



Introduzione allo storage e alla virtualizzazione

NetApp virtualization solutions

NetApp
July 21, 2026

Sommario

Introduzione allo storage e alla virtualizzazione	1
Scopri di più sull'integrazione di NetApp Trident con Red Hat OpenShift	1
NetApp Trident	1
Trident Protect	1
NetApp ONTAP modelli hardware e protocolli supportati	2
Driver Trident per lo storage ONTAP	2
Installare Trident su un cluster OpenShift e creare oggetti di storage	4
Passaggio 1: installa Trident	5
Passaggio 2: Preparare i file di backend dello storage e di configurazione di StorageClass per il proprio ambiente	9
Passaggio 3: Creare un file di configurazione VolumeSnapshotClass	29
Passaggio 4: applica i file di configurazione al tuo cluster	29
Passaggio 5: Imposta le classi di storage e snapshot predefinite di Trident	31
Requisiti per distribuire Red Hat OpenShift Virtualization con ONTAP	32
Prerequisiti	32
NetApp Trident CSI, Storage Profiles, modalità di accesso ai volumi per la virtualizzazione OpenShift	33
Crea una macchina virtuale utilizzando lo storage ONTAP con Red Hat OpenShift Virtualization	37
Crea VM	38
Dimostrazione video	41
Crea una macchina virtuale con Red Hat OpenShift Virtualization utilizzando lo storage ONTAP tramite metodi alternativi	41
Crea una macchina virtuale da un file YAML con Red Hat OpenShift Virtualization	41
Crea una VM da una copia snapshot con Red Hat OpenShift Virtualization	43
Clona una VM in OpenShift Virtualization usando Trident	47
Migrare una VM tra due nodi in un cluster Red Hat OpenShift	51
Migrazione live della VM	51
Migrazione dello storage delle VM tra classi di storage in OpenShift Virtualization	53

Introduzione allo storage e alla virtualizzazione

Scopri di più sull'integrazione di NetApp Trident con Red Hat OpenShift

Lo storage è fondamentale nelle applicazioni containerizzate, garantendo che i dati siano sempre disponibili in modo persistente e gestiti in modo efficiente in ambienti dinamici e spesso effimeri. Le moderne soluzioni di storage per container sono progettate per scalare senza problemi insieme alle applicazioni, offrendo la flessibilità di allocare dinamicamente risorse di storage in base alle esigenze dell'applicazione, supportando sia la crescita che carichi di lavoro variabili. Poiché la OpenShift Virtualization consente alle VM di essere eseguite come container, utilizzano lo stesso storage delle altre applicazioni nei OpenShift Clusters.

NetApp ONTAP è un software di gestione dei dati robusto e altamente scalabile che costituisce la spina dorsale delle soluzioni di storage enterprise di NetApp. Offre un'architettura di storage unificata che si integra perfettamente con diversi ambienti di storage, inclusi on-premises, cloud e configurazioni ibride. ONTAP supporta una varietà di protocolli come NFS, CIFS/SMB, iSCSI, FC e NVMe over TCP, consentendo un'integrazione fluida con diverse applicazioni e carichi di lavoro. Permette di consolidare SAN, NAS e storage a oggetti in un'unica piattaforma, semplificando la gestione e riducendo i costi. NetApp ONTAP è disponibile on-premises così come come servizio gestito nel cloud.

NetApp Trident

Trident è un orchestratore di storage open-source sviluppato da NetApp, progettato per facilitare l'utilizzo dello storage ONTAP in ambienti containerizzati. Consente il provisioning dinamico e la gestione dello storage persistente per applicazioni containerizzate.

Caratteristiche principali di Trident

Conformità CSI Garantisce la compatibilità con le più recenti funzionalità e standard di storage di Kubernetes. **Configurazione dichiarativa** Consente agli utenti di definire i requisiti di storage in file YAML, che vengono poi gestiti automaticamente da Trident. **Driver** Trident supporta i driver per i protocolli NFS, CIFS/SMB, iSCSI, FC e NVMe/TCP supportati da ONTAP. **Supporto ibrido e multi cloud** Trident supporta i driver per lo storage ONTAP on-premises e per i servizi di storage gestiti negli hyperscaler, ovvero FSx per NetApp ONTAP in AWS, Azure NetApp Files in Azure e Google Cloud NetApp Volumes in Google Cloud. **Gestione avanzata dei dati** Sfrutta le funzionalità di snapshot e clonazione di ONTAP per una protezione efficiente dei dati e un provisioning rapido degli ambienti di test e sviluppo.

Per tutti i dettagli su Trident, consulta NetApp Trident ["documentazione qui"](#)

Trident Protect

Trident Protect è una funzionalità avanzata di Trident che offre una protezione e una sicurezza dei dati superiori per gli ambienti containerizzati. È progettata per garantire che i dati critici siano protetti da potenziali minacce e scenari di perdita di dati. È disponibile gratuitamente per tutti gli utenti di Trident e può essere facilmente abilitata in qualsiasi OpenShift cluster con Trident installato.

Caratteristiche principali di Trident Protect

Backup automatici Trident Protect consente backup automatici basati su policy di applicazioni container e macchine virtuali, garantendo che vengano sottoposti a backup regolarmente senza intervento manuale.

Snapshot Management Sfrutta le funzionalità di snapshot di ONTAP per creare copie point-in-time frequenti di applicazioni container e VM, fornendo un meccanismo affidabile per il ripristino.

Crittografia dei dati Assicura che i dati siano crittografati sia a riposo che in transito, soddisfacendo rigorosi requisiti di sicurezza e proteggendo le informazioni sensibili.

Conformità e audit Fornisce strumenti e funzionalità che aiutano le organizzazioni a soddisfare i requisiti di conformità e a condurre audit periodici delle proprie pratiche di protezione dei dati.

Ripristino di emergenza Si integra con le funzionalità di replica di ONTAP per fornire soluzioni di ripristino di emergenza robuste, garantendo la continuità operativa anche in caso di eventi catastrofici.

Per informazioni complete su Trident Protect, consulta NetApp Trident Protect ["documentazione qui"](#).

NetApp ONTAP modelli hardware e protocolli supportati

NetApp ONTAP software è eseguibile su una varietà di modelli hardware, ciascuno progettato per soddisfare diverse esigenze di prestazioni e capacità. Trident estende le sue robuste funzionalità per supportare diversi sistemi NetApp ONTAP, inclusi All Flash FAS (AFF), All SAN Array (ASA) e ASA R2. Questo supporto garantisce che le organizzazioni possano sfruttare appieno il potenziale delle loro soluzioni di storage dalle performance elevate in ambienti containerizzati.

All Flash FAS (AFF) Systems I sistemi AFF offrono prestazioni eccezionali e bassa latenza, rendendoli ideali per applicazioni ad alta richiesta.

All SAN Array (ASA) Systems I sistemi ASA sono progettati per ambienti SAN dedicati, offrendo prestazioni e affidabilità elevate.

ASA R2 Systems I sistemi ASA R2 offrono funzionalità e capacità avanzate per gli ambienti SAN.

AFX NetApp AFX è un'architettura di storage all-flash disaggregata costruita ad hoc, progettata per i carichi di lavoro AI enterprise. Offre NAS dalle performance elevate, consentendo la scalabilità indipendente di calcolo e capacità. I sistemi di storage NetApp AFX si differenziano dagli altri sistemi basati su ONTAP (ASA, AFF e FAS) per l'implementazione del loro storage layer. AFX utilizza un file system distribuito che consente prestazioni elevate e scalabilità, mentre gli altri sistemi basati su ONTAP utilizzano un'architettura di storage tradizionale.

1. Protocolli supportati per AFF: NFS, CIFS/SMB, iSCSI, FC, NVMe/TCP
2. Protocolli supportati per ASA: iSCSI, FC, NVMe/TCP
3. Protocolli supportati per ASA R2: iSCSI, FC, NVMe/TCP
4. Protocolli supportati per AFX: a partire da Trident 25.10, è supportato solo il protocollo NFS; il protocollo SMB non è supportato.

Driver Trident per lo storage ONTAP

Trident include i seguenti driver CSI, ognuno dei quali può effettuare il provisioning di un volume nello storage di backend

Per i protocolli NAS sui modelli hardware supportati

1. ontap-nas: Un pvc mappa su 1 pv che è un flexvolume dedicato
2. ontap-nas-economy: Ogni PV allocato è un qtree, con un numero configurabile di qtree per FlexVolume (il valore predefinito è 200, configurabile tra 50 e 300). Un pvc corrisponde a 1 pv che è un qtree su un flexvolume condiviso



Puoi posizionare tutti i dischi delle VM come qtree in un singolo volume. Tuttavia, tieni presente che le funzionalità di Snapshot, Cloni e Replica non sono supportate da Trident. Quando queste funzionalità sono gestite dall'amministratore dello storage, non sono granulari a livello di PV.

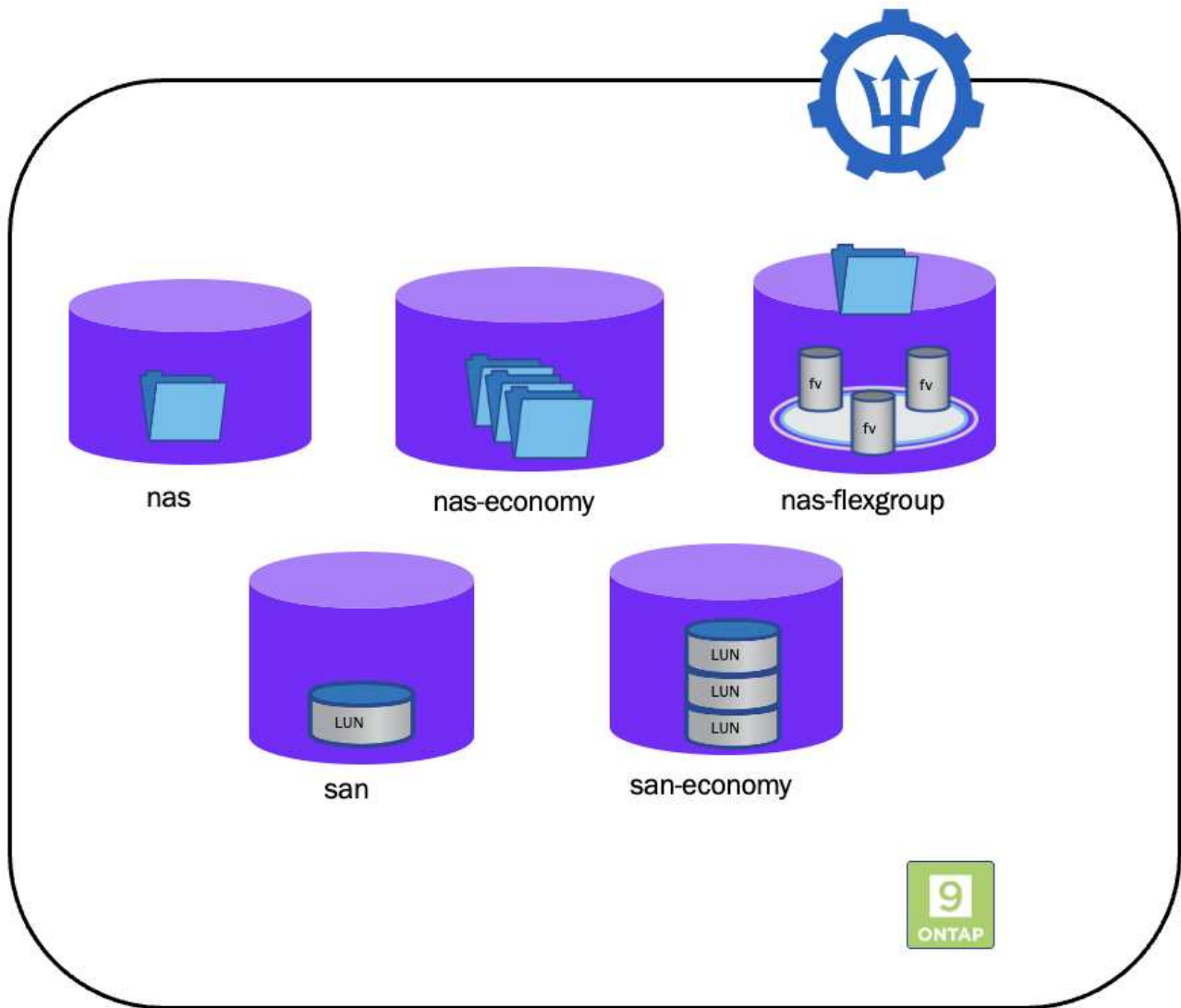
1. ontap-nas-flexgroup: Ogni PV di cui è stato effettuato il provisioning è un ONTAP FlexGroup, e vengono utilizzati tutti gli aggregati assegnati a un SVM.

Per i protocolli SAN sui modelli hardware supportati

1. ontap-san: Un pvc mappa su 1 pv che è un LUN su un flexvolume dedicato
2. ontap-san-economy: Un PVC corrisponde a 1 PV, che è una LUN su un flexvolume condiviso. Puoi avere più LUN nel flexvolume condiviso (il valore predefinito è 100, configurabile tra 50 e 200 LUN/PV per volume flexvol).



Puoi posizionare tutti i dischi delle VM come LUN in un singolo volume. Tuttavia, questo driver, pur supportando Snapshot e Cloni, non supporta la Replication. Quando la Replication è gestita dall'amministratore dello storage, non è granulare a livello di PV.



Per l'elenco completo delle funzionalità esposte tramite i driver Trident e di quelle che devono essere applicate dall'amministratore dello storage, consulta la sezione "[Driver backend ONTAP](#)" nella documentazione di Trident.

Installare Trident su un cluster OpenShift e creare oggetti di storage

Installa Trident con il Red Hat Certified Trident Operator e crea oggetti di storage per ONTAP e Amazon FSx for NetApp ONTAP per abilitare il provisioning dinamico dei volumi per container e VM. Prepara i nodi worker per l'accesso a blocchi quando richiesto.

Prima di iniziare

- Completa le procedure in questa pagina prima di installare OpenShift Virtualization. OpenShift Virtualization richiede un StorageClass predefinito basato su Trident e un VolumeSnapshotClass per creare immagini golden per i template delle VM.
- Se hai già installato OpenShift Virtualization prima di configurare Trident, elimina tutte le golden images

create con una classe di storage diversa. Dopo aver impostato Trident come predefinito, OpenShift Virtualization ricrea le golden images utilizzando lo storage Trident.

```
oc delete dv,VolumeSnapshot -n openshift-virtualization-os-images  
--selector=cdi.kubevirt.io/dataImportCron
```

Passaggio 1: installa Trident

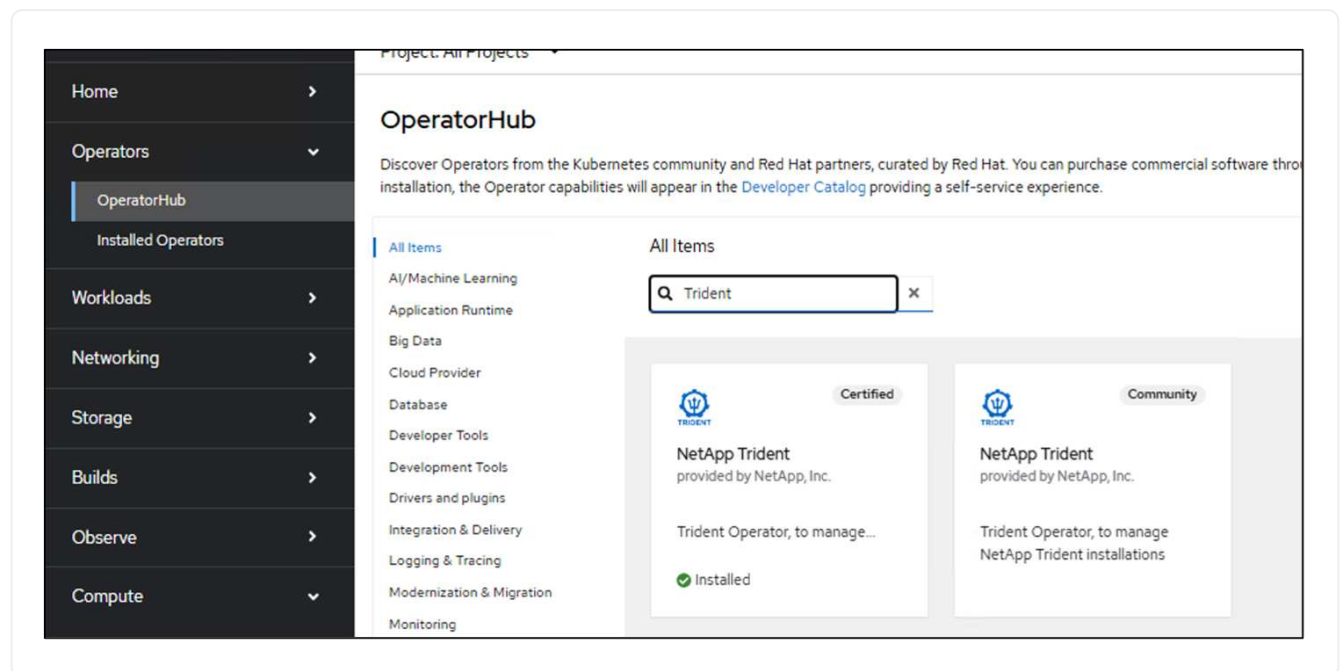
L'operatore Trident certificato da Red Hat è supportato da NetApp per OpenShift on-premises, nei cloud pubblici e nei servizi gestiti come ROSA. A partire da Trident 25.02, l'operatore può anche preparare i nodi worker per iSCSI quando si utilizza Amazon FSx for NetApp ONTAP e si prevede di eseguire carichi di lavoro VM di OpenShift Virtualization.

