



Introducción al almacenamiento y la virtualización

NetApp virtualization solutions

NetApp
July 21, 2026

This PDF was generated from <https://docs.netapp.com/es-es/netapp-solutions-virtualization/openshift/osv-storage-getting-started.html> on July 21, 2026. Always check docs.netapp.com for the latest.

Tabla de contenidos

Introducción al almacenamiento y la virtualización	1
Obtenga más información sobre la integración de NetApp Trident con Red Hat OpenShift	1
NetApp Trident	1
Trident Protect	2
NetApp Modelos de hardware de ONTAP y protocolos compatibles	2
Controladores Trident para almacenamiento ONTAP	3
Instala Trident en un clúster Red Hat OpenShift y crea objetos de almacenamiento	4
Paso 1: instalar Trident	5
Paso 2: prepara el backend de almacenamiento y los archivos de configuración de StorageClass para tu entorno	9
Paso 3: crea un archivo de configuración VolumeSnapshotClass	29
Paso 4: aplica los archivos de configuración a tu clúster	29
Paso 5: establece las clases predeterminadas de almacenamiento e instantáneas de Trident	31
Requisitos para implementar Red Hat OpenShift Virtualization con ONTAP	32
Prerrequisitos	32
NetApp Trident CSI, perfiles de almacenamiento, modos de acceso a volúmenes para OpenShift Virtualization	33
Crea una máquina virtual usando almacenamiento ONTAP con Red Hat OpenShift Virtualization	38
Crear máquina virtual	39
Demostración en video	42
Crea una máquina virtual con Red Hat OpenShift Virtualization usando almacenamiento ONTAP mediante métodos alternativos	42
Crear una máquina virtual a partir de un archivo YAML con Red Hat OpenShift Virtualization	42
Cree una máquina virtual a partir de una copia instantánea con Red Hat OpenShift Virtualization	45
Clona una máquina virtual en OpenShift Virtualization usando Trident	49
Migrar una máquina virtual entre dos nodos en un clúster de Red Hat OpenShift	53
Migración en vivo de máquinas virtuales	53
Migración de máquinas virtuales entre clases de almacenamiento en OpenShift Virtualization	55

Introducción al almacenamiento y la virtualización

Obtenga más información sobre la integración de NetApp Trident con Red Hat OpenShift

El almacenamiento es fundamental en las aplicaciones en contenedores, ya que garantiza que los datos estén disponibles de forma permanente y se gestionen de manera eficiente en entornos dinámicos y, a menudo, efímeros. Las soluciones de almacenamiento modernas para contenedores están diseñadas para escalar de forma fluida con las aplicaciones, lo que proporciona la flexibilidad necesaria para asignar dinámicamente los recursos de almacenamiento en función de las necesidades de las aplicaciones, dando soporte tanto al crecimiento como a las cargas de trabajo variables. Como OpenShift Virtualization permite que las máquinas virtuales se ejecuten como contenedores, usan el mismo almacenamiento que otras aplicaciones en los clústeres de OpenShift.

NetApp ONTAP es un software de gestión de datos robusto y altamente escalable que constituye la columna vertebral de las soluciones de almacenamiento de clase empresarial de NetApp. Ofrece una arquitectura de almacenamiento unificada que se integra a la perfección con diversos entornos de almacenamiento, incluyendo configuraciones locales, en la nube e híbridas. ONTAP es compatible con una amplia variedad de protocolos, como NFS, CIFS/SMB, iSCSI, FC y NVMe sobre TCP, lo que permite una integración perfecta con diferentes aplicaciones y cargas de trabajo. Permite la consolidación de SAN, NAS y almacenamiento de objetos en una plataforma única, lo que simplifica la gestión y reduce los costes. NetApp ONTAP está disponible tanto en las instalaciones como en forma de servicios gestionados en la nube.

NetApp Trident

Trident es un orquestador de almacenamiento de código abierto desarrollado por NetApp, diseñado para facilitar el uso del almacenamiento ONTAP en entornos de contenedores. Permite el aprovisionamiento y la gestión dinámicos del almacenamiento persistente para aplicaciones en contenedores.

Funciones clave de Trident

Cumplimiento de CSI Garantiza la compatibilidad con las últimas funciones y estándares de almacenamiento de Kubernetes. **Configuración declarativa** Permite a los usuarios definir los requisitos de almacenamiento en archivos YAML, que luego son gestionados automáticamente por Trident. **Controladores** Trident admite controladores para los protocolos NFS, CIFS/SMB, iSCSI, FC y NVMe/TCP compatibles con ONTAP. **Compatibilidad con entornos híbridos y multinube** Trident admite controladores para almacenamiento ONTAP local y servicios de almacenamiento gestionados en los hiperescaladores, concretamente FSx para NetApp ONTAP en AWS, Azure NetApp Files en Azure y Google Cloud NetApp Volumes en Google Cloud. **Gestión avanzada de datos** Aprovechando las capacidades de instantáneas y clonación de ONTAP para una protección de datos eficiente y un aprovisionamiento rápido de entornos de prueba y desarrollo.

Para obtener información detallada sobre Trident, consulta NetApp Trident "[documentación aquí](#)"

Trident Protect

Trident Protect es una función avanzada de Trident que ofrece una mayor protección y seguridad de los datos en entornos contenedorizados. Está diseñada para garantizar la protección de los datos críticos frente a posibles amenazas y situaciones de pérdida de datos. Está disponible de forma gratuita para todos los usuarios de Trident y se puede activar fácilmente en cualquier clúster de OpenShift que tenga instalado Trident.

Funciones clave de Trident Protect

Copias de seguridad automáticas Trident Protect permite realizar copias de seguridad automáticas, condicionadas por políticas, de aplicaciones en contenedores y máquinas virtuales, asegurando que se respalden regularmente sin intervención manual.

Gestión de instantáneas Aprovecha las capacidades de instantáneas de ONTAP para crear copias frecuentes de un momento específico de aplicaciones en contenedores y máquinas virtuales, proporcionando un mecanismo fiable para la recuperación.

Cifrado de datos Garantiza que los datos estén cifrados tanto en reposo como en tránsito, cumpliendo estrictos requisitos de seguridad y protegiendo la información confidencial.

Cumplimiento normativo y auditorías Ofrece herramientas y funciones que ayudan a las organizaciones a cumplir los requisitos de cumplimiento y a realizar auditorías periódicas de sus prácticas de protección de datos.

Recuperación ante desastres se integra con las funciones de replicación de ONTAP para ofrecer soluciones sólidas de recuperación ante desastres, garantizando la continuidad del negocio incluso ante eventos catastróficos.

Para obtener información detallada sobre Trident Protect, consulta NetApp Trident Protect ["documentación aquí"](#).

NetApp Modelos de hardware de ONTAP y protocolos compatibles

El software NetApp ONTAP se ejecuta en diversos modelos de hardware, cada uno diseñado para satisfacer distintos requisitos de rendimiento y capacidad. Trident amplía sus sólidas capacidades para dar soporte a varios sistemas NetApp ONTAP, incluidos All Flash FAS (AFF), All SAN Array (ASA) y ASA R2. Este soporte garantiza que las organizaciones puedan aprovechar todo el potencial de sus soluciones de almacenamiento de alto rendimiento en entornos contenedorizados.

Sistemas All Flash FAS (AFF) Los sistemas AFF ofrecen un rendimiento excepcional y baja latencia, lo que los hace ideales para aplicaciones de demanda elevada.

Todos los sistemas de matriz SAN (ASA) Los sistemas ASA están diseñados para entornos SAN dedicados, ofreciendo un rendimiento y una fiabilidad robustos.

Sistemas ASA R2 Los sistemas ASA R2 ofrecen funciones y prestaciones mejoradas para entornos SAN.

AFX NetApp AFX es una arquitectura de almacenamiento all-flash desagregada y diseñada específicamente para cargas de trabajo de IA empresarial. Ofrece un NAS de alto rendimiento, lo que permite escalar de forma independiente la capacidad de cálculo y el almacenamiento. NetApp AFX se diferencia de otros sistemas basados en ONTAP (ASA, AFF y FAS) en la implementación de su capa de almacenamiento. AFX utiliza un sistema de archivos distribuido que permite un alto rendimiento y escalabilidad, mientras que otros sistemas basados en ONTAP utilizan una arquitectura de almacenamiento tradicional.

1. Protocolos compatibles con AFF: NFS, CIFS/SMB, iSCSI, FC, NVMe/TCP
2. Protocolos compatibles con ASA: iSCSI, FC, NVMe/TCP
3. Protocolos compatibles con ASA R2: iSCSI, FC, NVMe/TCP
4. Protocolos compatibles con AFX: a partir de Trident 25.10, solo se admite el protocolo NFS; el protocolo SMB no es compatible.

Controladores Trident para almacenamiento ONTAP

Trident incluye los siguientes controladores CSI, cada uno de los cuales puede aprovisionar un volumen en el almacenamiento backend

Para los protocolos NAS en los modelos de hardware compatibles

1. `ontap-nas`: Cada pvc se asigna a 1 pv, que es un flexvolume dedicado
2. `ontap-nas-economy`: Cada PV aprovisionado es un qtree, con un número configurable de qtrees por FlexVolume (el valor por defecto es 200, configurable entre 50 y 300). Cada pvc se asigna a 1 pv, que es un qtree en un flexvolume compartido.



Puedes colocar todos los discos de las máquinas virtuales como qtrees en un único volumen. Sin embargo, ten en cuenta que las funciones de instantáneas, clones y replicación no son compatibles con Trident. Cuando esas funciones son gestionadas por el administrador de almacenamiento, no se aplican de forma granular a los PV.

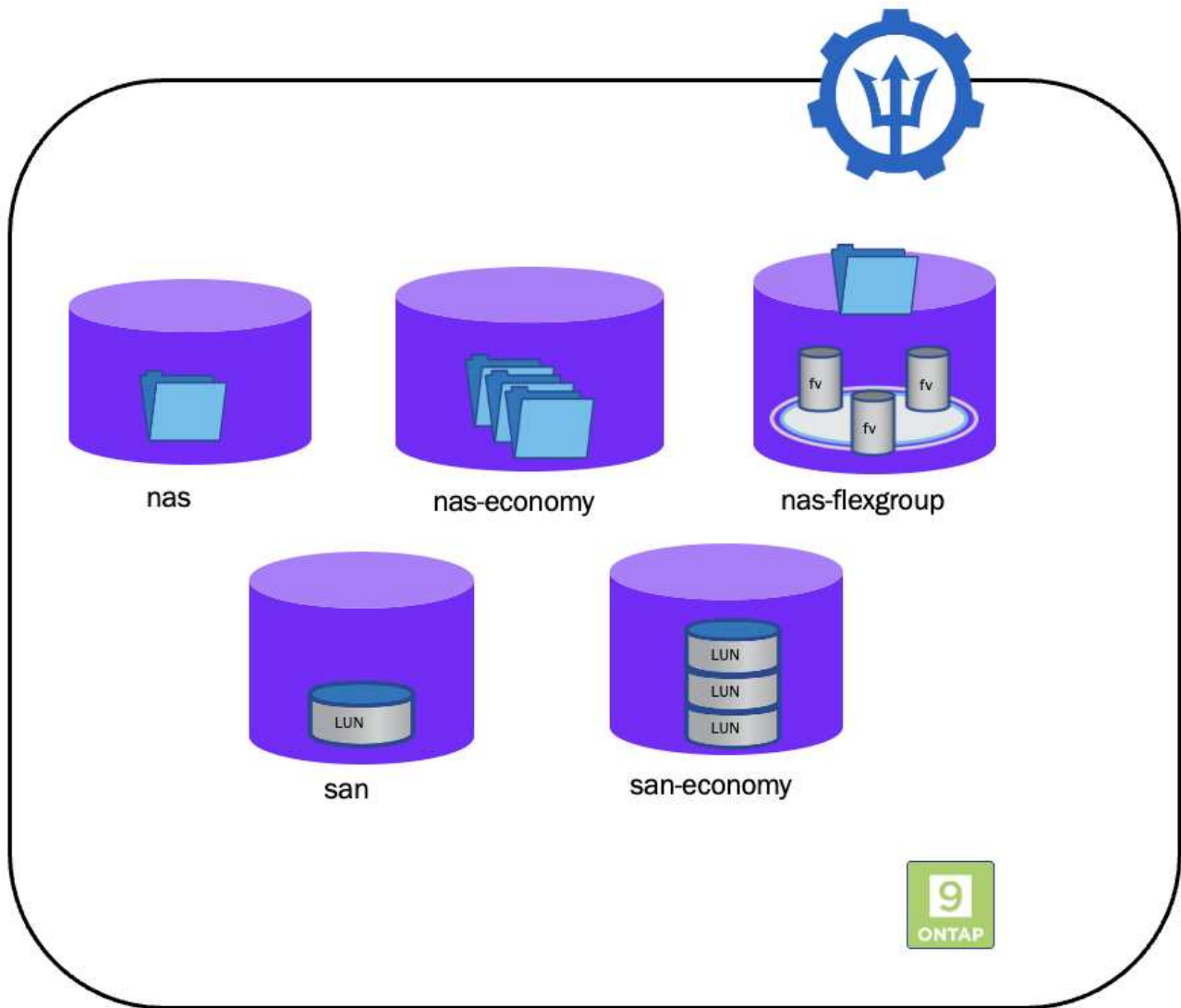
1. `ontap-nas-flexgroup`: Cada PV aprovisionado es un ONTAP completo FlexGroup, y se utilizan todos los agregados asignados a un SVM.

