Buenas prácticas y recomendaciones

Trident

NetApp
July 01, 2026

Tabla de contenidos

Buenas prácticas y recomendaciones

Despliegue

Usa las recomendaciones que se indican aquí cuando despliegues Trident.

Implementa en un espacio de nombres dedicado

"Espacios de nombres" proporcionan separación administrativa entre diferentes aplicaciones y son una barrera para compartir recursos. Por ejemplo, una PVC de un espacio de nombres no puede consumirse desde otro. Trident proporciona recursos de PV a todos los espacios de nombres en el clúster de Kubernetes y, en consecuencia, aprovecha una cuenta de servicio que tiene privilegios elevados.

Además, el acceso al pod Trident podría permitir que un usuario acceda a las credenciales del sistema de almacenamiento y a otra información sensible. Es importante asegurarse de que los usuarios de las aplicaciones y las aplicaciones de gestión no tengan la capacidad de acceder a las definiciones de objetos Trident o a los propios pods.

Usa cuotas y límites de rango para controlar el consumo de almacenamiento

Kubernetes cuenta con dos funciones que, cuando se combinan, proporcionan un potente mecanismo para limitar el consumo de recursos por parte de las aplicaciones. El "mecanismo de cuota de almacenamiento" permite al administrador implementar límites de consumo de capacidad y recuento de objetos globales y específicos de la clase de almacenamiento por espacio de nombres. Además, usar un "límite de rango" garantiza que las solicitudes de PVC estén dentro de un valor mínimo y máximo antes de que la solicitud se reenvíe al provisionador.

Estos valores se definen por espacio de nombres, lo que significa que cada espacio de nombres debe tener valores definidos que se ajusten a sus necesidades de recursos. Mira aquí para obtener información sobre "cómo aprovechar las cuotas".

Configuración de almacenamiento

Cada plataforma de almacenamiento en el portafolio de NetApp tiene capacidades únicas que benefician a las aplicaciones, estén o no en contenedores.

Descripción general de la plataforma

Trident es compatible con ONTAP y Element. No existe una plataforma que se adapte mejor a todas las aplicaciones y escenarios que otra, sin embargo, al elegir una plataforma, se deben tener en cuenta las necesidades de la aplicación y del equipo que administra el dispositivo.

Deberías seguir las prácticas recomendadas básicas para el sistema operativo host con el protocolo que estás usando. Opcionalmente, podrías considerar incorporar las prácticas recomendadas de la aplicación, cuando estén disponibles, junto con la configuración de backend, storage class y PVC para optimizar el almacenamiento para aplicaciones específicas.

Mejores prácticas de ONTAP y Cloud Volumes ONTAP

Aprende las mejores prácticas para configurar ONTAP y Cloud Volumes ONTAP para Trident.

Las siguientes recomendaciones son pautas para configurar ONTAP para cargas de trabajo en contenedores que consumen volúmenes aprovisionados dinámicamente por Trident. Cada una debe considerarse y evaluarse para determinar su idoneidad en tu entorno.

Usa SVM dedicados a Trident

Las máquinas virtuales de almacenamiento (SVM) proporcionan aislamiento y separación administrativa entre los inquilinos en un sistema ONTAP. Dedicar una SVM a las aplicaciones permite la delegación de privilegios y permite aplicar las mejores prácticas para limitar el consumo de recursos.

Hay varias opciones disponibles para la gestión del SVM:

- Proporciona la interfaz de administración del clúster en la configuración del backend, junto con las credenciales adecuadas, y especifica el nombre de SVM.
- Crea una interfaz de administración dedicada para la SVM usando ONTAP System Manager o la CLI.
- Comparte el rol de administración con una interfaz de datos NFS.

En cada caso, la interfaz debe estar en DNS y el nombre DNS debe usarse al configurar Trident. Esto ayuda a facilitar algunos escenarios de recuperación ante desastres, por ejemplo, SVM-DR sin usar la retención de identidad de red.

No hay preferencia entre tener un LIF de administración dedicado o compartido para la SVM, pero debes asegurarte de que tus políticas de seguridad de red se alineen con el enfoque que elijas. De todas formas, el LIF de administración debe ser accesible mediante DNS para facilitar la máxima flexibilidad si "SVM-DR" se usa junto con Trident.