Per altre opzioni di installazione, vedere ["la documentazione di Trident"](#).

Passi

1. In **OperatorHub**, seleziona **Certified NetApp Trident**.

Mostra esempio



2. Nella pagina **Install**, mantieni la versione più recente e seleziona **Install**.

Mostra esempio

[OperatorHub](#) > Operator Installation

Install Operator

Install your Operator by subscribing to one of the update channels to keep the Operator up to date. The strategy determines either manual or automatic updates.

Update channel * ⓘ
stable

Version *
25.2.1
25.2.1
25.2.0

Operator will be available in all namespaces.

☐ A specific namespace on the cluster
This mode is not supported by this Operator

Installed Namespace *
PR openshift-operators

Update approval * ⓘ
☒ Automatic
☐ Manual

Install

Cancel

3. Dopo l'installazione dell'operatore, selezionare **View operator** e creare un'istanza di Trident Orchestrator.

Se vuoi preparare i nodi worker per iSCSI, passa alla visualizzazione YAML e aggiungi `iscsi` a `nodePrep`.

Mostra esempio

Create TridentOrchestrator

Create by completing the form. Default values may be provided by the Operator authors.

Configure via: ☐ Form view ☒ YAML view

```
1 kind: TridentOrchestrator
2 apiVersion: trident.netapp.io/v1
3 metadata:
4   name: trident
5 spec:
6   IPv6: false
7   debug: true
8   nodePrep:
9     - iscsi
10  imagePullSecrets: []
11  imageRegistry: ''
12  namespace: trident
13  silenceAutosupport: false
14
```

4. Verificare che tutti i pod di Trident siano in esecuzione nel cluster.

Mostra esempio

```
[root@localhost ~]# oc get pods -n trident
NAME                                READY   STATUS    RESTARTS   AGE
trident-controller-84cb9bff89-lkx6k 6/6     Running   0           16h
trident-node-linux-d88b9             2/2     Running   0           16h
trident-node-linux-ld4b8             2/2     Running   0           16h
trident-node-linux-mj5r8             2/2     Running   0           16h
trident-node-linux-mkmmmp            2/2     Running   0           16h
trident-node-linux-qhgr7             2/2     Running   0           16h
trident-node-linux-vt9tp             2/2     Running   0           16h
[root@localhost ~]#
```

5. Se hai abilitato la preparazione del nodo iSCSI, accedi ai nodi worker e verifica che `iscsid` e `multipathd` siano attivi e che `multipath.conf` abbia delle voci.

Mostra esempio

```
sh-5.1# systemctl status iscsid
● iscsid.service - Open-iSCSI
   Loaded: loaded (/usr/lib/systemd/system/iscsid.service; enabled; preset: disabled)
   Active: active (running) since Fri 2025-04-25 00:23:49 UTC; 3 days ago
 TriggeredBy: ● iscsid.socket
    Docs: man:iscsid(8)
          man:iscsiuio(8)
          man:iscsiadm(8)
   Main PID: 74787 (iscsid)
   Status: "Ready to process requests"
   Tasks: 1 (limit: 410912)
  Memory: 1.8M
    CPU: 6ms
   CGroup: /system.slice/iscsid.service
           └─74787 /usr/sbin/iscsid -f

Apr 25 00:23:49 ocp11-worker1 systemd[1]: Starting Open-iSCSI...
Apr 25 00:23:49 ocp11-worker1 systemd[1]: Started Open-iSCSI.
sh-5.1#
```

Mostra esempio

```
sh-5.1# systemctl status multipathd
● multipathd.service - Device-Mapper Multipath Device Controller
   Loaded: loaded (/usr/lib/systemd/system/multipathd.service; enabled; preset: enabled)
   Active: active (running) since Fri 2025-04-25 00:23:50 UTC; 3 days ago
 TriggeredBy: ● multipathd.socket
   Process: 74905 ExecStartPre=/sbin/modprobe -a scsi_dh_alua scsi_dh_emc scsi_dh_rdac dm-multipath (code=exited, status=0/SUCCESS)
   Process: 74906 ExecStartPre=/sbin/multipath -A (code=exited, status=0/SUCCESS)
   Main PID: 74907 (multipathd)
   Status: "up"
   Tasks: 7
  Memory: 18.3M
    CPU: 23.008s
   CGroup: /system.slice/multipathd.service
           └─74907 /sbin/multipathd -d -s

Apr 25 00:23:50 ocp11-worker1 systemd[1]: Starting Device-Mapper Multipath Device Controller...
Apr 25 00:23:50 ocp11-worker1 multipathd[74907]: -----start up-----
Apr 25 00:23:50 ocp11-worker1 multipathd[74907]: read /etc/multipath.conf
Apr 25 00:23:50 ocp11-worker1 multipathd[74907]: path checkers start up
Apr 25 00:23:50 ocp11-worker1 systemd[1]: Started Device-Mapper Multipath Device Controller.
sh-5.1#
```

```
sh-5.1# cat /etc/multipath.conf
defaults {
    find_multipaths no
}
blacklist {
    device {
        product .*
        vendor  .*
    }
}
blacklist_exceptions {
    device {
        product LUN
        vendor  NETAPP
    }
}
sh-5.1# █
```

Dimostrazione video

Il seguente video mostra una dimostrazione dell'installazione di Trident utilizzando il Red Hat Certified Trident Operator.

[Installazione di Trident 25.02.1 utilizzando l'operatore Trident certificato in OpenShift](#)

Passaggio 2: Preparare i file di backend dello storage e di configurazione di StorageClass per il proprio ambiente

Crea le definizioni di TridentBackendConfig e StorageClass per il tuo ambiente. Puoi configurare più protocolli di storage all'interno del tuo ambiente. Crea file YAML per ogni protocollo che desideri utilizzare e sostituisci i valori segnaposto con i dettagli specifici della tua configurazione.



Completa la sezione relativa all'ambiente locale o ROSA riportata di seguito in base al tuo ambiente, quindi procedi al passaggio 3.

Cluster OpenShift locali

Crea file YAML per ogni protocollo che desideri configurare. Puoi configurare uno o più dei seguenti protocolli: NAS per file storage basato su NFS, iSCSI per storage a blocchi iSCSI, NVMe/TCP per storage a blocchi NVMe over TCP dalle performance elevate o FC per storage a blocchi Fibre Channel.

Crea un TridentBackendConfig e StorageClass per ogni protocollo. La configurazione del backend include le credenziali memorizzate in un Secret di Kubernetes.

Linee guida essenziali per le opzioni di base della configurazione del backend

1. I file di configurazione del backend fanno riferimento al tuo **ONTAP SVM** e al **management LIF**. +
2. Puoi usare **nome utente e password** per autenticarti con ONTAP. Può trattarsi del nome utente e della password dell'amministratore del cluster o dell'amministratore del vserver. +
3. Puoi anche usare un **certificato client** per l'autenticazione con ONTAP. Per dettagli su come generare il certificato client, la chiave privata e il certificato CA, consulta "[Documentazione di Trident qui](#)".

Linee guida essenziali per le opzioni di configurazione del backend per il driver NAS economy

Per la configurazione del **driver NAS economy** è possibile includere i seguenti parametri:

1. **qtreesPerFlexvol** Il valore deve essere compreso tra 50 e 300, il valore predefinito è 200. +
2. **limitVolumePoolSize** Il valore è la dimensione massima di FlexVol che richiedi quando usi i qtree nel backend ontap-nas-economy. Non è impostato alcun valore predefinito e questo parametro non viene applicato per impostazione predefinita. Se imposti questo parametro, l'operatore Trident crea un nuovo FlexVol quando quello corrente FlexVol raggiunge la dimensione specificata. Questo parametro può aiutarti a controllare l'utilizzo della capacità nel tuo cluster limitando la dimensione dei FlexVols usati per il provisioning dei PV. +
3. **denyNewVolumePools** Questo parametro impedisce ai backend ontap-nas-economy di creare nuovi FlexVol volumi per contenere i loro qtree. Solo i FlexVol preesistenti vengono utilizzati per il provisioning di nuovi PV.

Linee guida essenziali per le opzioni di configurazione del backend per il driver SAN

Per la configurazione del **driver SAN** puoi includere i seguenti parametri:

1. **limitVolumeSize** Il valore è la dimensione massima del volume che richiedi quando usi il backend SAN. Non è impostato alcun valore predefinito e questo parametro non viene applicato per impostazione predefinita. Se imposti questo parametro, l'operatore Trident non esegue il provisioning se la dimensione del volume richiesta supera questo valore. Puoi usare questo parametro per limitare la dimensione massima dei volumi che gestisce per le LUN. +
2. **lunsPerFlexvol** Numero massimo di LUN per FlexVol, deve essere compreso nell'intervallo [50, 200], il valore predefinito è 100. Se imposti questo parametro, l'operatore Trident crea un nuovo FlexVol quando quello corrente FlexVol raggiunge il numero di LUN specificato. Questo parametro può aiutarti a controllare l'utilizzo della capacità nel tuo cluster limitando il numero di LUN per FlexVol utilizzati per il provisioning dei PV.

Linee guida essenziali per le opzioni di configurazione del backend per il driver SAN economy

Per la configurazione del **SAN economy driver** è possibile includere i seguenti parametri:

1. **limitVolumePoolSize** Il valore è la dimensione massima di FlexVol che richiedi quando usi il backend SAN economy. Non è impostato alcun valore predefinito e questo parametro non viene applicato per impostazione predefinita. Se imposti questo parametro, l'operatore Trident limita la dimensione massima del flexvol che può essere creata per la creazione di LUN nel flexvol. Questo parametro può aiutarti a controllare l'utilizzo della capacità nel tuo cluster limitando la dimensione delle FlexVols usate per il provisioning dei PV. +
2. **denyNewVolumePools** Questo parametro impedisce ai backend SAN economy di creare nuovi FlexVol volumi per contenere i propri LUN. Solo i FlexVol preesistenti vengono utilizzati per il provisioning di nuovi PV.



Nei file yaml di esempio forniti per ciascun protocollo, se storagePrefix non viene specificato, viene utilizzato il prefisso predefinito "trident". Quando usi ontap-nas-economy e un storagePrefix di 24 o più caratteri, i qtree non avranno il prefisso di storage incorporato, anche se sarà presente nel nome del volume.

Quando crei un volume in ONTAP direttamente da System Manager o dalla CLI, senza fornire valori per tutti i parametri opzionali, alcuni valori predefiniti vengono impostati automaticamente durante il provisioning di un volume come mostrato di seguito.

```
NSOL-NetApp-A70-T19U05::> volume create -vserver oc-test-automation -volume test_ontap_default -aggregate NSOL_NetApp_A70_T19U05a_NVME_SSD_1 -size 10GB
[Job 22836] Job succeeded: Successful
```

```
NSOL-NetApp-A70-T19U05::> volume show -vserver oc-test-automation -volume test_ontap_default -fields snapshot-policy,qos-adaptive-policy-group,snapshot-reserve-available,tiering-policy,unix-permissions,snapdir-access,space-guarantee
vserver      volume      unix-permissions  space-guarantee  snapdir-access  snapshot-policy  qos-adaptive-policy-group  snapshot-reserve-available  tiering-policy
oc-test-automation test_ontap_default ---rwxr-xr-x      none             true             default          -                      511.8MB                  none
```

Ecco un esempio di volume creato utilizzando la configurazione backend di Trident. Non presenta opzioni impostate esplicitamente nella sezione delle impostazioni predefinite, ad eccezione del modello di nome.

```
oc describe tbc -n trident tbc-nas
### output truncated for brevity
Spec:
  Backend Name:  tbc-nas
  Credentials:
    Name:  tbc-nas-secret
  Defaults: # There are no defaults set except for the volume name
template.
    Name Template:      {{ .config.StoragePrefix }}_{{
.volume.Namespace }}_{{ .volume.RequestName }}
  Management LIF:      10.192.102.50
  Storage Driver Name:  ontap-nas
  Storage Prefix:      testautomation
  Svm:                  oc-test-automation
```

Quando il volume viene creato utilizzando questa configurazione di backend, le opzioni che non sono impostate nel file di configurazione del backend vengono impostate sui valori predefiniti come mostrato di seguito.

```
NSOL-NetApp-A70-T19U05::> volume show -vserver oc-test-automation -volume testautomation_default_test_trident_defaults_91653 -fields snapshot-policy,qos-adaptive-policy-group,snapshot-reserve-available,tiering-policy,unix-permissions,snapdir-access,space-guarantee
vserver      volume      unix-permissions  space-guarantee  snapdir-access  snapshot-policy  qos-adaptive-policy-group  snapshot-reserve-available  tiering-policy
oc-test-automation testautomation_default_test_trident_defaults_91653 ---rwxrwxrwx      none             false            none                  -                      0B                      none
```

Tuttavia, quando imposti i valori nella sezione predefinita della configurazione del backend di Trident, puoi creare volumi che possono avere valori specifici per quei parametri. Nell'esempio seguente, snapshotPolicy è impostato sulla policy predefinita (è none se non imposti nulla esplicitamente) e snapshotDir è impostato su true nella sezione predefinita del file di configurazione yaml del backend. Con queste impostazioni, puoi vedere che il volume del backend ottiene quei valori. (snapdir-access è true e

snapshot-policy è default)

```
[root@00-50-56-b5-01-5b pvcs-br]# oc describe tbc -n trident tbc-nas
Name:          tbc-nas
Namespace:     trident
Labels:        <none>
Annotations:   <none>
API Version:   trident.netapp.io/v1
Kind:          TridentBackendConfig
Metadata:
  Creation Timestamp:  2026-03-30T20:06:59Z
  Finalizers:
    trident.netapp.io
  Generation:          3
  Resource Version:     81358056
  UID:                 4dc24339-ff66-4345-acc1-37ec73d14140
Spec:
  Backend Name:  tbc-nas
  Credentials:
    Name:  tbc-nas-secret
  Defaults:
    Name Template:  {{ .config.StoragePrefix }}_{{ .volume.Namespace }}_{{ .volume.RequestName }}
    Snapshot Dir:   true
    Snapshot Policy: default
    Management LIF: 10.192.102.50
    Storage Driver Name: ontap-nas
    Storage Prefix:  testautomation
    SVM:             oc-test-automation
    Version:          1
Status:
  Backend Info:
    Backend Name:      tbc-nas
    Backend UUID:      210cc722-b394-489d-8e9e-49018d3e8710
    Deletion Policy:   delete
    Last Operation Status: Success
    Message:           Backend 'tbc-nas' updated
    Phase:             Bound
```

```
NSOL-NetApp-A70-T19U05:> volume show -vserver oc-test-automation -volume testautomation_default_test_trident_set_defaults_7567a -fields snapshot-policy,qos-adaptive-policy-group,snapshot-reserve-available,tiering-policy,unix-permissions,snapdir-access,space-guarantee
vserver      volume                                     unix-permissions space-guarantee snapdir-access snapshot-policy qos-adaptive-policy-group snapshot-reserve-availab
le tiering-policy
-----
oc-test-automation testautomation_default_test_trident_set_defaults_7567a ---RWXIRWX none true default - 269.5MB
none
```

```
NSOL-NetApp-A70-T19U05:> volume show -vserver oc-test-automation -volume testautomation_default_test_trident_set_defaults_7567a -fields snapshot-policy,snapdir-access
vserver      volume                                     snapdir-access snapshot-policy
oc-test-automation testautomation_default_test_trident_set_defaults_7567a true default
```

Per tutti gli altri parametri aggiuntivi, consulta la documentazione di Trident:

1. ["Opzioni di configurazione del backend NAS di Trident"](#) +
2. ["Opzioni di configurazione del backend SAN di Trident"](#)

NAS

Crea un `TridentBackendConfig` e un `StorageClass` per ONTAP NAS per abilitare il provisioning di storage persistente basato su NFS. La configurazione del backend include le credenziali memorizzate in un `Secret` di Kubernetes e fa riferimento al tuo ONTAP SVM e al LIF di gestione.