Para los protocolos SAN en los modelos de hardware compatibles

1. `ontap-san`: Un PVC se asigna a 1 PV, que es un LUN en un FlexVolume dedicado
2. `ontap-san-economy`: Cada pvc se asigna a 1 pv, que es un LUN en un flexvolume compartido. Puedes tener varios LUN en el flexvolume compartido (el valor predeterminado es 100, configurable entre 50 y 200 LUN/PV por volumen flexvol).



Puedes colocar todos los discos de las máquinas virtuales como LUN en un único volumen. Sin embargo, este controlador, aunque admite Snapshots y Clones, no admite Replication. Cuando Replication la gestiona el administrador de almacenamiento, no es granular a nivel de PV.



Para consultar la lista completa de funciones disponibles a través de los controladores de Trident y aquellas que debe aplicar el administrador de almacenamiento, consulta la sección "[Controladores de backend de ONTAP](#)" en la documentación de Trident.

Instala Trident en un clúster Red Hat OpenShift y crea objetos de almacenamiento

Instala Trident con Red Hat Certified Trident Operator y crea objetos de almacenamiento para ONTAP y Amazon FSx for NetApp ONTAP para habilitar el aprovisionamiento dinámico de volúmenes para contenedores y VMs. Prepara los nodos trabajadores para el acceso a bloques cuando sea necesario.

Antes de empezar

- Completa los procedimientos en esta página antes de instalar OpenShift Virtualization. OpenShift Virtualization requiere un StorageClass respaldado por Trident por defecto y un VolumeSnapshotClass para crear imágenes golden para plantillas de máquinas virtuales.
- Si ya instalaste OpenShift Virtualization antes de configurar Trident, elimina cualquier imagen golden

creada con una clase de almacenamiento diferente. Después de que configures Trident como predeterminado, OpenShift Virtualization vuelve a crear las imágenes golden usando el almacenamiento de Trident.

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

Paso 1: instalar Trident

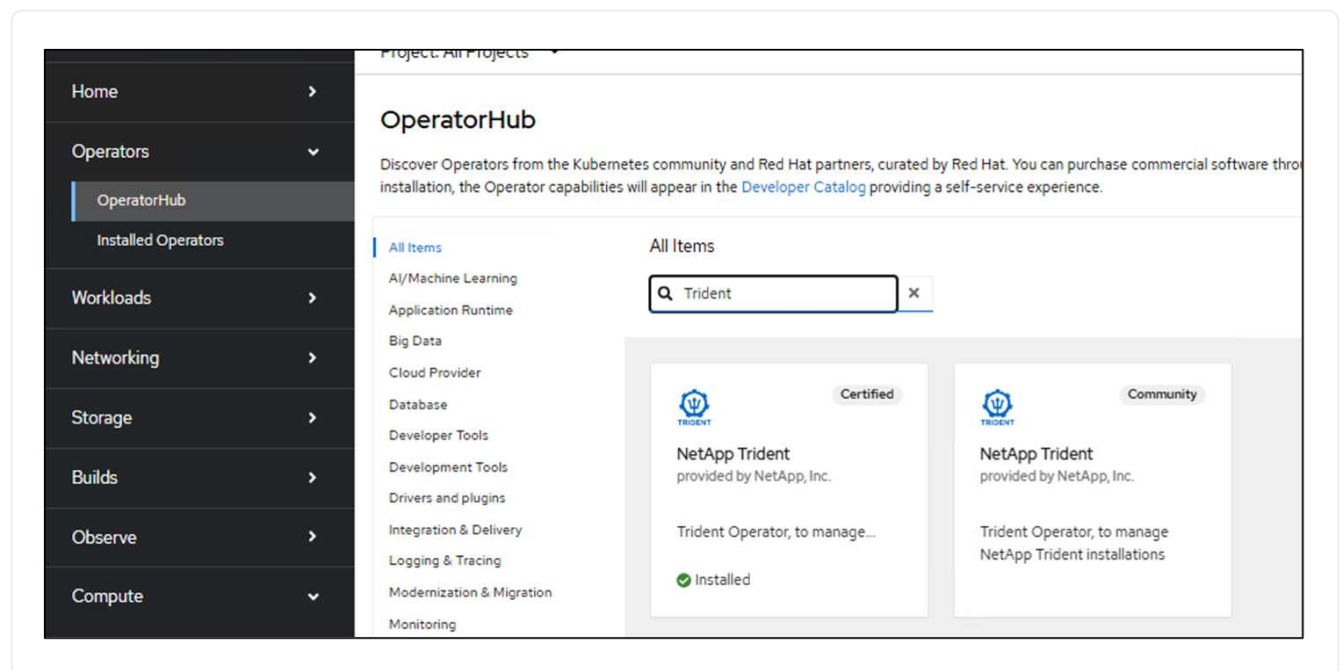
El Red Hat Certified Trident Operator es compatible con NetApp para OpenShift en las instalaciones, en nubes públicas y en servicios gestionados como ROSA. A partir de Trident 25.02, el operador también puede preparar nodos de trabajo para iSCSI cuando usas Amazon FSx for NetApp ONTAP y planeas ejecutar cargas de trabajo de VM de virtualización de OpenShift.

Para otras opciones de instalación, mira ["la documentación de Trident"](#).

Pasos

1. En **OperatorHub**, selecciona **Certified NetApp Trident**.

Mostrar ejemplo



2. En la página **Instalar**, mantén la última versión y selecciona **Instalar**.

Mostrar ejemplo

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

- Después de que el operador se instale, selecciona **View operator** y crea una instancia de Trident Orchestrator.

Si quieres preparar los nodos trabajadores para iSCSI, cambia a la vista YAML y añade `iscsi` a `nodePrep`.

Mostrar ejemplo

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. Confirma que todos los pods Trident están en ejecución en el clúster.

Mostrar ejemplo

```
[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. Si activaste la preparación de nodos iSCSI, inicia sesión en los nodos trabajadores y verifica que `iscsid` y `multipathd` estén activos y que `multipath.conf` tenga entradas.

Mostrar ejemplo

```
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#
```

Mostrar ejemplo

```
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# █
```

Demostración en video

El siguiente vídeo muestra una demostración de cómo instalar Trident usando el Red Hat Certified Trident Operator.

[Instalación de Trident 25.02.1 utilizando el operador Trident certificado en OpenShift](#)

Paso 2: prepara el backend de almacenamiento y los archivos de configuración de StorageClass para tu entorno

Crea definiciones de TridentBackendConfig y StorageClass para tu entorno. Puedes configurar varios protocolos de almacenamiento dentro de tu entorno. Crea archivos YAML para cada protocolo que quieras usar y reemplaza los valores de los marcadores de posición por los detalles específicos de tu configuración.



Completa la sección local o ROSA según tu entorno y luego sigue con el paso 3.

Clústeres on-premises OpenShift

Crea archivos YAML para cada protocolo que quieras configurar. Puedes configurar uno o más de los siguientes protocolos: NAS para almacenamiento de archivos basado en NFS, iSCSI para almacenamiento de bloques iSCSI, NVMe/TCP para almacenamiento de bloques NVMe sobre TCP de alto rendimiento o FC para almacenamiento de bloques Fibre Channel.

Crea un TridentBackendConfig y un StorageClass para cada protocolo. La configuración del backend incluye credenciales almacenadas en un secreto de Kubernetes.

Pautas esenciales para las opciones básicas de configuración del backend

1. Los archivos de configuración del backend hacen referencia a tu **ONTAP SVM** y a tu **LIF de gestión**.
+
2. Puedes utilizar **nombre de usuario y contraseña** para autenticarte en ONTAP. Pueden ser el nombre de usuario y la contraseña del administrador del clúster o del administrador del vserver. +
3. También puedes usar un **certificado de cliente** para la autenticación con ONTAP. Para detalles sobre cómo generar el certificado de cliente, la clave privada y el certificado de la CA, revisa ["Documentación de Trident aquí"](#).

Pautas esenciales para las opciones de configuración del backend para el controlador economy de NAS

Se pueden incluir los siguientes parámetros para la configuración del **NAS economy driver**:

1. **qtreesPerFlexvol** El valor debe estar en el rango de 50 a 300, el valor predeterminado es 200. +
2. **limitVolumePoolSize** El valor es el tamaño máximo de FlexVol que solicitas cuando usas qtrees en el backend ontap-nas-economy. No se establece ningún valor predeterminado y este parámetro no se aplica por defecto. Si configuras este parámetro, el operador Trident crea un nuevo FlexVol cuando el FlexVol actual alcanza el tamaño especificado. Este parámetro puede ayudarte a controlar el uso de la capacidad en tu clúster al restringir el tamaño de los FlexVols usados para el aprovisionamiento de PVs. +
3. **denyNewVolumePools** Este parámetro impide que los backends de ontap-nas-economy creen nuevos volúmenes FlexVol para alojar sus qtrees. Solo se utilizan los FlexVol ya existentes para el aprovisionamiento de nuevos PV.

Directrices esenciales para las opciones de configuración del backend para el controlador SAN

Se pueden incluir los siguientes parámetros para la configuración del **controlador SAN**:

1. **limitVolumeSize** El valor es el tamaño máximo de volumen que solicitas cuando usas un backend SAN. No se establece ningún valor predeterminado y este parámetro no se aplica por defecto. Si configuras este parámetro, el operador de Trident falla el aprovisionamiento si el tamaño de volumen solicitado supera este valor. Puedes usar este parámetro para restringir el tamaño máximo de los volúmenes que gestiona para LUN. +
2. **lunsPerFlexvol** Número máximo de LUN por FlexVol, debe estar en el rango [50, 200], el valor predeterminado es 100. Si configuras este parámetro, el operador de Trident crea un nuevo FlexVol cuando el FlexVol actual alcanza el número especificado de LUN. Este parámetro puede ayudarte a controlar el uso de la capacidad en tu clúster al restringir el número de LUN por FlexVol usado para el aprovisionamiento de PV.

Se pueden incluir los siguientes parámetros para la configuración del **SAN economy driver**:

1. **limitVolumePoolSize** El valor es el tamaño máximo de FlexVol que solicitas cuando usas el backend SAN economy. No se establece ningún valor predeterminado y este parámetro no se aplica por defecto. Si configuras este parámetro, el operador de Trident limita el tamaño máximo de flexvol que se puede crear para crear LUNs en el flexvol. Este parámetro puede ayudarte a controlar el uso de la capacidad en tu clúster al restringir el tamaño de los FlexVols usados para el aprovisionamiento de PVs. +
2. **denyNewVolumePools** Este parámetro impide que los backends económicos de SAN creen nuevos volúmenes FlexVol para alojar sus LUN. Solo se utilizan los FlexVol ya existentes para el aprovisionamiento de nuevos PV.

Directrices esenciales para las opciones de configuración del backend para el aprovisionamiento de volúmenes



En los archivos yaml de ejemplo proporcionados para cada protocolo, si no se proporciona storagePrefix, se utiliza el prefijo predeterminado “trident”. Cuando usas ontap-nas-economy y un storagePrefix de 24 caracteres o más, los qtrees no tendrán el prefijo de almacenamiento incluido, aunque sí estará en el nombre del volumen.

Cuando creas un volumen en ONTAP directamente desde system manager o desde la CLI, sin proporcionar valores para todos los parámetros opcionales, se establecen automáticamente algunos valores predeterminados durante el aprovisionamiento de un volumen, como se muestra a continuación.

