ad Astra Control Service che non ha Astra Trident precedentemente installato, il cluster verrà contrassegnato come `Eligible`. Dopo di voi "Aggiungi il cluster a Astra Control", Astra Control Provivisioner verrà abilitato automaticamente.

Se il cluster non è contrassegnato `Eligible`, verrà contrassegnato `Partially eligible` a causa di uno dei seguenti fattori:

- Sta utilizzando una versione meno recente di Astra Trident
- Sta utilizzando un Astra Trident 23,10 che non ha ancora attivato l'opzione di provisioning

- Si tratta di un tipo di cluster che non consente l'abilitazione automatica

Per `Partially eligible` i casi, usa queste istruzioni per abilitare manualmente Astra Control Provisioner per il tuo cluster.



Prima di attivare Astra Control Provisioner

Se hai già un Astra Trident senza Astra Control Provivisioner e vuoi abilitare Astra Control Provivisioner, esegui prima quanto segue:

- **Se Astra Trident è installato, conferma che la sua versione si trovi all'interno di una finestra a quattro release**: Puoi eseguire un aggiornamento diretto a Astra Trident 24,02 con Astra Control Provisioner se il tuo Astra Trident si trova all'interno di una finestra a quattro release della versione 24,02. Ad esempio, puoi eseguire l'upgrade direttamente da Astra Trident 23,04 a 24,02.
- **Confirmi che il tuo cluster ha un'architettura di sistema AMD64**: L'immagine Astra Control Provivisioner è fornita in entrambe le architetture CPU AMD64 e ARM64, ma solo AMD64 è supportato da Astra Control.

Fasi

1. Accedere al Registro di sistema dell'immagine di controllo Astra di NetApp:
   a. Effettua l'accesso all'interfaccia utente di Astra Control Service e registra l'ID account Astra Control.
      i. Selezionare l'icona della figura in alto a destra nella pagina.
      ii. Selezionare **API access**.
      iii. Annotare l'ID account.
   b. Nella stessa pagina, selezionare **generate API token**, copiare la stringa del token API negli Appunti e salvarla nell'editor.

c. Accedere al registro Astra Control utilizzando il metodo preferito:

```
docker login cr.astra.netapp.io -u <account-id> -p <api-token>
```

```
crane auth login cr.astra.netapp.io -u <account-id> -p <api-token>
```

2. (Solo registri personalizzati) attenersi alla seguente procedura per spostare l'immagine nel registro personalizzato. Se non si utilizza un registro, seguire i passaggi dell'operatore Trident nella sezione successiva.

> Per i seguenti comandi, puoi utilizzare Podman al posto di Docker. Se si utilizza un ambiente Windows, si consiglia di utilizzare PowerShell.

Docker

a. Estrarre l'immagine di Astra Control provisioner dal Registro di sistema:

> L'immagine estratta non supporta più piattaforme e supporta solo la stessa piattaforma dell'host che ha estratto l'immagine, ad esempio Linux AMD64.

```
docker pull cr.astra.netapp.io/astra/trident-acp:24.02.0
--platform <cluster platform>
```

Esempio:

```
docker pull cr.astra.netapp.io/astra/trident-acp:24.02.0
--platform linux/amd64
```

b. Contrassegnare l'immagine:

```
docker tag cr.astra.netapp.io/astra/trident-acp:24.02.0
<my_custom_registry>/trident-acp:24.02.0
```

c. Inviare l'immagine al registro personalizzato:

```
docker push <my_custom_registry>/trident-acp:24.02.0
```

Gru

a. Copiare il manifesto di Astra Control Provisioner nel registro personalizzato:

```
crane copy cr.astra.netapp.io/astra/trident-acp:24.02.0
<my_custom_registry>/trident-acp:24.02.0
```

3. Determinare se il metodo di installazione originale di Astra Trident ha utilizzato un.
4. Abilita Astra Control Provisioner in Astra Trident utilizzando il metodo di installazione utilizzato originariamente:

Operatore Astra Trident

a. "Scaricare il programma di installazione di Astra Trident ed estrarlo".

b. Completa questi passaggi se non hai ancora installato Astra Trident o se hai rimosso l'operatore dall'implementazione originale di Astra Trident:

   i. Creare il CRD:

```
kubectl create -f
deploy/crds/trident.netapp.io_tridentorchestrators_crd_post1.1
6.yaml
```

   ii. Creare lo spazio dei nomi tridente (`kubectl create namespace trident`) o confermare che lo spazio dei nomi tridente esista ancora (`kubectl get all -n trident`). Se lo spazio dei nomi è stato rimosso, crearlo di nuovo.

c. Aggiorna Astra Trident alla versione 24.02.0:

> Per i cluster che eseguono Kubernetes 1,24 o versione precedente, utilizzare `bundle_pre_1_25.yaml`. Per i cluster che eseguono Kubernetes 1,25 o versione successiva, utilizzare `bundle_post_1_25.yaml`.

```
kubectl -n trident apply -f trident-installer/deploy/<bundle-
name.yaml>
```

d. Verificare che Astra Trident sia in esecuzione:

```
kubectl get torc -n trident
```

Risposta:

```
NAME      AGE
trident   21m
```

e. se si dispone di un registro che utilizza segreti, creare un segreto da utilizzare per estrarre l'immagine di Astra Control Provisioner:

```
kubectl create secret docker-registry <secret_name> -n trident
--docker-server=<my_custom_registry> --docker-username=<username>
--docker-password=<token>
```

f. Modificare il TridentOrchestrator CR e apportare le seguenti modifiche:

```
kubectl edit torc trident -n trident
```

i. Impostare una posizione del Registro di sistema personalizzata per l'immagine Astra Trident o estrarla dal Registro di sistema Astra Control (`tridentImage: <my_custom_registry>/trident:24.02.0` o `tridentImage: netapp/trident:24.02.0`).

ii. Attiva Astra Control Provivioner (`enableACP: true`).

iii. Impostare la posizione del Registro di sistema personalizzata per l'immagine Astra Control Provivioner o estrarla dal Registro di sistema Astra Control (`acpImage: <my_custom_registry>/trident-acp:24.02.0` o `acpImage: cr.astra.netapp.io/astra/trident-acp:24.02.0`).

iv. Se è stato stabilito segreti di estrazione delle immagini precedentemente in questa procedura, è possibile impostarli qui (`imagePullSecrets: - <secret_name>`). Usare lo stesso nome segreto che hai stabilito nei passaggi precedenti.

```
apiVersion: trident.netapp.io/v1
kind: TridentOrchestrator
metadata:
  name: trident
spec:
  debug: true
  namespace: trident
  tridentImage: <registry>/trident:24.02.0
  enableACP: true
  acpImage: <registry>/trident-acp:24.02.0
  imagePullSecrets:
  - <secret_name>
```

g. Salvare e uscire dal file. Il processo di distribuzione si avvia automaticamente.

h. Verificare che l'operatore, la distribuzione e i replicaset siano stati creati.

```
kubectl get all -n trident
```

> In un cluster Kubernetes dovrebbe esserci solo **un'istanza** dell'operatore. Non creare implementazioni multiple dell'operatore Astra Trident.

i. Verificare che il `trident-acp` contenitore sia in funzione e che `acpVersion` lo stato sia `24.02.0 Installed`:

```
kubectl get torc -o yaml
```

Risposta:

```
status:
  acpVersion: 24.02.0
  currentInstallationParams:
    ...
    acpImage: <registry>/trident-acp:24.02.0
    enableACP: "true"
    ...
  ...
  status: Installed
```

tridentctl

a. "Scaricare il programma di installazione di Astra Trident ed estrarlo".

b. "Se disponi già di un Astra Trident, disinstallarlo dal cluster che lo ospita".

c. Installare Astra Trident con Astra Control Provivisioner abilitato (`--enable-acp=true`):

```
./tridentctl -n trident install --enable-acp=true --acp
-image=mycustomregistry/trident-acp:24.02
```

d. Confermare che Astra Control Provisioner è stato abilitato:

```
./tridentctl -n trident version
```

Risposta:

```
+----------------+----------------+-------------+ | SERVER
VERSION | CLIENT VERSION | ACP VERSION | +----------------
+----------------+-------------+ | 24.02.0 | 24.02.0 | 24.02.0. |
+----------------+----------------+-------------+
```

Timone

a. Se Astra Trident 23.07.1 o versione precedente è installato, "disinstallazione" l'operatore e altri componenti.

b. Se il cluster Kubernetes esegue la versione 1,24 o precedente, elimina psp:

```
kubectl delete psp tridentoperatorpod
```

c. Aggiungere il repository Astra Trident Helm:

```
helm repo add netapp-trident https://netapp.github.io/trident-
helm-chart
```

d. Aggiornare il grafico Helm:

```
helm repo update netapp-trident
```

Risposta:

```
Hang tight while we grab the latest from your chart
repositories...
...Successfully got an update from the "netapp-trident" chart
repository
Update Complete. □Happy Helming!□
```

e. Elencare le immagini:

```
./tridentctl images -n trident
```

Risposta:

```
| v1.28.0            | netapp/trident:24.02.0|
|                    | docker.io/netapp/trident-
autosupport:24.02|
|                    | registry.k8s.io/sig-storage/csi-
provisioner:v4.0.0|
|                    | registry.k8s.io/sig-storage/csi-
attacher:v4.5.0|
|                    | registry.k8s.io/sig-storage/csi-
resizer:v1.9.3|
|                    | registry.k8s.io/sig-storage/csi-
snapshotter:v6.3.3|
|                    | registry.k8s.io/sig-storage/csi-node-
driver-registrar:v2.10.0 |
|                    | netapp/trident-operator:24.02.0 (optional)
```

f. Assicurarsi che l’operatore di tridente 24.02.0 sia disponibile:

```
helm search repo netapp-trident/trident-operator --versions
```

