在 Red Hat OpenShift 集群上安装 Trident并创建存储对象

NetApp virtualization solutions

NetApp
August 18, 2025

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

目录

在 Red Hat OpenShift 集群上安装Trident并创建存储对象

使用 Red Hat 认证的Trident Operator 在 OpenShift 集群上安装Trident ，并准备工作节点以进行块访问。为ONTAP和 FSxN 存储创建Trident后端和存储类对象，以实现容器和虚拟机的动态卷配置。

如果您需要在 OpenShift Virtualization 中创建虚拟机，则必须安装Trident ，并且必须在集群（本地和 ROSA）上安装 OpenShift Virtualization 之前在 openShift 集群中创建后端对象和存储类对象。默认存储类和默认卷快照类必须设置为集群中的Trident存储和快照类。仅当配置完成后，OpenShift Virtualization 才能在本地提供黄金映像，以便使用模板创建 VM。

如果在安装Trident之前安装了 OpenShift Virtualization Operator，则可以使用以下命令删除使用不同存储类创建的黄金映像，然后通过确保设置了Trident Storage 和 Volume Snapshot 类默认值，让 OpenShift Virtualization 使用Trident存储类创建黄金映像。

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

要获取示例 yaml 文件来为 ROSA 集群的 FSxN 存储创建 trident 对象，以及获取 VolumeSnapshotClass 的示例 yaml 文件，请向下滚动此页面。