```
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
```

A continuación se muestra un ejemplo de un volumen creado mediante la configuración del backend Trident. No tiene ninguna opción configurada explícitamente en la sección defaults, salvo la plantilla de nombre.

```
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
```

Cuando se crea el volumen utilizando esta configuración del backend, las opciones que no se hayan establecido en el archivo de configuración del backend se fijan en los valores por defecto como se muestra a continuación.

```
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
```

Sin embargo, cuando estableces valores en la sección default en la configuración del backend de Trident, puedes crear volúmenes que pueden tener valores específicos para esos parámetros. En el siguiente ejemplo, snapshotPolicy se establece en default policy (es none si no configuras nada explícitamente) y

snapshotDir se establece en true en la sección default del archivo de configuración yaml del backend. Con esta configuración, puedes ver que el volumen del backend obtiene esos valores. (snapdir-access es true y snapshot-policy es 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 ---rwxrwxrwx  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
```

Para el resto de parámetros adicionales, consulta la documentación de Trident:

1. "Opciones de configuración del backend NAS de Trident" +
2. "Opciones de configuración del backend de Trident SAN"

NAS

Crea un TridentBackendConfig y un StorageClass para ONTAP NAS para habilitar el aprovisionamiento de almacenamiento persistente basado en NFS. La configuración del backend incluye credenciales almacenadas en un Kubernetes Secret y hace referencia a tu SVM de ONTAP y a tu LIF de gestión.

Ejemplo de secreto de backend y archivo de configuración de backend con autenticación mediante nombre de usuario y contraseña (guardar como `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
```

Ejemplo de secreto de backend y archivo de configuración de backend con autenticación mediante certificado de cliente (guardar como `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>
```

Definición de StorageClass (guarda como `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` y un `StorageClass` para el controlador ONTAP NAS economy, con el fin de habilitar el aprovisionamiento de almacenamiento persistente basado en NFS mediante qtrees. La

configuración del backend incluye credenciales almacenadas en un Kubernetes Secret y hace referencia a tu ONTAP SVM y a tu LIF de gestión.

Ejemplo de secreto de backend y archivo de configuración de backend con autenticación mediante nombre de usuario y contraseña (guardar como `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
  }}"
    credentials:
      name: tbc-nas-economy-secret
```

StorageClass definición (guarda como `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 las directrices esenciales para las opciones de configuración del backend del controlador NAS economy en la sección de arriba para ver detalles sobre los parámetros adicionales que puedes incluir en el archivo de configuración del backend para el controlador ontap-nas-economy.

SAN iSCSI

Crea un TridentBackendConfig y un StorageClass para ONTAP SAN con iSCSI para habilitar el aprovisionamiento de almacenamiento en bloques basado en iSCSI. La configuración del backend utiliza el controlador ontap-san e incluye credenciales almacenadas en un secreto de Kubernetes. El parámetro sanType usa por defecto el protocolo iSCSI si se omite.

Secreto del backend y archivo de configuración del backend con autenticación mediante nombre de usuario y contraseña (guardar como 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

```

Secreto del backend y archivo de configuración del backend usando autenticación mediante certificado de cliente (guardar como 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>
```

Definición de StorageClass (guarda como `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 un `TridentBackendConfig` y un `StorageClass` para ONTAP SAN con NVMe sobre TCP para habilitar el aprovisionamiento de almacenamiento en bloque de alto rendimiento. La configuración del backend usa el controlador `ontap-san` optimizado para el transporte NVMe/TCP e incluye credenciales almacenadas en un Kubernetes Secret.

Secreto del backend y archivo de configuración del backend con autenticación mediante nombre de usuario y contraseña (guardar como `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
```

Secreto del backend y archivo de configuración del backend usando autenticación mediante certificado de cliente (guardar como `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>
```

Definición de StorageClass (guarda como `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
```



Si planeas usar volúmenes ReadWriteMany (RWX) con el protocolo NVMe, un solo volumen NVMe RWX no puede montarse por más de 64 nodos en un clúster de Kubernetes. Para más información sobre esta limitación y sobre el modelo de supersubsistema introducido en Trident 26.02 y el modelo de subsistema por volumen anterior a Trident 26.02, consulta la documentación de Trident: <https://docs.netapp.com/us-en/trident/trident-use/ontap-san-prep.html#nvme-tcp-considerations>.

FC SAN

Crea un `TridentBackendConfig` y un `StorageClass` para ONTAP SAN con Fibre Channel para habilitar el aprovisionamiento de almacenamiento en bloques basado en FC. La configuración del backend usa el controlador `ontap-san` con el protocolo FCP especificado e incluye credenciales almacenadas en un Kubernetes Secret.

Utiliza el método de autenticación con nombre de usuario y contraseña o el método de autenticación con certificado de cliente para autenticarte con ONTAP. Para ver detalles sobre cómo generar el certificado de cliente, la clave privada y el certificado de la CA, revisa "[Documentación de Trident aquí](#)". Aquí tienes ejemplos para ambos métodos.

Mostrar ejemplo

Secreto del backend y archivo de configuración del backend con autenticación mediante nombre de usuario y contraseña de ONTAP (guardar como `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
  }}_{{ .volume.RequestName }}"
  credentials:
    name: tbc-fc-secret
```

Secreto del backend y archivo de configuración del backend usando autenticación mediante certificado de cliente (guardar como `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
  }}"
    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` y un `StorageClass` para ONTAP SAN economy con iSCSI para habilitar el aprovisionamiento de almacenamiento en bloques basado en iSCSI. La configuración del backend utiliza el controlador `ontap-san-economy` con el protocolo iSCSI e incluye credenciales almacenadas en un secreto de Kubernetes.

Secreto del backend y archivo de configuración del backend con autenticación mediante certificado de cliente (guardar como `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 documentación de Trident para ver más ejemplos de archivos yaml e información sobre los modos de acceso y los modos de volumen compatibles con los controladores SAN y NAS.

1. ["Ejemplos con controladores NAS"](#)
2. ["Ejemplos con controladores SAN"](#)

Clústeres ROSA con Amazon FSx for NetApp ONTAP

Crea archivos YAML para cada protocolo que quieras configurar. Puedes configurar uno o ambos de los siguientes protocolos: NAS para almacenamiento de archivos basado en NFS o iSCSI para almacenamiento en bloques.

NAS

Crea un TridentBackendConfig y un StorageClass para Amazon FSx for NetApp ONTAP con ONTAP NAS para habilitar el aprovisionamiento de almacenamiento persistente basado en NFS en clústeres ROSA. La configuración del backend usa los nombres DNS de Amazon FSx for NetApp ONTAP para la gestión y los LIF de datos, e incluye credenciales almacenadas en un Secret de Kubernetes en el namespace de trident.

Secreto de backend y archivo de configuración de backend (guarda como `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 definición (guarda como `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 un TridentBackendConfig y un StorageClass para Amazon FSx for NetApp ONTAP con ONTAP SAN para habilitar el aprovisionamiento de almacenamiento en bloques basado en iSCSI en clústeres ROSA. La configuración del backend usa el controlador ontap-san e incluye credenciales almacenadas en un Kubernetes Secret. Asegúrate de que los nodos worker estén preparados para el acceso a iSCSI.

Secreto de backend y archivo de configuración de backend (guárdalo como `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 definición (guarda como `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
```

Paso 3: crea un archivo de configuración VolumeSnapshotClass

Crea una definición de VolumeSnapshotClass tanto para implementaciones locales como para ROSA. Esta configuración permite operaciones basadas en instantáneas para volúmenes persistentes.

Definición de VolumeSnapshotClass (guarda como `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
```

Paso 4: aplica los archivos de configuración a tu clúster

Aplica los archivos de configuración que creaste en los pasos anteriores a tu clúster OpenShift.

Pasos

1. Aplica los archivos TridentBackendConfig y StorageClass para cada protocolo que configuraste.

Para clústeres locales:

```
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
```

Para los clústeres 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. Aplica la configuración de VolumeSnapshotClass.

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

3. Verifica que los recursos se hayan creado correctamente.

Revisa los objetos TridentBackendConfig:

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

Comprueba los objetos de StorageClass:

```
oc get storageclass
```

Revisa VolumeSnapshotClass:

```
oc get volumesnapshotclass
```

Paso 5: establece las clases predeterminadas de almacenamiento e instantáneas de Trident

Configura Trident StorageClass y VolumeSnapshotClass como los predeterminados en el clúster OpenShift. Esto es necesario para que OpenShift Virtualization cree fuentes de imágenes doradas para plantillas de máquinas virtuales.

Pasos

1. Configura el StorageClass predeterminado de Trident.

Configura un StorageClass respaldado por Trident como predeterminado del clúster para que las PersistentVolumeClaims lo usen automáticamente cuando no se especifique ninguna clase de almacenamiento. Necesitas configurar dos anotaciones: una para el valor predeterminado de todo el clúster y otra específica para OpenShift Virtualization.

- a. Establece la anotación predeterminada de StorageClass para todo el clúster.

Asegúrate de que solo un StorageClass esté configurado como predeterminado. Si otro StorageClass ya está configurado como predeterminado, pon su anotación en false.

En la consola, edita la anotación:

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

Desde la CLI:

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

- b. Establece la anotación predeterminada específica de OpenShift Virtualization.

OpenShift Virtualization utiliza una anotación específica que tiene prioridad sobre la anotación general del clúster `is-default-class`. Si otro StorageClass ya está configurado como predeterminado, pon su anotación en false.

En la consola, edita la anotación:

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

Desde la CLI:

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

2. Establece el Trident VolumeSnapshotClass predeterminado.

Establece un VolumeSnapshotClass respaldado por Trident como el predeterminado del clúster para habilitar operaciones basadas en instantáneas para volúmenes persistentes. Esto asegura que VolumeSnapshots use automáticamente el controlador CSI de Trident cuando no se especifica ninguna clase de instantánea y permite que OpenShift Virtualization cree instantáneas de imágenes doradas.

Asegúrate de que solo un VolumeSnapshotClass esté configurado como predeterminado. Si ya hay otro VolumeSnapshotClass configurado como predeterminado, pon su anotación en false.

En la consola, edita la anotación:

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

Desde la CLI:

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

Requisitos para implementar Red Hat OpenShift Virtualization con ONTAP

Consulta los requisitos previos y la información adicional para implementar la virtualización de OpenShift con sistemas de almacenamiento ONTAP.

Prerrequisitos

- Un clúster Red Hat OpenShift (posterior a la versión 4.6) instalado en una infraestructura física con nodos de trabajo RHCOS
- Implementar comprobaciones del estado de la máquina para mantener la alta disponibilidad de las máquinas virtuales
- Un clúster NetApp ONTAP , con SVM configurado con el protocolo correcto.
- Trident instalado en el clúster OpenShift
- Se creó una configuración de backend de Trident
- Una StorageClass configurada en el clúster OpenShift con Trident como aprovisionador
- Un Trident VolumeSnapshotClass configurado en el clúster OpenShift con Trident como el provisioner
- Trident StorageClass configurado como predeterminado para el clúster de OpenShift o OpenShift Virtualization.

Para conocer los requisitos previos de Trident mencionados anteriormente, consulte ["Sección de instalación del Trident"](#) Para más detalles.

- Acceso de administrador de clúster al clúster Red Hat OpenShift
- Acceso de administrador al clúster NetApp ONTAP
- Una estación de trabajo de administrador con herramientas tridentctl y oc instaladas y agregadas a \$PATH

Dado que OpenShift Virtualization se gestiona mediante un operador instalado en el clúster OpenShift, esto supone una carga adicional para la memoria, la CPU y el almacenamiento, lo cual debe tenerse en cuenta al planificar los requisitos de hardware del clúster. Consulta la documentación ["aquí"](#) para más detalles.

Si lo deseas, también puedes especificar un subconjunto de los nodos del clúster de OpenShift para alojar los operadores, controladores y máquinas virtuales de OpenShift Virtualization configurando las reglas de ubicación de nodos. Para configurar las reglas de ubicación de nodos para OpenShift Virtualization, sigue la documentación ["aquí"](#).

Para el respaldo de almacenamiento de OpenShift Virtualization, NetApp recomienda tener una StorageClass dedicada que solicita almacenamiento de un backend Trident particular, que a su vez está respaldado por un SVM dedicado. Esto mantiene un nivel de multitenencia con respecto a los datos que se sirven para cargas de trabajo basadas en VM en el clúster OpenShift.

NetApp Trident CSI, perfiles de almacenamiento, modos de acceso a volúmenes para OpenShift Virtualization

En OpenShift Virtualization, las fuentes de arranque son imágenes de disco que se usan para aprovisionar máquinas virtuales (VM) a partir de plantillas. Normalmente se almacenan como DataVolumes o Persistent Volume Claims (PVC) dentro del namespace openshift-virtualization-os-images. Para que OpenShift Virtualization funcione correctamente, hay dos tareas de configuración de almacenamiento que son obligatorias antes de instalar OpenShift Virtualization usando el operador:

1. Debes configurar un StorageClass predeterminado para el clúster. De lo contrario, OpenShift Virtualization no puede importar automáticamente las imágenes de origen de arranque.
2. Debes configurar perfiles de almacenamiento si CDI no reconoce a tu proveedor de almacenamiento. Un perfil de almacenamiento proporciona ajustes de almacenamiento recomendados basados en el StorageClass asociado. Una de las funciones fundamentales de StorageProfile es definir estrategias de clonación para un StorageClass. Esto determina cómo se clonan los volúmenes (por ejemplo, mediante instantáneas, clones CSI o copias asistidas por el host), optimizando el rendimiento y el uso de los recursos según las capacidades del proveedor de almacenamiento subyacente. + El controlador Trident CSI es reconocido por CDI y se crea automáticamente un perfil de almacenamiento predeterminado para cualquier StorageClass de Trident creado en el clúster. La estrategia de clonación para este perfil de almacenamiento predeterminado está configurada como instantánea, que es la estrategia de clonación recomendada para el controlador Trident CSI. Debes crear el VolumeSnapshotClass y el StorageClass para Trident y establecer el StorageClass como predeterminado antes de instalar OpenShift Virtualization. Esto asegura que las imágenes de origen de arranque estén disponibles como VolumeSnapshots en el espacio de nombres openshift-virtualization-os-images, para poder aprovisionar máquinas virtuales de forma eficiente y sin errores.

En determinados sistemas operativos compatibles (como RHEL, Fedora o CentOS), OpenShift Virtualization proporciona y actualiza automáticamente las fuentes de arranque. Estas se descargan cuando se instala OpenShift Virtualization y se gestionan y actualizan automáticamente a la última versión del sistema operativo. Si no se proporciona automáticamente un sistema operativo, un administrador debe preparar e incorporar manualmente una imagen personalizada. Cuando usas NetApp Trident como el CSI, obtienes los siguientes beneficios adicionales:

1. Las imágenes de origen de arranque estarán disponibles como instantáneas en el espacio de nombres openshift-virtualization-os-images. Cuando se aprovisionan las máquinas virtuales, la imagen de arranque se clona a partir de la instantánea, lo que hace que el arranque de la máquina virtual sea mucho más rápido.
2. Se crea automáticamente un perfil de almacenamiento para cada Trident StorageClass y se configura automáticamente con la estrategia de clonación establecida en snapshot. El modo de acceso está

configurado en `ReadWriteMany`. No es necesario realizar ninguna configuración manual.

Es fundamental que crees un `Trident StorageClass` y un `Trident VolumeSnapshotClass` y los configures como predeterminados antes de instalar la OpenShift Virtualization en tu clúster. De este modo, te asegurarás de que las imágenes de origen de arranque estén disponibles como `VolumeSnapshots` en el espacio de nombres `openshift-virtualization-os-images`, lo que permitirá aprovisionar máquinas virtuales de forma eficiente y sin errores.

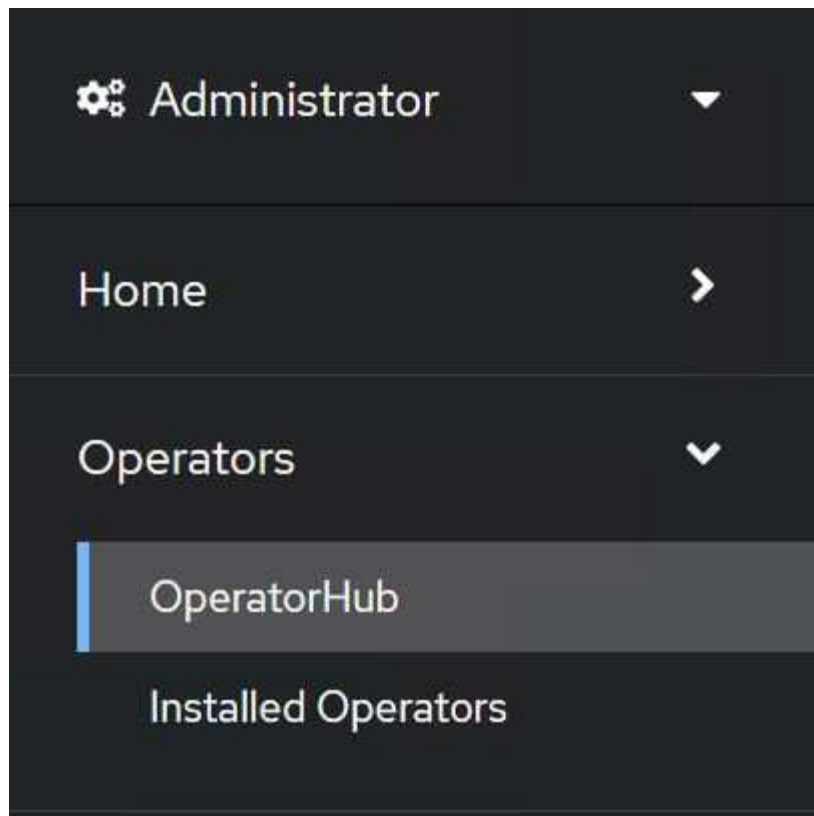
Otra ventaja de usar `Trident CSI`: . En OpenShift Virtualization, los discos de arranque de las máquinas virtuales necesitan tener modos de acceso `RWX` para hacer la migración en vivo de las máquinas virtuales. `Trident` admite el modo de acceso **`ReadWriteMany`** para volúmenes aprovisionados usando protocolos `NAS`, así como usando protocolos `SAN` (cuando `volumeMode` está configurado en `Block` y no en `Filesystem`). Puedes usar cualquiera de los dos protocolos para el disco de arranque. El objeto de perfil de almacenamiento se configurará automáticamente según la configuración de `StorageClass` y los modos de acceso al disco de arranque se establecerán adecuadamente al aprovisionar las máquinas virtuales.

