



在 Red Hat OpenShift 叢集上安裝 Trident並建立儲存對象 NetApp virtualization solutions

NetApp
August 18, 2025

This PDF was generated from <https://docs.netapp.com/zh-tw/netapp-solutions-virtualization/openshift/osv-trident-install.html> on January 12, 2026. Always check docs.netapp.com for the latest.

目錄

在 Red Hat OpenShift 叢集上安裝Trident並建立儲存對象	1
影片示範	6
本地 OpenShift 叢集的Trident配置	6
使用 FSxN 儲存的 ROSA 叢集的Trident配置	11
建立Trident磁碟區快照類	12
使用Trident Storage 和 Snapshot Class 設定預設值	13

在 Red Hat OpenShift 叢集上安裝Trident並建立儲存對象

使用 Red Hat 認證的Trident Operator 在 OpenShift 叢集上安裝Trident，並準備工作節點以進行區塊存取。為ONTAP和 FSxN 儲存建立Trident後端和儲存類別對象，以實現容器和虛擬機器的動態磁碟區配置。



如果您需要在 OpenShift Virtualization 中建立虛擬機，則必須安裝Trident，並且必須在叢集（本機和 ROSA）上安裝 OpenShift Virtualization 之前在 openShift 叢集中建立後端物件和儲存類別物件。預設儲存類別和預設磁碟區快照類別必須設定為叢集中的Trident儲存和快照類別。只有在配置完成後，OpenShift Virtualization 才能在本機提供黃金映像，以便使用範本建立 VM。



如果在安裝Trident之前安裝了 OpenShift Virtualization Operator，則可以使用以下命令刪除使用不同儲存類別建立的黃金映像，然後透過確保設定了Trident Storage 和 Volume Snapshot 類別預設值，讓 OpenShift Virtualization 使用Trident儲存類別建立黃金映像。

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



若要取得範例 yaml 檔案來為 ROSA 叢集的 FSxN 儲存建立 trident 對象，以及取得 VolumeSnapshotClass 的範例 yaml 文件，請向下捲動此頁面。

安裝Trident

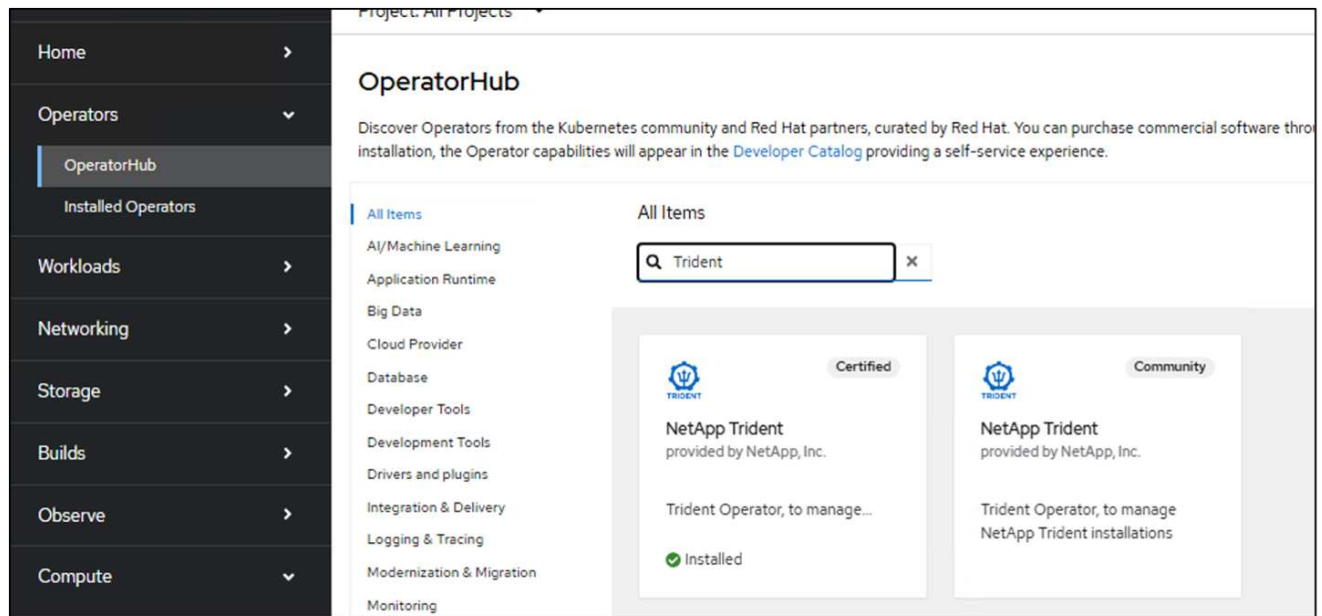
本節將提供使用 Red Hat 認證Trident Operator 安裝Trident的詳細信息"請參閱Trident文檔"了解安裝Trident的其他方法。隨著Trident 25.02 的發布，Red Hat OpenShift 本地和雲端的Trident用戶以及 AWS 上的 Red Hat OpenShift Service 等託管服務現在可以使用來自 Operator Hub 的Trident Certified Operator 安裝Trident。這對 OpenShift 用戶社群來說意義重大，因為Trident之前僅作為社群營運商提供。

Red Hat 認證Trident操作員的優勢在於，當與 OpenShift 一起使用時（無論是在本地、在雲端，還是作為 ROSA 的託管服務），操作員及其容器的基礎完全受到NetApp的支援。此外，NetApp Trident對客戶免費，因此您只需使用經過驗證可與 Red Hat OpenShift 無縫協作且打包以便於生命週期管理的認證操作員進行安裝即可。

