# Best practice e consigli

## Astra Trident

NetApp
November 14, 2025

This PDF was generated from https://docs.netapp.com/it-it/trident-2301/trident-reco/deploy-reco.html on November 14, 2025. Always check docs.netapp.com for the latest.

# Sommario


Best practice e consigli . . . 1
- Implementazione . . . 1
  - Eseguire l'implementazione in uno spazio dei nomi dedicato . . . 1
  - Utilizza quote e limiti di intervallo per controllare il consumo dello storage . . . 1
- Configurazione dello storage . . . 1
  - Panoramica della piattaforma . . . 1
  - Best practice per ONTAP e Cloud Volumes ONTAP . . . 1
  - Best practice di SolidFire . . . 6
  - Dove trovare ulteriori informazioni? . . . 8
- Integrare Astra Trident . . . 8
  - Selezione e implementazione dei driver . . . 8
  - Design di classe storage . . . 12
  - Progettazione di un pool virtuale . . . 13
  - Operazioni di volume . . . 14
  - Implementare i servizi OpenShift . . . 15
  - Servizio di metriche . . . 17
- Protezione dei dati . . . 19
  - Eseguire il backup di `etcd` dati del cluster . . . 19
  - Ripristinare la data utilizzando le snapshot ONTAP . . . 20
  - Replicare i dati utilizzando ONTAP . . . 20
  - Ripristinare i dati utilizzando le snapshot Element . . . 24
- Sicurezza . . . 24
  - Sicurezza . . . 24
  - Linux Unified Key Setup (LUKS) . . . 25

# Best practice e consigli

## Implementazione

Quando si implementa Astra Trident, utilizzare i consigli elencati di seguito.

### Eseguire l'implementazione in uno spazio dei nomi dedicato

"Spazi dei nomi" fornire una separazione amministrativa tra diverse applicazioni e costituire una barriera per la condivisione delle risorse. Ad esempio, un PVC di uno spazio dei nomi non può essere utilizzato da un altro. Astra Trident fornisce risorse PV a tutti gli spazi dei nomi nel cluster Kubernetes e sfrutta di conseguenza un account di servizio con privilegi elevati.

Inoltre, l'accesso al pod Trident potrebbe consentire a un utente di accedere alle credenziali del sistema di storage e ad altre informazioni sensibili. È importante assicurarsi che gli utenti delle applicazioni e le applicazioni di gestione non abbiano la possibilità di accedere alle definizioni degli oggetti Trident o ai pod stessi.

### Utilizza quote e limiti di intervallo per controllare il consumo dello storage

Kubernetes dispone di due funzionalità che, se combinate, offrono un potente meccanismo per limitare il consumo di risorse da parte delle applicazioni. Il "meccanismo di quota dello storage" consente all'amministratore di implementare limiti di consumo di capacità e numero di oggetti globali e specifici per classe di storage in base allo spazio dei nomi. Inoltre, utilizzando un "limite di intervallo" Garantisce che le richieste PVC rientrino in un valore minimo e massimo prima che la richiesta venga inoltrata al provisioning.

Questi valori sono definiti in base allo spazio dei nomi, il che significa che ogni spazio dei nomi deve avere valori definiti che sono in linea con i requisiti delle risorse. Vedere qui per informazioni su "come sfruttare le quote".

## Configurazione dello storage

Ogni piattaforma di storage del portfolio NetApp dispone di funzionalità uniche che offrono vantaggi alle applicazioni, containerizzate o meno.

### Panoramica della piattaforma

Trident funziona con ONTAP ed Element. Non esiste una piattaforma più adatta a tutte le applicazioni e gli scenari rispetto all'altra, tuttavia, è necessario tenere conto delle esigenze dell'applicazione e del team che amministra il dispositivo quando si sceglie una piattaforma.

Seguire le Best practice di base per il sistema operativo host con il protocollo che si sta sfruttando. Se lo si desidera, si consiglia di includere Best practice applicative, se disponibili, con impostazioni di backend, classe di storage e PVC per ottimizzare lo storage per applicazioni specifiche.

### Best practice per ONTAP e Cloud Volumes ONTAP

Scopri le Best practice per la configurazione di ONTAP e Cloud Volumes ONTAP per Trident.

I seguenti consigli sono linee guida per la configurazione di ONTAP per i carichi di lavoro containerizzati, che consumano volumi che vengono forniti dinamicamente da Trident. Ciascuno di essi deve essere considerato e

valutato per l'adeguatezza nel proprio ambiente.

### Utilizzare SVM dedicate a Trident

Le macchine virtuali di storage (SVM) forniscono isolamento e separazione amministrativa tra tenant su un sistema ONTAP. Dedicare una SVM alle applicazioni consente la delega dei privilegi e l'applicazione di Best practice per limitare il consumo delle risorse.

Sono disponibili diverse opzioni per la gestione di SVM:

- Fornire l'interfaccia di gestione del cluster nella configurazione back-end, insieme alle credenziali appropriate, e specificare il nome SVM.
- Creare un'interfaccia di gestione dedicata per la SVM utilizzando Gestione di sistema di ONTAP o l'interfaccia CLI.
- Condividere il ruolo di gestione con un'interfaccia dati NFS.

In ogni caso, l'interfaccia deve essere in DNS e il nome DNS deve essere utilizzato durante la configurazione di Trident. In questo modo è possibile semplificare alcuni scenari di disaster recovery, ad esempio SVM-DR, senza utilizzare la conservazione delle identità di rete.

Non esiste alcuna preferenza tra avere una LIF di gestione dedicata o condivisa per SVM, tuttavia, è necessario assicurarsi che le policy di sicurezza della rete siano allineate con l'approccio scelto. In ogni caso, la LIF di gestione deve essere accessibile via DNS per facilitare la massima flessibilità dovrebbe "SVM-DR" essere utilizzata insieme a Trident.

### Limitare il numero massimo di volumi

I sistemi storage ONTAP hanno un numero massimo di volumi, che varia in base alla versione software e alla piattaforma hardware. Vedere "NetApp Hardware Universe" Per la piattaforma e la versione di ONTAP specifiche per determinare i limiti esatti. Una volta esaurito il numero di volumi, le operazioni di provisioning non vengono eseguite solo per Trident, ma per tutte le richieste di storage.

Di Trident `ontap-nas` e. `ontap-san` I driver forniscono un FlexVolume per ogni volume persistente Kubernetes (PV) creato. Il `ontap-nas-economy` Il driver crea circa un FlexVolume ogni 200 PVS (configurabile tra 50 e 300). Il `ontap-san-economy` Il driver crea circa un FlexVolume ogni 100 PVS (configurabile tra 50 e 200). Per evitare che Trident utilizzi tutti i volumi disponibili sul sistema storage, è necessario impostare un limite per SVM. È possibile eseguire questa operazione dalla riga di comando:

```
vserver modify -vserver <svm_name> -max-volumes <num_of_volumes>
```

Il valore per `max-volumes` varia in base a diversi criteri specifici per l'ambiente:

- Il numero di volumi esistenti nel cluster ONTAP
- Il numero di volumi che si prevede di eseguire il provisioning al di fuori di Trident per altre applicazioni
- Il numero di volumi persistenti che si prevede siano utilizzati dalle applicazioni Kubernetes

Il `max-volumes` Il valore è il totale dei volumi forniti in tutti i nodi del cluster ONTAP e non in un singolo nodo ONTAP. Di conseguenza, potrebbero verificarsi alcune condizioni in cui un nodo del cluster ONTAP potrebbe avere volumi con provisioning Trident molto più o meno elevati rispetto a un altro nodo.

Ad esempio, un cluster ONTAP a due nodi può ospitare un massimo di 2000 FlexVolumes. Il fatto che il

numero massimo di volumi sia impostato su 1250 appare molto ragionevole. Tuttavia, se solo "aggregati" Da un nodo vengono assegnati alla SVM oppure gli aggregati assegnati da un nodo non possono essere sottoposti a provisioning (ad esempio, a causa della capacità), quindi l'altro nodo diventa la destinazione di tutti i volumi con provisioning Trident. Ciò significa che il limite di volume potrebbe essere raggiunto per quel nodo prima di `max-volumes` Viene raggiunto il valore, con conseguente impatto sulle operazioni di Trident e di altri volumi che utilizzano tale nodo. **È possibile evitare questa situazione assicurandosi che gli aggregati di ciascun nodo del cluster siano assegnati alla SVM utilizzata da Trident in numeri uguali.**

#### Limitare le dimensioni massime dei volumi creati da Trident

Per configurare le dimensioni massime dei volumi che possono essere creati da Trident, utilizzare `limitVolumeSize` nel `backend.json` definizione.

Oltre a controllare le dimensioni del volume nell'array di storage, è necessario sfruttare le funzionalità di Kubernetes.

#### Configurare Trident per l'utilizzo di CHAP bidirezionale

È possibile specificare i nomi utente e le password dell'iniziatore CHAP e di destinazione nella definizione di backend e impostare Trident per abilitare CHAP su SVM. Utilizzando il `useCHAP` Parametro nella configurazione back-end, Trident autentica le connessioni iSCSI per i backend ONTAP con CHAP. Il supporto CHAP bidirezionale è disponibile con Trident 20.04 e versioni successive.

#### Creare e utilizzare una policy di QoS SVM

L'utilizzo di una policy di qualità del servizio ONTAP, applicata alla SVM, limita il numero di IOPS consumabili dai volumi sottoposti a provisioning Trident. In questo modo è possibile impedire a un "prevenire un bullismo" container fuori controllo di influenzare i carichi di lavoro esterni alla SVM di Trident.

È possibile creare una policy QoS per SVM in pochi passaggi. Per informazioni più precise, consultare la documentazione relativa alla versione di ONTAP in uso. Nell'esempio riportato di seguito viene creata una policy di QoS che limita a 5000 gli IOPS totali disponibili per la SVM.

```
# create the policy group for the SVM
qos policy-group create -policy-group <policy_name> -vserver <svm_name>
-max-throughput 5000iops

# assign the policy group to the SVM, note this will not work
# if volumes or files in the SVM have existing QoS policies
vserver modify -vserver <svm_name> -qos-policy-group <policy_name>
```

Inoltre, se la tua versione di ONTAP lo supporta, puoi considerare l'utilizzo di un QoS minimo per garantire una quantità di throughput per i carichi di lavoro containerizzati. QoS adattiva non è compatibile con una policy di livello SVM.