安装**Trident**

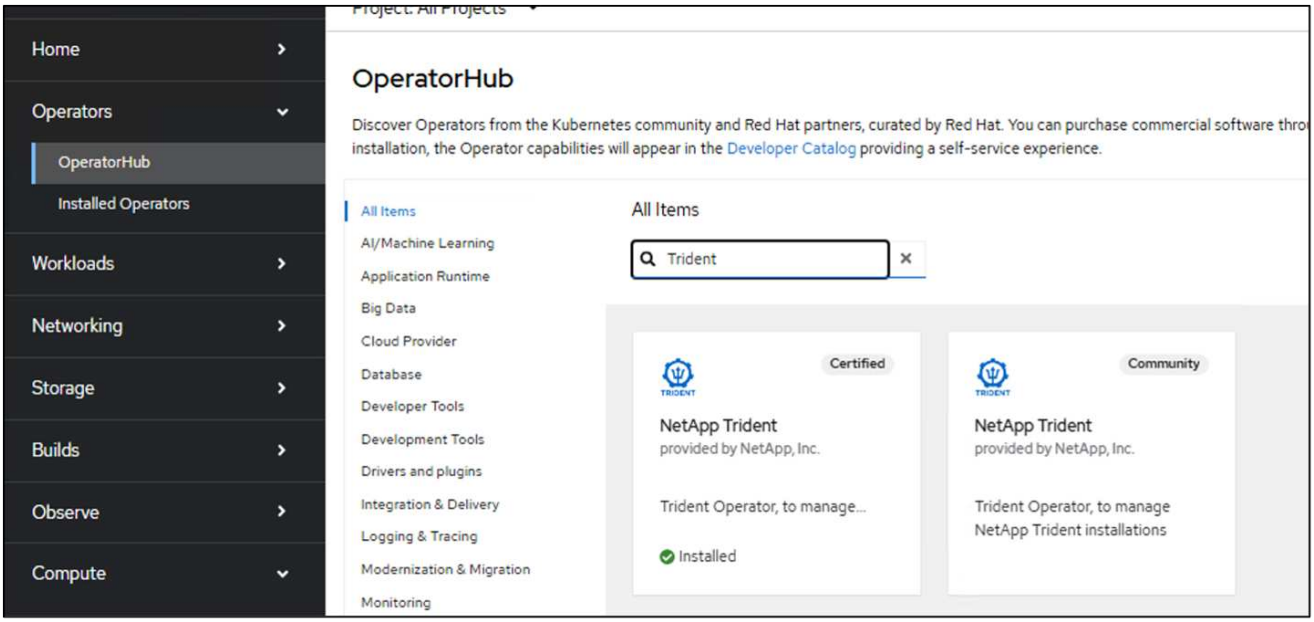
使用 **Red Hat** 认证操作员安装**Trident**

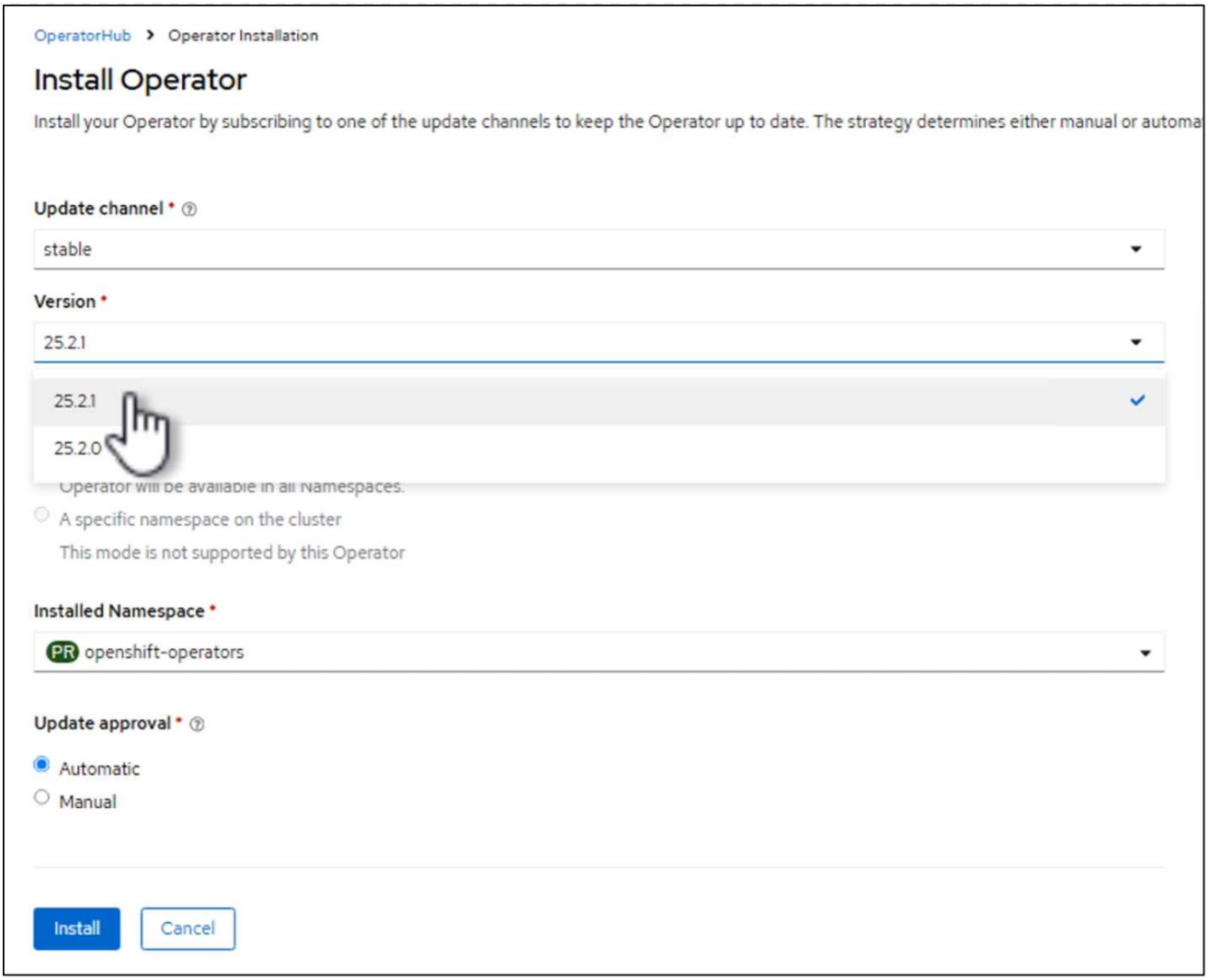
本节将提供使用 Red Hat 认证Trident Operator 安装Trident的详细信息"请参阅Trident文档"了解安装Trident的其他方法。随着Trident 25.02 的发布，Red Hat OpenShift 本地和云端的Trident用户以及 AWS 上的 Red Hat OpenShift Service 等托管服务现在可以使用来自 Operator Hub 的Trident Certified Operator 安装Trident 。这对于 OpenShift 用户社区来说意义重大，因为Trident之前仅作为社区运营商提供。

Red Hat 认证Trident操作员的优势在于，当与 OpenShift 一起使用时（无论是在本地、在云端，还是作为 ROSA 的托管服务），操作员及其容器的基础完全受到NetApp的支持。此外， NetApp Trident对客户免费，因此您只需使用经过验证可与 Red Hat OpenShift 无缝协作且打包以便于生命周期管理的认证操作员进行安装即可。