Una vez que hayas configurado el `Trident StorageClass` como predeterminado para el clúster, puedes crear `StorageClasses` adicionales para `Trident` con diferentes parámetros y serán reconocidas automáticamente por OpenShift Virtualization, con los perfiles de almacenamiento apropiados creados para ellas. Luego puedes especificar el `StorageClass` que se usará para el disco de arranque y los discos de datos de la máquina virtual en la definición de la VM.

Instalar OpenShift Virtualization en el clúster de OpenShift es un proceso sencillo usando el operador OpenShift Virtualization disponible en el OpenShift OperatorHub. Para ver los pasos detallados sobre cómo instalar OpenShift Virtualization, consulta la documentación ["aquí"](#).

Instalar OpenShift Virtualization en un clúster bare-metal de Red Hat OpenShift. Este procedimiento incluye iniciar sesión con acceso de administrador del clúster, navegar al OperatorHub e instalar el operador de OpenShift Virtualization.

1. Inicie sesión en el clúster bare-metal de Red Hat OpenShift con acceso de administrador del clúster.
2. Seleccione Administrador en el menú desplegable Perspectiva.
3. Vaya a Operadores > OperatorHub y busque OpenShift Virtualization.



4. Seleccione el mosaico OpenShift Virtualization y haga clic en Instalar.



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. En la pantalla Instalar operador, deje todos los parámetros predeterminados y haga clic en Instalar.

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

i 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

HC OpenShift
Virtualization
Deployment **Required**

Represents the deployment of
OpenShift Virtualization

6. Espere a que se complete la instalación del operador.



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. Una vez instalado el operador, haga clic en Crear HyperConverged.



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. En la pantalla Crear HyperConverged, haga clic en Crear, aceptando todos los parámetros predeterminados. Este paso inicia la instalación de OpenShift Virtualization.

Name *

kubevirt-hyperconverged

Labels

app=frontend

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.

Create **Cancel**

9. Una vez que todos los pods pasan al estado En ejecución en el espacio de nombres openshift-cnv y el operador de virtualización OpenShift está en el estado Correcto, el operador está listo para usarse. Ahora se pueden crear máquinas virtuales en el clúster 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 ▾	Search by name...	
Name ↑	Managed Namespaces ⓘ	Status
 OpenShift Virtualization 2.6.2 provided by Red Hat	 openshift-cnv	 Succeeded Up to date
		Last updated  May 18, 8:02 pm
		Provided APIs OpenShift Virtualization Deployment HostPathProvisioner deployment

Crea una máquina virtual usando almacenamiento ONTAP con Red Hat OpenShift Virtualization

Crea una máquina virtual con OpenShift Virtualization. Este procedimiento incluye seleccionar una plantilla de sistema operativo, configurar StorageClasses y personalizar los parámetros de la máquina virtual para cumplir requisitos específicos. Como requisito

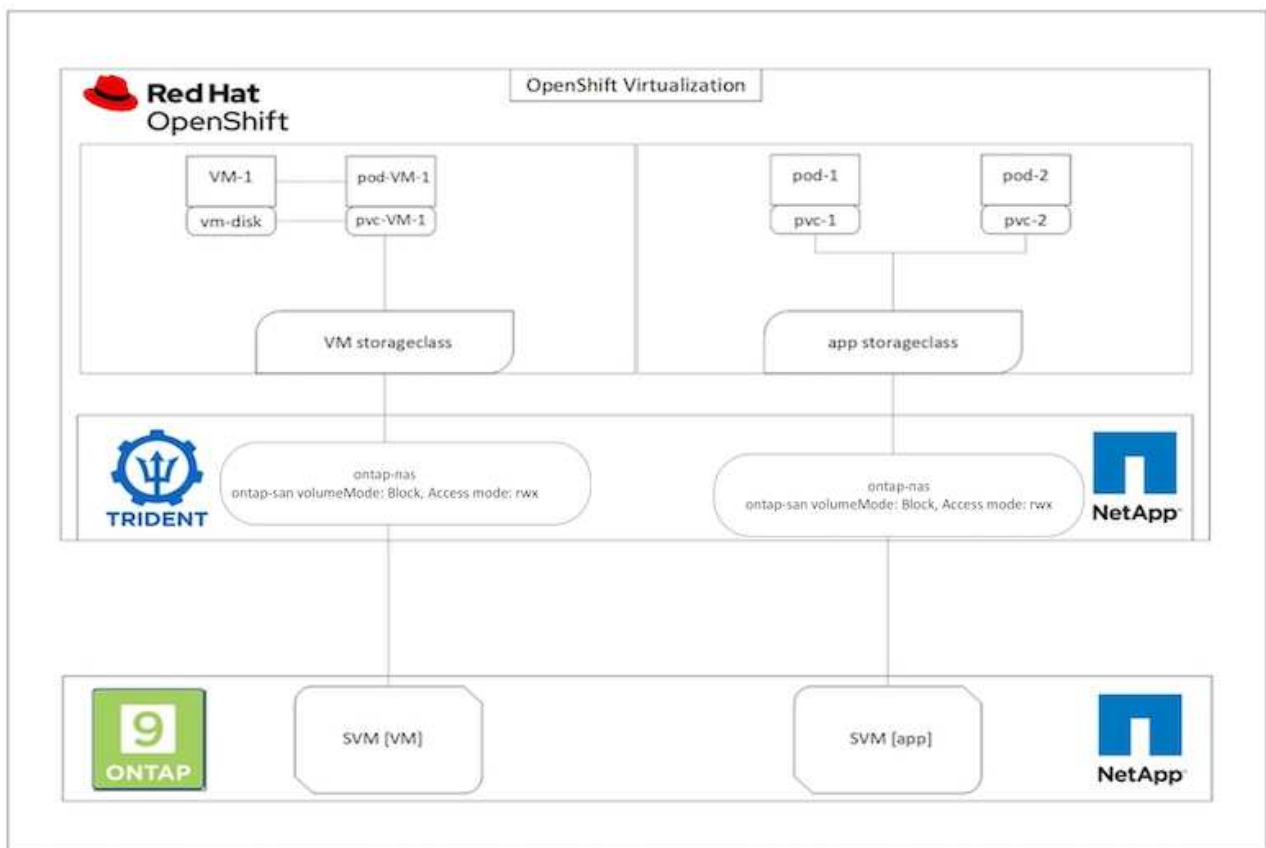
previo, ya debes haber creado el backend de Trident, el StorageClass y los objetos de clase de instantáneas de volumen. Puedes consultar "[Sección de instalación del Trident](#)" para más detalles.

Crear máquina virtual

Las máquinas virtuales son implementaciones con estado que requieren volúmenes para alojar el sistema operativo y los datos. Con CNV, debido a que las máquinas virtuales se ejecutan como pods, las máquinas virtuales están respaldadas por PV alojados en NetApp ONTAP a través de Trident. Estos volúmenes se conectan como discos y almacenan todo el sistema de archivos, incluida la fuente de arranque de la máquina virtual.



Trident StorageClass debe configurarse como StorageClass predeterminado en el clúster para asegurar que los PVC creados para los discos de las máquinas virtuales se aprovisionen usando el backend de Trident. Puedes tener varios StorageClasses en el clúster, pero el StorageClass predeterminado debe usarse para aprovisionar el PVC de los discos de arranque de las máquinas virtuales. Los discos adicionales pueden usar otros StorageClasses si es necesario.



Para crear rápidamente una máquina virtual en el clúster OpenShift, complete los siguientes pasos:

1. Vaya a Virtualización > Máquinas virtuales y haga clic en Crear.
2. Seleccionar desde plantilla.
3. Seleccione el sistema operativo deseado para el cual está disponible la fuente de arranque.

4. Marque la casilla de verificación Iniciar la máquina virtual después de la creación.
5. Haga clic en Crear máquina virtual rápidamente.

La máquina virtual se crea y se inicia y pasa al estado **En ejecución**. Se crea automáticamente un PVC y un PV correspondiente para el disco de arranque utilizando el StorageClass predeterminado. Para poder migrar en vivo la máquina virtual en el futuro, debes asegurarte de que el StorageClass utilizado para los discos pueda admitir volúmenes RWX. Este es un requisito para la migración en vivo. `ontap-nas` y `ontap-san` (`volumeMode block` para los protocolos iSCSI y NVMe/TCP) pueden admitir modos de acceso RWX para los volúmenes creados usando los StorageClasses respectivos.

Para configurar `ontap-san` StorageClass en el clúster, consulta el ["Sección para configurar StorageClass"](#).



Al hacer clic en Quick create VirtualMachine se usará la StorageClass predeterminada para crear el PVC y el PV del disco raíz de arranque de la VM. Puedes seleccionar una StorageClass diferente para el disco, seleccionando Customize VirtualMachine > Customize VirtualMachine parameters > Disks y luego editando el disco para usar la StorageClass requerida.

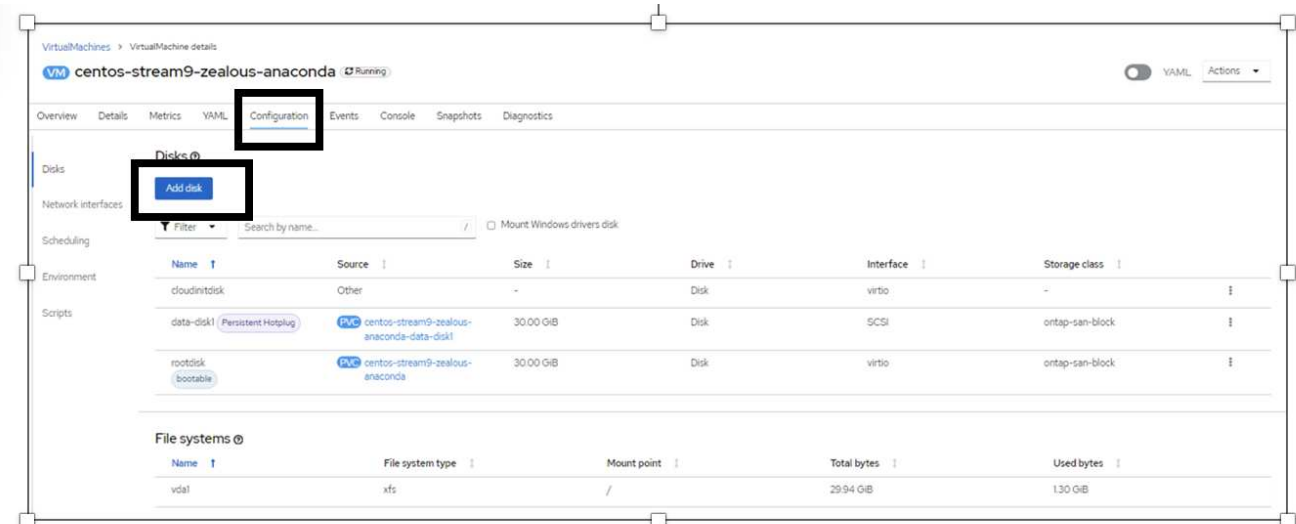
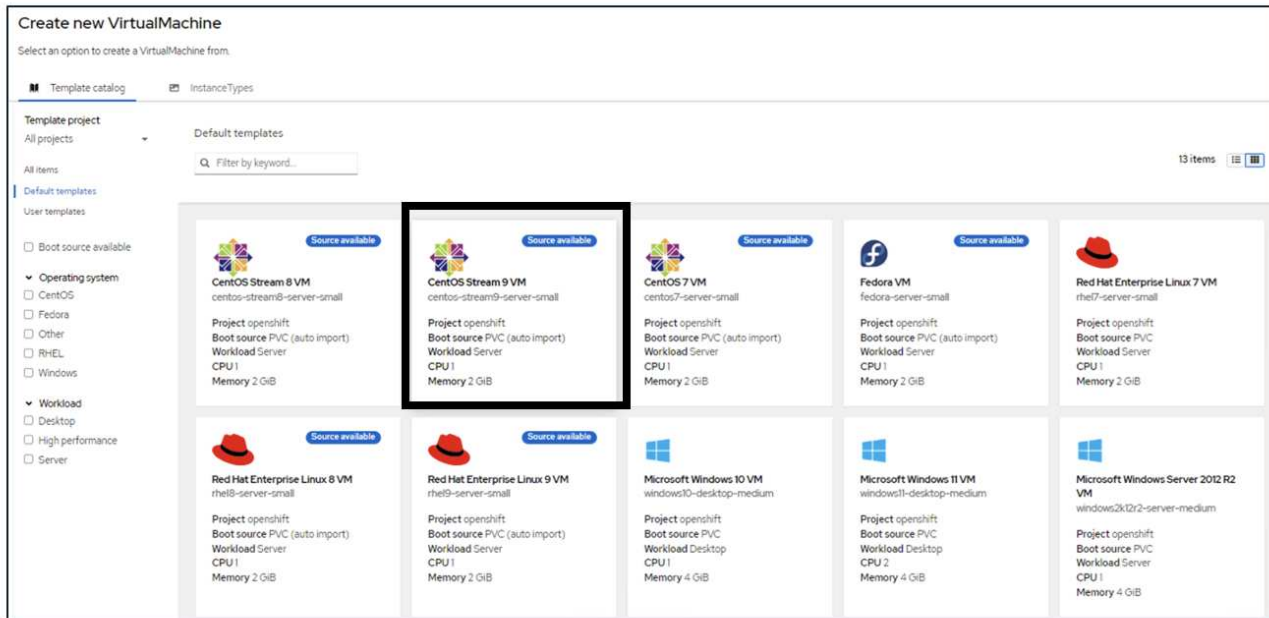
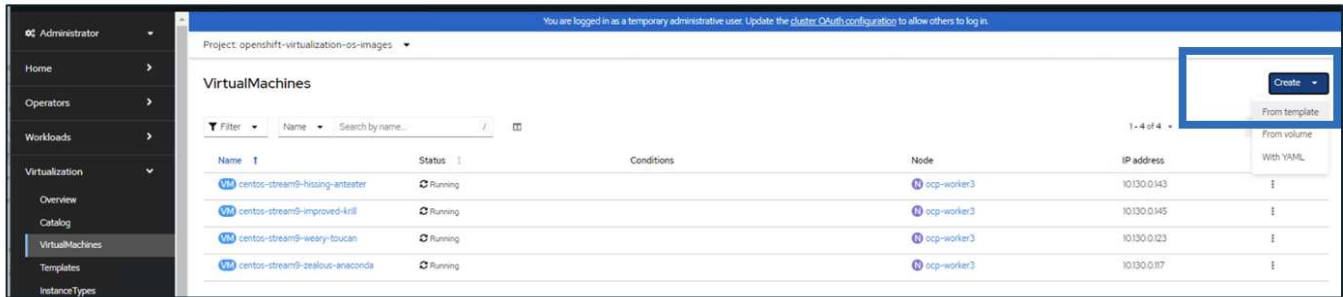
Generalmente, se prefiere el modo de acceso en bloque en comparación con los sistemas de archivos al aprovisionar los discos de VM.

Para personalizar la creación de la máquina virtual después de haber seleccionado la plantilla del sistema operativo, haga clic en Personalizar máquina virtual en lugar de Creación rápida.

1. Si el sistema operativo seleccionado tiene una fuente de arranque configurada, puede hacer clic en **Personalizar parámetros de VirtualMachine**.
2. Si el sistema operativo seleccionado no tiene ninguna fuente de arranque configurada, deberá configurarla. Puede ver detalles sobre los procedimientos que se muestran en el ["documentación"](#).
3. Después de configurar el disco de arranque, puede hacer clic en **Personalizar parámetros de VirtualMachine**.
4. Puede personalizar la VM desde las pestañas de esta página. Por ejemplo, haga clic en la pestaña **Discos** y luego haga clic en **Agregar disco** para agregar otro disco a la VM.
5. Haga clic en Crear máquina virtual para crear la máquina virtual; esto activa un pod correspondiente en segundo plano.



Cuando se configura una fuente de arranque para una plantilla o un sistema operativo desde una URL o desde un registro, se crea una PVC en el `openshift-virtualization-os-images` proyecto y descarga la imagen de invitado KVM al PVC. Debe asegurarse de que las PVC de plantilla tengan suficiente espacio provisto para acomodar la imagen de invitado KVM para el sistema operativo correspondiente. Luego, estos PVC se clonan y se adjuntan como discos raíz a las máquinas virtuales cuando se crean utilizando las plantillas respectivas en cualquier proyecto.



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

Demostración en video

El siguiente vídeo muestra una demostración de cómo crear una máquina virtual en OpenShift Virtualization usando almacenamiento iSCSI.

[Crear una máquina virtual en OpenShift Virtualization usando almacenamiento en bloque](#)

Crea una máquina virtual con Red Hat OpenShift Virtualization usando almacenamiento ONTAP mediante métodos alternativos

Además del método para crear una máquina virtual mediante la interfaz de usuario de OpenShift Virtualization, existen otros métodos para crear máquinas virtuales en OpenShift Virtualization. Esta sección ofrece una descripción general de algunos de esos métodos.

Crear una máquina virtual a partir de un archivo YAML con Red Hat OpenShift Virtualization

También puedes crear una máquina virtual aplicando un archivo YAML con el comando `oc apply`. Este método es útil para los usuarios que prefieren usar la CLI o quieren automatizar la creación de máquinas virtuales usando scripts. Para crear una máquina virtual con este método, puedes crear un archivo YAML que defina la máquina virtual y sus recursos asociados, como los PVC para los discos de la máquina virtual.

Luego, puedes aplicar el archivo YAML usando el comando `oc apply -f <yaml-file>` para crear la máquina virtual en el clúster OpenShift.

Mostrar ejemplo