Il numero di IOPS dedicati ai carichi di lavoro containerizzati dipende da molti aspetti. Tra le altre cose, queste includono:

- Altri carichi di lavoro che utilizzano lo storage array. Se sono presenti altri carichi di lavoro, non correlati all'implementazione di Kubernetes, che utilizzano le risorse di storage, è necessario prestare attenzione a garantire che tali carichi di lavoro non vengano accidentalmente influenzati negativamente.

- Carichi di lavoro previsti eseguiti in container. Se i carichi di lavoro con requisiti IOPS elevati verranno eseguiti in container, una policy QoS bassa comporta un'esperienza negativa.

È importante ricordare che una policy di QoS assegnata a livello di SVM comporta la condivisione dello stesso pool di IOPS di tutti i volumi forniti a SVM. Se una, o un numero limitato, delle applicazioni containerizzate presenta un elevato requisito di IOPS, potrebbe diventare un problema per gli altri carichi di lavoro containerizzati. In questo caso, è possibile utilizzare l'automazione esterna per assegnare policy QoS per volume.

> È necessario assegnare il gruppo di criteri QoS a SVM **only** se la versione di ONTAP è precedente alla 9.8.

**Creare gruppi di policy QoS per Trident**

La qualità del servizio (QoS) garantisce che le performance dei carichi di lavoro critici non vengano degradate da carichi di lavoro concorrenti. I gruppi di policy QoS di ONTAP offrono opzioni di QoS per i volumi e consentono agli utenti di definire il limite massimo di throughput per uno o più carichi di lavoro. Per ulteriori informazioni su QoS, vedere "Garanzia di throughput con QoS". È possibile specificare i gruppi di policy QoS nel backend o in un pool di storage, che vengono applicati a ciascun volume creato in quel pool o backend.

ONTAP dispone di due tipi di gruppi di policy QoS: Tradizionale e adattiva. I gruppi di policy tradizionali forniscono un throughput massimo (o minimo, nelle versioni successive) costante negli IOPS. La QoS adattiva scala automaticamente il throughput in base alle dimensioni del carico di lavoro, mantenendo il rapporto tra IOPS e TB|GB in base alle dimensioni del carico di lavoro. Questo offre un vantaggio significativo quando si gestiscono centinaia o migliaia di carichi di lavoro in un'implementazione di grandi dimensioni.

Quando si creano gruppi di criteri QoS, considerare quanto segue:

- Impostare `qosPolicy` digitare `defaults` blocco della configurazione back-end. Vedere il seguente esempio di configurazione del backend:

```
  ---
version: 1
storageDriverName: ontap-nas
managementLIF: 0.0.0.0
dataLIF: 0.0.0.0
svm: svm0
username: user
password: pass
defaults:
  qosPolicy: standard-pg
storage:
- labels:
    performance: extreme
  defaults:
    adaptiveQosPolicy: extremely-adaptive-pg
- labels:
    performance: premium
  defaults:
    qosPolicy: premium-pg
```

- È necessario applicare i gruppi di criteri per volume, in modo che ogni volume ottenga l'intero throughput come specificato dal gruppo di criteri. I gruppi di criteri condivisi non sono supportati.

Per ulteriori informazioni sui gruppi di criteri QoS, vedere "Comandi QoS di ONTAP 9.8".

**Limitare l'accesso alle risorse di storage ai membri del cluster Kubernetes**

Limitare l'accesso ai volumi NFS e alle LUN iSCSI create da Trident è un componente critico della posizione di sicurezza per l'implementazione di Kubernetes. In questo modo si impedisce agli host che non fanno parte del cluster Kubernetes di accedere ai volumi e di modificare i dati in modo imprevisto.

È importante comprendere che gli spazi dei nomi sono il limite logico delle risorse in Kubernetes. L'ipotesi è che le risorse nello stesso namespace siano in grado di essere condivise, tuttavia, cosa importante, non esiste alcuna funzionalità di spazio dei nomi incrociato. Ciò significa che anche se i PVS sono oggetti globali, quando sono associati a un PVC sono accessibili solo da pod che si trovano nello stesso namespace. **È fondamentale assicurarsi che gli spazi dei nomi siano utilizzati per fornire la separazione quando appropriato.**

La preoccupazione principale per la maggior parte delle organizzazioni in relazione alla sicurezza dei dati in un contesto Kubernetes è che un processo in un container può accedere allo storage montato sull'host, ma non è destinato al container. "Spazi dei nomi" sono progettati per evitare questo tipo di compromesso. Tuttavia, esiste un'eccezione: I container con privilegi.

Un container con privilegi è un container che viene eseguito con un numero di autorizzazioni a livello di host sostanzialmente superiore al normale. Per impostazione predefinita, questi elementi non vengono rifiutati, quindi disattivare la funzionalità utilizzando "policy di sicurezza pod".

Per i volumi in cui si desidera accedere sia da Kubernetes che da host esterni, lo storage deve essere gestito in modo tradizionale, con il PV introdotto dall'amministratore e non gestito da Trident. In questo modo, il volume di storage viene distrutto solo quando Kubernetes e gli host esterni si sono disconnessi e non

utilizzano più il volume. Inoltre, è possibile applicare una policy di esportazione personalizzata, che consente l'accesso dai nodi del cluster Kubernetes e dai server di destinazione all'esterno del cluster Kubernetes.

Per le implementazioni che hanno nodi di infrastruttura dedicati (ad esempio, OpenShift) o altri nodi che non sono in grado di pianificare le applicazioni utente, è necessario utilizzare policy di esportazione separate per limitare ulteriormente l'accesso alle risorse di storage. Ciò include la creazione di una policy di esportazione per i servizi implementati nei nodi dell'infrastruttura (ad esempio, i servizi OpenShift Metrics e Logging) e le applicazioni standard implementate nei nodi non dell'infrastruttura.

#### Utilizzare una policy di esportazione dedicata

È necessario verificare l'esistenza di una policy di esportazione per ciascun backend che consenta l'accesso solo ai nodi presenti nel cluster Kubernetes. Trident può creare e gestire automaticamente le policy di esportazione a partire dalla release 20.04. In questo modo, Trident limita l'accesso ai volumi che fornisce ai nodi nel cluster Kubernetes e semplifica l'aggiunta/eliminazione dei nodi.

In alternativa, è anche possibile creare manualmente una policy di esportazione e compilarla con una o più regole di esportazione che elaborano ogni richiesta di accesso al nodo:

- Utilizzare `vserver export-policy create` Comando ONTAP CLI per creare il criterio di esportazione.
- Aggiungere regole ai criteri di esportazione utilizzando `vserver export-policy rule create` Comando CLI ONTAP.

L'esecuzione di questi comandi consente di limitare i nodi Kubernetes che hanno accesso ai dati.

#### Disattiva `showmount` Per l'applicazione SVM

Il `showmount` Questa funzione consente a un client NFS di eseguire query su SVM per un elenco delle esportazioni NFS disponibili. Un pod implementato nel cluster Kubernetes può emettere `showmount -e` Eseguire il comando in base al LIF dei dati e ricevere un elenco di montaggi disponibili, inclusi quelli a cui non ha accesso. Sebbene questo, di per sé, non sia un compromesso in termini di sicurezza, fornisce informazioni non necessarie che potrebbero aiutare un utente non autorizzato a connettersi a un'esportazione NFS.

Disattivare `showmount` Utilizzando il comando CLI ONTAP a livello di SVM:

```
vserver nfs modify -vserver <svm_name> -showmount disabled
```

## Best practice di SolidFire

Scopri le Best practice per la configurazione dello storage SolidFire per Trident.

#### Crea account SolidFire

Ogni account SolidFire rappresenta un unico proprietario di volume e riceve un proprio set di credenziali CHAP (Challenge-Handshake Authentication Protocol). È possibile accedere ai volumi assegnati a un account utilizzando il nome dell'account e le relative credenziali CHAP o un gruppo di accesso al volume. A un account possono essere assegnati fino a duemila volumi, ma un volume può appartenere a un solo account.

### Creare una policy QoS

Utilizzare le policy di qualità del servizio (QoS) di SolidFire se si desidera creare e salvare un'impostazione di qualità del servizio standardizzata che può essere applicata a molti volumi.

È possibile impostare i parametri QoS in base al volume. Le performance per ciascun volume possono essere garantite impostando tre parametri configurabili che definiscono la QoS: Min IOPS, Max IOPS e Burst IOPS.

Di seguito sono riportati i possibili valori IOPS minimi, massimi e burst per la dimensione del blocco di 4 Kb.

| Parametro IOPS | Definizione | Min. valore | Valore predefinito | Max. Valore (4 Kb) |
|---|---|---|---|---|
| IOPS minimi | Il livello garantito di performance per un volume. | 50 | 50 | 15000 |
| IOPS max | Le performance non supereranno questo limite. | 50 | 15000 | 200,000 |
| IOPS burst | IOPS massimi consentiti in uno scenario a burst breve. | 50 | 15000 | 200,000 |

Anche se i massimi IOPS e burst IOPS possono essere impostati su 200,000, le performance massime reali di un volume sono limitate dall'utilizzo del cluster e dalle performance per nodo.

Le dimensioni dei blocchi e la larghezza di banda influiscono direttamente sul numero di IOPS. Con l'aumentare delle dimensioni dei blocchi, il sistema aumenta la larghezza di banda fino a raggiungere un livello necessario per elaborare blocchi di dimensioni maggiori. Con l'aumentare della larghezza di banda, il numero di IOPS che il sistema è in grado di raggiungere diminuisce. Vedere "Qualità del servizio SolidFire" Per ulteriori informazioni su QoS e performance.

### Autenticazione SolidFire

Element supporta due metodi di autenticazione: CHAP e VAG (Volume Access Group). CHAP utilizza il protocollo CHAP per autenticare l'host nel backend. I gruppi di accesso ai volumi controllano l'accesso ai volumi previsti dall'IT. NetApp consiglia di utilizzare CHAP per l'autenticazione, poiché è più semplice e non ha limiti di scalabilità.

Trident con il provisioning CSI avanzato supporta l'utilizzo dell'autenticazione CHAP. I VAG devono essere utilizzati solo nella modalità operativa tradizionale non CSI.