Esempio di segreto del backend e file di configurazione con metodo di autenticazione tramite nome utente e password (salva come `tbc-nas.yaml`):

```
# tbc-nas.yaml
apiVersion: v1
kind: Secret
metadata:
  name: tbc-nas-secret
type: Opaque
stringData:
  username: <cluster admin username>
  password: <cluster admin password>
---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: tbc-nas
spec:
  version: 1
  storageDriverName: ontap-nas
  managementLIF: <ONTAP management LIF>
  backendName: tbc-nas
  svm: zoneb #<replace with your SVM name>
  storagePrefix: testzoneb #<replace with your prefix>
  defaults:
    nameTemplate: "{{ .config.StoragePrefix }}_{{ .volume.Namespace }}"
    credentials:
      name: tbc-nas-secret
```

Esempio di segreto del backend e file di configurazione che utilizza l'autenticazione tramite certificato client (salva come `tbc-nas.yaml`):

```

# tbc-nas.yaml
apiVersion: v1
kind: Secret
metadata:
  name: ontap-nas-secret
  namespace: trident
type: Opaque
stringData:
  clientPrivateKey: <client private>
---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: tbc-nas
  namespace: trident
spec:
  version: 1
  storageDriverName: ontap-nas
  managementLIF: <ONTAP management LIF>
  backendName: tbc-nas
  svm: zoneb #<replace with your SVM name>
  storagePrefix: testzoneb #<replace with your prefix>
  credentials:
    name: ontap-nas-secret
    clientCertificate: <client certificate>

```

StorageClass definition (salva come `sc-nas.yaml`):

```

# sc-nas.yaml
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: sc-nas
provisioner: csi.trident.netapp.io
parameters:
  backendType: "ontap-nas"
  media: "ssd"
  provisioningType: "thin"
  snapshots: "true"
allowVolumeExpansion: true

```

NAS economy

Crea un `TridentBackendConfig` e un `StorageClass` per il driver ONTAP NAS economy per abilitare il provisioning di storage persistente basato su NFS utilizzando i qtree. La configurazione del backend

include le credenziali memorizzate in un Secret di Kubernetes e fa riferimento al tuo ONTAP SVM e al LIF di gestione.

Esempio di segreto del backend e file di configurazione con metodo di autenticazione tramite nome utente e password (salva come `tbc-nas-economy.yaml`):

```
# tbc-nas-economy.yaml
apiVersion: v1
kind: Secret
metadata:
  name: tbc-nas-economy-secret
type: Opaque
stringData:
  username: <cluster admin username>
  password: <cluster admin password>
---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: tbc-nas-economy
spec:
  version: 1
  storageDriverName: ontap-nas-economy
  managementLIF: <ONTAP management LIF>
  backendName: tbc-nas-eco
  svm: <replace with your SVM name>
  storagePrefix: <replace with your prefix>
  defaults:
    nameTemplate: "{{ .config.StoragePrefix }}_{{ .volume.Namespace
  }}_{{ .volume.RequestName }}"
  credentials:
    name: tbc-nas-economy-secret
```

StorageClass definition (salva come `sc-nas-economy.yaml`):

```
# sc-nas-economy.yaml
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: sc-nas-economy
provisioner: csi.trident.netapp.io
parameters:
  backendType: "ontap-nas-economy"
  media: "ssd"
  provisioningType: "thin"
  snapshots: "true"
allowVolumeExpansion: true
```



Consulta la sezione "Linee guida essenziali per le opzioni di configurazione del backend per il driver NAS economy" sopra per i dettagli sui parametri aggiuntivi che puoi includere nel file di configurazione del backend per il driver ontap-nas-economy.

iSCSI SAN

Crea un TridentBackendConfig e un StorageClass per ONTAP SAN con iSCSI per abilitare il provisioning dello storage a blocchi basato su iSCSI. La configurazione del backend utilizza il driver ontap-san e include le credenziali memorizzate in un Secret di Kubernetes. Il parametro sanType utilizza per impostazione predefinita il protocollo iSCSI se omissso.

Segreto del backend e file di configurazione del backend che utilizzano il metodo di autenticazione tramite nome utente e password (salva come `tbc-iscsi.yaml`):

```

# tbc-iscsi.yaml
apiVersion: v1
kind: Secret
metadata:
  name: backend-tbc-ontap-iscsi-secret
type: Opaque
stringData:
  username: <cluster admin username>
  password: <cluster admin password>
---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: ontap-iscsi
spec:
  version: 1
  storageDriverName: ontap-san
  managementLIF: <management LIF>
  backendName: ontap-iscsi
  svm: <SVM name>
  credentials:
    name: backend-tbc-ontap-iscsi-secret

```

Segreto del backend e file di configurazione del backend che utilizzano il metodo di autenticazione tramite certificato client (salva come `tbc-iscsi.yaml`):

```
# tbc-iscsi.yaml
apiVersion: v1
kind: Secret
metadata:
  name: backend-tbc-ontap-iscsi-secret
type: Opaque
stringData:
  clientPrivateKey: <client private key>
---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: ontap-iscsi
spec:
  version: 1
  storageDriverName: ontap-san
  managementLIF: <management LIF>
  backendName: ontap-iscsi
  svm: <SVM name>
  credentials:
    name: backend-tbc-ontap-iscsi-secret
    clientCertificate: <client certificate>
```

StorageClass definition (salva come sc-iscsi.yaml):

```
# sc-iscsi.yaml
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: sc-iscsi
provisioner: csi.trident.netapp.io
parameters:
  backendType: "ontap-san"
  media: "ssd"
  provisioningType: "thin"
  fsType: ext4
  snapshots: "true"
allowVolumeExpansion: true
```

NVMe/TCP SAN

Crea una TridentBackendConfig e una StorageClass per ONTAP SAN con NVMe su TCP per abilitare il provisioning di storage a blocchi dalle performance elevate. La configurazione del backend utilizza il driver ontap-san ottimizzato per il trasporto NVMe/TCP e include le credenziali memorizzate in un Secret di Kubernetes.

Segreto del backend e file di configurazione del backend che utilizzano il metodo di autenticazione tramite nome utente e password (salva come `tbc-nvme.yaml`):

```
# tbc-nvme.yaml
apiVersion: v1
kind: Secret
metadata:
  name: backend-tbc-ontap-nvme-secret
type: Opaque
stringData:
  username: <cluster admin username>
  password: <cluster admin password>
---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-tbc-ontap-nvme
spec:
  version: 1
  storageDriverName: ontap-san
  sanType: nvme
  managementLIF: <ONTAP management LIF>
  backendName: backend-tbc-ontap-nvme
  svm: <SVM name>
  credentials:
    name: backend-tbc-ontap-nvme-secret
```

Segreto del backend e file di configurazione del backend che utilizzano il metodo di autenticazione tramite certificato client (salva come `tbc-nvme.yaml`):

```

# tbc-nvme.yaml
apiVersion: v1
kind: Secret
metadata:
  name: backend-tbc-ontap-nvme-secret
type: Opaque
stringData:
  clientPrivateKey: <client private key>
---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-tbc-ontap-nvme
spec:
  version: 1
  storageDriverName: ontap-san
  sanType: nvme
  managementLIF: <ONTAP management LIF>
  backendName: backend-tbc-ontap-nvme
  svm: <SVM name>
  credentials:
    name: backend-tbc-ontap-nvme-secret
    clientCertificate: <client certificate>

```

StorageClass definition (salva come `sc-nvme.yaml`):

```

# sc-nvme.yaml
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: sc-nvme
provisioner: csi.trident.netapp.io
parameters:
  backendType: "ontap-san"
  media: "ssd"
  provisioningType: "thin"
  fsType: ext4
  snapshots: "true"
allowVolumeExpansion: true

```



Se prevedi di usare i volumi ReadWriteMany (RWX) con il protocollo NVMe, un singolo volume RWX NVMe non può essere montato da più di 64 nodi in un cluster Kubernetes. Per maggiori informazioni su questa limitazione e sui dettagli del modello Super-subsystem introdotto in Trident 26.02 e del modello Per-volume subsystem precedente a Trident 26.02, consulta la documentazione di Trident: <https://docs.netapp.com/us-en/trident/trident-use/ontap-san-prep.html#nvme-tcp-considerations>.

FC SAN

Crea una `TridentBackendConfig` e una `StorageClass` per ONTAP SAN con Fibre Channel per abilitare il provisioning dello storage a blocchi basato su FC. La configurazione del backend utilizza il driver `ontap-san` con il protocollo FCP specificato e include le credenziali memorizzate in un `Secret` di Kubernetes.

Usa il metodo di autenticazione `username/password` o il metodo di autenticazione con certificato client per autenticarti con ONTAP. Per dettagli su come generare il certificato client, la chiave privata e il certificato CA, consulta "[Documentazione di Trident qui](#)". Qui sotto trovi esempi per entrambi i metodi.

Mostra esempio

Segreto del backend e file di configurazione del backend che utilizzano il metodo di autenticazione ONTAP con nome utente e password (salva come `tbc-fc.yaml`):

```
# tbc-fc.yaml
apiVersion: v1
kind: Secret
metadata:
  name: tbc-fc-secret
type: Opaque
stringData:
  username: <cluster admin username>
  password: <cluster admin password>
---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: tbc-fc
spec:
  version: 1
  storageDriverName: ontap-san
  managementLIF: <ONTAP management LIF>
  backendName: tbc-fc
  svm: openshift-fc #<replace with your SVM name>
  sanType: fcp
  storagePrefix: demofc #<replace with your prefix>
  defaults:
    nameTemplate: "{{ .config.StoragePrefix }}_{{ .volume.Namespace
  }}"
    credentials:
      name: tbc-fc-secret
```

Segreto del backend e file di configurazione del backend che utilizzano il metodo di autenticazione tramite certificato client (salva come `tbc-fc.yaml`):

```
# tbc-fc.yaml
apiVersion: v1
kind: Secret
metadata:
  name: tbc-fc-secret
type: Opaque
stringData:
  clientPrivateKey: <client private key>
---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: tbc-fc
spec:
  version: 1
  storageDriverName: ontap-san
  managementLIF: <ONTAP management LIF>
  backendName: tbc-fc
  svm: openshift-fc #<replace with your SVM name>
  sanType: fcp
  storagePrefix: demofc #<replace with your prefix>
  defaults:
    nameTemplate: "{{ .config.StoragePrefix }}_{{ .volume.Namespace
  }}"_{{ .volume.RequestName }}"
  credentials:
    name: tbc-fc-secret
    clientCertificate: <client certificate>
```

```
# sc-fc.yaml
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: sc-fc
provisioner: csi.trident.netapp.io
parameters:
  backendType: "ontap-san"
  media: "ssd"
  provisioningType: "thin"
  fsType: ext4
  snapshots: "true"
allowVolumeExpansion: true
```

Crea un `TridentBackendConfig` e un `StorageClass` per ONTAP SAN economy con iSCSI per abilitare il provisioning dello storage a blocchi basato su iSCSI. La configurazione del backend utilizza il driver `ontap-san-economy` con il protocollo iSCSI e include le credenziali memorizzate in un `Secret` di Kubernetes.

Segreto del backend e file di configurazione del backend che utilizzano il metodo di autenticazione tramite certificato client (salva come `tbc-san-eco.yaml`):

```
# tbc-san-eco.yaml
---
apiVersion: v1
kind: Secret
metadata:
  name: ontap-san-cert-auth
  namespace: trident
type: Opaque
stringData:
  clientPrivateKey: <client private key>
---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: tbc-san-eco-cert
  namespace: trident
spec:
  version: 1
  storageDriverName: ontap-san-economy
  managementLIF: "<ONTAP management LIF>"
  backendName: tbc_san_eco
  svm: <replace with your SVM name>
  storagePrefix: <replace with your prefix>
  credentials:
    name: ontap-san-cert-auth
    clientCertificate: <client certificate>
```

```
# sc-san-eco.yaml
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: sc-san-eco
provisioner: csi.trident.netapp.io
parameters:
  backendType: "ontap-san-economy"
  provisioningType: "thin"
  snapshots: "true"
```

Consulta la documentazione di Trident per ulteriori esempi di file yaml e informazioni sulle modalità di accesso e sulle modalità di volume supportate dai driver SAN e NAS.

1. ["Esempi che utilizzano driver NAS"](#)
2. ["Esempi che utilizzano driver SAN"](#)

Cluster ROSA con Amazon FSx for NetApp ONTAP

Crea file YAML per ogni protocollo che desideri configurare. Puoi configurare uno o entrambi i seguenti protocolli: NAS per file storage basato su NFS o iSCSI per storage a blocchi.

NAS

Crea un `TridentBackendConfig` e un `StorageClass` per Amazon FSx for NetApp ONTAP con ONTAP NAS per abilitare il provisioning di storage persistente basato su NFS sui cluster ROSA. La configurazione del backend utilizza i nomi DNS di Amazon FSx for NetApp ONTAP per le LIF di gestione e dati, e include le credenziali memorizzate in un `Secret` di Kubernetes nel namespace `trident`.

Segreto del backend e file di configurazione del backend (salva come `tbc-fsx-nas.yaml`):

```
# tbc-fsx-nas.yaml
apiVersion: v1
kind: Secret
metadata:
  name: backend-fsx-ontap-nas-secret
type: Opaque
stringData:
  username: <FSx for ONTAP, for example fsxadmin>
  password: <FSx for ONTAP password>
---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-fsx-ontap-nas
spec:
  version: 1
  backendName: fsx-ontap
  storageDriverName: ontap-nas
  managementLIF: <Management DNS name>
  dataLIF: <NFS DNS name>
  svm: <SVM NAME>
  credentials:
    name: backend-fsx-ontap-nas-secret
```

`StorageClass` definition (salva come `sc-fsx-nas.yaml`):

```
# sc-fsx-nas.yaml (storage class name is trident-csi)
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: trident-csi
provisioner: csi.trident.netapp.io
parameters:
  backendType: "ontap-nas"
  fsType: "ext4"
allowVolumeExpansion: true
reclaimPolicy: Retain
```

iSCSI

Crea una TridentBackendConfig e una StorageClass per Amazon FSx for NetApp ONTAP con ONTAP SAN per abilitare il provisioning dello storage a blocchi basato su iSCSI sui cluster ROSA. La configurazione del backend utilizza il driver ontap-san e include le credenziali memorizzate in un Secret di Kubernetes. Assicurati che i nodi worker siano predisposti per l'accesso iSCSI.