此外，Trident 25.02 操作員（以及未來版本）提供了為 iSCSI 準備工作節點的可選好處。如果您打算在 ROSA 叢集上部署工作負載並打算將 iSCSI 協定與 FSxN 一起使用，這尤其有利，尤其是對於 OpenShift Virtualization VM 工作負載。在叢集上安裝Trident時，此功能減輕了使用 FSxN 在 ROSA 叢集上為 iSCSI 準備工作節點的挑戰。

無論您是在本機叢集還是在 ROSA 上安裝，使用操作員的安裝步驟都是相同的。若要使用 Operator 安裝Trident，請按一下 Operator hub 並選擇 Certified NetApp Trident。在安裝頁面中，預設選擇最新版本。按一下“安裝”

。



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

安裝操作員後，按一下查看操作員，然後建立Trident Orchestrator 的實例。如果要為 iSCSI 儲存存取準備工作節點，請前往 yaml 視圖並透過新增 iscsi 來修改 nodePrep 參數。

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
```

現在，所有 trident pod 都應該在叢集中執行。

```
[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 ~]#
```

若要驗證 OpenShift 叢集的工作節點上是否已啟用 iSCSI 工具，請登入工作節點並驗證您是否看到 iscsid、multipathd active 和 multipath.conf 檔案中的項目，如圖所示。

```

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#
```

影片示範

以下影片示範如何使用 Red Hat 認證Trident Operator 安裝Trident

在 [OpenShift](#) 中使用經過認證的Trident Operator 安裝Trident 25.02.1

本地 OpenShift 叢集的Trident配置


```
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
```

```
# 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
```

```
# 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
```

```
# 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
```

使用 FSxN 儲存的 ROSA 叢集的Trident配置

FSxN NAS 的Trident後端和儲存類

```
#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
```

建立Trident磁碟區快照類

Trident卷快照類

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

準備好後端設定、儲存類別配置和快照配置所需的 yaml 檔案後，您可以使用下列命令建立 trident 後端、儲存類別和快照類別對象

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

使用Trident Storage 和 Snapshot Class 設定預設值

使用Trident Storage 和 Snapshot Class 設定預設值

現在，您可以將所需的 trident 儲存類別和磁碟區快照類別設為 OpenShift 叢集中的預設類別。如前所述，需要設定預設儲存類別和磁碟區快照類，以允許 OpenShift Virtualization 使黃金映像來源可用於從預設範本建立虛擬機器。

您可以從控制台編輯註解或從命令列進行修補，將Trident儲存類別和快照類別設定為預設值。

```
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"}}}'
```

設定完成後，您可以使用以下命令刪除任何預先存在的 dv 和 VolumeSnapshot 物件：

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


版權資訊

Copyright © 2026 NetApp, Inc. 版權所有。台灣印製。非經版權所有人事先書面同意，不得將本受版權保護文件的任何部分以任何形式或任何方法（圖形、電子或機械）重製，包括影印、錄影、錄音或儲存至電子檢索系統中。

由 NetApp 版權資料衍伸之軟體必須遵守下列授權和免責聲明：

此軟體以 NETAPP「原樣」提供，不含任何明示或暗示的擔保，包括但不限於有關適售性或特定目的適用性之擔保，特此聲明。於任何情況下，就任何已造成或基於任何理論上責任之直接性、間接性、附隨性、特殊性、懲罰性或衍生性損害（包括但不限於替代商品或服務之採購；使用、資料或利潤上的損失；或企業營運中斷），無論是在使用此軟體時以任何方式所產生的契約、嚴格責任或侵權行為（包括疏忽或其他）等方面，NetApp 概不負責，即使已被告知有前述損害存在之可能性亦然。

NetApp 保留隨時變本文所述之任何產品的權利，恕不另行通知。NetApp 不承擔因使用本文所述之產品而產生的責任或義務，除非明確經過 NetApp 書面同意。使用或購買此產品並不會在依據任何專利權、商標權或任何其他 NetApp 智慧財產權的情況下轉讓授權。

本手冊所述之產品受到一項（含）以上的美國專利、國外專利或申請中專利所保障。

有限權利說明：政府機關的使用、複製或公開揭露須受 DFARS 252.227-7013（2014 年 2 月）和 FAR 52.227-19（2007 年 12 月）中的「技術資料權利 - 非商業項目」條款 (b)(3) 小段所述之限制。

此處所含屬於商業產品和 / 或商業服務（如 FAR 2.101 所定義）的資料均為 NetApp, Inc. 所有。根據本協議提供的所有 NetApp 技術資料和電腦軟體皆屬於商業性質，並且完全由私人出資開發。美國政府對於該資料具有非專屬、非轉讓、非轉授權、全球性、有限且不可撤銷的使用權限，僅限於美國政府為傳輸此資料所訂合約所允許之範圍，並基於履行該合約之目的方可使用。除非本文另有規定，否則未經 NetApp Inc. 事前書面許可，不得逕行使用、揭露、重製、修改、履行或展示該資料。美國政府授予國防部之許可權利，僅適用於 DFARS 條款 252.227-7015(b)（2014 年 2 月）所述權利。

商標資訊

NETAPP、NETAPP 標誌及 <http://www.netapp.com/TM> 所列之標章均為 NetApp, Inc. 的商標。文中所涉及的所有其他公司或產品名稱，均為其各自所有者的商標，不得侵犯。