L'autenticazione CHAP (verifica che l'iniziatore sia l'utente del volume desiderato) è supportata solo con il controllo degli accessi basato su account. Se si utilizza CHAP per l'autenticazione, sono disponibili due opzioni: CHAP unidirezionale e CHAP bidirezionale. CHAP unidirezionale autentica l'accesso al volume utilizzando il nome account SolidFire e il segreto dell'iniziatore. L'opzione CHAP bidirezionale rappresenta il metodo più sicuro per autenticare il volume, in quanto il volume autentica l'host tramite il nome account e il segreto dell'iniziatore, quindi l'host autentica il volume tramite il nome account e il segreto di destinazione.

Tuttavia, se non è possibile attivare CHAP e sono richiesti VAG, creare il gruppo di accesso e aggiungere gli

iniziatori host e i volumi al gruppo di accesso. Ogni IQN aggiunto a un gruppo di accesso può accedere a ciascun volume del gruppo con o senza autenticazione CHAP. Se iSCSI Initiator è configurato per utilizzare l'autenticazione CHAP, viene utilizzato il controllo degli accessi basato sull'account. Se iSCSI Initiator non è configurato per utilizzare l'autenticazione CHAP, viene utilizzato il controllo di accesso del gruppo di accesso al volume.

### Dove trovare ulteriori informazioni?

Di seguito sono elencate alcune delle Best practice. Eseguire una ricerca in "Libreria NetApp" per le versioni più recenti.

**ONTAP**

- "Guida alle Best practice e all'implementazione di NFS"
- "Gestione dello storage SAN" (Per iSCSI)
- "Configurazione iSCSI Express per RHEL"

**Software Element**

- "Configurazione di SolidFire per Linux"

**NetApp HCI**

- "Prerequisiti per l'implementazione di NetApp HCI"
- "Accedi al NetApp Deployment Engine"

**Informazioni sulle Best practice applicative**

- "Best practice per MySQL su ONTAP"
- "Best practice per MySQL su SolidFire"
- "NetApp SolidFire e Cassandra"
- "Best practice Oracle su SolidFire"
- "Best practice PostgreSQL su SolidFire"

Non tutte le applicazioni hanno linee guida specifiche, è importante collaborare con il team NetApp e utilizzare "Libreria NetApp" per trovare la documentazione più aggiornata.

# Integrare Astra Trident

Per integrare Astra Trident, è necessario integrare i seguenti elementi di progettazione e architettura: Selezione e implementazione dei driver, progettazione della classe di storage, progettazione del pool virtuale, impatto del PVC (Persistent Volume Claim) sul provisioning dello storage, sulle operazioni dei volumi e sull'implementazione dei servizi OpenShift utilizzando Astra Trident.

### Selezione e implementazione dei driver

Selezionare e implementare un driver back-end per il sistema storage.

### Driver backend ONTAP

I driver di back-end ONTAP si differenziano in base al protocollo utilizzato e al modo in cui i volumi vengono forniti nel sistema di storage. Pertanto, prendere in considerazione attentamente quando si decide quale driver implementare.

A un livello superiore, se l'applicazione dispone di componenti che richiedono storage condiviso (diversi pod che accedono allo stesso PVC), i driver basati su NAS sarebbero la scelta predefinita, mentre i driver iSCSI basati su blocchi soddisfano le esigenze dello storage non condiviso. Scegli il protocollo in base ai requisiti dell'applicazione e al livello di comfort dei team di storage e infrastruttura. In generale, la differenza tra le due applicazioni è minima, quindi spesso la decisione si basa sulla necessità o meno di uno storage condiviso (in cui più di un pod necessitano di accesso simultaneo).

I driver backend ONTAP disponibili sono:

- `ontap-nas`: Ogni PV fornito è un FlexVolume ONTAP completo.
- `ontap-nas-economy`: Ogni PV fornito è un qtree, con un numero configurabile di qtree per FlexVolume (il valore predefinito è 200).
- `ontap-nas-flexgroup`: Vengono utilizzati tutti i PV forniti come ONTAP FlexGroup completo e tutti gli aggregati assegnati a una SVM.
- `ontap-san`: Ogni PV fornito è un LUN all'interno del proprio FlexVolume.
- `ontap-san-economy`: Ogni PV fornito è un LUN, con un numero configurabile di LUN per FlexVolume (il valore predefinito è 100).

La scelta tra i tre driver NAS ha alcune ramificazioni alle funzionalità, che sono rese disponibili per l'applicazione.

Si noti che, nelle tabelle seguenti, non tutte le funzionalità sono esposte attraverso Astra Trident. Alcuni devono essere applicati dall'amministratore dello storage dopo il provisioning, se si desidera questa funzionalità. Le note a piè di pagina in superscript distinguono le funzionalità per funzionalità e driver.

| Driver NAS ONTAP | Snapshot | Cloni | Policy di esportazione dinamiche | Multi-attach | QoS | Ridimensionare | Replica |
|---|---|---|---|---|---|---|---|
| `ontap-nas` | Sì | Sì | Yes [5] | Sì | Yes [1] | Sì | Yes [1] |
| `ontap-nas-economy` | Yes [3] | Yes [3] | Yes [5] | Sì | Yes [3] | Sì | Yes [3] |
| `ontap-nas-flexgroup` | Yes [1] | No | Yes [5] | Sì | Yes [1] | Sì | Yes [1] |

Astra Trident offre 2 driver SAN per ONTAP, le cui funzionalità sono illustrate di seguito.

| Driver SAN ONTAP | Snapshot | Cloni | Multi-attach | CHAP bidirezionale | QoS | Ridimensionare | Replica |
|---|---|---|---|---|---|---|---|
| `ontap-san` | Sì | Sì | Yes [4] | Sì | Yes [1] | Sì | Yes [1] |
| `ontap-san-economy` | Sì | Sì | Yes [4] | Sì | Yes [3] | Sì | Yes [3] |

> Nota a piè di pagina per le tabelle precedenti: Yes [1]: Non gestito da Astra Trident Yes [2]: Gestito da Astra Trident, ma non da PV Granular Yesnota a piè di pagina:3[]: Non gestito da Astra Trident e non da PV Granular Yesnota a piè di pagina:4[]: Supportato da CSI Trident

Le funzionalità non granulari PV vengono applicate all'intero FlexVolume e tutti i PVS (ovvero qtree o LUN in FlexVol condivisi) condividono una pianificazione comune.

Come si può vedere nelle tabelle precedenti, gran parte delle funzionalità tra `ontap-nas` e. `ontap-nas-economy` è lo stesso. Tuttavia, perché il `ontap-nas-economy` Driver limita la capacità di controllare la pianificazione in base alla granularità per PV, questo può influire in particolare sul disaster recovery e sulla pianificazione del backup. Per i team di sviluppo che desiderano sfruttare la funzionalità dei cloni PVC sullo storage ONTAP, ciò è possibile solo quando si utilizza `ontap-nas`, `ontap-san` oppure `ontap-san-economy` driver.

> Il `solidfire-san` Il driver è anche in grado di clonare i PVC.

#### Driver backend Cloud Volumes ONTAP

Cloud Volumes ONTAP offre il controllo dei dati e funzionalità di storage di livello Enterprise per diversi casi di utilizzo, tra cui condivisioni di file e storage a livello di blocco che servono protocolli NAS e SAN (NFS, SMB/CIFS e iSCSI). I driver compatibili per Cloud Volume ONTAP sono `ontap-nas`, `ontap-nas-economy`, `ontap-san` e. `ontap-san-economy`. Questi sono validi per Cloud Volume ONTAP per Azure, Cloud Volume ONTAP per GCP.

#### Driver backend Amazon FSX per ONTAP

Amazon FSX per ONTAP consente ai clienti di sfruttare le funzionalità, le performance e le funzionalità amministrative di NetApp che conoscono, sfruttando al contempo la semplicità, l'agilità, la sicurezza e la scalabilità dell'archiviazione dei dati su AWS. FSX per ONTAP supporta molte delle funzionalità del file system e delle API di amministrazione di ONTAP. I driver compatibili per Cloud Volume ONTAP sono `ontap-nas`, `ontap-nas-economy`, `ontap-nas-flexgroup`, `ontap-san` e. `ontap-san-economy`.

#### Driver backend NetApp HCI/SolidFire

Il `solidfire-san` Il driver utilizzato con le piattaforme NetApp HCI/SolidFire aiuta l'amministratore a configurare un backend elemento per Trident in base ai limiti di QoS. Se si desidera progettare il backend per impostare i limiti di QoS specifici sui volumi forniti da Trident, utilizzare `type` nel file backend. L'amministratore può inoltre limitare le dimensioni del volume che è possibile creare sullo storage utilizzando `limitVolumeSize` parametro. Attualmente, le funzionalità di storage degli elementi come il ridimensionamento del volume e la replica del volume non sono supportate da `solidfire-san` driver. Queste operazioni devono essere eseguite manualmente tramite l'interfaccia utente Web di Element Software.

| Driver SolidFire | Snapshot | Cloni | Multi-attach | CAP | QoS | Ridimensionare | Replica |
|---|---|---|---|---|---|---|---|
| `solidfire-san` | Sì | Sì | Yes [2] | Sì | Sì | Sì | Yes [1] |

> Nota a piè di pagina: Yesnota a piè di pagina:1[]: Non gestito da Astra Trident Yesnota a piè di pagina:2[]: Supportato per i volumi raw-block

### Driver backend Azure NetApp Files

Astra Trident utilizza `azure-netapp-files` driver per la gestione di "Azure NetApp Files" servizio.

Per ulteriori informazioni su questo driver e su come configurarlo, consultare "Configurazione backend Astra Trident per Azure NetApp Files".

| Driver Azure NetApp Files | Snapshot | Cloni | Multi-attach | QoS | Espandere | Replica |
|---|---|---|---|---|---|---|
| `azure-netapp-files` | Sì | Sì | Sì | Sì | Sì | Yes [1] |

> Nota a piè di pagina: Yesnota a piè di pagina:1[]: Non gestita da Astra Trident

### Driver backend Cloud Volumes Service su Google Cloud

Astra Trident utilizza `gcp-cvs` Driver per il collegamento a Cloud Volumes Service su Google Cloud.