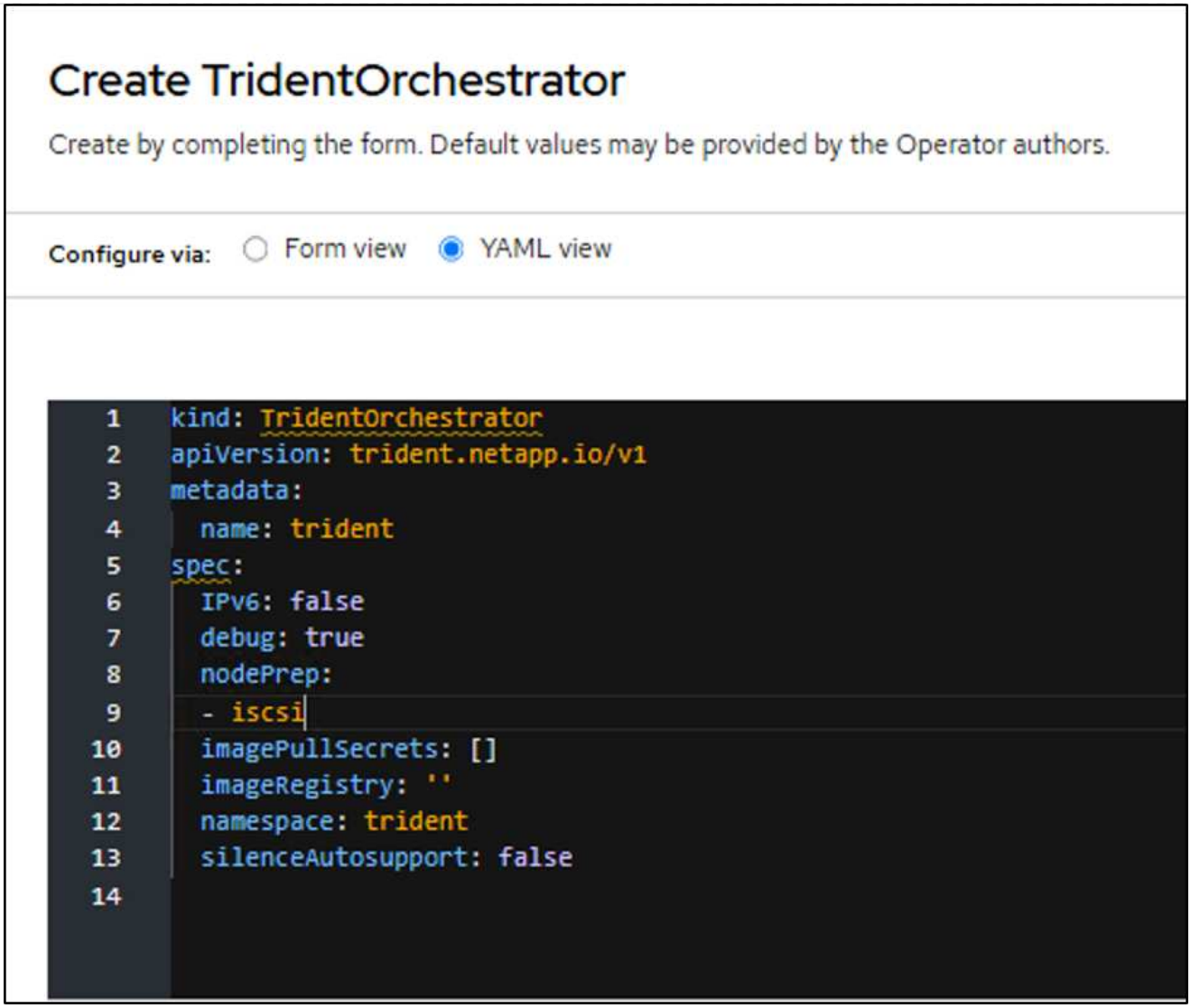
此外， Trident 25.02 操作员（以及未来版本）提供了为 iSCSI 准备工作节点的可选好处。如果您计划在 ROSA 集群上部署工作负载并打算将 iSCSI 协议与 FSxN 一起使用，这尤其有利，尤其是对于 OpenShift Virtualization VM 工作负载。在集群上安装Trident时，此功能减轻了使用 FSxN 在 ROSA 集群上为 iSCSI 准备工作节点的挑战。

无论您是在本地集群还是在 ROSA 上安装，使用操作员的安装步骤都是相同的。要使用 Operator 安装Trident ，请单击 Operator hub 并选择 Certified NetApp Trident。在安装页面中，默认选择最新版本。单击“安装”
。