Segreto del backend e file di configurazione del backend (salva come `tbc-fsx-iscsi.yaml`):

```
# tbc-fsx-iscsi.yaml
apiVersion: v1
kind: Secret
metadata:
  name: backend-tbc-fsx-iscsi-secret
type: Opaque
stringData:
  username: <FSx for ONTAP, for example fsxadmin>
  password: <FSx for ONTAP password>
---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: fsx-iscsi
spec:
  version: 1
  storageDriverName: ontap-san
  managementLIF: <Management DNS name>
  backendName: fsx-iscsi
  svm: <SVM name>
  credentials:
    name: backend-tbc-fsx-iscsi-secret
```

StorageClass definition (salva come `sc-fsx-iscsi.yaml`):

```
# sc-fsx-iscsi.yaml
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: sc-fsx-iscsi
provisioner: csi.trident.netapp.io
parameters:
  backendType: "ontap-san"
  media: "ssd"
  provisioningType: "thin"
  fsType: ext4
  snapshots: "true"
allowVolumeExpansion: true
```

Passaggio 3: Creare un file di configurazione VolumeSnapshotClass

Crea una definizione VolumeSnapshotClass sia per le implementazioni on-premises che per quelle ROSA. Questa configurazione abilita le operazioni basate su snapshot per i volumi persistenti.

VolumeSnapshotClass definition (salva come `snapshot-class.yaml`):

```
# snapshot-class.yaml
apiVersion: snapshot.storage.k8s.io/v1
kind: VolumeSnapshotClass
metadata:
  name: trident-snapshotclass
driver: csi.trident.netapp.io
deletionPolicy: Retain
```

Passaggio 4: applica i file di configurazione al tuo cluster

Applica i file di configurazione che hai creato nei passaggi precedenti al tuo OpenShift cluster.

Passi

1. Applica i file TridentBackendConfig e StorageClass per ciascun protocollo configurato.

Per i cluster on-premises:

```
oc create -f tbc-nas.yaml -n trident
oc create -f sc-nas.yaml

oc create -f tbc-nas-economy.yaml -n trident
oc create -f sc-nas-economy.yaml

oc create -f tbc-iscsi.yaml -n trident
oc create -f sc-iscsi.yaml

oc create -f tbc-nvme.yaml -n trident
oc create -f sc-nvme.yaml

oc create -f tbc-fc.yaml -n trident
oc create -f sc-fc.yaml
```

Per i cluster ROSA:

```
oc create -f tbc-fsx-nas.yaml -n trident
oc create -f sc-fsx-nas.yaml

oc create -f tbc-fsx-iscsi.yaml -n trident
oc create -f sc-fsx-iscsi.yaml
```

2. Applica la configurazione VolumeSnapshotClass.

```
oc create -f snapshot-class.yaml
```

3. Verificare che le risorse siano state create correttamente.

Controlla gli oggetti TridentBackendConfig:

```
oc get tbc -n trident
```

Controlla gli oggetti StorageClass:

```
oc get storageclass
```

Verifica VolumeSnapshotClass:

```
oc get volumesnapshotclass
```

Passaggio 5: Imposta le classi di storage e snapshot predefinite di Trident

Imposta StorageClass e VolumeSnapshotClass di Trident come predefiniti nel cluster OpenShift. Questo è necessario affinché OpenShift Virtualization crei sorgenti di immagini golden per i template delle VM.

Passi

1. Imposta il StorageClass predefinito di Trident.

Imposta una StorageClass supportata da Trident come predefinita del cluster, in modo che le PersistentVolumeClaims la utilizzino automaticamente quando non viene specificata alcuna classe di storage. È necessario configurare due annotazioni: una per l'impostazione predefinita a livello di cluster e una specifica per la virtualizzazione OpenShift.

- a. Imposta l'annotazione StorageClass predefinita a livello di cluster.

Assicurati che solo uno StorageClass sia impostato come predefinito. Se un altro StorageClass è già impostato come predefinito, imposta la relativa annotazione su false.

Nella console, modifica l'annotazione:

```
storageclass.kubernetes.io/is-default-class: "true"
```

Dalla CLI:

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

- b. Imposta l'annotazione predefinita specifica per la OpenShift Virtualization.

OpenShift Virtualization utilizza un'annotazione specifica che ha la precedenza sull'annotazione generale del cluster `is-default-class`. Se un'altra StorageClass è già impostata come predefinita, imposta la relativa annotazione su false.

Nella console, modifica l'annotazione:

```
storageclass.kubevirt.io/is-default-virt-class: "true"
```

Dalla CLI:

```
kubectl patch storageclass <storage class name> -p '{"metadata":  
{"annotations":{"storageclass.kubevirt.io/is-default-virt-class":  
"true"}}}'
```

2. Imposta la VolumeSnapshotClass predefinita di Trident.

Imposta un VolumeSnapshotClass supportato da Trident come predefinito del cluster per abilitare le

operazioni basate su snapshot per i volumi persistenti. Questo garantisce che i VolumeSnapshots utilizzino automaticamente il driver CSI Trident quando non viene specificata alcuna classe di snapshot e consente a OpenShift Virtualization di creare snapshot di immagini golden.

Assicurati che solo un VolumeSnapshotClass sia impostato come predefinito. Se un altro VolumeSnapshotClass è già impostato come predefinito, imposta la relativa annotazione su false.

Nella console, modifica l'annotazione:

```
snapshot.storage.kubernetes.io/is-default-class: "true"
```

Dalla CLI:

```
oc patch volumesnapshotclass <snapshot class name> --type=merge -p
'{"metadata":{"annotations":{"snapshot.storage.kubernetes.io/is-default-
class":"true"}}}'
```

Requisiti per distribuire Red Hat OpenShift Virtualization con ONTAP

Esamina i prerequisiti e i dettagli aggiuntivi per implementare la virtualizzazione OpenShift con i sistemi di storage ONTAP.

Prerequisiti

- Un cluster Red Hat OpenShift (successivo alla versione 4.6) installato su un'infrastruttura bare-metal con nodi worker RHCOS
- Distribuisce controlli di integrità delle macchine per mantenere l'alta disponibilità per le VM
- Un cluster NetApp ONTAP , con SVM configurato con il protocollo corretto.
- Trident installato sul cluster OpenShift
- Una configurazione backend Trident creata
- Una StorageClass configurata sul cluster OpenShift con Trident come provisioner
- Un Trident VolumeSnapshotClass configurato sul cluster OpenShift con Trident come provisioner
- Trident StorageClass configurato come predefinito per il cluster OpenShift o per OpenShift Virtualization.

Per i prerequisiti Trident sopra indicati, vedere ["Sezione di installazione Trident"](#) per i dettagli.

- Accesso di amministrazione del cluster al cluster Red Hat OpenShift
- Accesso amministratore al cluster NetApp ONTAP
- Una postazione di lavoro di amministrazione con gli strumenti tridentctl e oc installati e aggiunti a \$PATH

Poiché la OpenShift Virtualization è gestita da un operatore installato sul cluster OpenShift, impone un overhead aggiuntivo su memoria, CPU e storage, che devi considerare quando pianifichi i requisiti hardware del cluster. Consulta la documentazione ["Qui"](#) per maggiori dettagli.

Facoltativamente, puoi anche specificare un sottoinsieme dei nodi del cluster OpenShift per ospitare gli operatori, i controller e le VM di OpenShift Virtualization configurando le regole di posizionamento dei nodi. Per configurare le regole di posizionamento dei nodi per OpenShift Virtualization, segui la documentazione ["Qui"](#).

Per lo storage che supporta OpenShift Virtualization, NetApp consiglia di avere una StorageClass dedicata che richiede storage da un particolare backend Trident, che a sua volta è supportato da una SVM dedicata. In questo modo si mantiene un livello di multi-tenancy per quanto riguarda i dati forniti per i carichi di lavoro basati su VM sul cluster OpenShift.

NetApp Trident CSI, Storage Profiles, modalità di accesso ai volumi per la virtualizzazione OpenShift

In OpenShift Virtualization, le sorgenti di avvio sono immagini disco utilizzate per il provisioning di macchine virtuali (VM) da template. Di solito sono archiviate come DataVolumes o Persistent Volume Claims (PVC) all'interno dello spazio dei nomi `openshift-virtualization-os-images`. Perché OpenShift Virtualization funzioni correttamente, ci sono due attività di configurazione dello storage che sono obbligatorie prima di installare OpenShift Virtualization tramite l'operatore:

1. Devi configurare un StorageClass predefinito per il cluster. Altrimenti, OpenShift Virtualization non può importare automaticamente le immagini di origine di avvio.
2. Devi configurare i profili di storage se il tuo provider di storage non è riconosciuto da CDI. Un profilo di storage fornisce impostazioni di storage consigliate in base allo StorageClass associato. Una delle funzioni critiche di StorageProfile è definire le strategie di clonazione per uno StorageClass. Questo determina come vengono clonati i volumi (ad esempio, utilizzando snapshot, cloni CSI o copie assistite dall'host), ottimizzando le prestazioni e l'utilizzo delle risorse in base alle capacità dello storage sottostante. Il driver Trident CSI è riconosciuto da CDI e un profilo di storage predefinito viene creato automaticamente per qualsiasi StorageClass di Trident creato nel cluster. La strategia di clonazione per questo profilo di storage predefinito è impostata su `snapshot`, che è la strategia di clonazione consigliata per il driver Trident CSI. Devi creare la `VolumeSnapshotClass` e la `StorageClass` per Trident e impostare la `StorageClass` come predefinita prima di installare OpenShift Virtualization. Questo garantisce che le immagini di origine di avvio siano disponibili come `VolumeSnapshots` nel namespace `openshift-virtualization-os-images`, così da poter effettuare il provisioning delle VM in modo efficiente e senza errori.

Per alcuni sistemi operativi supportati (come RHEL, Fedora o CentOS), OpenShift Virtualization fornisce e aggiorna automaticamente le sorgenti di avvio. Queste vengono scaricate quando OpenShift Virtualization viene installato e sono gestite e aggiornate automaticamente all'ultima versione del sistema operativo. Se un sistema operativo non viene fornito automaticamente, un amministratore deve preparare e collegare manualmente un'immagine personalizzata. Quando usi NetApp Trident come CSI, ottieni i seguenti vantaggi aggiuntivi:

1. Le immagini sorgente di avvio saranno disponibili come snapshot nello spazio dei nomi `openshift-virtualization-os-images`. Quando vengono create le macchine virtuali, l'immagine di avvio viene clonata dallo snapshot, rendendo così l'avvio della macchina virtuale molto più veloce.
2. Un profilo di storage viene creato automaticamente per ogni Trident StorageClass creato ed è automaticamente configurato con la strategia di clonazione impostata su `snapshot`. La modalità di accesso è impostata su `ReadWriteMany`. Non serve alcuna configurazione manuale.

È fondamentale che tu crei un Trident StorageClass e un Trident VolumeSnapshotClass e li imposti come predefiniti prima di installare OpenShift Virtualization nel tuo cluster. Questo garantirà che le immagini di origine di avvio siano disponibili come `VolumeSnapshots` nel namespace `openshift-virtualization-os-images`, così da poter effettuare il provisioning delle VM in modo efficiente e senza errori.

Un altro vantaggio dell'utilizzo di Trident CSI: . In OpenShift Virtualization, i dischi di avvio delle VM devono avere modalità di accesso `RWX` per eseguire la migrazione live delle VM. Trident supporta la modalità di

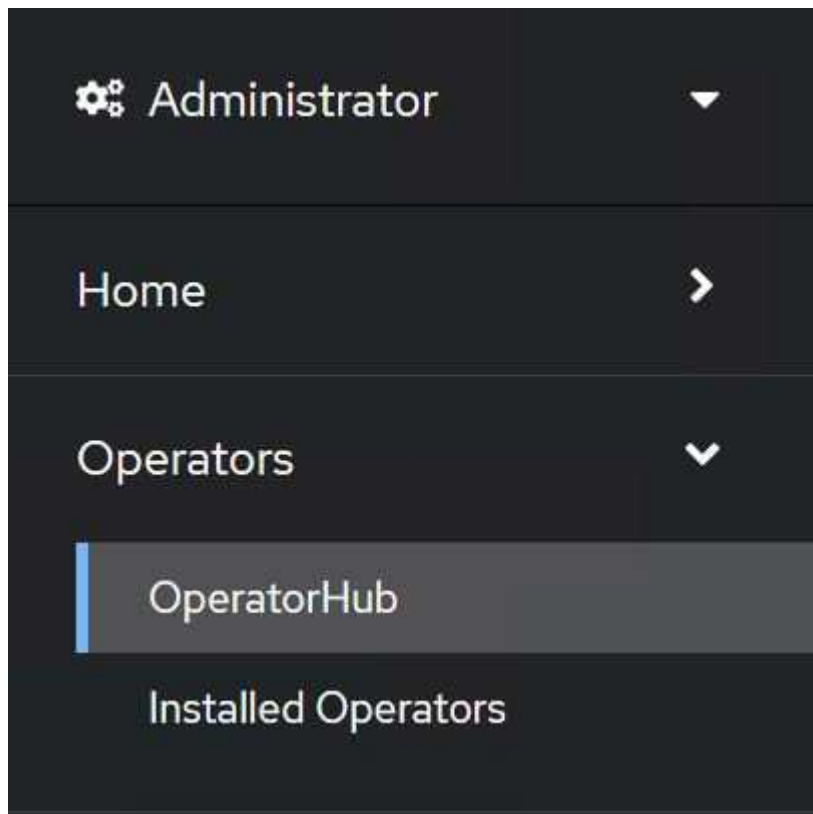
accesso **ReadWriteMany** per i volumi di cui è stato eseguito il provisioning utilizzando protocolli NAS così come protocolli SAN (quando volumeMode è impostato su Block e non su FileSystem). Puoi usare entrambi i protocolli per il disco di avvio. L'oggetto Storage profile verrà configurato automaticamente in base alle impostazioni di StorageClass e le modalità di accesso del disco di avvio verranno impostate in modo appropriato durante il provisioning delle VM.

Dopo che hai impostato Trident StorageClass come predefinito per il cluster, puoi creare ulteriori StorageClasses per Trident con parametri diversi e questi verranno automaticamente riconosciuti da OpenShift Virtualization con i profili di storage appropriati creati per loro. Puoi quindi specificare la StorageClass da usare per il disco di avvio e i dischi dati della VM nella definizione della VM.

Installare OpenShift Virtualization sul OpenShift cluster è un processo semplice utilizzando l'operatore OpenShift Virtualization disponibile in OpenShift OperatorHub. Per i passaggi dettagliati su come installare OpenShift Virtualization, consulta la documentazione ["Qui"](#).

Installa OpenShift Virtualization su un cluster bare-metal Red Hat OpenShift. Questa procedura include l'accesso con privilegi di amministratore del cluster, la navigazione verso OperatorHub e l'installazione dell'operatore OpenShift Virtualization.

1. Accedi al cluster bare-metal di Red Hat OpenShift con accesso cluster-admin.
2. Selezionare Amministratore dal menu a discesa Prospettiva.
3. Vai su Operatori > OperatorHub e cerca OpenShift Virtualization.



4. Selezionare il riquadro OpenShift Virtualization e fare clic su Installa.



OpenShift Virtualization

2.6.2 provided by Red Hat



Install

Latest version

2.6.2

Capability level

- ☒ Basic Install
- ☒ Seamless Upgrades
- ☒ Full Lifecycle
- ☐ Deep Insights
- ☐ Auto Pilot

Provider type

Red Hat

Provider

Red Hat

Requirements

Your cluster must be installed on bare metal infrastructure with Red Hat Enterprise Linux CoreOS workers.

Details

OpenShift Virtualization extends Red Hat OpenShift Container Platform, allowing you to host and manage virtualized workloads on the same platform as container-based workloads. From the OpenShift Container Platform web console, you can import a VMware virtual machine from vSphere, create new or clone existing VMs, perform live migrations between nodes, and more. You can use OpenShift Virtualization to manage both Linux and Windows VMs.

The technology behind OpenShift Virtualization is developed in the [KubeVirt](#) open source community. The KubeVirt project extends [Kubernetes](#) by adding additional virtualization resource types through [Custom Resource Definitions](#) (CRDs). Administrators can use Custom Resource Definitions to manage [VirtualMachine](#) resources alongside all other resources that Kubernetes provides.