Limitar el recuento máximo de volúmenes

Los sistemas de almacenamiento ONTAP tienen un recuento máximo de volúmenes, que varía según la versión del software y la plataforma de hardware. Consulta "NetApp Hardware Universe" para tu plataforma específica y versión de ONTAP para determinar los límites exactos. Cuando se agota el recuento de volúmenes, las operaciones de aprovisionamiento fallan no solo para Trident, sino para todas las solicitudes de almacenamiento.

Los controladores de Trident `ontap-nas` y `ontap-san` aprovisionan un FlexVolume por cada volumen persistente (PV) de Kubernetes que se crea. El controlador `ontap-nas-economy` crea aproximadamente un FlexVolume por cada 200 PV (configurable entre 50 y 300). El controlador `ontap-san-economy` crea aproximadamente un FlexVolume por cada 100 PV (configurable entre 50 y 200). Para evitar que Trident consuma todos los volúmenes disponibles en el sistema de almacenamiento, deberías establecer un límite en la SVM. Puedes hacer esto desde la línea de comandos:

```
vserver modify -vserver <svm_name> -max-volumes <num_of_volumes>
```

El valor de `max-volumes` varía según varios criterios específicos de tu entorno:

- La cantidad de volúmenes existentes en el clúster de ONTAP
- La cantidad de volúmenes que esperas aprovisionar fuera de Trident para otras aplicaciones
- La cantidad de volúmenes persistentes que se espera que consuman las aplicaciones de Kubernetes

El `max-volumes` valor es la cantidad total de volúmenes aprovisionados en todos los nodos del clúster de ONTAP, y no en un nodo individual de ONTAP. Como resultado, podrías encontrarte con situaciones en las

que un nodo del clúster de ONTAP tenga muchos más o menos volúmenes aprovisionados con Trident que otro nodo.

Por ejemplo, un clúster ONTAP de dos nodos tiene la capacidad de alojar un máximo de 2000 FlexVol volúmenes. Tener el recuento máximo de volúmenes establecido en 1250 parece muy razonable. Sin embargo, si solo "agregados" de un nodo se asignan a la SVM, o los agregados asignados de un nodo no pueden aprovisionarse (por ejemplo, debido a la capacidad), entonces el otro nodo se convierte en el destino de todos los volúmenes aprovisionados por Trident. Esto significa que el límite de volúmenes podría alcanzarse para ese nodo antes de que se alcance el valor de `max-volumes`, lo que impacta tanto a Trident como a otras operaciones de volúmenes que usan ese nodo. **Puedes evitar esta situación asegurándote de que los agregados de cada nodo del clúster se asignen a la SVM que usa Trident en cantidades iguales.**

Clonar un volumen

NetApp Trident admite la clonación de volúmenes cuando se usan los `ontap-nas`, `ontap-san` y `solidfire-san` controladores de almacenamiento. Cuando se usan los `ontap-nas-flexgroup` o `ontap-nas-economy` controladores, la clonación no está soportada. Crear un volumen nuevo a partir de un volumen existente hará que se cree una nueva instantánea.

> Evita clonar un PVC que esté asociado con un StorageClass diferente. Realiza operaciones de clonación dentro del mismo StorageClass para garantizar la compatibilidad y evitar comportamientos inesperados.

Limita el tamaño máximo de los volúmenes creados por Trident

Para configurar el tamaño máximo para los volúmenes que puede crear Trident, usa el `limitVolumeSize` parámetro en tu `backend.json` definición.

Además de controlar el tamaño del volumen en la matriz de almacenamiento, también deberías aprovechar las capacidades de Kubernetes.

Limita el tamaño máximo de los FlexVols creados por Trident

Para configurar el tamaño máximo para los FlexVols usados como pools para los controladores ontap-san-economy y ontap-nas-economy, usa el parámetro `limitVolumePoolSize` en tu definición `backend.json`.

Configura Trident para usar CHAP bidireccional

Puedes especificar los nombres de usuario y las contraseñas del iniciador y del destino CHAP en la definición del backend y hacer que Trident habilite CHAP en la SVM. Usando el `useCHAP` parámetro en la configuración del backend, Trident autentica las conexiones iSCSI para backends ONTAP con CHAP.