```
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

```

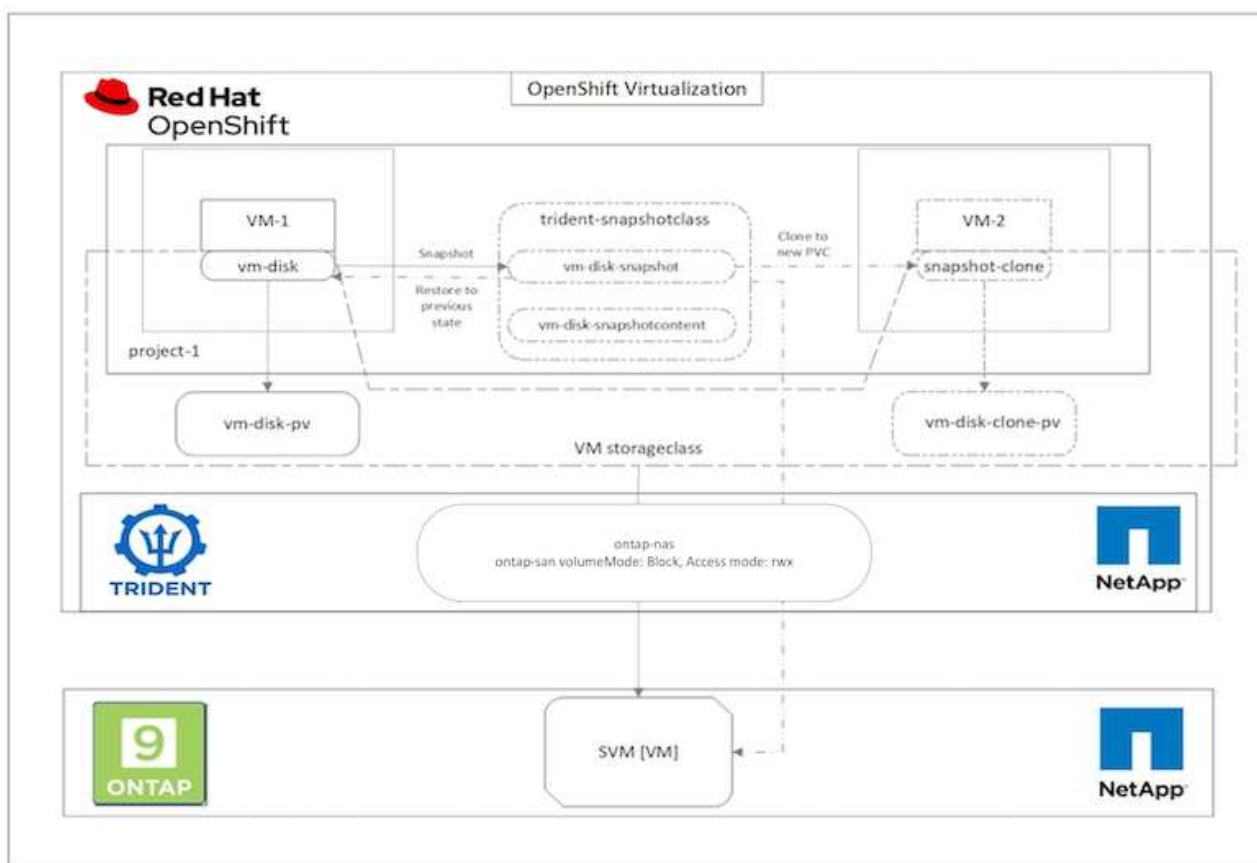
Cree una máquina virtual a partir de una copia instantánea con Red Hat OpenShift Virtualization

Puedes crear una máquina virtual a partir de una instantánea con OpenShift Virtualization. Este procedimiento incluye crear una `VolumeSnapshotClass`, tomar una instantánea de la reclamación de volumen persistente (PVC) de la máquina virtual, restaurar la instantánea en una nueva PVC y desplegar una nueva máquina virtual que use la PVC restaurada como disco raíz.

Con Trident y Red Hat OpenShift, los usuarios pueden tomar una instantánea de un volumen persistente en las clases de almacenamiento aprovisionadas por él. Con esta función, los usuarios pueden tomar una copia de un punto en el tiempo de un volumen y usarla para crear un nuevo volumen o restaurar el mismo volumen a un estado anterior. Esto habilita o respalda una variedad de casos de uso, desde reversión hasta clones y restauración de datos.

Para las operaciones de instantáneas en OpenShift, se deben definir los recursos `VolumeSnapshotClass`, `VolumeSnapshot` y `VolumeSnapshotContent`.

- Un `VolumeSnapshotContent` es la instantánea real tomada de un volumen en el clúster. Es un recurso de todo el clúster análogo a `PersistentVolume` para el almacenamiento.
- Una `VolumeSnapshot` es una solicitud para crear la instantánea de un volumen. Es análogo a un `PersistentVolumeClaim`.
- `VolumeSnapshotClass` permite al administrador especificar diferentes atributos para un `VolumeSnapshot`. Le permite tener diferentes atributos para diferentes instantáneas tomadas del mismo volumen.



Mostrar ejemplo

Requisito previo

Debes tener una máquina virtual existente con un PVC adjunto que esté aprovisionado en una Trident Storage Class, y la máquina virtual debe estar en estado de ejecución antes de que puedas tomar una instantánea del PVC. Debes tener un VolumeSnapshotClass creado que luego se pueda usar para crear un VolumeSnapshot. Esto se habría creado como parte de los requisitos previos para instalar OpenShift Virtualization. Puedes consultar el ["Paso 3 para crear una configuración de VolumeSnapshotClass"](#) para más detalles.

Para crear una instantánea de una máquina virtual, sigue estos pasos:

1. Identifique el PVC que está conectado a la máquina virtual de origen y luego cree una instantánea de ese PVC. Navegar a `Storage > VolumeSnapshots` y haga clic en `Crear instantáneas de volumen`.
2. Seleccione el PVC para el cual desea crear la instantánea, ingrese el nombre de la instantánea o acepte el valor predeterminado y seleccione la VolumeSnapshotClass adecuada. Luego haga clic en `Crear`.
3. Esto crea la instantánea del PVC en ese 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

Crear una nueva máquina virtual a partir de la instantánea

1. Primero, restaure la instantánea en un nuevo PVC. Vaya a `Almacenamiento > Instantáneas de volumen`, haga clic en los puntos suspensivos junto a la instantánea que desea restaurar y haga clic en `Restaurar como nueva PVC`.
2. Ingrese los detalles del nuevo PVC y haga clic en `Restaurar`. Esto crea un nuevo 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. A continuación, cree una nueva VM a partir de esta PVC. Vaya a Virtualización > Máquinas virtuales y haga clic en Crear > Con YAML.
2. En la sección especificación > plantilla > especificación > volúmenes, especifique el nuevo PVC creado a partir de una instantánea en lugar de desde el disco contenedor. Proporcione todos los demás detalles para la nueva máquina virtual según sus requisitos.

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

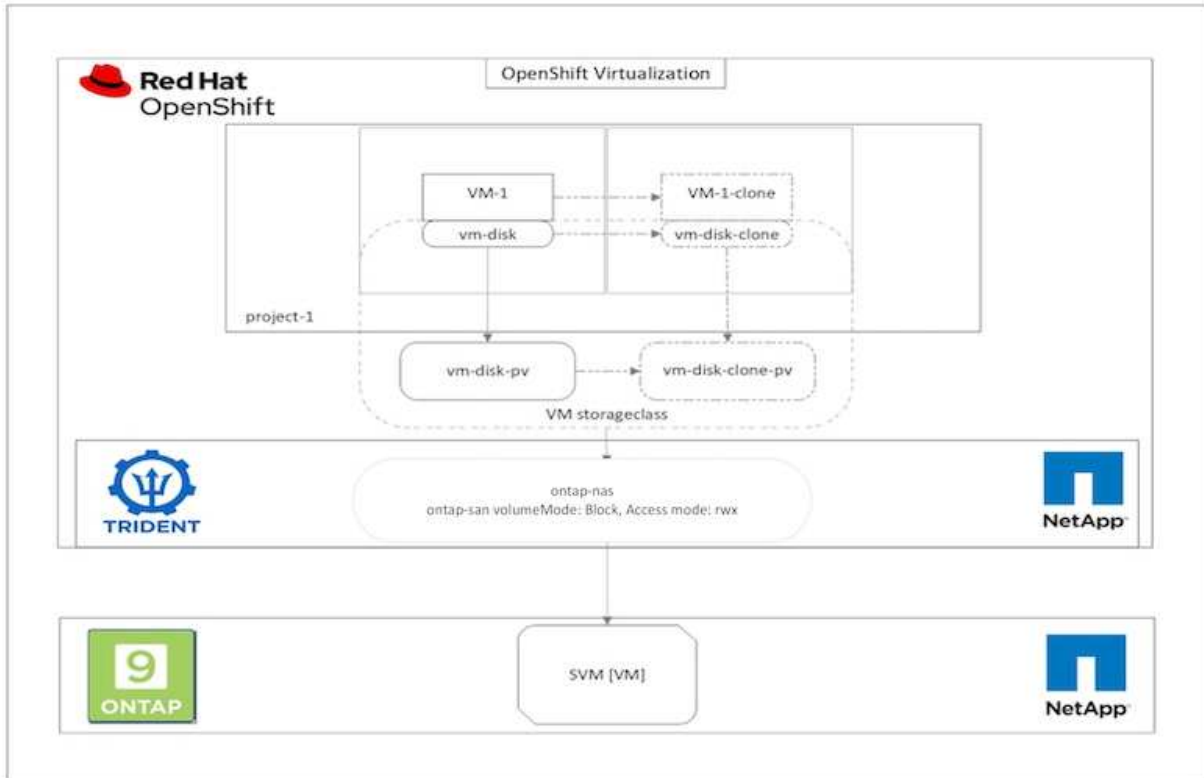
1. Haga clic en Crear para crear la nueva máquina virtual.
2. Una vez creada correctamente la VM, acceda y verifique que la nueva VM tenga el mismo estado que la VM cuyo PVC se utilizó para crear la instantánea en el momento en que se creó esta.

Clona una máquina virtual en OpenShift Virtualization usando Trident

Este procedimiento incluye el uso de la clonación de volúmenes de Trident CSI, lo que te permite crear una nueva máquina virtual apagando la máquina virtual de origen o dejándola en funcionamiento.

Clonación de máquinas virtuales

La clonación de una máquina virtual existente en OpenShift se logra con el soporte de la función de clonación Volume CSI de Trident. La clonación de volumen CSI permite la creación de un nuevo PVC utilizando un PVC existente como fuente de datos mediante la duplicación de su PV. Una vez creado el nuevo PVC, funciona como una entidad separada y sin ningún vínculo o dependencia del PVC de origen.



Existen ciertas restricciones a tener en cuenta con la clonación de volúmenes CSI:

1. El PVC de origen y el PVC de destino deben estar en el mismo proyecto.
2. Se admite la clonación dentro de la misma clase de almacenamiento.
3. La clonación solo se puede realizar cuando los volúmenes de origen y destino usan la misma configuración VolumeMode; por ejemplo, un volumen de bloque solo se puede clonar en otro volumen de bloque.

Las máquinas virtuales en un clúster OpenShift se pueden clonar de dos maneras:

1. Apagando la máquina virtual de origen
2. Manteniendo activa la máquina virtual de origen