Risposta:

```
NAME                            CHART VERSION   APP VERSION
DESCRIPTION
netapp-trident/trident-operator 100.2402.0      24.02.0         A
```

g. Utilizzare `helm install` ed eseguire una delle seguenti opzioni che includono queste impostazioni:
   - Un nome per la posizione di distribuzione
   - La versione di Astra Trident
   - Il nome dell'immagine di Astra Control provisioner
   - Il flag per abilitare il provisioner
   - (Facoltativo) percorso del Registro di sistema locale. Se si utilizza un registro locale, lo "Immagini Trident" può trovarsi in un registro o in registri diversi, ma tutte le immagini CSI devono trovarsi nello stesso registro.
   - Il namespace Trident

Opzioni

- Immagini senza registro

```
helm install trident netapp-trident/trident-operator --version
100.2402.0 --set acpImage=cr.astra.netapp.io/astra/trident-
acp:24.02.0 --set enableACP=true --set operatorImage=netapp/trident-
operator:24.02.0 --set
tridentAutosupportImage=docker.io/netapp/trident-autosupport:24.02
--set tridentImage=netapp/trident:24.02.0 --namespace trident
```

- Immagini in uno o più registri

```
helm install trident netapp-trident/trident-operator --version
100.2402.0 --set acpImage=<your-registry>:<acp image> --set
enableACP=true --set imageRegistry=<your-registry>/sig-storage --set
operatorImage=netapp/trident-operator:24.02.0 --set
tridentAutosupportImage=docker.io/netapp/trident-autosupport:24.02
--set tridentImage=netapp/trident:24.02.0 --namespace trident
```

È possibile utilizzare `helm list` per rivedere i dettagli dell'installazione, ad esempio nome, spazio dei nomi, grafico, stato, versione dell'applicazione, e numero di revisione.

Se hai problemi nell'implementazione di Trident utilizzando Helm, esegui questo comando per disinstallare completamente Astra Trident:

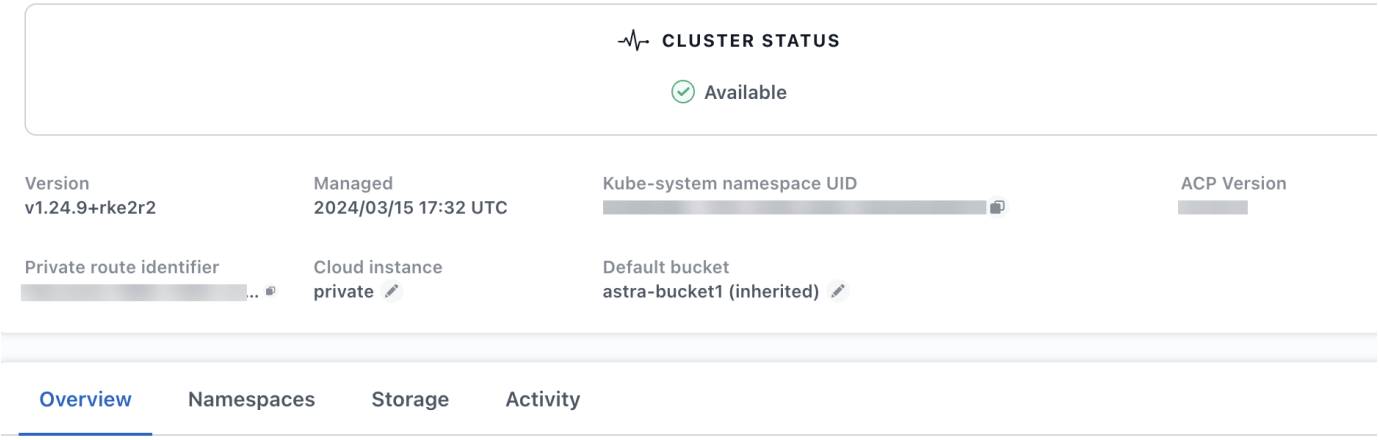
```
./tridentctl uninstall -n trident
```

Non "Rimuovere completamente i CRD Astra Trident" come parte della disinstallazione prima di tentare di abilitare nuovamente Astra Control Provivisioner.

Risultato

La funzionalità Astra Control Provisioner è abilitata ed è possibile utilizzare qualsiasi funzionalità disponibile per la versione in esecuzione.

Dopo l'installazione di Astra Control provisioner, il cluster che ospita il provisioner nell'interfaccia utente di Astra Control mostrerà un `ACP version` numero di versione installata piuttosto che `Trident version` sul campo.