Crear y usar una política de QoS de SVM

El uso de una política de QoS de ONTAP aplicada a la SVM limita la cantidad de IOPS consumibles por los volúmenes aprovisionados por Trident. Esto ayuda a "prevenir a un acosador"evitar que un contenedor fuera de control afecte las cargas de trabajo fuera de la SVM de Trident.

Puedes crear una política de QoS para la SVM en pocos pasos. Consulta la documentación de tu versión de ONTAP para la información más precisa. El siguiente ejemplo crea una política de QoS que limita el total de IOPS disponibles para la SVM a 5000.

```
# create the policy group for the SVM
qos policy-group create -policy-group <policy_name> -vserver <svm_name>
-max-throughput 5000iops

# assign the policy group to the SVM, note this will not work
# if volumes or files in the SVM have existing QoS policies
vserver modify -vserver <svm_name> -qos-policy-group <policy_name>
```

Además, si tu versión de ONTAP lo admite, puedes considerar usar un mínimo de QoS para garantizar una cantidad de ancho de banda para las cargas de trabajo en contenedores. La QoS adaptativa no es compatible con una política a nivel de SVM.

La cantidad de IOPS dedicadas a las cargas de trabajo en contenedores depende de muchos aspectos. Entre otras cosas, se incluyen:

- Otras cargas de trabajo que utilizan la matriz de almacenamiento. Si existen otras cargas de trabajo, no relacionadas con la implementación de Kubernetes, que utilizan los recursos de almacenamiento, se debe tener cuidado para asegurarse de que esas cargas de trabajo no se vean afectadas accidentalmente.
- Cargas de trabajo esperadas ejecutándose en contenedores. Si cargas de trabajo con altos requisitos de IOPS se ejecutan en contenedores, una política de QoS baja resulta en una mala experiencia.

Es importante recordar que una política de QoS asignada a nivel de SVM implica que todos los volúmenes aprovisionados en la SVM compartan el mismo grupo de IOPS. Si una o un pequeño número de aplicaciones contenedorizadas tienen un alto requerimiento de IOPS, podría convertirse en un obstáculo para las demás cargas de trabajo contenedorizadas. Si este es el caso, podrías considerar usar automatización externa para asignar políticas de QoS por volumen.

> Deberías asignar el grupo de políticas QoS al SVM **solo** si tu versión de ONTAP es anterior a 9.8.

Crea grupos de políticas de QoS para Trident

La calidad de servicio (QoS) garantiza que el rendimiento de las cargas de trabajo críticas no se degrade por cargas de trabajo competidoras. Los grupos de políticas de QoS de ONTAP ofrecen opciones de QoS para volúmenes y permiten a los usuarios definir el límite de rendimiento para una o más cargas de trabajo. Para más información sobre QoS, consulta "Garantizar el rendimiento con QoS". Puedes especificar grupos de políticas de QoS en el backend o en un pool de almacenamiento, y se aplican a cada volumen creado en ese pool o backend.

ONTAP cuenta con dos tipos de grupos de políticas de QoS: tradicionales y adaptativos. Los grupos de políticas tradicionales proporcionan un máximo fijo (o mínimo, en versiones posteriores) de rendimiento en IOPS. La QoS adaptativa ajusta automáticamente el rendimiento al tamaño de la carga de trabajo, manteniendo la proporción de IOPS a TB o GB a medida que cambia el tamaño de la carga de trabajo. Esto proporciona una ventaja significativa cuando gestionas cientos o miles de cargas de trabajo en una implementación grande.

Ten en cuenta lo siguiente al crear grupos de políticas de QoS:

- Debes configurar la `qosPolicy`clave en el bloque `defaults` de la configuración del backend. Consulta el siguiente ejemplo de configuración del backend:

```
---
version: 1
storageDriverName: ontap-nas
managementLIF: 0.0.0.0
dataLIF: 0.0.0.0
svm: svm0
username: user
password: pass
defaults:
  qosPolicy: standard-pg
storage:
  - labels:
      performance: extreme
    defaults:
      adaptiveQosPolicy: extremely-adaptive-pg
  - labels:
      performance: premium
    defaults:
      qosPolicy: premium-pg
```

- Deberías aplicar los grupos de políticas por volumen, así cada volumen obtiene todo el rendimiento especificado por el grupo de políticas. No se admiten los grupos de políticas compartidos.