安装操作员后，单击查看操作员，然后创建Trident Orchestrator 的实例。如果要为 iSCSI 存储访问准备工作节点，请转到 yaml 视图并通过添加 iscsi 来修改 nodePrep 参数。



现在，所有 trident pod 都应该在集群中运行。

```
[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-mkmmp              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 ~]#
```

要验证 OpenShift 集群的工作节点上是否已启用 iSCSI 工具，请登录工作节点并验证您是否看到 iscsid 、multipathd active 以及 multipath.conf 文件中的条目，如图所示。

```
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#
```

视频演示

以下视频演示了如何使用 Red Hat 认证Trident Operator 安装Trident

在 OpenShift 中使用经过认证的Trident Operator 安装Trident 25.02.1

本地 OpenShift 集群的Trident配置

Trident后端和 NAS 存储类

```
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
```

用于 **iSCSI** 的**Trident**后端和存储类

```
# 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
```

NVMe/TCP 的Trident后端和存储类

```
# 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
```

FC 的Trident后端和存储类

```
# 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 存储的 ROSA 集群的Trident配置

FSxN NAS 的Trident后端和存储类

```
#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
```

FSxN iSCSI 的Trident后端和存储类

```
# 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
```

创建Trident卷快照类

Trident卷快照类

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

准备好后端配置、存储类配置和快照配置所需的 yaml 文件后，您可以使用以下命令创建 trident 后端、存储类和快照类对象

```
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 设置默认值

使用**Trident Storage** 和 **Snapshot Class** 设置默认值

现在，您可以将所需的 trident 存储类和卷快照类设为 OpenShift 集群中的默认类。如前所述，需要设置默认存储类和卷快照类，以允许 OpenShift Virtualization 使黄金映像源可用于从默认模板创建虚拟机。

您可以通过从控制台编辑注释或从命令行进行修补，将Trident存储类和快照类设置为默认值。

```
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"}}}'
```

设置完成后，您可以使用以下命令删除任何预先存在的 dv 和 VolumeSnapShot 对象：

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

版权信息

版权所有 © 2026 NetApp, Inc.。保留所有权利。中国印刷。未经版权所有者事先书面许可，本文档中受版权保护的任何部分不得以任何形式或通过任何手段（图片、电子或机械方式，包括影印、录音、录像或存储在电子检索系统中）进行复制。

从受版权保护的 NetApp 资料派生的软件受以下许可和免责声明的约束：

本软件由 NetApp 按“原样”提供，不含任何明示或暗示担保，包括但不限于适销性以及针对特定用途的适用性的隐含担保，特此声明不承担任何责任。在任何情况下，对于因使用本软件而以任何方式造成的任何直接性、间接性、偶然性、特殊性、惩罚性或后果性损失（包括但不限于购买替代商品或服务；使用、数据或利润方面的损失；或者业务中断），无论原因如何以及基于何种责任理论，无论出于合同、严格责任或侵权行为（包括疏忽或其他行为），NetApp 均不承担责任，即使已被告知存在上述损失的可能性。

NetApp 保留在不另行通知的情况下随时对本文档所述的任何产品进行更改的权利。除非 NetApp 以书面形式明确同意，否则 NetApp 不承担因使用本文档所述产品而产生的任何责任或义务。使用或购买本产品不表示获得 NetApp 的任何专利权、商标权或任何其他知识产权许可。

本手册中描述的产品可能受一项或多项美国专利、外国专利或正在申请的专利的保护。

有限权利说明：政府使用、复制或公开本文档受 DFARS 252.227-7013（2014 年 2 月）和 FAR 52.227-19（2007 年 12 月）中“技术数据权利 — 非商用”条款第 (b)(3) 条规定的限制条件的约束。

本文档中所含数据与商业产品和/或商业服务（定义见 FAR 2.101）相关，属于 NetApp, Inc. 的专有信息。根据本协议提供的所有 NetApp 技术数据和计算机软件具有商业性质，并完全由私人出资开发。 美国政府对这些数据的使用权具有非排他性、全球性、受限且不可撤销的许可，该许可既不可转让，也不可再许可，但仅限在与交付数据所依据的美国政府合同有关且受合同支持的情况下使用。除本文档规定的情形外，未经 NetApp, Inc. 事先书面批准，不得使用、披露、复制、修改、操作或显示这些数据。美国政府对国防部的授权仅限于 DFARS 的第 252.227-7015(b)（2014 年 2 月）条款中明确的权利。

商标信息

NetApp、NetApp 标识和 http://www.netapp.com/TM 上所列的商标是 NetApp, Inc. 的商标。其他公司和产品名称可能是其各自所有者的商标。