Il `gcp-cvs` Il driver utilizza pool virtuali per astrarre il backend e consentire ad Astra Trident di determinare il posizionamento del volume. L'amministratore definisce i pool virtuali in `backend.json` file. Le classi di storage utilizzano selettori per identificare i pool virtuali in base all'etichetta.

- Se i pool virtuali sono definiti nel backend, Astra Trident tenterà di creare un volume nei pool di storage di Google Cloud a cui tali pool virtuali sono limitati.
- Se i pool virtuali non sono definiti nel backend, Astra Trident selezionerà un pool di storage Google Cloud dai pool di storage disponibili nella regione.

Per configurare il backend di Google Cloud su Astra Trident, è necessario specificare `projectNumber`, `apiRegion`, e. `apiKey` nel file backend. Il numero del progetto si trova nella console di Google Cloud. La chiave API viene presa dal file della chiave privata dell'account di servizio creato durante la configurazione dell'accesso API per Cloud Volumes Service su Google Cloud.

Per ulteriori informazioni sui tipi di servizio e i livelli di servizio di Cloud Volumes Service su Google Cloud, vedere "Scopri di più sul supporto di Astra Trident per CVS per GCP".

| Driver Cloud Volumes Service per Google Cloud | Snapshot | Cloni | Multi-attach | QoS | Espandere | Replica |
|---|---|---|---|---|---|---|
| `gcp-cvs` | Sì | Sì | Sì | Sì | Sì | Disponibile solo sul tipo di servizio CVS-Performance. |

**Note sulla replica**

- La replica non è gestita da Astra Trident.
- Il clone verrà creato nello stesso pool di storage del volume di origine.

## Design di classe storage

È necessario configurare e applicare singole classi di storage per creare un oggetto Kubernetes Storage Class. In questa sezione viene descritto come progettare una classe di storage per l'applicazione.

### Utilizzo specifico del back-end

Il filtraggio può essere utilizzato all'interno di un oggetto specifico della classe di storage per determinare quale pool o insieme di pool di storage utilizzare con tale classe di storage specifica. Nella classe di storage è possibile impostare tre set di filtri: `storagePools`, `additionalStoragePools`, e/o. `excludeStoragePools`.

Il `storagePools` parametro consente di limitare lo storage al set di pool che corrispondono a qualsiasi attributo specificato. Il `additionalStoragePools` Il parametro viene utilizzato per estendere il set di pool che Astra Trident utilizzerà per il provisioning insieme al set di pool selezionato dagli attributi e. `storagePools` parametri. È possibile utilizzare i parametri singolarmente o entrambi insieme per assicurarsi che sia selezionato il set appropriato di pool di storage.

Il `excludeStoragePools` il parametro viene utilizzato per escludere in modo specifico il set di pool elencato che corrispondono agli attributi.

### Emulare le policy di QoS

Se si desidera progettare classi di storage per emulare le policy di qualità del servizio, creare una classe di storage con `media` attributo come `hdd` oppure `ssd`. Basato su `media` Attributo menzionato nella classe di storage, Trident selezionerà il backend appropriato che serve `hdd` oppure `ssd` aggregato in modo da corrispondere all'attributo di supporto e indirizzare il provisioning dei volumi sull'aggregato specifico. Pertanto, possiamo creare una classe di storage PREMIUM che avrebbe `media` attributo impostato come `ssd` Che potrebbero essere classificati come policy DI qualità del servizio PREMIUM. È possibile creare un altro STANDARD di classe storage con l'attributo media impostato come `hdd' che potrebbe essere classificato come policy standard di QoS. Potremmo anche utilizzare l'attributo ``IOPS'' nella classe di storage per reindirizzare il provisioning a un'appliance Element che può essere definita come policy QoS.

### Utilizzare il back-end in base a funzionalità specifiche

Le classi di storage possono essere progettate per indirizzare il provisioning dei volumi su un backend specifico in cui sono abilitate funzionalità come thin provisioning e thick provisioning, snapshot, cloni e crittografia. Per specificare lo storage da utilizzare, creare classi di storage che specifichino il backend appropriato con la funzionalità richiesta attivata.

### Pool virtuali

Sono disponibili pool virtuali per tutti i backend Astra Trident. È possibile definire i pool virtuali per qualsiasi backend, utilizzando qualsiasi driver fornito da Astra Trident.

I pool virtuali consentono a un amministratore di creare un livello di astrazione sui backend a cui si può fare riferimento attraverso le classi di storage, per una maggiore flessibilità e un posizionamento efficiente dei volumi sui backend. È possibile definire backend diversi con la stessa classe di servizio. Inoltre, è possibile creare più pool di storage sullo stesso backend, ma con caratteristiche diverse. Quando una classe di storage viene configurata con un selettore con le etichette specifiche, Astra Trident sceglie un backend che corrisponde a tutte le etichette del selettore per posizionare il volume. Se le etichette del selettore Storage Class corrispondono a più pool di storage, Astra Trident sceglierà una di queste da cui eseguire il provisioning del volume.

## Progettazione di un pool virtuale

Durante la creazione di un backend, in genere è possibile specificare un set di parametri. Per l'amministratore non era possibile creare un altro backend con le stesse credenziali di storage e con un set di parametri diverso. Con l'introduzione dei pool virtuali, questo problema è stato risolto. Virtual Pools è un'astrazione di livello introdotta tra il backend e Kubernetes Storage Class, in modo che l'amministratore possa definire i parametri insieme alle etichette a cui si può fare riferimento attraverso le classi di storage di Kubernetes come un selettore, in modo indipendente dal backend. È possibile definire i pool virtuali per tutti i backend NetApp supportati con Astra Trident. L'elenco include SolidFire/NetApp HCI, ONTAP, Cloud Volumes Service su GCP e Azure NetApp Files.

Quando si definiscono i pool virtuali, si consiglia di non tentare di riorganizzare l'ordine dei pool virtuali esistenti in una definizione di backend. Si consiglia inoltre di non modificare/modificare gli attributi di un pool virtuale esistente e di non definire un nuovo pool virtuale.

### Emulazione di diversi livelli di servizio/QoS

È possibile progettare pool virtuali per l'emulazione delle classi di servizio. Utilizzando l'implementazione del pool virtuale per il servizio volume cloud per Azure NetApp Files, esaminiamo come possiamo configurare diverse classi di servizio. Configurare il backend ANF con più etichette, che rappresentano diversi livelli di performance. Impostare `servicelevel` aspect al livello di performance appropriato e aggiungere altri aspetti richiesti sotto ogni etichetta. Creare ora diverse classi di storage Kubernetes che si mappano a diversi pool virtuali. Utilizzando il `parameters.selector` Ciascun StorageClass richiama i pool virtuali che possono essere utilizzati per ospitare un volume.

### Assegnazione di un insieme specifico di aspetti

È possibile progettare più pool virtuali con un set specifico di aspetti da un singolo backend di storage. A tale scopo, configurare il backend con più etichette e impostare gli aspetti richiesti sotto ciascuna etichetta. Ora è possibile creare diverse classi di storage Kubernetes utilizzando `parameters.selector` campo che viene mappato a diversi pool virtuali. I volumi con cui viene eseguito il provisioning sul back-end avranno gli aspetti definiti nel pool virtuale scelto.

### Caratteristiche del PVC che influiscono sul provisioning dello storage

Alcuni parametri oltre la classe di storage richiesta possono influire sul processo decisionale di provisioning di Astra Trident durante la creazione di un PVC.

### Modalità di accesso

Quando si richiede lo storage tramite PVC, uno dei campi obbligatori è la modalità di accesso. La modalità desiderata può influire sul backend selezionato per ospitare la richiesta di storage.

Astra Trident tenterà di associare il protocollo di storage utilizzato al metodo di accesso specificato in base alla matrice seguente. Ciò è indipendente dalla piattaforma di storage sottostante.

| | ReadWriteOnce | ReadOnlyMany | ReadWriteMany |
|---|---|---|---|
| ISCSI | Sì | Sì | Sì (blocco raw) |
| NFS | Sì | Sì | Sì |

Una richiesta di ReadWriteMany PVC inviata a un'implementazione Trident senza un backend NFS configurato non comporterà il provisioning di alcun volume. Per questo motivo, il richiedente deve utilizzare la modalità di

accesso appropriata per la propria applicazione.

## Operazioni di volume

### Modificare i volumi persistenti

I volumi persistenti sono, con due eccezioni, oggetti immutabili in Kubernetes. Una volta creata, la policy di recupero e le dimensioni possono essere modificate. Tuttavia, questo non impedisce che alcuni aspetti del volume vengano modificati al di fuori di Kubernetes. Ciò può essere utile per personalizzare il volume per applicazioni specifiche, per garantire che la capacità non venga accidentalmente consumata o semplicemente per spostare il volume in un controller di storage diverso per qualsiasi motivo.

Attualmente, i provisioning in-tree di Kubernetes non supportano le operazioni di ridimensionamento dei volumi per NFS o iSCSI PVS. Astra Trident supporta l'espansione dei volumi NFS e iSCSI.

I dettagli di connessione del PV non possono essere modificati dopo la creazione.

### Creazione di snapshot di volumi on-demand

Astra Trident supporta la creazione on-demand di snapshot di volumi e la creazione di PVC da snapshot utilizzando il framework CSI. Gli snapshot offrono un metodo pratico per mantenere copie point-in-time dei dati e hanno un ciclo di vita indipendente dal PV di origine in Kubernetes. Queste snapshot possono essere utilizzate per clonare i PVC.

### Creare volumi da snapshot

Astra Trident supporta anche la creazione di PersistentVolumes da snapshot di volumi. A tale scopo, è sufficiente creare un PersistentVolumeClaim e citare il `datasource` come snapshot richiesto da cui è necessario creare il volume. Astra Trident gestirà questo PVC creando un volume con i dati presenti nello snapshot. Con questa funzionalità, è possibile duplicare i dati tra regioni, creare ambienti di test, sostituire un volume di produzione danneggiato o corrotto nella sua interezza o recuperare file e directory specifici e trasferirli in un altro volume collegato.