Para obtener más información sobre los grupos de políticas de QoS, consulta "Referencia de comandos de ONTAP".

Limita el acceso a los recursos de almacenamiento solo a los miembros del clúster de Kubernetes

Limitar el acceso a los volúmenes NFS, LUN iSCSI y LUN FC creados por Trident es un componente fundamental de la seguridad para tu implementación de Kubernetes. Hacer esto evita que los hosts que no forman parte del clúster de Kubernetes accedan a los volúmenes y puedan modificar los datos de forma inesperada.

Es importante comprender que los espacios de nombres son el límite lógico para los recursos en Kubernetes. Se asume que los recursos en el mismo espacio de nombres se pueden compartir, pero, lo importante es que no existe capacidad entre espacios de nombres. Esto significa que, aunque los PV son objetos globales, cuando se vinculan a un PVC solo son accesibles por pods que están en el mismo espacio de nombres. **Es fundamental asegurarse de que los espacios de nombres se usen para proporcionar separación cuando corresponde.**

La principal preocupación de la mayoría de las organizaciones con respecto a la seguridad de los datos en un contexto de Kubernetes es que un proceso en un contenedor pueda acceder al almacenamiento montado en el host, pero que no está destinado al contenedor. "Espacios de nombres" están diseñados para evitar este tipo de vulnerabilidad. Sin embargo, hay una excepción: los contenedores privilegiados.

Un contenedor privilegiado es aquel que se ejecuta con muchos más permisos a nivel de host de lo normal. Estos no se deniegan por defecto, así que asegúrate de deshabilitar la capacidad usando "políticas de seguridad de pod".

Para volúmenes a los que se desea acceder tanto desde Kubernetes como desde hosts externos, el

almacenamiento debe gestionarse de forma tradicional, con el PV introducido por el administrador y no gestionado por Trident. Esto asegura que el volumen de almacenamiento se destruya solo cuando tanto Kubernetes como los hosts externos se hayan desconectado y ya no estén usando el volumen. Además, se puede aplicar una política de exportación personalizada, que permite el acceso desde los nodos del clúster de Kubernetes y los servidores de destino fuera del clúster de Kubernetes.

Para implementaciones con nodos de infraestructura dedicados (por ejemplo, OpenShift) u otros nodos que no pueden programar aplicaciones de usuario, se deben usar políticas de exportación independientes para limitar aún más el acceso a los recursos de almacenamiento. Esto incluye crear una política de exportación para los servicios que se implementan en esos nodos de infraestructura (por ejemplo, los servicios de Métricas y Registro de OpenShift) y para las aplicaciones estándar que se implementan en nodos que no son de infraestructura.

Usa una política de exportación dedicada

Debes asegurarte de que exista una política de exportación para cada backend que solo permita el acceso a los nodos presentes en el clúster de Kubernetes. Trident puede crear y administrar políticas de exportación automáticamente. Así, Trident limita el acceso a los volúmenes que aprovisiona a los nodos del clúster de Kubernetes y simplifica la adición y eliminación de nodos.

Como alternativa, también puedes crear una política de exportación manualmente y llenarla con una o más reglas de exportación que procesen cada solicitud de acceso de nodo:

- Usa el `vserver export-policy create` comando CLI de ONTAP para crear la política de exportación.
- Agrega reglas a la política de exportación usando el comando CLI de `vserver export-policy rule create` ONTAP.

Al ejecutar estos comandos puedes restringir qué nodos de Kubernetes tienen acceso a los datos.

Deshabilita `showmount` para la aplicación SVM

La `showmount`función permite que un cliente de NFS consulte la SVM para obtener una lista de las exportaciones de NFS disponibles. Un pod implementado en el clúster de Kubernetes puede ejecutar el `showmount -e`comando contra la SVM y recibir una lista de los montajes disponibles, incluidos aquellos a los que no tiene acceso. Aunque esto por sí solo no es una vulnerabilidad de seguridad, sí proporciona información innecesaria que podría ayudar a un usuario no autorizado a conectarse a una exportación de NFS.