5. Nella schermata Installa operatore, lasciare tutti i parametri predefiniti e fare clic su Installa.

Update channel *

- ☐ 2.1
- ☐ 2.2
- ☐ 2.3
- ☐ 2.4
- ☒ stable

Installation mode *

- ☐ All namespaces on the cluster (default)
This mode is not supported by this Operator
- ☒ A specific namespace on the cluster
Operator will be available in a single Namespace only.

Installed Namespace *

- ☒ Operator recommended Namespace: **PR** openshift-cnv



Namespace creation

Namespace **openshift-cnv** does not exist and will be created.

- ☐ Select a Namespace

Approval strategy *

- ☒ Automatic
- ☐ Manual

Install

Cancel



OpenShift Virtualization
provided by Red Hat

Provided APIs



OpenShift
Virtualization
Deployment

Required

Represents the deployment of
OpenShift Virtualization

6. Attendere il completamento dell'installazione dell'operatore.



OpenShift Virtualization
2.6.2 provided by Red Hat

Installing Operator
The Operator is being installed. This may take a few minutes.
[View installed Operators in Namespace openshift-cnv](#)

7. Dopo aver installato l'operatore, fare clic su Crea iperconvergente.



OpenShift Virtualization
2.6.2 provided by Red Hat

Installed operator - operand required
The Operator has installed successfully. Create the required custom resource to be able to use this Operator.

 **HyperConverged**  **Required**

Creates and maintains an OpenShift Virtualization Deployment

Create HyperConverged

[View installed Operators in Namespace openshift-cnv](#)

8. Nella schermata Crea iperconvergente, fare clic su Crea, accettando tutti i parametri predefiniti. Questo passaggio avvia l'installazione di OpenShift Virtualization.

Name *

Labels

Infra >

infra HyperConvergedConfig influences the pod configuration (currently only placement) for all the infra components needed on the virtualization enabled cluster but not necessarily directly on each node running VMs/VMLs.

Workloads >

workloads HyperConvergedConfig influences the pod configuration (currently only placement) of components which need to be running on a node where virtualization workloads should be able to run. Changes to Workloads HyperConvergedConfig can be applied only without existing workload.

Bare Metal Platform

☒ true

BareMetalPlatform indicates whether the infrastructure is baremetal.

Feature Gates >

featureGates is a map of feature gate flags. Setting a flag to "true" will enable the feature. Setting "false" or removing the feature gate, disables the feature.

Local Storage Class Name

LocalStorageClassName the name of the local storage class.

9. Dopo che tutti i pod passano allo stato In esecuzione nello spazio dei nomi openshift-cnv e l'operatore OpenShift Virtualization è nello stato Riuscito, l'operatore è pronto per l'uso. Ora è possibile creare VM sul cluster OpenShift.

Project: openshift-cnv ▾

Installed Operators

Installed Operators are represented by ClusterServiceVersions within this Namespace. For more information, see the [Understanding Operators documentation](#) or create an Operator and ClusterServiceVersion using the [Operator SDK](#).

Name ▾	Managed Namespaces ⓘ	Status	Last updated	Provided APIs
 OpenShift Virtualization 2.6.2 provided by Red Hat	 openshift-cnv	 Succeeded Up to date	 May 18, 8:02 pm	OpenShift Virtualization Deployment HostPathProvisioner deployment

Crea una macchina virtuale utilizzando lo storage ONTAP con Red Hat OpenShift Virtualization

Crea una VM con OpenShift Virtualization. Questa procedura include la selezione di un modello di sistema operativo, la configurazione delle StorageClasses e la personalizzazione dei parametri della VM per soddisfare requisiti specifici. Come

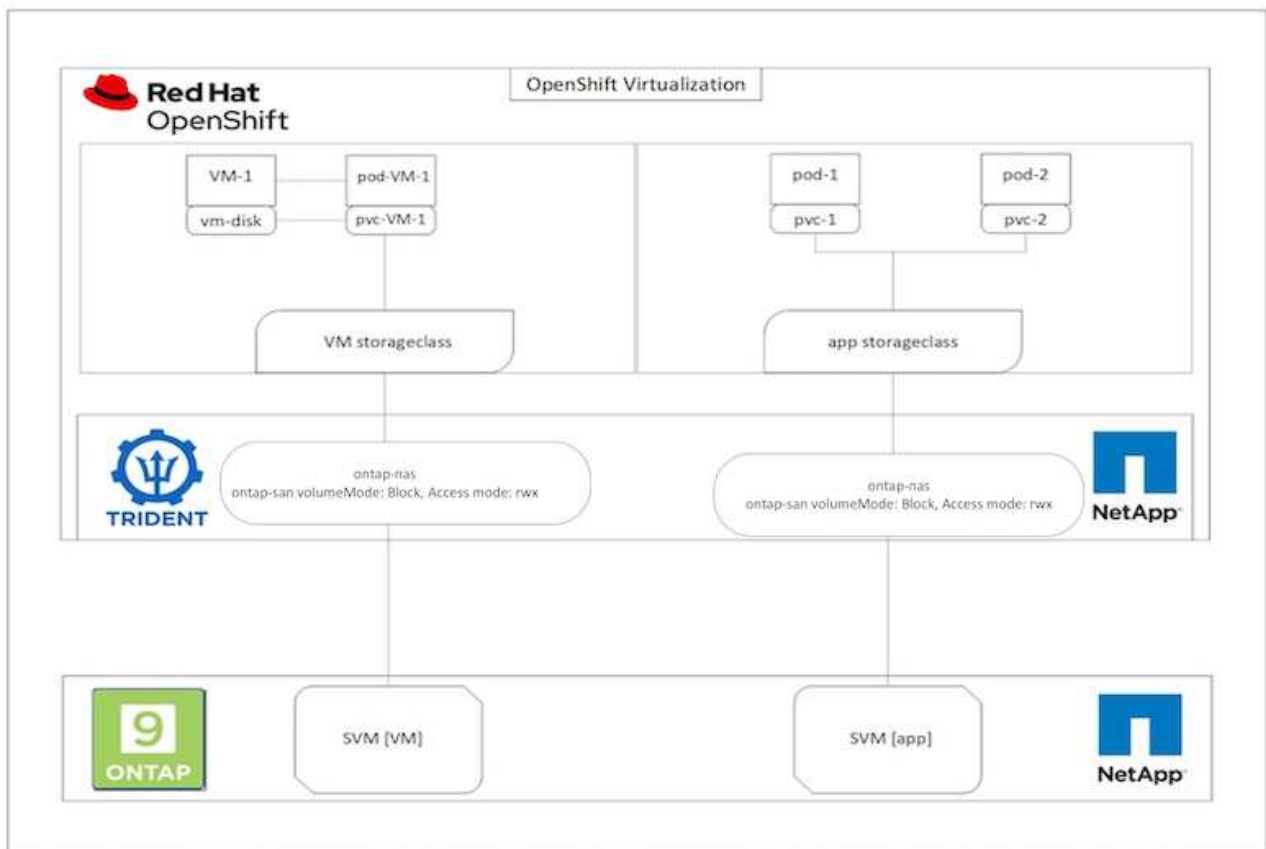
prerequisito, dovresti aver già creato il backend Trident, la StorageClass e gli oggetti della volume snapshot class. Puoi fare riferimento a ["Sezione di installazione Trident"](#) per i dettagli.

Crea VM

Le VM sono distribuzioni con stato che richiedono volumi per ospitare il sistema operativo e i dati. Con CNV, poiché le VM vengono eseguite come pod, sono supportate da PV ospitati su NetApp ONTAP tramite Trident. Questi volumi sono collegati come dischi e memorizzano l'intero file system, inclusa la sorgente di avvio della VM.



La StorageClass di Trident deve essere impostata come StorageClass predefinita sul cluster per garantire che i PVC creati per i dischi delle VM siano sottoposti a provisioning utilizzando il backend Trident. Puoi avere più StorageClasses sul cluster, ma la StorageClass predefinita deve essere utilizzata per il provisioning del PVC per i dischi di avvio delle VM. I dischi aggiuntivi possono utilizzare altre StorageClasses se necessario.



Per creare rapidamente una macchina virtuale sul cluster OpenShift, completare i seguenti passaggi:

1. Vai su Virtualizzazione > Macchine virtuali e fai clic su Crea.
2. Seleziona Da modello.
3. Selezionare il sistema operativo desiderato per il quale è disponibile la sorgente di avvio.
4. Selezionare la casella di controllo Avvia la macchina virtuale dopo la creazione.

5. Fare clic su Creazione rapida macchina virtuale.

La macchina virtuale viene creata e avviata e passa allo stato **Running**. Crea automaticamente un PVC e un PV corrispondente per il disco di avvio utilizzando la StorageClass predefinita. Per poter eseguire la migrazione live della VM in futuro, devi assicurarti che la StorageClass utilizzata per i dischi supporti i volumi RWX. Questo è un requisito per la migrazione live. `ontap-nas` e `ontap-san` (volumeMode block per i protocolli iSCSI e NVMe/TCP) possono supportare le modalità di accesso RWX per i volumi creati utilizzando le rispettive StorageClasses.

Per configurare `ontap-san` StorageClass sul cluster, vedi il ["Sezione per la configurazione di StorageClass"](#).



Facendo clic su Creazione rapida VirtualMachine verrà utilizzato il StorageClass predefinito per creare il PVC e il PV per il disco root avviabile della VM. Puoi selezionare un StorageClass diverso per il disco selezionando Personalizza VirtualMachine > Personalizza parametri VirtualMachine > Dischi e poi modificando il disco per usare il StorageClass richiesto.

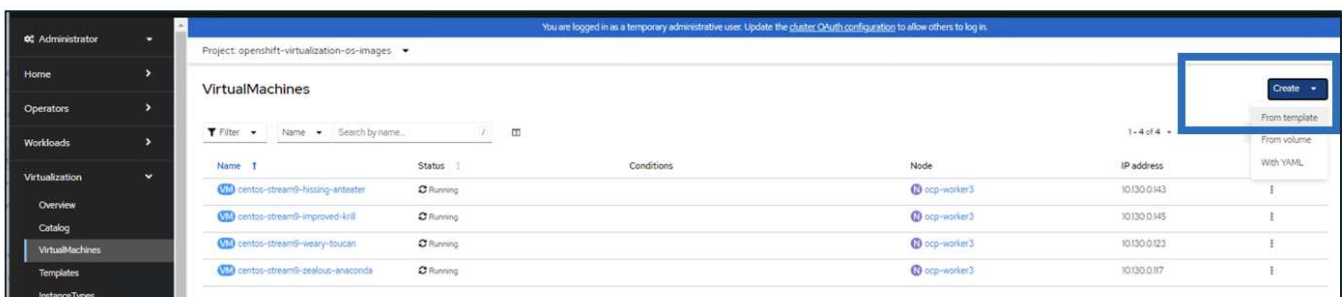
In genere, durante il provisioning dei dischi della VM, si preferisce la modalità di accesso a blocchi rispetto ai file system.

Per personalizzare la creazione della macchina virtuale dopo aver selezionato il modello del sistema operativo, fare clic su Personalizza macchina virtuale anziché su Creazione rapida.

1. Se il sistema operativo selezionato ha configurato la sorgente di avvio, è possibile fare clic su **Personalizza parametri VirtualMachine**.
2. Se il sistema operativo selezionato non ha alcuna sorgente di avvio configurata, è necessario configurarla. È possibile visualizzare i dettagli sulle procedure illustrate nel ["documentazione"](#).
3. Dopo aver configurato il disco di avvio, puoi fare clic su **Personalizza parametri VirtualMachine**.
4. È possibile personalizzare la VM dalle schede presenti in questa pagina. Ad esempio, fare clic sulla scheda **Dischi** e quindi su **Aggiungi disco** per aggiungere un altro disco alla VM.
5. Fare clic su Crea macchina virtuale per creare la macchina virtuale; in questo modo verrà avviato un pod corrispondente in background.



Quando una sorgente di avvio viene configurata per un modello o un sistema operativo da un URL o da un registro, crea un PVC nel `openshift-virtualization-os-images` progetto e scarica l'immagine guest KVM sul PVC. È necessario assicurarsi che i PVC modello dispongano di spazio sufficiente per ospitare l'immagine guest KVM per il sistema operativo corrispondente. Questi PVC vengono quindi clonati e collegati come rootdisk alle macchine virtuali quando vengono create utilizzando i rispettivi modelli in qualsiasi progetto.



Create new VirtualMachine

Select an option to create a VirtualMachine from.

Template catalog InstanceTypes

Template project: All projects Default templates

All items Filter by keyword...

13 items

Default templates

User templates

☐ Boot source available

Operating system

☐ CentOS

☐ Fedora

☐ Other

☐ RHEL

☐ Windows

Workload

☐ Desktop

☐ High performance

☐ Server

CentOS Stream 8 VM
centos-stream8-server-small

Project openshift
Boot source PVC (auto import)
Workload Server
CPU 1
Memory 2 GiB

CentOS Stream 9 VM
centos-stream9-server-small

Project openshift
Boot source PVC (auto import)
Workload Server
CPU 1
Memory 2 GiB

CentOS 7 VM
centos7-server-small

Project openshift
Boot source PVC (auto import)
Workload Server
CPU 1
Memory 2 GiB

Fedora VM
fedora-server-small

Project openshift
Boot source PVC (auto import)
Workload Server
CPU 1
Memory 2 GiB

Red Hat Enterprise Linux 7 VM
rhel7-server-small

Project openshift
Boot source PVC (auto import)
Workload Server
CPU 1
Memory 2 GiB

Red Hat Enterprise Linux 8 VM
rhel8-server-small

Project openshift
Boot source PVC (auto import)
Workload Server
CPU 1
Memory 2 GiB

Red Hat Enterprise Linux 9 VM
rhel9-server-small

Project openshift
Boot source PVC (auto import)
Workload Server
CPU 1
Memory 2 GiB

Microsoft Windows 10 VM
windows10-desktop-medium

Project openshift
Boot source PVC
Workload Desktop
CPU 1
Memory 4 GiB

Microsoft Windows 11 VM
windows11-desktop-medium

Project openshift
Boot source PVC
Workload Desktop
CPU 2
Memory 4 GiB

Microsoft Windows Server 2012 R2 VM
windows2k12r2-server-medium

Project openshift
Boot source PVC
Workload Server
CPU 1
Memory 4 GiB

VirtualMachines > VirtualMachine details

VM centos-stream9-zealous-anaconda Running VAML Actions

Overview Details Metrics VAML **Configuration** Events Console Snapshots Diagnostics

Disk **Add disk**

Network interfaces

Scheduling

Environment

Scripts

Name	Source	Size	Drive	Interface	Storage class
cloudinitdisk	Other	-	Disk	virtio	-
data-disk1 Persistent Hotplug	centos-stream9-zealous-anaconda-data-disk1	30.00 GiB	Disk	SCSI	ontap-san-block
rootdisk bootable	centos-stream9-zealous-anaconda	30.00 GiB	Disk	virtio	ontap-san-block

File systems

Name	File system type	Mount point	Total bytes	Used bytes
vdal	xfs	/	29.94 GiB	1.30 GiB

Project: openshift-virtualization-os-images

Catalog > Customize template parameters > Customize VirtualMachine

Customize and create VirtualMachine

Template: CentOS Stream 9 VM

Overview | **YAML** | Scheduling | Environment | Network interfaces | Disks | Scripts | Metadata

Name

centos-stream9-pleased-hamster

Namespace

openshift-virtualization-os-images

Description

Not available

Operating system

CentOS Stream 9 VM

CPU | Memory

1 CPU | 2 GiB Memory

Machine type

pc-q35-rhel9.2.0

Boot mode

BIOS

Start in pause mode

☐

Workload profile

Server

Network interfaces (1)

Name	Network	Type
default	Pod networking	Masquerade

Disks (2)

Name	Drive	Size
rootdisk	Disk	30 GiB
cloudinitdisk	Disk	-

Hardware devices

GPU devices

Not available

Host devices

Not available

Headless mode

☐

Hostname

centos-stream9-pleased-hamster

☒ Start this VirtualMachine after creation

Create VirtualMachine Cancel

Dimostrazione video

Il seguente video mostra una dimostrazione della creazione di una VM in OpenShift Virtualization utilizzando l'archiviazione iSCSI.