### Spostare i volumi nel cluster

Gli amministratori dello storage hanno la possibilità di spostare i volumi tra aggregati e controller nel cluster ONTAP senza interruzioni per il consumatore di storage. Questa operazione non influisce su Astra Trident o sul cluster Kubernetes, purché l'aggregato di destinazione sia un aggregato a cui ha accesso la SVM utilizzata da Astra Trident. Cosa importante, se l'aggregato è stato aggiunto di recente alla SVM, il backend dovrà essere aggiornato aggiungendolo nuovamente ad Astra Trident. In questo modo Astra Trident reinventarierà la SVM in modo che il nuovo aggregato venga riconosciuto.

Tuttavia, Astra Trident non supporta automaticamente lo spostamento dei volumi tra backend. Ciò include le SVM nello stesso cluster, tra cluster o su una piattaforma storage diversa (anche se il sistema storage è collegato ad Astra Trident).

Se un volume viene copiato in un'altra posizione, la funzione di importazione del volume può essere utilizzata per importare i volumi correnti in Astra Trident.

### Espandere i volumi

Astra Trident supporta il ridimensionamento di NFS e iSCSI PVS. Ciò consente agli utenti di ridimensionare i propri volumi direttamente attraverso il livello Kubernetes. L'espansione dei volumi è possibile per tutte le

principali piattaforme di storage NetApp, inclusi i backend ONTAP, SolidFire/NetApp HCI e Cloud Volumes Service. Per consentire la possibile espansione in un secondo momento, impostare `allowVolumeExpansion` a. `true` Nel StorageClass associato al volume. Ogni volta che è necessario ridimensionare il volume persistente, modificare `spec.resources.requests.storage` Annotazione nella richiesta di rimborso del volume persistente sulla dimensione del volume richiesta. Trident si occuperà automaticamente del ridimensionamento del volume sul cluster di storage.

#### Importare un volume esistente in Kubernetes

L'importazione dei volumi consente di importare un volume di storage esistente in un ambiente Kubernetes. Questa funzione è attualmente supportata da `ontap-nas`, `ontap-nas-flexgroup`, `solidfire-san`, `azure-netapp-files`, e. `gcp-cvs` driver. Questa funzionalità è utile quando si esegue il porting di un'applicazione esistente in Kubernetes o durante scenari di disaster recovery.

Quando si utilizza ONTAP e. `solidfire-san` driver, utilizzare il comando `tridentctl import volume <backend-name> <volume-name> -f /path/pvc.yaml` Per importare un volume esistente in Kubernetes da gestire da Astra Trident. Il file PVC YAML o JSON utilizzato nel comando del volume di importazione punta a una classe di storage che identifica Astra Trident come provider. Quando si utilizza un backend NetApp HCI/SolidFire, assicurarsi che i nomi dei volumi siano univoci. Se i nomi dei volumi sono duplicati, clonare il volume con un nome univoco in modo che la funzione di importazione dei volumi possa distinguerli.

Se il `azure-netapp-files` oppure `gcp-cvs` driver, utilizzare il comando `tridentctl import volume <backend-name> <volume path> -f /path/pvc.yaml` Importare il volume in Kubernetes da gestire da Astra Trident. In questo modo si garantisce un riferimento di volume univoco.

Quando viene eseguito il comando precedente, Astra Trident troverà il volume sul backend e ne leggerà le dimensioni. Aggiunge automaticamente (e sovrascrive se necessario) le dimensioni del volume PVC configurato. Astra Trident crea quindi il nuovo PV e Kubernetes lega il PVC al PV.

Se un container fosse stato implementato in modo da richiedere lo specifico PVC importato, rimarrebbe in sospeso fino a quando la coppia PVC/PV non sarà legata tramite il processo di importazione del volume. Una volta rilegata la coppia PVC/PV, il container dovrebbe salire, a condizione che non vi siano altri problemi.

## Implementare i servizi OpenShift

I servizi cluster OpenShift a valore aggiunto offrono funzionalità importanti agli amministratori dei cluster e alle applicazioni ospitate. Lo storage utilizzato da questi servizi può essere fornito utilizzando le risorse locali del nodo, tuttavia, questo spesso limita la capacità, le performance, la ripristinabilità e la sostenibilità del servizio. Sfruttando un array di storage Enterprise per fornire la capacità a questi servizi è possibile migliorare drasticamente il servizio, tuttavia, come per tutte le applicazioni, OpenShift e gli amministratori dello storage dovrebbero collaborare strettamente per determinare le opzioni migliori per ciascuno di essi. La documentazione di Red Hat deve essere sfruttata in maniera significativa per determinare i requisiti e garantire che le esigenze di dimensionamento e performance siano soddisfatte.

#### Servizio di registro

La distribuzione e la gestione dello storage per il registro sono state documentate su "netapp.io" in "blog".

#### Servizio di registrazione

Come gli altri servizi OpenShift, il servizio di logging viene implementato utilizzando Ansible con parametri di configurazione forniti dal file di inventario, ovvero host, forniti al playbook. Sono previsti due metodi di installazione: Distribuzione del logging durante l'installazione iniziale di OpenShift e distribuzione del logging

dopo l'installazione di OpenShift.

> A partire dalla versione 3.9 di Red Hat OpenShift, la documentazione ufficiale consiglia NFS per il servizio di logging a causa di problemi legati alla corruzione dei dati. Questo si basa sui test Red Hat dei loro prodotti. Il server NFS di ONTAP non presenta questi problemi e può facilmente eseguire il backup di un'implementazione di logging. In definitiva, la scelta del protocollo per il servizio di logging dipende da voi, sappiate che entrambi funzioneranno benissimo quando si utilizzano le piattaforme NetApp e che non vi è alcun motivo per evitare NFS se questa è la vostra preferenza.

Se si sceglie di utilizzare NFS con il servizio di registrazione, è necessario impostare la variabile Ansible `openshift_enable_unsupported_configurations` a. `true` per impedire il malfunzionamento del programma di installazione.

#### Inizia subito

Il servizio di logging può, facoltativamente, essere implementato per entrambe le applicazioni e per le operazioni principali del cluster OpenShift stesso. Se si sceglie di implementare la registrazione delle operazioni, specificando la variabile `openshift_logging_use_ops` come `true`, verranno create due istanze del servizio. Le variabili che controllano l'istanza di logging per le operazioni contengono "Ops" al loro interno, mentre l'istanza per le applicazioni non lo fa.

La configurazione delle variabili Ansible in base al metodo di implementazione è importante per garantire che lo storage corretto venga utilizzato dai servizi sottostanti. Diamo un'occhiata alle opzioni per ciascuno dei metodi di implementazione.

> Le tabelle seguenti contengono solo le variabili rilevanti per la configurazione dello storage in relazione al servizio di registrazione. Altre opzioni sono disponibili in "Documentazione di registrazione di RedHat OpenShift" che devono essere esaminate, configurate e utilizzate in base all'implementazione.

Le variabili riportate nella tabella seguente determineranno la creazione di un PV e di un PVC per il servizio di registrazione utilizzando i dettagli forniti. Questo metodo è notevolmente meno flessibile rispetto all'utilizzo del playbook di installazione dei componenti dopo l'installazione di OpenShift, tuttavia, se si dispone di volumi esistenti, si tratta di un'opzione.

| Variabile | Dettagli |
| --- | --- |
| `openshift_logging_storage_kind` | Impostare su `nfs` Per fare in modo che il programma di installazione crei un NFS PV per il servizio di registrazione. |
| `openshift_logging_storage_host` | Il nome host o l'indirizzo IP dell'host NFS. Questa opzione deve essere impostata sul LIF dei dati per la macchina virtuale. |
| `openshift_logging_storage_nfs_directory` | Il percorso di montaggio per l'esportazione NFS. Ad esempio, se il volume è giuntato come `/openshift_logging`, utilizzare tale percorso per questa variabile. |
| `openshift_logging_storage_volume_name` | Il nome, ad esempio `pv_ose_logs`, Del PV da creare. |

| Variabile | Dettagli |
| --- | --- |
| `openshift_logging_storage_volume_size` | Le dimensioni dell'esportazione NFS, ad esempio `100Gi`. |

Se il cluster OpenShift è già in esecuzione e quindi Trident è stato implementato e configurato, l'installatore può utilizzare il provisioning dinamico per creare i volumi. È necessario configurare le seguenti variabili.

| Variabile | Dettagli |
| --- | --- |
| `openshift_logging_es_pvc_dynamic` | Impostare su true per utilizzare volumi con provisioning dinamico. |
| `openshift_logging_es_pvc_storage_class_name` | Il nome della classe di storage che verrà utilizzata nel PVC. |
| `openshift_logging_es_pvc_size` | La dimensione del volume richiesto nel PVC. |
| `openshift_logging_es_pvc_prefix` | Prefisso dei PVC utilizzati dal servizio di registrazione. |
| `openshift_logging_es_ops_pvc_dynamic` | Impostare su `true` per utilizzare volumi con provisioning dinamico per l'istanza di logging ops. |
| `openshift_logging_es_ops_pvc_storage_class_name` | Il nome della classe di storage per l'istanza di logging di Ops. |
| `openshift_logging_es_ops_pvc_size` | La dimensione della richiesta di volume per l'istanza Ops. |
| `openshift_logging_es_ops_pvc_prefix` | Un prefisso per i PVC di istanza di Ops. |

#### Implementare lo stack di logging

Se si sta implementando la registrazione come parte del processo di installazione iniziale di OpenShift, è sufficiente seguire il processo di distribuzione standard. Ansible configurerà e implementerà i servizi e gli oggetti OpenShift necessari in modo che il servizio sia disponibile non appena Ansible sarà completato.

Tuttavia, se si esegue l'implementazione dopo l'installazione iniziale, Ansible dovrà utilizzare il playbook dei componenti. Questo processo potrebbe cambiare leggermente con diverse versioni di OpenShift, quindi assicurati di leggere e seguire "Documentazione di RedHat OpenShift Container Platform 3.11" per la versione in uso.

## Servizio di metriche

Il servizio Metrics fornisce all'amministratore informazioni preziose sullo stato, l'utilizzo delle risorse e la disponibilità del cluster OpenShift. È inoltre necessario per la funzionalità di scalabilità automatica di Pod e molte organizzazioni utilizzano i dati del servizio di metriche per le proprie applicazioni di riaccredito e/o visualizzazione.