Al apagar la máquina virtual de origen

Clonar una máquina virtual existente apagándola es una función nativa de OpenShift que se implementa con el soporte de Trident. Complete los siguientes pasos para clonar una máquina virtual.

1. Vaya a Cargas de trabajo > Virtualización > Máquinas virtuales y haga clic en los puntos suspensivos junto a la máquina virtual que desea clonar.
2. Haga clic en Clonar máquina virtual y proporcione los detalles de la nueva máquina virtual.

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. Haga clic en Clonar máquina virtual; esto apaga la máquina virtual de origen e inicia la creación de la máquina virtual clonada.
4. Una vez completado este paso, podrá acceder y verificar el contenido de la máquina virtual clonada.

Al mantener activa la máquina virtual de origen

También se puede clonar una VM existente clonando el PVC existente de la VM de origen y luego creando una nueva VM usando el PVC clonado. Este método no requiere que apague la máquina virtual de origen. Complete los siguientes pasos para clonar una máquina virtual sin apagarla.

1. Vaya a Almacenamiento > PersistentVolumeClaims y haga clic en los puntos suspensivos junto a la PVC que está conectada a la máquina virtual de origen.
2. Haga clic en Clonar PVC y proporcione los detalles del nuevo PVC.

Clone

Name *

Access Mode *

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

Size *

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. Luego haga clic en Clonar. Esto crea una PVC para la nueva VM.
4. Vaya a Cargas de trabajo > Virtualización > Máquinas virtuales y haga clic en Crear > Con YAML.
5. En la sección especificación > plantilla > especificación > volúmenes, adjunte el PVC clonado en lugar del disco contenedor. Proporcione todos los demás detalles para la nueva máquina virtual según sus requisitos.

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

1. Haga clic en Crear para crear la nueva máquina virtual.
2. Una vez creada correctamente la máquina virtual, acceda y verifique que la nueva máquina virtual sea un clon de la máquina virtual de origen.

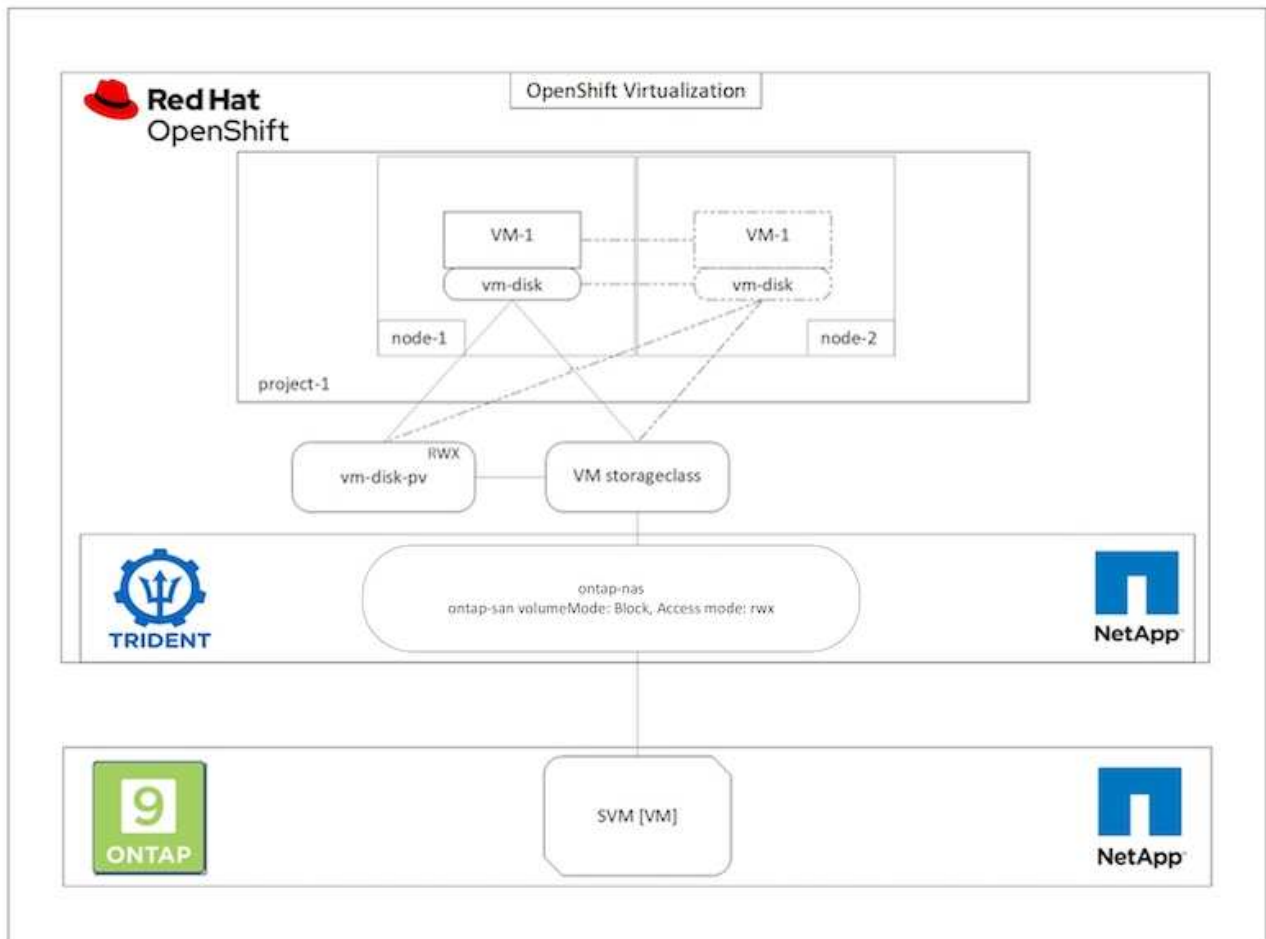
Migrar una máquina virtual entre dos nodos en un clúster de Red Hat OpenShift

Migre una VM en OpenShift Virtualization entre dos nodos en el clúster sin tiempo de inactividad. Este procedimiento incluye confirmar que los discos utilizan clases de almacenamiento compatibles con RWX, iniciar la migración y supervisar el progreso.

Migración en vivo de máquinas virtuales

La migración en vivo es un proceso de migración de una instancia de VM de un nodo a otro en un clúster OpenShift sin tiempo de inactividad. Para que la migración en vivo funcione en un clúster OpenShift, las máquinas virtuales deben estar vinculadas a PVC con modo de acceso ReadWriteMany compartido. Los backends Trident configurados mediante controladores ontap-nas admiten el modo de acceso RWX para los protocolos de sistema de archivos NFS y SMB. Consulte la documentación [aquí](#) . Los backends Trident configurados mediante controladores ontap-san admiten el modo de acceso RWX para el modo de volumen de bloque para los protocolos iSCSI y NVMe/TCP. Consulte la documentación [aquí](#) .

Por lo tanto, para que la migración en vivo tenga éxito, las máquinas virtuales deben estar provistas de discos (discos de arranque y discos hot plug adicionales) con PVC utilizando clases de almacenamiento ontap-nas o ontap-san (volumeMode: Block). Cuando se crean las PVC, Trident crea volúmenes ONTAP en un SVM habilitado para NFS o iSCSI.

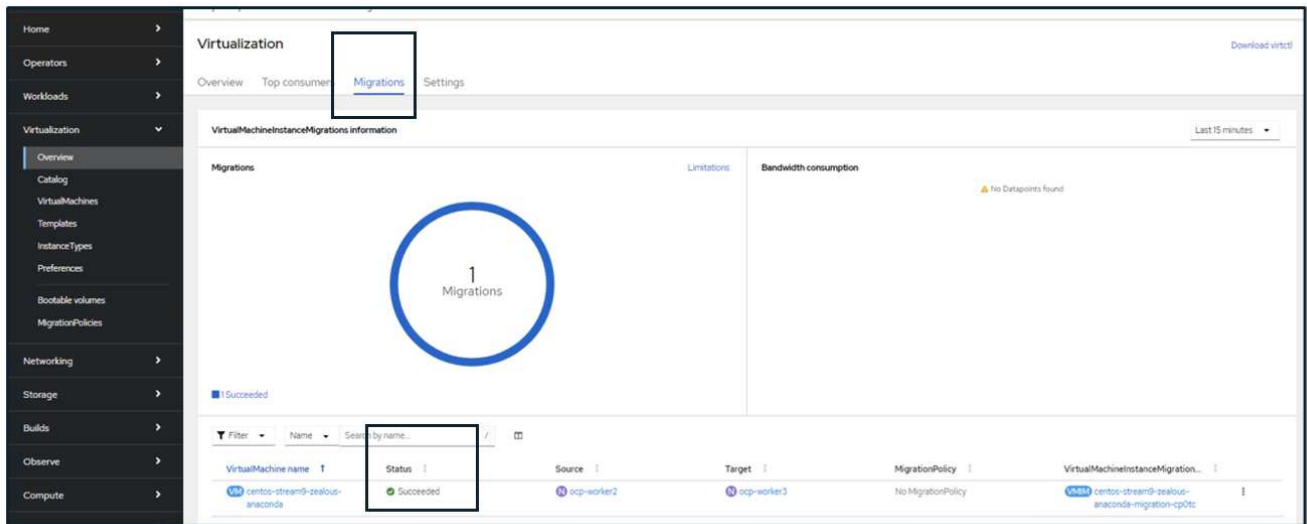
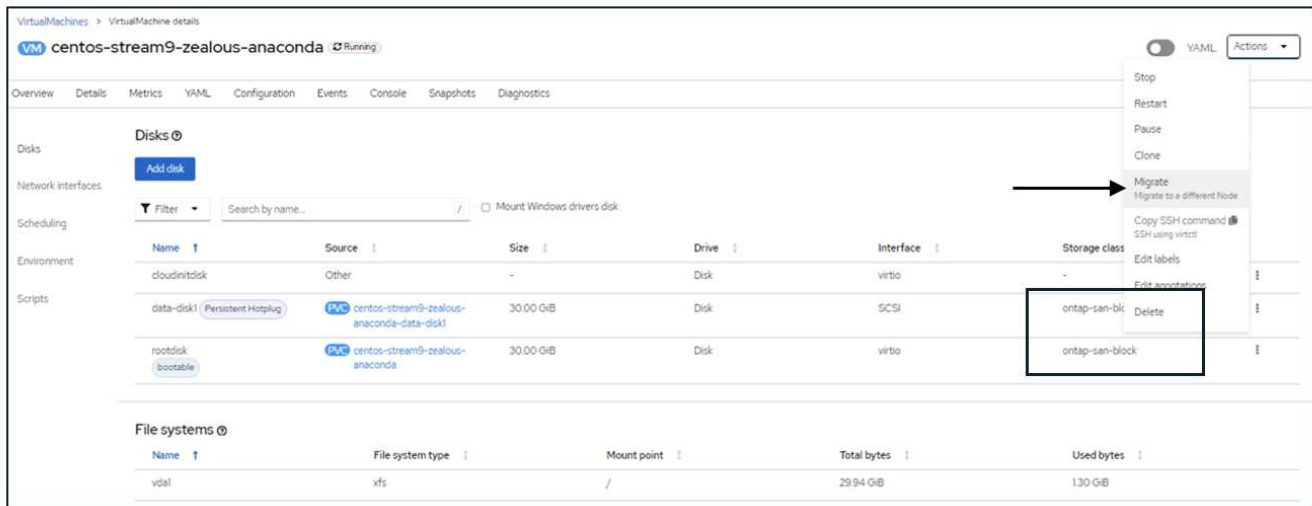


Para realizar una migración en vivo de una máquina virtual que se ha creado previamente y se encuentra en estado de ejecución, realice los siguientes pasos:

1. Seleccione la VM que desea migrar en vivo.
2. Haga clic en la pestaña **Configuración**.
3. Asegúrese de que todos los discos de la máquina virtual se creen utilizando clases de almacenamiento que admitan el modo de acceso RWX.
4. Haga clic en **Acciones** en la esquina derecha y luego seleccione **Migrar**.
5. Para ver el progreso de la migración, vaya a Virtualización > Descripción general en el menú del lado izquierdo y luego haga clic en la pestaña **Migraciones**. La migración de la máquina virtual pasará de **Pendiente** a **Programando** a **Realizada**.



Una instancia de VM en un clúster de OpenShift migra automáticamente a otro nodo cuando el nodo original se coloca en modo de mantenimiento si evictionStrategy está configurado en LiveMigrate.



Migración de máquinas virtuales entre clases de almacenamiento en OpenShift Virtualization

En OpenShift Virtualization, puedes migrar los discos de una máquina virtual (VM) a una clase de almacenamiento diferente directamente a través de la consola web sin eliminar la VM. Este proceso te permite mover todo el almacenamiento de la VM o volúmenes específicos.

Los siguientes pasos describen cómo migrar la clase de almacenamiento de una VM:

1. Asegúrate de que la clase de almacenamiento de destino esté correctamente configurada y disponible en tu clúster de OpenShift. Esto incluye tener configurado y accesible el backend de almacenamiento necesario (por ejemplo, NetApp ONTAP).

Mostrar ejemplo