[Creare una VM in OpenShift Virtualization utilizzando Block Storage](#)

Crea una macchina virtuale con Red Hat OpenShift Virtualization utilizzando lo storage ONTAP tramite metodi alternativi

Oltre al metodo di creazione di una macchina virtuale tramite l'interfaccia utente di OpenShift Virtualization, ci sono altri metodi per creare macchine virtuali su OpenShift Virtualization. Questa sezione fornisce una panoramica di alcuni di questi metodi.

Crea una macchina virtuale da un file YAML con Red Hat OpenShift Virtualization

Puoi anche creare una VM applicando un file YAML con il comando `oc apply`. Questo metodo è utile per chi preferisce usare la CLI o vuole automatizzare la creazione della VM tramite script. Per creare una VM con questo metodo, puoi creare un file YAML che definisce la VM e le risorse associate, come i PVC per i dischi della VM. Poi puoi applicare il file YAML usando il comando `oc apply -f <yaml-file>` per creare la VM sul cluster OpenShift.

Mostra esempio

```
apiVersion: kubevirt.io/v1
kind: VirtualMachine
metadata:
  name: test-vms-vm
  namespace: test-5vms
  labels:
    app: test-vms-vm
    vm.kubevirt.io/template: centos-stream9-server-small
spec:
  dataVolumeTemplates:
    - apiVersion: cdi.kubevirt.io/v1beta1
      kind: DataVolume
      metadata:
        name: test-vms-vm-rootdisk
      spec:
        sourceRef:
          kind: DataSource
          name: centos-stream9
          namespace: openshift-virtualization-os-images
        storage:
          resources:
            requests:
              storage: 30Gi
  runStrategy: Always
  template:
    metadata:
      labels:
        kubevirt.io/domain: test-vms-vm
    spec:
      domain:
        cpu:
          cores: 1
          sockets: 1
          threads: 1
        devices:
          disks:
            - bootOrder: 1
              disk:
                bus: virtio
                name: rootdisk
            - disk:
                bus: virtio
                name: cloudinitdisk
          interfaces:
```

```

- masquerade: {}
  model: virtio
  name: default
resources:
  requests:
    memory: 2Gi
networks:
- name: default
  pod: {}
terminationGracePeriodSeconds: 180
volumes:
- dataVolume:
    name: test-vms-vm-rootdisk
    name: rootdisk
- cloudInitNoCloud:
    userData: |-
      #cloud-config
      user: admin
      password: <replace with your password>
      chpasswd: { expire: False }
    name: cloudinitdisk

```

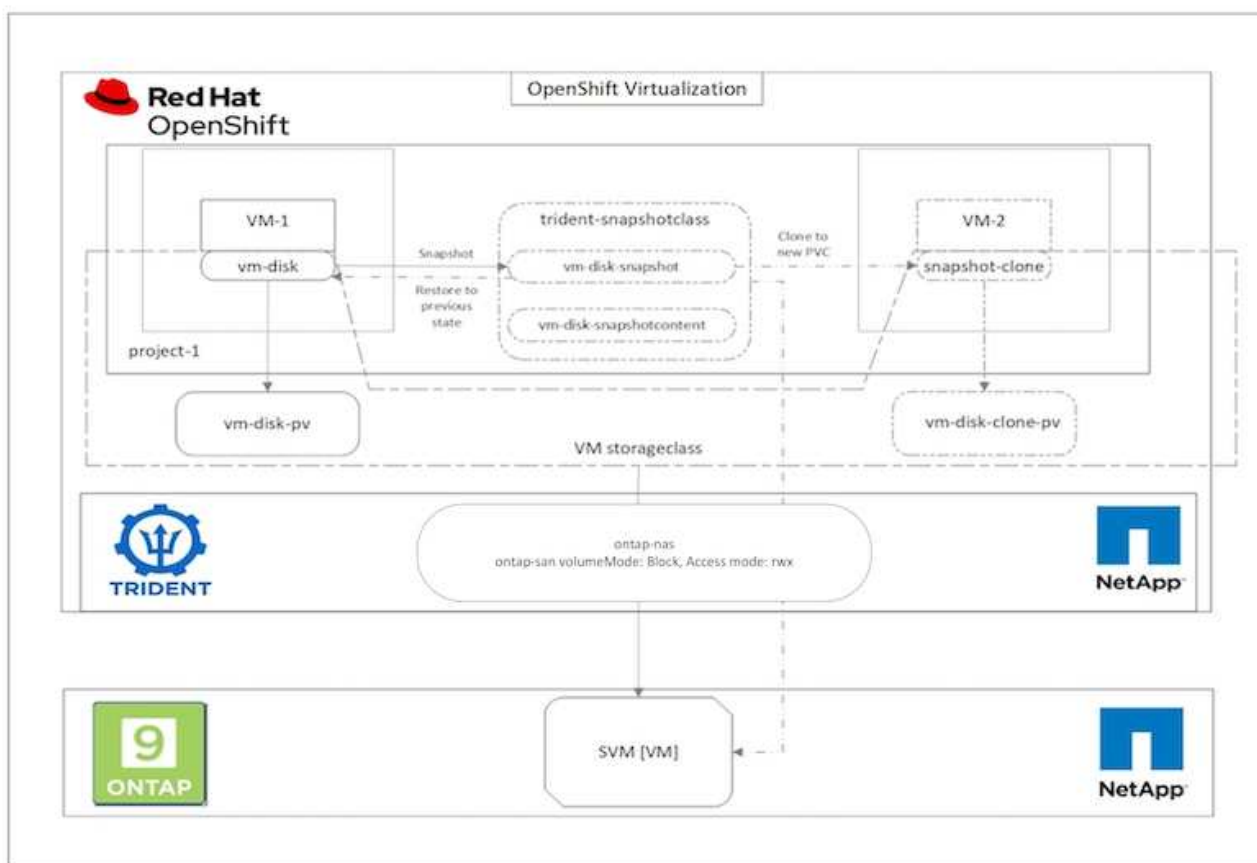
Crea una VM da una copia snapshot con Red Hat OpenShift Virtualization

Puoi creare una VM da uno snapshot con OpenShift Virtualization. Questa procedura include la creazione di una VolumeSnapshotClass, l'acquisizione di uno snapshot della persistent volume claim (PVC) della VM, il ripristino dello snapshot su un nuovo PVC e il deployment di una nuova VM che utilizza il PVC ripristinato come disco di root.

Con Trident e Red Hat OpenShift, gli utenti possono acquisire uno snapshot di un volume persistente sulle classi di archiviazione da esso fornite. Grazie a questa funzionalità, gli utenti possono effettuare una copia di un volume in un dato momento e utilizzarla per creare un nuovo volume o ripristinare lo stesso volume a uno stato precedente. Ciò consente o supporta una varietà di casi d'uso, dal rollback alla clonazione al ripristino dei dati.

Per le operazioni Snapshot in OpenShift, è necessario definire le risorse VolumeSnapshotClass, VolumeSnapshot e VolumeSnapshotContent.

- Un VolumeSnapshotContent è l'istantanea effettiva acquisita da un volume nel cluster. Si tratta di una risorsa a livello di cluster analoga a PersistentVolume per l'archiviazione.
- Un VolumeSnapshot è una richiesta di creazione dello snapshot di un volume. È analogo a PersistentVolumeClaim.
- VolumeSnapshotClass consente all'amministratore di specificare attributi diversi per un VolumeSnapshot. Consente di avere attributi diversi per snapshot diversi acquisiti dallo stesso volume.



Mostra esempio

Prerequisito

Devi avere una VM esistente con un PVC collegato, effettuato il provisioning su una Trident Storage Class, e la VM deve essere in esecuzione prima che tu possa creare uno snapshot del PVC. Devi avere una VolumeSnapshotClass creata che può poi essere usata per creare una VolumeSnapshot. Questa sarebbe stata creata come parte dei prerequisiti per l'installazione di OpenShift Virtualization. Puoi fare riferimento alla "[Passaggio 3 per creare una configurazione VolumeSnapshotClass](#)" per i dettagli.

Per creare Snapshot di una macchina virtuale, completa i seguenti passaggi:

1. Identificare il PVC collegato alla VM di origine e quindi creare uno snapshot di tale PVC. Vai a Storage > VolumeSnapshots e fare clic su Crea VolumeSnapshot.
2. Selezionare il PVC per cui si desidera creare lo Snapshot, immettere il nome dello Snapshot o accettare quello predefinito e selezionare la VolumeSnapshotClass appropriata. Quindi fare clic su Crea.
3. In questo modo si crea un'istantanea del PVC in quel momento.

Create VolumeSnapshot

[Edit YAML](#)

PersistentVolumeClaim *

PVC rhel8-short-frog-rootdisk-28dvv

Name *

rhel8-short-frog-rootdisk-28dvv-snapshot

Snapshot Class *

VSC trident-snapshot-class

Create

Cancel

Crea una nuova VM dallo snapshot

1. Per prima cosa, ripristina lo Snapshot in un nuovo PVC. Passare a Storage > VolumeSnapshots, fare clic sui puntini di sospensione accanto allo Snapshot che si desidera ripristinare e fare clic su Ripristina come nuovo PVC.
2. Inserisci i dettagli del nuovo PVC e fai clic su Ripristina. In questo modo si crea un nuovo PVC.

Restore as new PVC

When restore action for snapshot **rhel8-short-frog-rootdisk-28dvb-snapshot** is finished a new crash-consistent PVC copy will be created.

Name *

rhel8-short-frog-rootdisk-28dvb-snapshot-restore

Storage Class *



basic



Access Mode *

☐ Single User (RWO) ☒ Shared Access (RWX) ☐ Read Only (ROX)

Size *

20

GiB



VolumeSnapshot details

Created at

 May 21, 12:46 am

Namespace

 default

Status

 Ready

API version

snapshot.storage.k8s.io/v1

Size

20 GiB

1. Successivamente, creare una nuova VM da questo PVC. Vai su Virtualizzazione > Macchine virtuali e fai clic su Crea > Con YAML.
2. Nella sezione spec > template > spec > volumi, specificare il nuovo PVC creato da Snapshot anziché dal disco del contenitore. Fornisci tutti gli altri dettagli per la nuova VM in base alle tue esigenze.

```
- name: rootdisk
  persistentVolumeClaim:
    claimName: rhel8-short-frog-rootdisk-28dvb-snapshot-restore
```

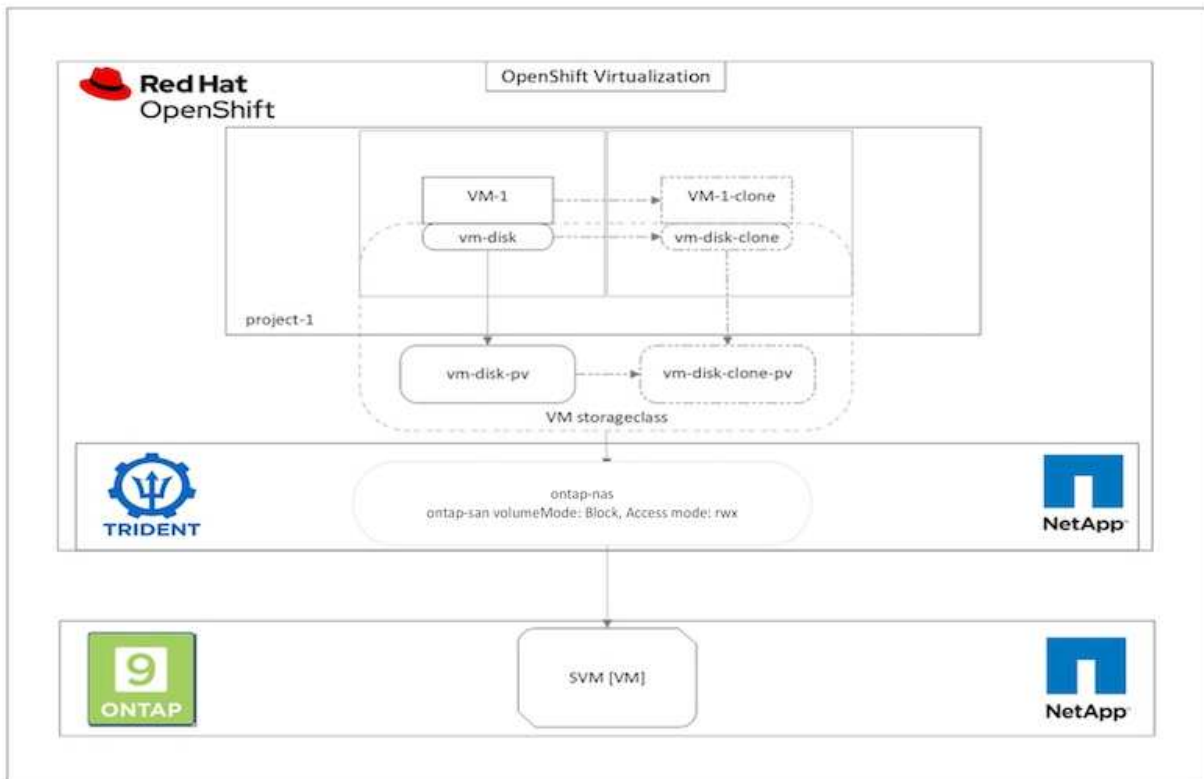
1. Fare clic su Crea per creare la nuova VM.
2. Dopo aver creato correttamente la VM, accedi e verifica che la nuova VM abbia lo stesso stato della VM il cui PVC è stato utilizzato per creare lo snapshot al momento della creazione dello snapshot.

Clona una VM in OpenShift Virtualization usando Trident

Questa procedura prevede l'utilizzo della clonazione dei volumi Trident CSI, che ti permette di creare una nuova macchina virtuale arrestando la macchina virtuale di origine oppure di mantenerla in esecuzione.

Clonazione di macchine virtuali

La clonazione di una VM esistente in OpenShift viene eseguita con il supporto della funzionalità di clonazione Volume CSI di Trident. La clonazione del volume CSI consente di creare un nuovo PVC utilizzando un PVC esistente come origine dati, duplicandone il PV. Una volta creato, il nuovo PVC funziona come un'entità separata, senza alcun collegamento o dipendenza dal PVC di origine.



Esistono alcune restrizioni da considerare per quanto riguarda la clonazione del volume CSI:

1. Il PVC di origine e quello di destinazione devono trovarsi nello stesso progetto.
2. La clonazione è supportata all'interno della stessa classe di archiviazione.
3. La clonazione può essere eseguita solo quando i volumi di origine e di destinazione utilizzano la stessa impostazione VolumeMode; ad esempio, un volume a blocchi può essere clonato solo su un altro volume a blocchi.

Le VM in un cluster OpenShift possono essere clonate in due modi:

1. Arrestando la VM di origine
2. Mantenendo attiva la VM di origine

Spegnendo la macchina virtuale di origine

La clonazione di una VM esistente tramite l'arresto della VM è una funzionalità nativa di OpenShift implementata con il supporto di Trident. Per clonare una VM, completare i seguenti passaggi.

1. Vai su Carichi di lavoro > Virtualizzazione > Macchine virtuali e fai clic sui puntini di sospensione accanto alla macchina virtuale che desideri clonare.
2. Fare clic su Clona macchina virtuale e fornire i dettagli per la nuova VM.

Clone Virtual Machine

Name *	rhel8-short-frog-clone
Description	
Namespace *	default
	☒ Start virtual machine on clone
Configuration	Operating System Red Hat Enterprise Linux 8.0 or higher Flavor Small: 1 CPU 2 GiB Memory Workload Profile server NICs default - virtio Disks cloudinitdisk - cloud-init disk rootdisk - 20Gi - basic

 The VM rhel8-short-frog is still running. It will be powered off while cloning.