Come nel caso del servizio di registrazione e di OpenShift nel suo complesso, Ansible viene utilizzato per implementare il servizio di metriche. Inoltre, come il servizio di registrazione, il servizio di metriche può essere implementato durante una configurazione iniziale del cluster o dopo che è operativo utilizzando il metodo di installazione dei componenti. Le seguenti tabelle contengono le variabili importanti per la configurazione dello storage persistente per il servizio di metriche.

Le tabelle seguenti contengono solo le variabili rilevanti per la configurazione dello storage in relazione al servizio di metriche. La documentazione contiene molte altre opzioni che devono essere esaminate, configurate e utilizzate in base all'implementazione.

| Variabile | Dettagli |
|---|---|
| `openshift_metrics_storage_kind` | Impostare su `nfs` Per fare in modo che il programma di installazione crei un NFS PV per il servizio di registrazione. |
| `openshift_metrics_storage_host` | Il nome host o l'indirizzo IP dell'host NFS. Questa opzione deve essere impostata sul valore LIF dei dati per SVM. |
| `openshift_metrics_storage_nfs_directory` | Il percorso di montaggio per l'esportazione NFS. Ad esempio, se il volume è giuntato come `/openshift_metrics`, utilizzare tale percorso per questa variabile. |
| `openshift_metrics_storage_volume_name` | Il nome, ad esempio `pv_ose_metrics`, Del PV da creare. |
| `openshift_metrics_storage_volume_size` | Le dimensioni dell'esportazione NFS, ad esempio `100Gi`. |

Se il cluster OpenShift è già in esecuzione e quindi Trident è stato implementato e configurato, l'installatore può utilizzare il provisioning dinamico per creare i volumi. È necessario configurare le seguenti variabili.

| Variabile | Dettagli |
|---|---|
| `openshift_metrics_cassandra_pvc_prefix` | Prefisso da utilizzare per i PVC di metriche. |
| `openshift_metrics_cassandra_pvc_size` | Le dimensioni dei volumi da richiedere. |
| `openshift_metrics_cassandra_storage_type` | Il tipo di storage da utilizzare per le metriche, deve essere impostato su dinamico per Ansible per creare PVC con la classe di storage appropriata. |
| `openshift_metrics_cassanda_pvc_storage_class_name` | Il nome della classe di storage da utilizzare. |

#### Implementare il servizio di metriche

Con le variabili Ansible appropriate definite nel file di host/inventario, implementare il servizio utilizzando Ansible. Se si esegue l'implementazione al momento dell'installazione di OpenShift, il PV verrà creato e utilizzato automaticamente. Se si esegue l'implementazione utilizzando i playbook dei componenti, dopo l'installazione di OpenShift, Ansible creerà tutti i PVC necessari e, dopo che Astra Trident ha eseguito il provisioning dello storage, implementerà il servizio.

Le variabili di cui sopra e il processo di implementazione possono cambiare con ogni versione di OpenShift. Verifica e segui "Guida all'implementazione di OpenShift di RedHat" per la versione in uso, in modo che sia configurata per l'ambiente in uso.

# Protezione dei dati

Scopri le opzioni di ripristino e protezione dei dati offerte dalle piattaforme di storage NetApp. Astra Trident può eseguire il provisioning di volumi che possono sfruttare alcune di queste funzionalità. È necessario disporre di una strategia di protezione e ripristino dei dati per ogni applicazione con un requisito di persistenza.

## Eseguire il backup di `etcd` dati del cluster

Astra Trident memorizza i propri metadati nel cluster Kubernetes `etcd` database. Eseguire periodicamente il backup di `etcd` I dati del cluster sono importanti per ripristinare i cluster Kubernetes in situazioni di emergenza.

**Fasi**

1. Il `etcdctl snapshot save` il comando consente di acquisire un'istantanea point-in-time di `etcd` cluster:

   ```
   sudo docker run --rm -v /backup:/backup \
     --network host \
     -v /etc/kubernetes/pki/etcd:/etc/kubernetes/pki/etcd \
     --env ETCDCTL_API=3 \
     registry.k8s.io/etcd-amd64:3.2.18 \
     etcdctl --endpoints=https://127.0.0.1:2379 \
     --cacert=/etc/kubernetes/pki/etcd/ca.crt \
     --cert=/etc/kubernetes/pki/etcd/healthcheck-client.crt \
     --key=/etc/kubernetes/pki/etcd/healthcheck-client.key \
     snapshot save /backup/etcd-snapshot.db
   ```

   Questo comando crea uno snapshot etcd creando un container etcd e salvandolo in `/backup` directory.

2. In caso di disastro, è possibile accelerare un cluster Kubernetes utilizzando le snapshot etcd. Utilizzare `etcdctl snapshot restore` comando per ripristinare uno snapshot specifico su `/var/lib/etcd` cartella. Dopo il ripristino, verificare se `/var/lib/etcd` la cartella è stata popolata con `member` cartella. Di seguito viene riportato un esempio di `etcdctl snapshot restore` comando:

   ```
   etcdctl snapshot restore '/backup/etcd-snapshot-latest.db' ; mv
   /default.etcd/member/ /var/lib/etcd/
   ```

3. Prima di inizializzare il cluster Kubernetes, copiare tutti i certificati necessari.
4. Creare il cluster con `--ignore-preflight-errors=DirAvailable—var-lib-etcd` allarme.
5. Una volta attivato il cluster, assicurarsi che i pod del sistema kube siano stati avviati.
6. Utilizzare `kubectl get crd` Per verificare se le risorse personalizzate create da Trident sono presenti e recuperare gli oggetti Trident per assicurarsi che tutti i dati siano disponibili.

## Ripristinare la data utilizzando le snapshot ONTAP

Le snapshot svolgono un ruolo importante fornendo opzioni di recovery point-in-time per i dati delle applicazioni. Tuttavia, gli snapshot non sono backup da soli, ma non proteggono da guasti del sistema di storage o altre catastrofi. Tuttavia, rappresentano un metodo pratico, rapido e semplice per ripristinare i dati nella maggior parte degli scenari. Scopri come utilizzare la tecnologia ONTAP Snapshot per eseguire backup del volume e come ripristinarli.

- Se il criterio di snapshot non è stato definito nel backend, per impostazione predefinita viene utilizzato il `none` policy. In questo modo, ONTAP non crea snapshot automatiche. Tuttavia, l'amministratore dello storage può eseguire snapshot manuali o modificare il criterio di snapshot tramite l'interfaccia di gestione di ONTAP. Ciò non influisce sul funzionamento di Trident.
- La directory di snapshot è nascosta per impostazione predefinita. In questo modo, è possibile semplificare la massima compatibilità dei volumi con provisioning tramite `ontap-nas` e. `ontap-nas-economy` driver. Attivare il `.snapshot` directory quando si utilizza `ontap-nas` e. `ontap-nas-economy` driver per consentire alle applicazioni di ripristinare direttamente i dati dalle snapshot.
- Ripristinare uno stato di un volume registrato in uno snapshot precedente utilizzando `volume snapshot restore` Comando CLI ONTAP. Quando si ripristina una copia snapshot, l'operazione di ripristino sovrascrive la configurazione del volume esistente. Tutte le modifiche apportate ai dati nel volume dopo la creazione della copia Snapshot andranno perse.

```
cluster1::*> volume snapshot restore -vserver vs0 -volume vol3 -snapshot
vol3_snap_archive
```

## Replicare i dati utilizzando ONTAP

La replica dei dati può svolgere un ruolo importante nella protezione contro la perdita di dati dovuta a guasti degli array di storage.

Per ulteriori informazioni sulle tecnologie di replica ONTAP, vedere "Documentazione ONTAP" .

### Replica di SnapMirror Storage Virtual Machine (SVM)

Puoi usare "SnapMirror" per replicare una SVM completa, che include le sue impostazioni di configurazione e i suoi volumi. In caso di disastro, è possibile attivare la SVM di destinazione di SnapMirror per iniziare a fornire i dati. Una volta ripristinati i sistemi, è possibile tornare al sistema primario.

Astra Trident non è in grado di configurare le relazioni di replica, pertanto l'amministratore dello storage può utilizzare la funzione di replica SVM di SnapMirror di ONTAP per replicare automaticamente i volumi in una destinazione di disaster recovery (DR).

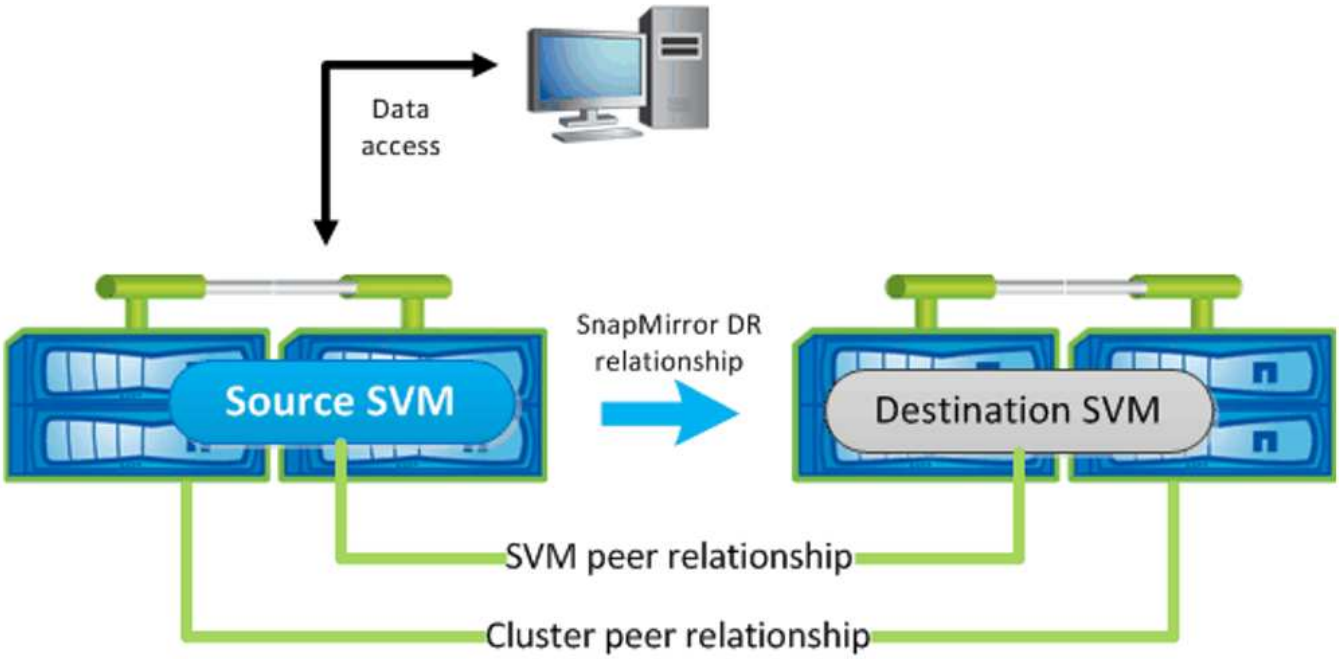
Considerare quanto segue se si intende utilizzare la funzione di replica SVM di SnapMirror o se si sta utilizzando la funzione:

- È necessario creare un backend distinto per ogni SVM, che ha SVM-DR abilitato.
- È necessario configurare le classi di storage in modo da non selezionare i backend replicati, tranne quando si desidera. Ciò è importante per evitare di disporre di volumi che non richiedono la protezione di una relazione di replica per il provisioning sul back-end che supporta SVM-DR.
- Gli amministratori delle applicazioni devono comprendere i costi e la complessità aggiuntivi associati alla replica dei dati e definire un piano di ripristino prima di sfruttare la replica dei dati.