Debes deshabilitar `showmount` usando el comando de la línea de comandos de ONTAP a nivel SVM:

```
vserver nfs modify -vserver <svm_name> -showmount disabled
```

SolidFire mejores prácticas

Conoce las mejores prácticas para configurar el almacenamiento SolidFire para Trident.

Crear cuenta de SolidFire

Cada cuenta de SolidFire representa a un propietario de volumen único y recibe su propio conjunto de credenciales de Challenge-Handshake Authentication Protocol (CHAP). Puedes acceder a los volúmenes asignados a una cuenta usando el nombre de la cuenta y las credenciales CHAP correspondientes o a través

de un grupo de acceso a volúmenes. Una cuenta puede tener hasta dos mil volúmenes asignados, pero un volumen solo puede pertenecer a una cuenta.

Crear una política de QoS

Usa las políticas de calidad de servicio (QoS) de SolidFire si quieres crear y guardar una configuración de calidad de servicio estandarizada que puedas aplicar a muchos volúmenes.

Puedes configurar los parámetros de QoS por volumen. El rendimiento de cada volumen se puede garantizar configurando tres parámetros configurables que definen la QoS: Min IOPS, Max IOPS y Burst IOPS.

Aquí tienes los posibles valores mínimos, máximos y de ráfaga de IOPS para el tamaño de bloque de 4Kb.

Parámetro IOPS	Definición	Valor mínimo	Valor predeterminado	Valor máximo (4Kb)
IOPS mínimas	El nivel de rendimiento garantizado para un volumen.	50	50	15000
IOPS máximo	El rendimiento no superará este límite.	50	15000	200.000
Burst IOPS	IOPS máximos permitidos en un escenario de ráfaga corta.	50	15000	200.000

> Aunque los IOPS máximos y los IOPS en ráfaga se pueden configurar hasta 200,000, el rendimiento máximo real de un volumen está limitado por el uso del clúster y el rendimiento por nodo.

El tamaño de bloque y el ancho de banda influyen directamente en el número de IOPS. A medida que aumenta el tamaño de bloque, el sistema aumenta el ancho de banda hasta el nivel necesario para procesar los tamaños de bloque más grandes. A medida que aumenta el ancho de banda, disminuye el número de IOPS que el sistema puede alcanzar. Consulta "Calidad de servicio de SolidFire" para más información sobre QoS y rendimiento.

Autenticación de SolidFire

Element admite dos métodos de autenticación: CHAP y Grupos de Acceso a Volumen (VAG). CHAP utiliza el protocolo CHAP para autenticar el host en el backend. Los Grupos de Acceso a Volumen controlan el acceso a los volúmenes que aprovisiona. NetApp recomienda usar CHAP para la autenticación, ya que es más sencillo y no tiene límites de escalabilidad.

> Trident con el aprovisionador CSI mejorado admite el uso de autenticación CHAP. Los VAG solo deben usarse en el modo de funcionamiento tradicional sin CSI.

La autenticación CHAP (verificación de que el iniciador es el usuario previsto del volumen) solo se admite con control de acceso basado en cuentas. Si usas CHAP para la autenticación, hay dos opciones disponibles: CHAP unidireccional y CHAP bidireccional. El CHAP unidireccional autentica el acceso al volumen usando el

nombre de la cuenta SolidFire y el secreto del iniciador. La opción CHAP bidireccional ofrece la forma más segura de autenticar el volumen porque el volumen autentica el host mediante el nombre de la cuenta y el secreto del iniciador, y luego el host autentica el volumen mediante el nombre de la cuenta y el secreto del destino.

Sin embargo, si no se puede habilitar CHAP y se requieren VAG, crea el grupo de acceso y añade los iniciadores de host y los volúmenes al grupo de acceso. Cada IQN que añades a un grupo de acceso puede acceder a cada volumen del grupo con o sin autenticación CHAP. Si el iniciador iSCSI está configurado para usar autenticación CHAP, se utiliza el control de acceso basado en cuentas. Si el iniciador iSCSI no está configurado para usar autenticación CHAP, entonces se utiliza el control de acceso del grupo de acceso a volúmenes.