3. Fare clic su Clona macchina virtuale; la macchina virtuale di origine viene arrestata e viene avviata la creazione della macchina virtuale clone.
4. Una volta completato questo passaggio, è possibile accedere e verificare il contenuto della VM

clonata.

Mantenendo attiva la macchina virtuale di origine

È anche possibile clonare una VM esistente clonando il PVC esistente della VM di origine e creando quindi una nuova VM utilizzando il PVC clonato. Con questo metodo non è necessario arrestare la VM di origine. Per clonare una VM senza arrestarla, completare i seguenti passaggi.

1. Passare a Storage > PersistentVolumeClaims e fare clic sui puntini di sospensione accanto al PVC associato alla VM di origine.
2. Fare clic su Clona PVC e fornire i dettagli per il nuovo PVC.

Clone

Name *

rhel8-short-frog-rootdisk-28dvv-clone

Access Mode *

☐ Single User (RWO) ☒ Shared Access (RWX) ☐ Read Only (ROX)

Size *

20

GiB

PVC details

Namespace

 default

Requested capacity

20 GiB

Access mode

Shared Access (RWX)

Storage Class

 basic

Used capacity

2.2 GiB

Volume mode

Filesystem

Cancel

Clone

3. Quindi fare clic su Clona. In questo modo si crea un PVC per la nuova VM.
4. Vai a Carichi di lavoro > Virtualizzazione > Macchine virtuali e fai clic su Crea > Con YAML.
5. Nella sezione spec > template > spec > volumi, allegare il PVC clonato al posto del disco contenitore. Fornisci tutti gli altri dettagli per la nuova VM in base alle tue esigenze.

```
- name: rootdisk
  persistentVolumeClaim:
    claimName: rhel8-short-frog-rootdisk-28dvv-clone
```

1. Fare clic su Crea per creare la nuova VM.
2. Dopo aver creato correttamente la VM, accedi e verifica che la nuova VM sia un clone della VM di origine.

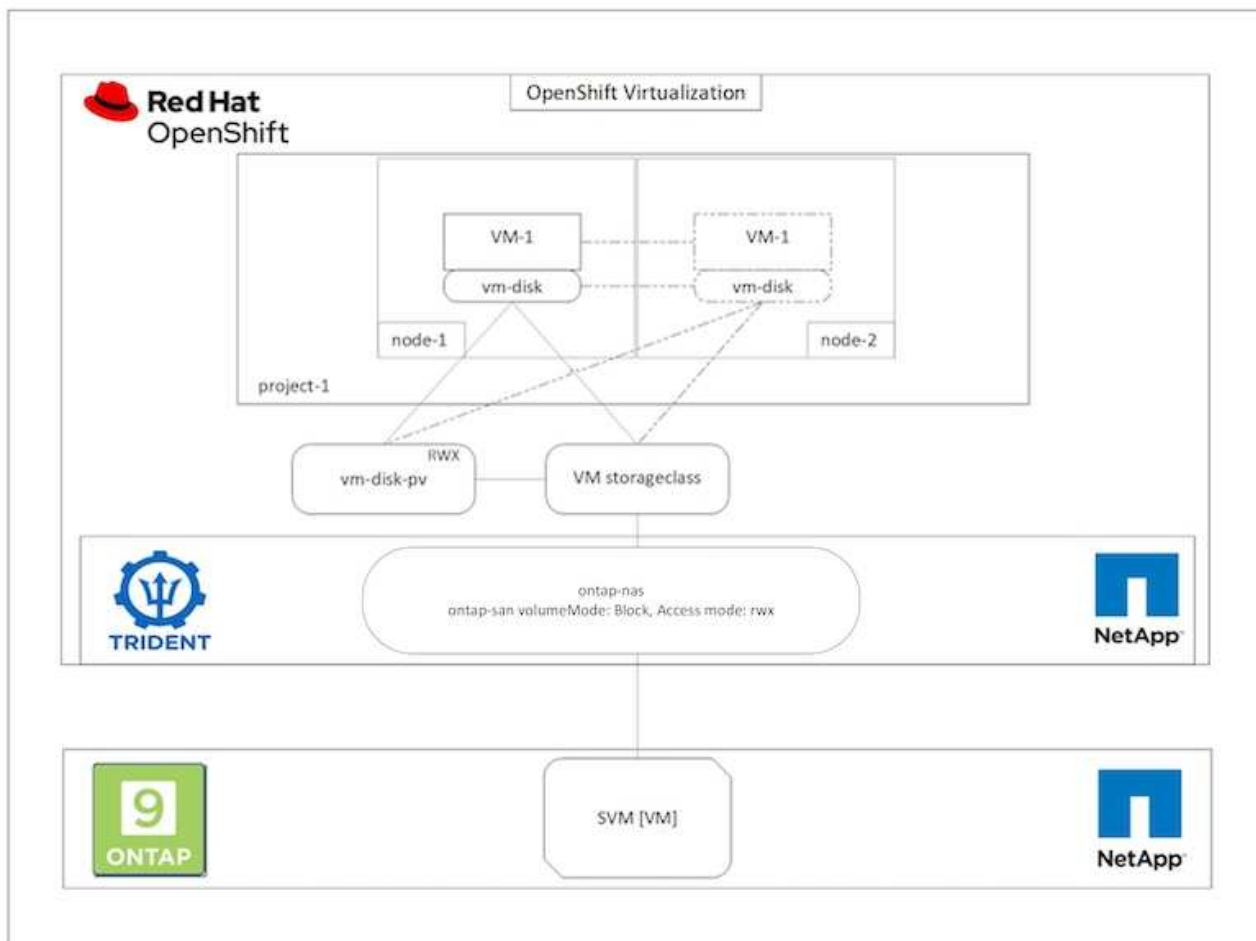
Migrare una VM tra due nodi in un cluster Red Hat OpenShift

Migra una VM in OpenShift Virtualization tra due nodi del cluster senza tempi di inattività. Questa procedura include la conferma che i dischi utilizzino classi di archiviazione compatibili con RWX, l'avvio della migrazione e il monitoraggio dell'avanzamento.

Migrazione live della VM

La migrazione in tempo reale è un processo di migrazione di un'istanza di macchina virtuale da un nodo a un altro in un cluster OpenShift senza tempi di inattività. Affinché la migrazione live funzioni in un cluster OpenShift, le VM devono essere associate a PVC con modalità di accesso ReadWriteMany condivisa. I backend Trident configurati tramite driver ontap-nas supportano la modalità di accesso RWX per i protocolli FileSystem nfs e smb. Fare riferimento alla documentazione ["Qui"](#) . I backend Trident configurati tramite driver ontap-san supportano la modalità di accesso RWX per il volume di blocco per i protocolli iSCSI e NVMe/TCP. Fare riferimento alla documentazione ["Qui"](#) .

Pertanto, affinché la migrazione live abbia successo, le VM devono essere dotate di dischi (dischi di avvio e dischi hot plug aggiuntivi) con PVC che utilizzano le classi di archiviazione ontap-nas o ontap-san (volumeMode: Block). Quando vengono creati i PVC, Trident crea volumi ONTAP in una SVM abilitata per NFS o iSCSI.

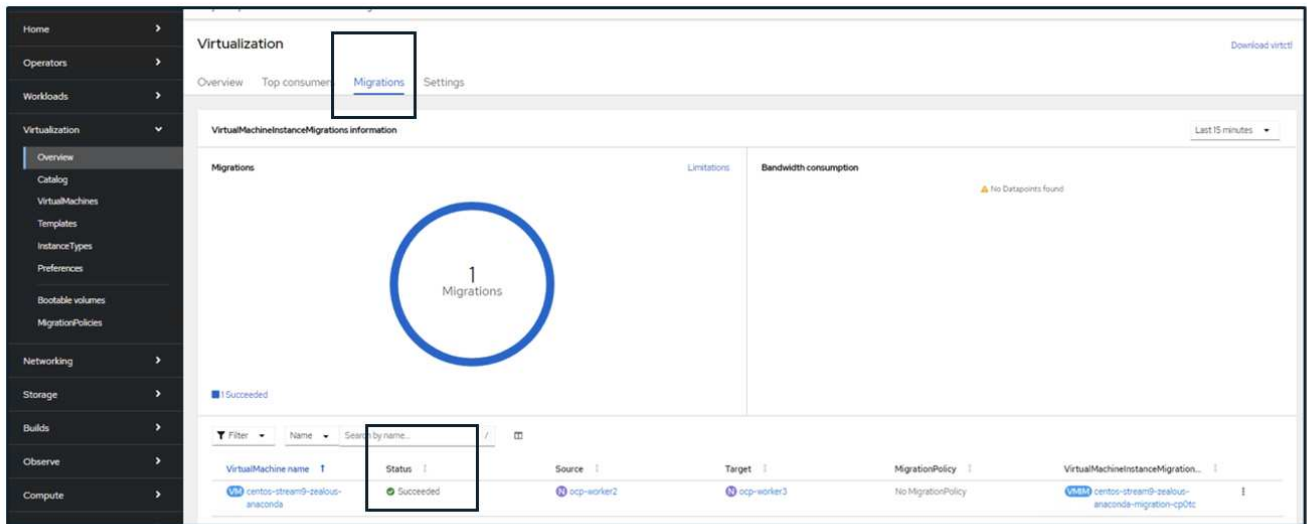
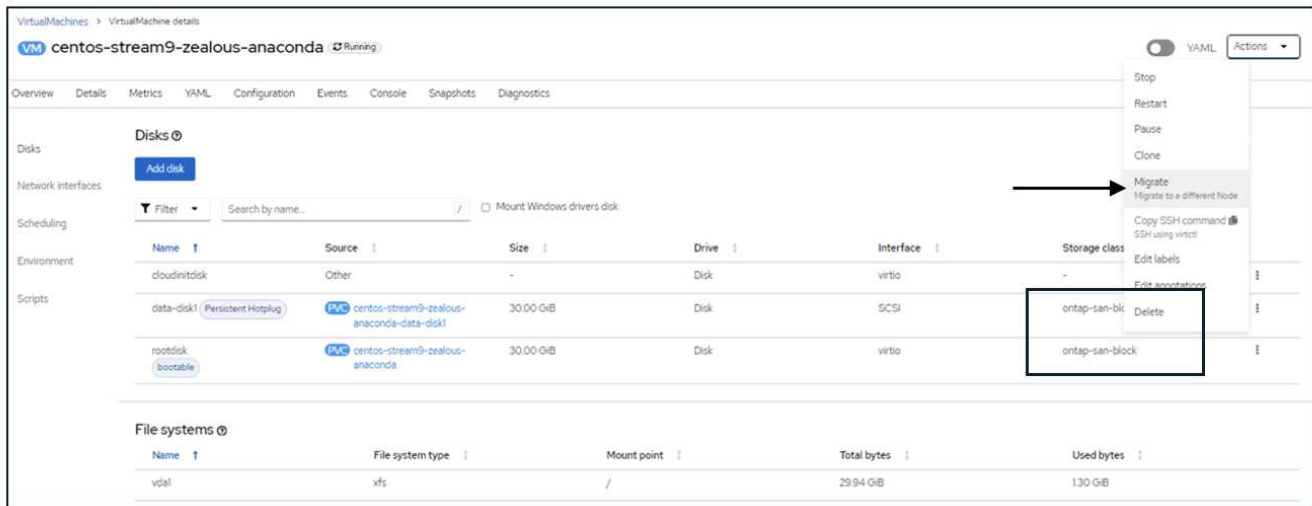


Per eseguire una migrazione live di una VM creata in precedenza e in stato In esecuzione, procedere come segue:

1. Seleziona la VM di cui vuoi eseguire la migrazione in tempo reale.
2. Fare clic sulla scheda **Configurazione**.
3. Assicurarsi che tutti i dischi della VM vengano creati utilizzando classi di archiviazione in grado di supportare la modalità di accesso RWX.
4. Fare clic su **Azioni** nell'angolo destro e quindi selezionare **Migra**.
5. Per visualizzare l'avanzamento della migrazione, vai su Virtualizzazione > Panoramica nel menu a sinistra e poi clicca sulla scheda **Migrazioni**. La migrazione della VM passerà da **In sospeso** a **Pianificazione** a **Riuscita**



Un'istanza VM in un cluster OpenShift migra automaticamente a un altro nodo quando il nodo originale viene messo in modalità di manutenzione se evictionStrategy è impostato su LiveMigrate.



Migrazione dello storage delle VM tra classi di storage in OpenShift Virtualization

In OpenShift Virtualization, puoi migrare i dischi di una macchina virtuale (VM) a una classe di storage diversa direttamente tramite la console web senza eliminare la VM. Questo processo ti permette di spostare l'intero storage della VM o volumi specifici.

I seguenti passaggi illustrano come migrare la storage class di una VM:

1. Assicurati che la classe di storage di destinazione sia configurata correttamente e disponibile nel tuo OpenShift cluster. Questo include la configurazione e l'accessibilità del backend di storage necessario (ad esempio, NetApp ONTAP).

Mostra esempio

```
[root@00-50-56-b5-2b-9b ~]# oc get tbe -n trident
NAME          BACKEND          BACKEND UUID
tbe-k2vqg     tbc-san         3e9dc8ae-3b29-44c8-99a9-affdeaa6b4cc
tbe-mtp5n     tbc-nas-eco     a59e8053-bb00-4410-9b12-1dc4d283928b
tbe-nwqvq     tbc-nas-50      8b49f02f-3bc3-4720-b60f-bf90c3600008
tbe-wtp9h     tbc_san_eco     61745526-04cd-4acb-8f1b-daa9aa79f05a
tbe-zfp46     tbc-nas         259a1c2a-0a84-488d-ab62-ccea45499e09
```

```
[root@00-50-56-b5-2b-9b ~]# oc get sc
NAME          PROVISIONER          RECLAIMPOLICY  VOLUMEBINDINGMODE  ALLOWVOLUMEEXPANSION  AGE
sc-nas (default)  csi.trident.netapp.io  Delete         Immediate          true                  35d
sc-nas-50        csi.trident.netapp.io  Delete         Immediate          true                  22d
sc-nas-economy   csi.trident.netapp.io  Delete         Immediate          true                  20d
sc-san           csi.trident.netapp.io  Delete         Immediate          false                 21d
```

2. Nella console web di OpenShift Container Platform, vai alla prospettiva Virtualizzazione e seleziona la VM che vuoi migrare.

Mostra esempio

VirtualMachines > VirtualMachine details

VM
rhel9-sc-migration
Running

Overview
Metrics
YAML
Configuration
Events
Console
Snapshots
Diagnostics

Details

Name
rhel9-sc-migration

Status
Running

Created
May 14, 2020, 1:31 PM (17 minutes ago)

Operating system
Red Hat Enterprise Linux 9.7 (Plow)

CPU | Memory
1 CPU | 2 GiB Memory

Time zone
EDT

Template
rhel9-server-small

Hostname
rhel9-sc-migration

VNC console

Open web console

Alerts (0)

General

Namespace
test-sc-migration

Node
00-50-56-b5-bd-fa

VirtualMachineInstance
rhel9-sc-migration

Pod
virt-launcher-rhel9-sc-migration-5jkdw

Owner
No owner

Snapshots (0)

Take snapshot

VirtualMachines > VirtualMachine details

VM
rhel9-sc-migration
Running

Overview
Metrics
YAML
Configuration
Events
Console
Snapshots
Diagnostics

Details

Storage

Add disk

Network

Filter
Search by name...
Mount Windows drivers disk

Scheduling

SSH

Initial run

Metadata

bootable

Disks

Name	Source	Size	Drive	Interface	Storage class
cloudinitdisk	Other	-	Disk	virtio	-
disk-rhel9-nas	PVC dv-rhel9-migration-disk-emerald-wren-75-d88usc	53 GiB	Disk	virtio	sc-nas
rootdisk	PVC rhel9-migration	31.8 GiB	Disk	virtio	sc-nas

```

[root@00-50-56-b5-2b-9b ~]# oc get vm,pvc -n test-sc-migration
NAME                                AGE    STATUS    READY
virtualmachine.kubevirt.io/rhel9-sc-migration 4m19s Running True

NAME                                STATUS    VOLUME                                CAPACITY    ACCESS MODES    STORAGECLASS    VOLUMEATTRIBUTESCLASS    AGE
persistentvolumeclaim/dv-rhel9-migration-disk-emerald-wren-75-d88usc Bound     pvc-bb54342f-c01f-453e-864d-e8f72e710a2f 53Gi        RWX              sc-nas          <unset>                  4m19s
persistentvolumeclaim/rhel9-migration Bound     pvc-46a34983-942f-4ddc-8e6d-974008f05d09 34144990004 34144990004    RWX              sc-nas          <unset>                  4m19s

```

- Fai clic su Azioni→Migra→Migra Storage. Seleziona i volumi da migrare e la classe di storage di destinazione, quindi fai clic su Migra. Il processo di migrazione inizierà e puoi monitorare l'avanzamento nella scheda Eventi della VM. Una volta completata la migrazione, i dischi della VM utilizzeranno la nuova classe di storage.