- Prima di attivare la SVM di destinazione di SnapMirror, interrompere tutti i trasferimenti pianificati di SnapMirror, interrompere tutti i trasferimenti in corso di SnapMirror, interrompere la relazione di replica, arrestare la SVM di origine e avviare la SVM di destinazione di SnapMirror.
- Astra Trident non rileva automaticamente gli errori SVM. Pertanto, in caso di errore, l'amministratore deve eseguire `tridentctl backend update` Comando per attivare il failover di Trident sul nuovo backend.

Di seguito viene riportata una panoramica delle fasi di installazione di SVM:

- Impostare il peering tra il cluster di origine e di destinazione e SVM.
- Creare la SVM di destinazione utilizzando `-subtype dp-destination` opzione.
- Creare una pianificazione dei processi di replica per garantire che la replica avvenga agli intervalli richiesti.
- Creare una replica SnapMirror dalla SVM di destinazione alla SVM di origine utilizzando `-identity -preserve true` Opzione per garantire che le configurazioni SVM di origine e le interfacce SVM di origine vengano copiate nella destinazione. Dalla SVM di destinazione, inizializzare la relazione di replica di SnapMirror SVM.




**Workflow di disaster recovery per Trident**

Astra Trident 19.07 e versioni successive utilizzano i CRD Kubernetes per memorizzare e gestire il proprio stato. Utilizza i cluster Kubernetes `etcd` per memorizzare i metadati. Supponiamo che i Kubernetes `etcd` I file di dati e i certificati vengono memorizzati su NetApp FlexVolume. Questo FlexVolume risiede in una SVM, che ha una relazione SnapMirror SVM-DR con una SVM di destinazione nel sito secondario.

I seguenti passaggi descrivono come ripristinare un singolo cluster Kubernetes master con Astra Trident in caso di disastro:

1. In caso di errore della SVM di origine, attivare la SVM di destinazione di SnapMirror. A tale scopo, è necessario interrompere i trasferimenti pianificati di SnapMirror, interrompere i trasferimenti in corso di SnapMirror, interrompere la relazione di replica, arrestare la SVM di origine e avviare la SVM di destinazione.

2. Dalla SVM di destinazione, montare il volume che contiene Kubernetes `etcd` file di dati e certificati sull'host che verrà configurato come nodo master.
3. Copiare tutti i certificati richiesti relativi al cluster Kubernetes in `/etc/kubernetes/pki` e l'etcd `member` file sotto `/var/lib/etcd`.
4. Creare un cluster Kubernetes utilizzando `kubeadm init` con il `--ignore-preflight -errors=DirAvailable—var-lib-etcd` allarme. I nomi host utilizzati per i nodi Kubernetes devono essere gli stessi del cluster Kubernetes di origine.
5. Eseguire `kubectl get crd` Comando per verificare se tutte le risorse personalizzate di Trident sono state create e recuperare gli oggetti Trident per verificare che tutti i dati siano disponibili.
6. Aggiornare tutti i backend richiesti per riflettere il nuovo nome SVM di destinazione eseguendo il `./tridentctl update backend <backend-name> -f <backend-json-file> -n <namespace>` comando.

> Per i volumi persistenti dell'applicazione, quando viene attivata la SVM di destinazione, tutti i volumi forniti da Trident iniziano a servire i dati. Dopo aver configurato il cluster Kubernetes sul lato di destinazione seguendo i passaggi descritti in precedenza, vengono avviate tutte le implementazioni e i pod e le applicazioni containerizzate devono essere eseguite senza problemi.

**Replica del volume SnapMirror**

La replica dei volumi SnapMirror di ONTAP è una funzionalità di disaster recovery che consente il failover verso lo storage di destinazione dallo storage primario a livello di volume. SnapMirror crea una replica di volume o un mirror dello storage primario sullo storage secondario sincronizzando gli snapshot.

Di seguito viene riportata una panoramica dei passaggi per la configurazione della replica del volume di ONTAP SnapMirror:

- Impostare il peering tra i cluster in cui risiedono i volumi e le SVM che servono i dati dei volumi.
- Creare un criterio SnapMirror che controlli il comportamento della relazione e specifichi gli attributi di configurazione per tale relazione.
- Creare una relazione SnapMirror tra il volume di destinazione e il volume di origine utilizzando il[`snapmirror create` comando^] e assegnare il criterio SnapMirror appropriato.
- Una volta creata la relazione SnapMirror, inizializzarla in modo da completare un trasferimento di riferimento dal volume di origine al volume di destinazione.

#### Workflow di disaster recovery del volume SnapMirror per Trident

I seguenti passaggi descrivono come ripristinare un singolo cluster Kubernetes master con Astra Trident.

1. In caso di disastro, interrompere tutti i trasferimenti SnapMirror pianificati e interrompere tutti i trasferimenti SnapMirror in corso. Interrompere la relazione di replica tra i volumi di destinazione e di origine in modo che il volume di destinazione diventi di lettura/scrittura.
2. Dalla SVM di destinazione, montare il volume che contiene Kubernetes `etcd` file di dati e certificati sull'host, che verrà impostato come nodo master.
3. Copiare tutti i certificati richiesti relativi al cluster Kubernetes in `/etc/kubernetes/pki` e l'etcd `member` file sotto `/var/lib/etcd`.
4. Creare un cluster Kubernetes eseguendo `kubeadm init` con il `--ignore-preflight -errors=DirAvailable—var-lib-etcd` allarme. I nomi host devono essere gli stessi del cluster Kubernetes di origine.
5. Eseguire `kubectl get crd` Per verificare se tutte le risorse personalizzate di Trident sono state create e recuperare gli oggetti Trident per assicurarsi che tutti i dati siano disponibili.
6. Ripulire i backend precedenti e creare nuovi backend su Trident. Specificare la nuova LIF di gestione, il nuovo nome SVM e la password della SVM di destinazione.

#### Workflow di disaster recovery per volumi persistenti delle applicazioni

I seguenti passaggi descrivono come rendere disponibili i volumi di destinazione di SnapMirror per i carichi di lavoro containerizzati in caso di disastro:

1. Interrompere tutti i trasferimenti SnapMirror pianificati e interrompere tutti i trasferimenti SnapMirror in corso. Interrompere la relazione di replica tra il volume di destinazione e quello di origine in modo che il volume di destinazione diventi di lettura/scrittura. Ripulire le implementazioni che consumavano PVC legato ai volumi sulla SVM di origine.
2. Dopo aver configurato il cluster Kubernetes sul lato di destinazione seguendo le procedure descritte in precedenza, ripulire le implementazioni, PVC e PV, dal cluster Kubernetes.
3. Creare nuovi backend su Trident specificando la nuova LIF di gestione e dati, il nuovo nome SVM e la password della SVM di destinazione.

4. Importare i volumi richiesti come PV associato a un nuovo PVC utilizzando la funzione di importazione Trident.
5. Ridistribuire le implementazioni applicative con i PVC appena creati.

### Ripristinare i dati utilizzando le snapshot Element

Eseguire il backup dei dati su un volume Element impostando una pianificazione di snapshot per il volume e garantendo che le snapshot vengano eseguite agli intervalli richiesti. È necessario impostare la pianificazione dello snapshot utilizzando l'interfaccia utente o le API di Element. Attualmente, non è possibile impostare una pianificazione snapshot su un volume tramite `solidfire-san` driver.

In caso di danneggiamento dei dati, è possibile scegliere uno snapshot specifico e eseguire il rollback del volume nello snapshot manualmente utilizzando l'interfaccia utente o le API Element. In questo modo vengono ripristinate le modifiche apportate al volume dalla creazione dello snapshot.

# Sicurezza

## Sicurezza

Utilizza i consigli elencati di seguito per assicurarti che l'installazione di Astra Trident sia sicura.

#### Eseguire Astra Trident nel proprio namespace

È importante impedire ad applicazioni, amministratori di applicazioni, utenti e applicazioni di gestione di accedere alle definizioni degli oggetti di Astra Trident o ai pod per garantire uno storage affidabile e bloccare potenziali attività dannose.

Per separare le altre applicazioni e gli utenti da Astra Trident, installare sempre Astra Trident nel proprio spazio dei nomi Kubernetes (`trident`). L'inserimento di Astra Trident nel proprio spazio dei nomi garantisce che solo il personale amministrativo di Kubernetes abbia accesso al pod Astra Trident e agli artefatti (come i segreti di backend e CHAP, se applicabili) memorizzati negli oggetti CRD con spazio dei nomi. È necessario garantire che solo gli amministratori possano accedere allo spazio dei nomi Astra Trident e quindi a. `tridentctl` applicazione.

#### Utilizza l'autenticazione CHAP con i backend SAN ONTAP

Astra Trident supporta l'autenticazione basata su CHAP per i carichi di lavoro SAN ONTAP (utilizzando il `ontap-san` e. `ontap-san-economy` driver). NetApp consiglia di utilizzare CHAP bidirezionale con Astra Trident per l'autenticazione tra un host e il backend dello storage.

