



Installa Trident sul cluster Red Hat OpenShift e crea oggetti di archiviazione

NetApp virtualization solutions

NetApp
August 18, 2025

Sommario

- Installa Trident sul cluster Red Hat OpenShift e crea oggetti di archiviazione 1
 - Dimostrazione video 6
 - Configurazione Trident per cluster OpenShift on-prem 6
 - Configurazione Trident per cluster ROSA utilizzando storage FSxN 11
 - Creazione della classe Snapshot del volume Trident 12
 - Impostazione dei valori predefiniti con Trident Storage e Snapshot Class 13

Installa Trident sul cluster Red Hat OpenShift e crea oggetti di archiviazione

Installa Trident utilizzando Red Hat Certified Trident Operator sui cluster OpenShift e prepara i nodi worker per l'accesso ai blocchi. Crea oggetti backend e classi di archiviazione Trident per l'archiviazione ONTAP e FSxN per abilitare il provisioning dinamico dei volumi per container e VM.



Se è necessario creare VM in OpenShift Virtualization, è necessario installare Trident e creare gli oggetti backend e gli oggetti della classe di archiviazione in OpenShift Cluster prima di installare OpenShift Virtualization sul cluster (in locale e ROSA). La classe di archiviazione predefinita e la classe di snapshot del volume predefinita devono essere impostate sull'archiviazione Trident e sulla classe di snapshot nel cluster. Solo se questa opzione è configurata, OpenShift Virtualization può rendere le immagini golden disponibili localmente per la creazione di VM tramite modelli.



Se l'operatore OpenShift Virtualization è installato prima di installare Trident, è possibile utilizzare il seguente comando per eliminare le immagini golden create utilizzando una classe di archiviazione diversa e quindi consentire a OpenShift Virtualization di creare le immagini golden utilizzando la classe di archiviazione Trident, assicurandosi che siano impostati i valori predefiniti delle classi di archiviazione Trident e snapshot del volume.

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



Per ottenere file yaml di esempio per creare oggetti trident per l'archiviazione FSxN per cluster ROSA e per ottenere file yaml di esempio per VolumeSnapshotClass, scorrere verso il basso in questa pagina.

Installazione di Trident

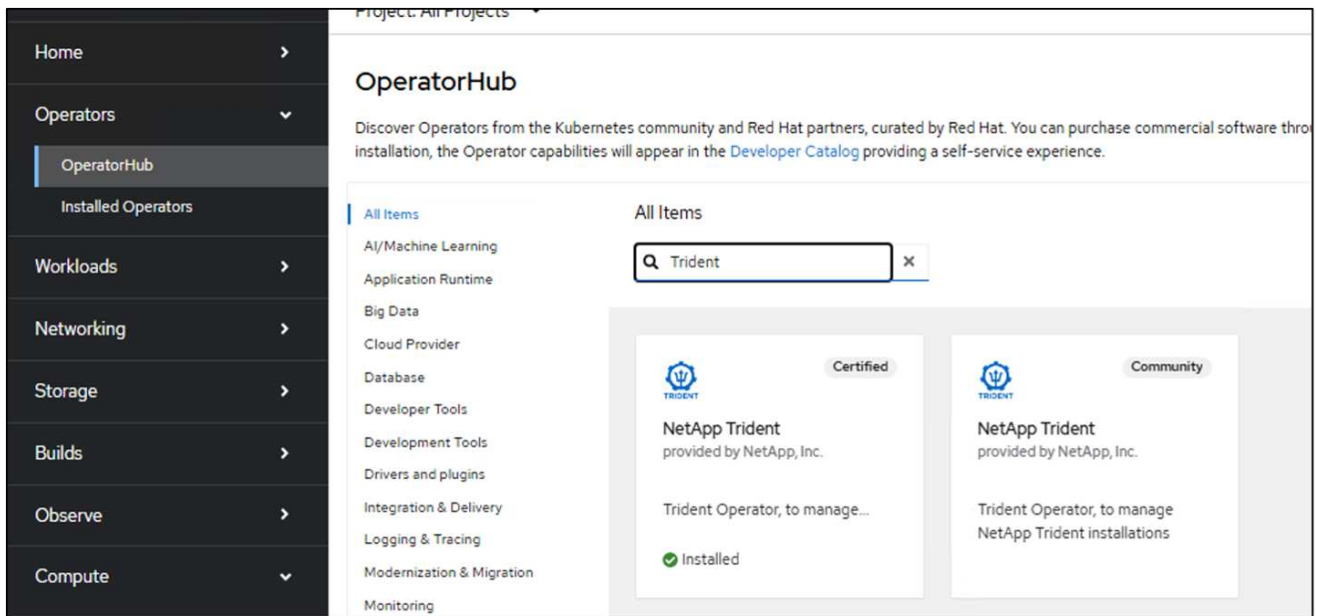
Installazione di Trident tramite Red Hat Certified Operator

In questa sezione vengono forniti i dettagli sull'installazione Trident utilizzando Red Hat Certified Trident Operator. [Fare riferimento alla documentazione Trident](#) per altri modi per installare Trident. Con il rilascio di Trident 25.02, gli utenti di Trident in Red Hat OpenShift on premises e nel cloud e nei servizi gestiti come Red Hat OpenShift Service su AWS possono ora installare Trident utilizzando Trident Certified Operator dall'Operator Hub. Ciò è significativo per la comunità di utenti di OpenShift, poiché in precedenza Trident era disponibile solo come operatore della comunità.