Per ulteriori informazioni

- "Documentazione sugli aggiornamenti di Astra Trident"

Informazioni sul copyright

Copyright © 2025 NetApp, Inc. Tutti i diritti riservati. Stampato negli Stati Uniti d'America. Nessuna porzione di questo documento soggetta a copyright può essere riprodotta in qualsiasi formato o mezzo (grafico, elettronico o meccanico, inclusi fotocopie, registrazione, nastri o storage in un sistema elettronico) senza previo consenso scritto da parte del detentore del copyright.

Il software derivato dal materiale sottoposto a copyright di NetApp è soggetto alla seguente licenza e dichiarazione di non responsabilità:

IL PRESENTE SOFTWARE VIENE FORNITO DA NETAPP "COSÌ COM'È" E SENZA QUALSIVOGLIA TIPO DI GARANZIA IMPLICITA O ESPRESSA FRA CUI, A TITOLO ESEMPLIFICATIVO E NON ESAUSTIVO, GARANZIE IMPLICITE DI COMMERCIABILITÀ E IDONEITÀ PER UNO SCOPO SPECIFICO, CHE VENGONO DECLINATE DAL PRESENTE DOCUMENTO. NETAPP NON VERRÀ CONSIDERATA RESPONSABILE IN ALCUN CASO PER QUALSIVOGLIA DANNO DIRETTO, INDIRETTO, ACCIDENTALE, SPECIALE, ESEMPLARE E CONSEQUENZIALE (COMPRESI, A TITOLO ESEMPLIFICATIVO E NON ESAUSTIVO, PROCUREMENT O SOSTITUZIONE DI MERCI O SERVIZI, IMPOSSIBILITÀ DI UTILIZZO O PERDITA DI DATI O PROFITTI OPPURE INTERRUZIONE DELL'ATTIVITÀ AZIENDALE) CAUSATO IN QUALSIVOGLIA MODO O IN RELAZIONE A QUALUNQUE TEORIA DI RESPONSABILITÀ, SIA ESSA CONTRATTUALE, RIGOROSA O DOVUTA A INSOLVENZA (COMPRESA LA NEGLIGENZA O ALTRO) INSORTA IN QUALSIASI MODO ATTRAVERSO L'UTILIZZO DEL PRESENTE SOFTWARE ANCHE IN PRESENZA DI UN PREAVVISO CIRCA L'EVENTUALITÀ DI QUESTO TIPO DI DANNI.

NetApp si riserva il diritto di modificare in qualsiasi momento qualunque prodotto descritto nel presente documento senza fornire alcun preavviso. NetApp non si assume alcuna responsabilità circa l'utilizzo dei prodotti o materiali descritti nel presente documento, con l'eccezione di quanto concordato espressamente e per iscritto da NetApp. L'utilizzo o l'acquisto del presente prodotto non comporta il rilascio di una licenza nell'ambito di un qualche diritto di brevetto, marchio commerciale o altro diritto di proprietà intellettuale di NetApp.

Il prodotto descritto in questa guida può essere protetto da uno o più brevetti degli Stati Uniti, esteri o in attesa di approvazione.

LEGENDA PER I DIRITTI SOTTOPOSTI A LIMITAZIONE: l'utilizzo, la duplicazione o la divulgazione da parte degli enti governativi sono soggetti alle limitazioni indicate nel sottoparagrafo (b)(3) della clausola Rights in Technical Data and Computer Software del DFARS 252.227-7013 (FEB 2014) e FAR 52.227-19 (DIC 2007).

I dati contenuti nel presente documento riguardano un articolo commerciale (secondo la definizione data in FAR 2.101) e sono di proprietà di NetApp, Inc. Tutti i dati tecnici e il software NetApp forniti secondo i termini del presente Contratto sono articoli aventi natura commerciale, sviluppati con finanziamenti esclusivamente privati. Il governo statunitense ha una licenza irrevocabile limitata, non esclusiva, non trasferibile, non cedibile, mondiale, per l'utilizzo dei Dati esclusivamente in connessione con e a supporto di un contratto governativo statunitense in base al quale i Dati sono distribuiti. Con la sola esclusione di quanto indicato nel presente documento, i Dati non possono essere utilizzati, divulgati, riprodotti, modificati, visualizzati o mostrati senza la previa approvazione scritta di NetApp, Inc. I diritti di licenza del governo degli Stati Uniti per il Dipartimento della Difesa sono limitati ai diritti identificati nella clausola DFARS 252.227-7015(b) (FEB 2014).

Informazioni sul marchio commerciale

NETAPP, il logo NETAPP e i marchi elencati alla pagina http://www.netapp.com/TM sono marchi di NetApp, Inc. Gli altri nomi di aziende e prodotti potrebbero essere marchi dei rispettivi proprietari.