Per i backend ONTAP che utilizzano i driver di storage SAN, Astra Trident può configurare CHAP bidirezionale e gestire i nomi utente e i segreti CHAP attraverso `tridentctl`. Vedere "qui" Per capire come Astra Trident configura CHAP sui backend ONTAP.

Il supporto CHAP per i backend ONTAP è disponibile con Trident 20.04 e versioni successive.

#### Utilizza l'autenticazione CHAP con backend NetApp HCI e SolidFire

NetApp consiglia di implementare CHAP bidirezionale per garantire l'autenticazione tra un host e i backend NetApp HCI e SolidFire. Astra Trident utilizza un oggetto segreto che include due password CHAP per tenant. Quando Trident viene installato come provider CSI, gestisce i segreti CHAP e li memorizza in un

`tridentvolume` Oggetto CR per il rispettivo PV. Quando si crea un PV, CSI Astra Trident utilizza i segreti CHAP per avviare una sessione iSCSI e comunicare con il sistema NetApp HCI e SolidFire su CHAP.

> I volumi creati da CSI Trident non sono associati a alcun gruppo di accesso al volume.

Nel frontend non CSI, l'attacco di volumi come dispositivi sui nodi di lavoro viene gestito da Kubernetes. Dopo la creazione del volume, Astra Trident effettua una chiamata API al sistema NetApp HCI/SolidFire per recuperare i segreti se il segreto per quel tenant non esiste già. Astra Trident poi passa i segreti su Kubernetes. Il kubelet situato su ciascun nodo accede ai segreti tramite l'API Kubernetes e li utilizza per eseguire/abilitare CHAP tra ciascun nodo che accede al volume e il sistema NetApp HCI/SolidFire in cui si trovano i volumi.

#### Utilizzare Astra Trident con NVE e NAE

NetApp ONTAP offre la crittografia dei dati inattivi per proteggere i dati sensibili in caso di furto, restituzione o riordinamento di un disco. Per ulteriori informazioni, fare riferimento a. "Panoramica sulla configurazione di NetApp Volume Encryption".

- Se NAE è attivato sul backend, qualsiasi volume fornito in Astra Trident sarà abilitato per NAE.
- Se NAE non è attivato sul backend, qualsiasi volume fornito in Astra Trident sarà abilitato per NVE, a meno che non si imposta il flag di crittografia NVE su `false` nella configurazione back-end.

> I volumi creati in Astra Trident su un backend abilitato NAE devono essere crittografati con NVE o NAE.
>
> - È possibile impostare il flag di crittografia NVE su `true` Nella configurazione backend Trident per eseguire l'override della crittografia NAE e utilizzare una chiave di crittografia specifica per volume.
> - Impostazione del flag di crittografia NVE su `false` Su un backend abilitato NAE verrà creato un volume abilitato NAE. Non è possibile disattivare la crittografia NAE impostando il flag di crittografia NVE su `false`.

- È possibile creare manualmente un volume NVE in Astra Trident impostando esplicitamente il flag di crittografia NVE su `true`.

Per ulteriori informazioni sulle opzioni di configurazione del backend, fare riferimento a:

- "Opzioni di configurazione SAN ONTAP"
- "Opzioni di configurazione NAS ONTAP"

## Linux Unified Key Setup (LUKS)

È possibile abilitare la configurazione delle chiavi unificate Linux (LUKS) per crittografare i volumi SAN ONTAP e SAN ONTAP SU Astra Trident. Astra Trident supporta la rotazione delle passphrase e l'espansione dei volumi con crittografia LUKS.

In Astra Trident, i volumi con crittografia LUKS utilizzano il cifrario aes-xts-plain64 e la modalità, come consigliato da "NIST".

#### Prima di iniziare

- Sui nodi di lavoro deve essere installata la crittografia 2.1 o superiore (ma inferiore a 3.0). Per ulteriori

informazioni, visitare il sito "Gitlab: Crittsetup".

- Per motivi di performance, consigliamo ai nodi di lavoro di supportare Advanced Encryption Standard New Instructions (AES-NI). Per verificare il supporto AES-NI, eseguire il seguente comando:

```
grep "aes" /proc/cpuinfo
```

Se non viene restituito nulla, il processore non supporta AES-NI. Per ulteriori informazioni su AES-NI, visita: "Intel: Advanced Encryption Standard Instructions (AES-NI)".

### Attivare la crittografia LUKS

È possibile attivare la crittografia lato host per volume utilizzando la configurazione unificata delle chiavi di Linux per volumi SAN ONTAP e SAN ONTAP.

**Fasi**

1. Definire gli attributi di crittografia LUKS nella configurazione del back-end. Per ulteriori informazioni sulle opzioni di configurazione back-end per ONTAP SAN, fare riferimento a. "Opzioni di configurazione SAN ONTAP".

```
"storage": [
    {
        "labels":{"luks": "true"},
        "zone":"us_east_1a",
        "defaults": {
            "luksEncryption": "true"
        }
    },
    {
        "labels":{"luks": "false"},
        "zone":"us_east_1a",
        "defaults": {
            "luksEncryption": "false"
        }
    },
]
```

2. Utilizzare `parameters.selector` Per definire i pool di storage utilizzando la crittografia LUKS. Ad esempio:

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: luks
provisioner: netapp.io/trident
parameters:
  selector: "luks=true"
  csi.storage.k8s.io/node-stage-secret-name: luks-${pvc.name}
  csi.storage.k8s.io/node-stage-secret-namespace: ${pvc.namespace}
```

3. Creare un segreto contenente la passphrase LUKS. Ad esempio:

```
kubectl -n trident create -f luks-pvc1.yaml
apiVersion: v1
kind: Secret
metadata:
  name: luks-pvc1
stringData:
  luks-passphrase-name: A
  luks-passphrase: secretA
```

#### Limitazioni

I volumi crittografati con LUKS non possono sfruttare la deduplica e la compressione ONTAP.

### Ruotare una passphrase LUKS

È possibile ruotare la passphrase LUKS e confermare la rotazione.

> Non dimenticare una passphrase fino a quando non viene verificata la mancanza di riferimenti da qualsiasi volume, snapshot o segreto. In caso di perdita di una passphrase di riferimento, potrebbe non essere possibile montare il volume e i dati resteranno crittografati e inaccessibili.

#### A proposito di questa attività

La rotazione della passphrase LUKS si verifica quando viene creato un pod che monta il volume dopo aver specificato una nuova passphrase LUKS. Quando viene creato un nuovo pod, Astra Trident confronta la passphrase LUKS sul volume con la passphrase attiva nel segreto.

- Se la passphrase sul volume non corrisponde alla passphrase attiva nel segreto, si verifica la rotazione.
- Se la passphrase sul volume corrisponde alla passphrase attiva nel segreto, il `previous-luks-passphrase` il parametro viene ignorato.

#### Fasi

1. Aggiungere il `node-publish-secret-name` e. `node-publish-secret-namespace` Parametri StorageClass. Ad esempio:

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: csi-san
provisioner: csi.trident.netapp.io
parameters:
  trident.netapp.io/backendType: "ontap-san"
  csi.storage.k8s.io/node-stage-secret-name: luks
  csi.storage.k8s.io/node-stage-secret-namespace: ${pvc.namespace}
  csi.storage.k8s.io/node-publish-secret-name: luks
  csi.storage.k8s.io/node-publish-secret-namespace: ${pvc.namespace}
```

2. Identificare le passphrase esistenti sul volume o sullo snapshot.

**Volume**

```
tridentctl -d get volume luks-pvc1
GET http://127.0.0.1:8000/trident/v1/volume/<volumeID>

...luksPassphraseNames:["A"]
```

**Snapshot**

```
tridentctl -d get snapshot luks-pvc1
GET http://127.0.0.1:8000/trident/v1/volume/<volumeID>/<snapshotID>

...luksPassphraseNames:["A"]
```

3. Aggiornare il segreto LUKS per il volume per specificare le passphrase nuove e precedenti. Assicurarsi `previous-luke-passphrase-name` e. `previous-luks-passphrase` associare la passphrase precedente.

```
apiVersion: v1
kind: Secret
metadata:
  name: luks-pvc1
stringData:
  luks-passphrase-name: B
  luks-passphrase: secretB
  previous-luks-passphrase-name: A
  previous-luks-passphrase: secretA
```

4. Creare un nuovo pod per il montaggio del volume. Questa operazione è necessaria per avviare la rotazione.

5. Verificare che la passphrase sia stata ruotata.

**Volume**

```
tridentctl -d get volume luks-pvc1
GET http://127.0.0.1:8000/trident/v1/volume/<volumeID>

...luksPassphraseNames:["B"]
```

**Snapshot**

```
tridentctl -d get snapshot luks-pvc1
GET http://127.0.0.1:8000/trident/v1/volume/<volumeID>/<snapshotID>

...luksPassphraseNames:["B"]
```

**Risultati**

La passphrase è stata ruotata quando viene restituita solo la nuova passphrase nel volume e nello snapshot.

> Se, ad esempio, vengono restituite due passphrase `luksPassphraseNames: ["B", "A"]`, la rotazione è incompleta. È possibile attivare un nuovo pod per tentare di completare la rotazione.

### Abilitare l'espansione dei volumi

È possibile attivare l'espansione del volume su un volume crittografato con LUKS.

**Fasi**

1. Attivare il `CSINodeExpandSecret` feature gate (beta 1.25+). Fare riferimento a. "Kubernetes 1.25: Utilizza Secrets per l'espansione basata su nodi di volumi CSI" per ulteriori informazioni.
2. Aggiungere il `node-expand-secret-name` e. `node-expand-secret-namespace` Parametri StorageClass. Ad esempio:

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: luks
provisioner: netapp.io/trident
parameters:
  selector: "luks=true"
  csi.storage.k8s.io/node-stage-secret-name: luks-${pvc.name}
  csi.storage.k8s.io/node-stage-secret-namespace: ${pvc.namespace}
  csi.storage.k8s.io/node-expand-secret-name: luks-${pvc.name}
  csi.storage.k8s.io/node-expand-secret-namespace: ${pvc.namespace}
allowVolumeExpansion: true
```

**Risultati**

Quando si avvia l’espansione dello storage online, il kubelet passa le credenziali appropriate al driver.

## 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.