Il vantaggio dell'operatore Red Hat Certified Trident è che la base per l'operatore e i suoi container è completamente supportata da NetApp quando utilizzata con OpenShift (sia in locale, nel cloud o come servizio gestito con ROSA). Inoltre, NetApp Trident non ha alcun costo per il cliente, quindi tutto ciò che devi fare è installarlo utilizzando l'operatore certificato che è stato verificato per funzionare senza problemi con Red Hat OpenShift e confezionato per una facile gestione del ciclo di vita.

Inoltre, l'operatore Trident 25.02 (e le versioni future) offre il vantaggio opzionale di preparare i nodi worker per iSCSI. Ciò è particolarmente vantaggioso se si prevede di distribuire i carichi di lavoro su cluster ROSA e si intende utilizzare il protocollo iSCSI con FSxN, in particolare per i carichi di lavoro delle VM di OpenShift Virtualization. La sfida della preparazione dei nodi worker per iSCSI sui cluster ROSA utilizzando FSxN è stata mitigata grazie a questa funzionalità durante l'installazione Trident sul cluster.

I passaggi di installazione tramite l'operatore sono gli stessi, indipendentemente dal fatto che l'installazione avvenga su un cluster locale o su ROSA. Per installare Trident tramite Operator, fare clic sull'hub Operator e selezionare Certified NetApp Trident. Nella pagina Installa, per impostazione predefinita è selezionata la versione più recente. Fare clic su Installa.



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 *

 openshift-operators

Update approval * ⓘ

☒ Automatic

☐ Manual

Install

Cancel

Una volta installato l'operatore, fare clic su Visualizza operatore e quindi creare un'istanza di Trident Orchestrator. Se si desidera preparare i nodi worker per l'accesso all'archiviazione iSCSI, andare alla vista yaml e modificare il parametro nodePrep aggiungendo iscsi.

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
```

Ora dovresti avere tutti i pod Trident in esecuzione nel tuo cluster.

```
[root@localhost ~]# oc get pods -n trident
NAME                                READY   STATUS    RESTARTS   AGE
trident-controller-84cb9bff89-1kx6k 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 ~]#
```

Per verificare che gli strumenti iSCSI siano stati abilitati sui nodi worker dell'OpenShift Cluster, accedi ai nodi worker e verifica di vedere iscsid, multipathd active e le voci nel file multipath.conf come mostrato.

```
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#
```

```
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 Red Hat Certified Trident Operator

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

Configurazione Trident per cluster OpenShift on-prem

Backend Trident e classe di archiviazione per NAS

```
cat 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: <cluster management lif>
  backendName: tbc-nas
  svm: zoneb
  storagePrefix: testzoneb
  defaults:
    nameTemplate: "{{ .config.StoragePrefix }}_{{ .volume.Namespace
  }}_{{ .volume.RequestName }}"
  credentials:
    name: tbc-nas-secret
```

```
cat 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
```

Backend Trident e classe di archiviazione per iSCSI

```
# cat 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
```

```
# cat 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
```

Backend Trident e classe di archiviazione per NVMe/TCP

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

```
# cat 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
```

Backend Trident e classe di archiviazione per FC

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

```
# cat 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
```

Configurazione Trident per cluster ROSA utilizzando storage FSxN

Backend Trident e classe di archiviazione per NAS FSxN

```
#cat tbc-fsx-nas.yaml
apiVersion: v1
kind: Secret
metadata:
  name: backend-fsx-ontap-nas-secret
  namespace: trident
type: Opaque
stringData:
  username: <cluster admin lif>
  password: <cluster admin passwd>
---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-fsx-ontap-nas
  namespace: trident
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
```

```
# cat sc-fsx-nas.yaml
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
```

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

```
# cat 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
```

Creazione della classe Snapshot del volume Trident

Classe di snapshot del volume Trident

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

Una volta che hai i file yaml necessari per la configurazione del backend, la configurazione della classe di archiviazione e le configurazioni degli snapshot, puoi creare gli oggetti backend trident, classe di archiviazione e classe snapshot utilizzando il seguente comando

```
oc create -f <backend-filename.yaml> -n trident
oc create -f <storageclass-filename.yaml>
oc create -f <snapshotclass-filename.yaml>
```

Impostazione dei valori predefiniti con Trident Storage e Snapshot Class

Impostazione dei valori predefiniti con Trident Storage e Snapshot Class

Ora è possibile impostare la classe di archiviazione Trident e la classe di snapshot del volume come predefinite in OpenShift Cluster. Come accennato in precedenza, è necessario impostare la classe di archiviazione predefinita e la classe di snapshot del volume per consentire a OpenShift Virtualization di rendere disponibile la sorgente dell'immagine golden per creare macchine virtuali da modelli predefiniti.

È possibile impostare la classe di archiviazione Trident e la classe snapshot come predefinite modificando l'annotazione dalla console o applicando la patch dalla riga di comando con quanto segue.

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

storageclass.kubevirt.io/is-default-virt-class: true
or
kubectl patch storageclass standard -p '{"metadata": {"annotations":{"storageclass.kubevirt.io/is-default-virt-class": "true"}}}'
```

Una volta impostata questa opzione, è possibile eliminare tutti gli oggetti dv e VolumeSnapshot preesistenti utilizzando il seguente comando:

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


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.