Mostra esempio

VirtualMachines > VirtualMachine details

VM **rhel9-sc-migration** Running

Overview

Metrics

YAML

Configuration

Events

Console

Snapshots

Diagnostics

Details

Name	rhel9-sc-migration
Status	Running
Created	May 14, 2026, 1:31 PM (19 minutes ago)
Operating system	Red Hat Enterprise Linux 9.7 (Plow)
CPU Memory	1 CPU 2 GiB Memory
Time zone	EDT
Template	rhel9-server-small
Hostname	rhel9-sc-migration

VNC console
[Open web console](#)

Alerts (0)	
General	Compute Migrate VirtualMachine to a different Node
Namespace	Storage Migrate VirtualMachine storage to a different StorageClass
Node	VirtualMachineInstance **rhel9-sc-migration**
Pod	virt-launcher-rhel9-sc-migration-5jkdw
Owner	No owner
Snapshots (0)	[Take snapshot](#)

Migrate VirtualMachine storage

Migrate VirtualMachine storage to a different StorageClass.

1 Migration details

2 Destination StorageClass

3 Review

Migration details

Select the storage to migrate for **rhel9-sc-migration**

☒ The entire VirtualMachine

☐ Selected volumes

Back

Next

Cancel

Migrate VirtualMachine storage

Migrate VirtualMachine storage to a different StorageClass.

1 Migration details

2 Destination StorageClass

3 Review

Destination StorageClass

Select the destination storage for the VirtualMachine storage migration.

sc-nas-economy

Back

Next

Cancel

56

×

Migrate VirtualMachine storage

Migrate VirtualMachine storage to a different StorageClass.

1 Migration details

2 Destination StorageClass

3 **Review**

Review

Verify the details and click **Migrate VirtualMachine storage** to start the migration

Migration type	VirtualMachine storage
Destination StorageClass	sc-nas-economy
Migrating 1 out of 3	disk-rhel9-nas

Back

Migrate VirtualMachine storage

Cancel

×

Migrate VirtualMachine storage

Migrate VirtualMachine storage to a different StorageClass.

In progress

Started on

 May 14, 2026, 1:54 PM

Status

Scheduling

Stop

Cancel

View report 

4. Al termine della migrazione, puoi verificare che i dischi della VM utilizzino ora la nuova storage class controllando i dettagli della VM nella console web o usando la OpenShift CLI.

57

Mostra esempio

```
[root@00-50-56-b5-2b-9b ~]# oc get vm,pvc -n test-sc-migration
NAME                                AGE    STATUS    READY
virtualmachine.kubevirt.io/rhel9-sc-migration    24m    Migrating True

NAME                                STATUS    VOLUME                                CAPACITY    ACCESS MODES    STORAGECLASS    VOLUMEATTRIBUTESCLASS    AGE
persistentvolumeclaim/dv-rhel9-migration-disk-emerald-wren-75-d88usc    Bound    pvc-bb54342f-c01f-453e-864d-e8f72e710a2f    53Gi    RWX    sc-nas    <unset>    24m
persistentvolumeclaim/dv-rhel9-migration-disk-emerald-wren-75-d88usc-mig-839zfb    Bound    pvc-75a885f0-824e-407c-ab29-8cc546dbdb3b    60322815673    RWX    sc-nas-economy    <unset>    2m17s
persistentvolumeclaim/rhel9-migration    Bound    pvc-46a34903-942f-4ddc-8e6d-974008f05d09    34144990004    RWX    sc-nas    <unset>    24m
[root@00-50-56-b5-2b-9b ~]#
```

```
[root@00-50-56-b5-2b-9b ~]# oc get all -n test-sc-migration
Warning: apps.openshift.io/v1 DeploymentConfig is deprecated in v4.14+, unavailable in v4.10000+
Warning: kubevirt.io/v1 VirtualMachineInstancePresets is now deprecated and will be removed in v2.
NAME                                READY    STATUS    RESTARTS    AGE
pod/virt-launcher-rhel9-sc-migration-4hd7    1/1    Running    0    2m50s
pod/virt-launcher-rhel9-sc-migration-5jkdw    0/1    Completed    0    25m

NAME                                TYPE    CLUSTER-IP    EXTERNAL-IP    PORT(S)    AGE
service/headless                    ClusterIP    None    <none>    5434/TCP    25m

NAME                                PHASE    PROGRESS    RESTARTS    AGE
datavolume.cdi.kubevirt.io/dv-rhel9-migration-disk-emerald-wren-75-d88usc    Succeeded    100.0%    0    25m
datavolume.cdi.kubevirt.io/dv-rhel9-migration-disk-emerald-wren-75-d88usc-mig-839zfb    Succeeded    100.0%    0    3m7s
datavolume.cdi.kubevirt.io/rhel9-migration    Succeeded    100.0%    0    25m

NAME                                PHASE    VMI
virtualmachineinstancemigration.kubevirt.io/kubevirt-workload-update-lbfj9    Succeeded    rhel9-sc-migration

NAME                                AGE    PHASE    IP    NODENAME    READY
virtualmachineinstance.kubevirt.io/rhel9-sc-migration    25m    Running    10.131.0.22    00-50-56-b5-84-74    True

NAME                                AGE    STATUS    READY
virtualmachine.kubevirt.io/rhel9-sc-migration    25m    Running    True
[root@00-50-56-b5-2b-9b ~]#
[root@00-50-56-b5-2b-9b ~]#
[root@00-50-56-b5-2b-9b ~]# oc get vm,pvc -n test-sc-migration
NAME                                AGE    STATUS    READY
virtualmachine.kubevirt.io/rhel9-sc-migration    26m    Running    True

NAME                                STATUS    VOLUME                                CAPACITY    ACCESS MODES    STORAGECLASS    VOLUMEATTRIBUTESCLASS    AGE
persistentvolumeclaim/dv-rhel9-migration-disk-emerald-wren-75-d88usc    Bound    pvc-bb54342f-c01f-453e-864d-e8f72e710a2f    53Gi    RWX    sc-nas    <unset>    26m
persistentvolumeclaim/dv-rhel9-migration-disk-emerald-wren-75-d88usc-mig-839zfb    Bound    pvc-75a885f0-824e-407c-ab29-8cc546dbdb3b    60322815673    RWX    sc-nas-economy    <unset>    3m28s
persistentvolumeclaim/rhel9-migration    Bound    pvc-46a34903-942f-4ddc-8e6d-974008f05d09    34144990004    RWX    sc-nas    <unset>    26m
[root@00-50-56-b5-2b-9b ~]#
```

```
[root@00-50-56-b5-2b-9b ~]# oc describe pv pvc-46a34903-942f-4ddc-8e6d-974008f05d09 -n test-sc-migration
Name:                pvc-46a34903-942f-4ddc-8e6d-974008f05d09
Labels:              <none>
Annotations:         pv.kubernetes.io/provisioned-by: csi.trident.netapp.io
                    volume.kubernetes.io/provisioner-deletion-secret-name:
                    volume.kubernetes.io/provisioner-deletion-secret-namespace:
Finalizers:          [external-provisioner.volume.kubernetes.io/finalizer kubernetes.io/pv-protection external-attacher/csi-trident-netapp-io]
StorageClass:        sc-nas
Status:              Bound
Claim:               test-sc-migration/rhel9-migration
Reclaim Policy:      Delete
Access Modes:        RWX
VolumeMode:          Filesystem
Capacity:            34144990004
Node Affinity:       <none>
Message:
Source:
  Type:               CSI (a Container Storage Interface (CSI) volume source)
  Driver:             csi.trident.netapp.io
  FSType:
  VolumeHandle:       pvc-46a34903-942f-4ddc-8e6d-974008f05d09
  ReadOnly:           false
  VolumeAttributes:    backendUUID=259a1c2a-0a84-488d-ab62-ccea45499e09
                    internalName=oc_automation_pvc_46a34903_942f_4ddc_8e6d_974008f05d09
                    name=pvc-46a34903-942f-4ddc-8e6d-974008f05d09
                    protocol=file
                    storage.kubernetes.io/csiProvisionerIdentity=1778505459781-8296-csi.trident.netapp.io
Events:              <none>
[root@00-50-56-b5-2b-9b ~]#
```

```

NSOL-NetApp-A70-T19U01:> volume show -vserver oc-automation-svm -volume oc_automation_pvc_46a34903_942f_4ddc_8e6d_974008f05d09

Vserver Name: oc-automation-svm
Volume Name: oc_automation_pvc_46a34903_942f_4ddc_8e6d_974008f05d09
Aggregate Name: NSOL_NetApp_A70_T19U01a_NVME_SSD_1
List of Aggregates for FlexGroup Constituents: NSOL_NetApp_A70_T19U01a_
NVME_SSD_1
Encryption Type: none
List of Nodes Hosting the Volume: NSOL-NetApp-A70-T19U01a
Volume Size: 31.80GB
Volume Data Set ID: 1716
Volume Master Data Set ID: 2152408880
Volume State: online
Volume Style: flex
Extended Volume Style: flexvol
FlexCache Endpoint Type: none
Is Cluster-Mode Volume: true
Is Constituent Volume: false
Number of Constituent Volumes: -
Export Policy: default
User ID: 0
Group ID: 0
Security Style: unix
UNIX Permissions: ---rwxrwxrwx
Junction Path: /oc_automation_pvc_46a34903_942f_4ddc_8e6d_974008f05d09
Junction Path Source: RW_volume
Junction Active: true
Junction Parent Volume: oc_automation-svm_root
Comment:
Available Size: 30.03GB
Filesystem Size: 31.80GB
Total User-Visible Size: 31.80GB
Used Size: 1.76GB
Used Percentage: 5%
Volume Nearly Full Threshold Percent: 95%
Volume Full Threshold Percent: 98%
Maximum Autosize: 38.16GB
Minimum Autosize: 31.80GB

```

```

[root@00-50-56-b5-2b-9b ~]# oc get all -n test-sc-migration
Warning: apps.openshift.io/v1 DeploymentConfig is deprecated in v4.14+, unavailable in v4.10000+
Warning: kubevirt.io/v1 VirtualMachineInstancePresets is now deprecated and will be removed in v2.
NAME                                READY    STATUS    RESTARTS   AGE
pod/virt-launcher-rhel9-sc-migration-44dc7    1/1      Running    0           2m50s
pod/virt-launcher-rhel9-sc-migration-5jkdw    0/1      Completed  0           25m

NAME                TYPE          CLUSTER-IP    EXTERNAL-IP    PORT(S)    AGE
service/headless    ClusterIP     None           <none>          5434/TCP    25m

NAME                PHASE    PROGRESS    RESTARTS   AGE
datavolume.cdi.kubevirt.io/dv-rhel9-migration-disk-emerald-wren-75-d88usc    Succeeded    100.0%      25m
datavolume.cdi.kubevirt.io/dv-rhel9-migration-disk-emerald-wren-75-d88usc-mig-839zfb    Succeeded    100.0%      3m7s
datavolume.cdi.kubevirt.io/rhel9-migration    Succeeded    100.0%      25m

NAME                PHASE    VMI
virtualmachineinstancemigration.kubevirt.io/kubevirt-workload-update-lbfj9    Succeeded    rhel9-sc-migration

NAME                AGE    PHASE    IP            NODENAME    READY
virtualmachineinstance.kubevirt.io/rhel9-sc-migration    25m    Running    10.131.0.22    00-50-56-b5-84-74    True

NAME                AGE    STATUS    READY
virtualmachine.kubevirt.io/rhel9-sc-migration    25m    Running    True
[root@00-50-56-b5-2b-9b ~]#
[root@00-50-56-b5-2b-9b ~]#
[root@00-50-56-b5-2b-9b ~]# oc get vm,pvc -n test-sc-migration
NAME                AGE    STATUS    READY
virtualmachine.kubevirt.io/rhel9-sc-migration    26m    Running    True

NAME                STATUS    VOLUME    CAPACITY    ACCESS MODES    STORAGECLASS    VOLUMEATTRIBUTESCLASS    AGE
persistentvolumeclaim/dv-rhel9-migration-disk-emerald-wren-75-d88usc    Bound    pvc-bb5d342f-c01f-453e-86ad-abf72e710a2f    536i    Rwx    sc-nas    <unset>    26m
persistentvolumeclaim/dv-rhel9-migration-disk-emerald-wren-75-d88usc-mig-839zfb    Bound    pvc-75a885f0-824e-407c-ab29-8cc546dbdb3b    60322815673    Rwx    sc-nas-economy    <unset>    3m28s
persistentvolumeclaim/rhel9-migration    Bound    pvc-46a34903-942f-4ddc-8e6d-974008f05d09    34144990004    Rwx    sc-nas    <unset>    26m
[root@00-50-56-b5-2b-9b ~]#

```

```

[root@00-50-56-b5-2b-9b ~]# oc describe pv -n test-sc-migration pvc-75a885f0-824e-407c-ab29-8cc546dbdb3b
Name:          pvc-75a885f0-824e-407c-ab29-8cc546dbdb3b
Labels:        <none>
Annotations:   pv.kubernetes.io/provisioned-by: csi.trident.netapp.io
               volume.kubernetes.io/provisioner-deletion-secret-name:
               volume.kubernetes.io/provisioner-deletion-secret-namespace:
Finalizers:    [external-provisioner.volume.kubernetes.io/finalizer kubernetes.io/pv-protection external-attacher/csi-trident-netapp-io]
StorageClass:  sc-nas-economy
Status:        Bound
Claim:         test-sc-migration/dv-rhel9-migration-disk-emerald-wren-75-d88usc-mig-839zfb
Reclaim Policy: Delete
Access Modes:  Rwx
Volume Mode:   Filesystem
Capacity:      60322815673
Node Affinity: <none>
Message:
Source:
  Type:        CSI (a Container Storage Interface (CSI) volume source)
  Driver:      csi.trident.netapp.io
  FSType:
  VolumeHandle: pvc-75a885f0-824e-407c-ab29-8cc546dbdb3b
  ReadOnly:    false
  VolumeAttributes:
    backendUUID=a59e8053-bb00-4410-9b12-1dc4d283928b
    internalID=/svm/oc-automation-svm/flexvol/trident_qtree_pool_oc_automation_OGBGMIAIOLG/qtree/oc_automation_pvc_75a885f0_824e_407c_ab29_8cc546...
    internalName=oc_automation_pvc_75a885f0_824e_407c_ab29_8cc546dbdb3b
    name=pvc-75a885f0-824e-407c-ab29-8cc546dbdb3b
    protocol=file
    storage.kubernetes.io/csiProvisionerIdentity=1778505459781-8296-csi.trident.netapp.io
Events:        <none>

```

Qtrees

+ Add

🔍 ⬇️ 🗃️ ☰

☐	Name	Storage VM	Volume name	Mount path
	oc_automation_pvc_75a885f0_824e_407c_ab29_8ccf			
▼	oc_automation_pvc_75a885f0_824e_407c_ab29_8cc546dbdb3b	oc-automation-svm	trident_qtree_pool_oc_automation_OGBGMIAOLG	/trident_qtree_pool_oc_automation_OGBGMIAOLG/oc_...

Informazioni sul copyright

Copyright © 2026 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 COMMERCIALIZZABILITÀ 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 CONSEGUENZIALE (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.