```
[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. En la consola web de OpenShift Container Platform, navega a la perspectiva de Virtualización y selecciona la máquina virtual que quieres migrar.

VirtualMachines

VirtualMachine details

VM

rhel9-sc-migration

Running

Actions

Overview

Metrics

YAML

Configuration

Events

Console

Snapshots

Diagnostics

Details

Name

rhel9-sc-migration

Status

Running

Created

May 14, 2026, 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-b5-8d-fa

VirtualMachineInstance

rhel9-sc-migration

Pod

virt-launcher-rhel9-sc-migration-Sjkdw

Owner

No owner

Snapshots (0)

Take snapshot

VirtualMachines

VirtualMachine details

VM

rhel9-sc-migration

Running

YAML

Overview

Metrics

YAML

Configuration

Events

Console

Snapshots

Diagnostics

Details

Storage

Network

Scheduling

SSH

Initial run

Metadata

Disks

Add disk

Filter

Search by name...

Mount Windows drivers disk

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  3414499000B   RWX          sc-nas          <unset>          4m19s
[root@00-50-56-b5-2b-9b ~]#
```

- 57

Mostrar ejemplo

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

Namespace

Node

VirtualMachineInstance

Pod

Owner

Migrate VirtualMachine to a different Node

Migrate VirtualMachine storage to a different StorageClass

The VirtualMachine is running

VM

rhel9-sc-migration

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

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

sc-nas-economy

Back

Next

Cancel

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. Una vez finalizada la migración, puedes comprobar que los discos de la máquina virtual utilizan ahora la nueva clase de almacenamiento consultando los detalles de la máquina virtual en la consola web o mediante la CLI de OpenShift.

```
[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-d8busc	Bound	pvc-bb54342f-c01f-453e-864d-e8f72e71a2f	5361	RWX	sc-nas	<unset>	24m
persistentvolumeclaim/dv-rhel9-migration-disk-emerald-wren-75-d8busc-mig-839zfb	Bound	pvc-78a885f0-824a-407c-ab29-bcc546d8db3b	68321815673	RWX	sc-nas-economy	<unset>	2m17s
persistentvolumeclaim/rhel9-migration	Bound	pvc-46a34903-942f-4ddc-8ed4-9780808f0d09	34140899004	RWX	sc-nas	<unset>	24m

```
[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:
               [external-provisioner.volume.kubernetes.io/finalizer kubernetes.io/pv-protection external-attacher/csi-trident-netapp-io]
Finalizers:    sc-nas
StorageClass:  Bound
Claim:         test-sc-migration/rhel9-migration
Reclaim Policy: Delete
Access Modes:  RWX
VolumeMode:   Filesystem
Capacity:      341144990004
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%      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-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:
Labels:
Annotations:
pv.kubernetes.io/provisioned-by: csi.trident.netapp.io
volume.kubernetes.io/provisioner-deletion-secret-name:
volume.kubernetes.io/provisioner-deletion-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
VolumeMode: 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_OGBGMIAOLG/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_...

Información de copyright

Copyright © 2026 NetApp, Inc. Todos los derechos reservados. Imprimido en EE. UU. No se puede reproducir este documento protegido por copyright ni parte del mismo de ninguna forma ni por ningún medio (gráfico, electrónico o mecánico, incluidas fotocopias, grabaciones o almacenamiento en un sistema de recuperación electrónico) sin la autorización previa y por escrito del propietario del copyright.

El software derivado del material de NetApp con copyright está sujeto a la siguiente licencia y exención de responsabilidad:

ESTE SOFTWARE LO PROPORCIONA NETAPP «TAL CUAL» Y SIN NINGUNA GARANTÍA EXPRESA O IMPLÍCITA, INCLUYENDO, SIN LIMITAR, LAS GARANTÍAS IMPLÍCITAS DE COMERCIALIZACIÓN O IDONEIDAD PARA UN FIN CONCRETO, CUYA RESPONSABILIDAD QUEDA EXIMIDA POR EL PRESENTE DOCUMENTO. EN NINGÚN CASO NETAPP SERÁ RESPONSABLE DE NINGÚN DAÑO DIRECTO, INDIRECTO, ESPECIAL, EJEMPLAR O RESULTANTE (INCLUYENDO, ENTRE OTROS, LA OBTENCIÓN DE BIENES O SERVICIOS SUSTITUTIVOS, PÉRDIDA DE USO, DE DATOS O DE BENEFICIOS, O INTERRUPCIÓN DE LA ACTIVIDAD EMPRESARIAL) CUALQUIERA SEA EL MODO EN EL QUE SE PRODUJERON Y LA TEORÍA DE RESPONSABILIDAD QUE SE APLIQUE, YA SEA EN CONTRATO, RESPONSABILIDAD OBJETIVA O AGRAVIO (INCLUIDA LA NEGLIGENCIA U OTRO TIPO), QUE SURJAN DE ALGÚN MODO DEL USO DE ESTE SOFTWARE, INCLUSO SI HUBIEREN SIDO ADVERTIDOS DE LA POSIBILIDAD DE TALES DAÑOS.

NetApp se reserva el derecho de modificar cualquiera de los productos aquí descritos en cualquier momento y sin aviso previo. NetApp no asume ningún tipo de responsabilidad que surja del uso de los productos aquí descritos, excepto aquello expresamente acordado por escrito por parte de NetApp. El uso o adquisición de este producto no lleva implícita ninguna licencia con derechos de patente, de marcas comerciales o cualquier otro derecho de propiedad intelectual de NetApp.

Es posible que el producto que se describe en este manual esté protegido por una o más patentes de EE. UU., patentes extranjeras o solicitudes pendientes.

LEYENDA DE DERECHOS LIMITADOS: el uso, la copia o la divulgación por parte del gobierno están sujetos a las restricciones establecidas en el subpárrafo (b)(3) de los derechos de datos técnicos y productos no comerciales de DFARS 252.227-7013 (FEB de 2014) y FAR 52.227-19 (DIC de 2007).

Los datos aquí contenidos pertenecen a un producto comercial o servicio comercial (como se define en FAR 2.101) y son propiedad de NetApp, Inc. Todos los datos técnicos y el software informático de NetApp que se proporcionan en este Acuerdo tienen una naturaleza comercial y se han desarrollado exclusivamente con fondos privados. El Gobierno de EE. UU. tiene una licencia limitada, irrevocable, no exclusiva, no transferible, no sublicenciable y de alcance mundial para utilizar los Datos en relación con el contrato del Gobierno de los Estados Unidos bajo el cual se proporcionaron los Datos. Excepto que aquí se disponga lo contrario, los Datos no se pueden utilizar, desvelar, reproducir, modificar, interpretar o mostrar sin la previa aprobación por escrito de NetApp, Inc. Los derechos de licencia del Gobierno de los Estados Unidos de América y su Departamento de Defensa se limitan a los derechos identificados en la cláusula 252.227-7015(b) de la sección DFARS (FEB de 2014).

Información de la marca comercial

NETAPP, el logotipo de NETAPP y las marcas que constan en <http://www.netapp.com/TM> son marcas comerciales de NetApp, Inc. El resto de nombres de empresa y de producto pueden ser marcas comerciales de sus respectivos propietarios.