¿Dónde encontrar más información?

A continuación, se incluye documentación sobre las mejores prácticas. Busca en "Biblioteca de NetApp" las versiones más recientes.

ONTAP

- "Guía de mejores prácticas e implementación de NFS"
- "Administración de SAN" (para iSCSI)
- "Configuración Express de iSCSI para RHEL"

Software Element

- "Configurar SolidFire para Linux"

NetApp HCI

- "Requisitos previos para la implementación de NetApp HCI"
- "Accede al NetApp Deployment Engine"

Información sobre las mejores prácticas de aplicación

- "Mejores prácticas para MySQL en ONTAP"
- "Prácticas recomendadas para MySQL en SolidFire"
- "NetApp SolidFire y Cassandra"
- "Prácticas recomendadas de Oracle en SolidFire"
- "Mejores prácticas de PostgreSQL en SolidFire"

No todas las aplicaciones tienen directrices específicas, es importante trabajar con tu equipo de NetApp y usar el "Biblioteca de NetApp" para encontrar la documentación más actualizada.

Integrar Trident

Para integrar Trident, los siguientes elementos de diseño y arquitectura requieren integración: selección e implementación de controladores, diseño de clases de almacenamiento, diseño de grupos virtuales, impactos de Persistent Volume Claim (PVC) en el aprovisionamiento de almacenamiento, operaciones de volumen y despliegue de servicios de OpenShift usando Trident.

Selección y despliegue de controladores

Selecciona e implementa un controlador de backend para tu sistema de almacenamiento.

Controladores de backend de ONTAP

Los controladores backend de ONTAP se diferencian por el protocolo utilizado y cómo se aprovisionan los volúmenes en el sistema de almacenamiento. Por lo tanto, piensa bien cuál controlador vas a implementar.

A un nivel superior, si tu aplicación tiene componentes que necesitan almacenamiento compartido (varios pods accediendo al mismo PVC), los controladores basados en NAS serían la opción predeterminada, mientras que los controladores iSCSI basados en bloques cubren las necesidades de almacenamiento no compartido. Elige el protocolo según los requisitos de la aplicación y el nivel de comodidad de los equipos de almacenamiento e infraestructura. En general, hay poca diferencia entre ellos para la mayoría de las aplicaciones, así que muchas veces la decisión se basa en si se necesita o no almacenamiento compartido (donde más de un pod necesitará acceso simultáneo).

Los controladores de backend ONTAP disponibles son:

- `ontap-nas`: Cada PV aprovisionado es un ONTAP completo FlexVolume.
- `ontap-nas-economy`: Cada PV aprovisionado es un qtree, con una cantidad configurable de qtrees por FlexVolume (el valor predeterminado es 200).
- `ontap-nas-flexgroup`: Cada PV se aprovisiona como un ONTAP FlexGroup completo, y se utilizan todos los agregados asignados a un SVM.
- `ontap-san`: cada PV aprovisionado es un LUN dentro de su propio FlexVolume.
- `ontap-san-economy`: cada PV aprovisionado es un LUN, con una cantidad configurable de LUN por FlexVolume (el valor predeterminado es 100).

Elegir entre los tres controladores NAS tiene algunas ramificaciones en las funciones que se ponen a disposición de la aplicación.

Ten en cuenta que, en las tablas a continuación, no todas las capacidades se exponen a través de Trident. Algunas deben ser aplicadas por el administrador de almacenamiento después del aprovisionamiento si se desea esa funcionalidad. Las notas al pie en superíndice distinguen la funcionalidad por característica y controlador.

Controladores NAS de ONTAP	Instantáneas	Clones	Políticas dinámicas de exportación	Conexión múltiple	QoS	Cambiar tamaño	Replicación
`ontap-nas`	Sí	Sí	Sí nota al pie: 5[]	Sí	Sí nota al pie: 1[]	Sí	Sí nota al pie: 1[]
`ontap-nas-economy`	NO [3]	NO [3]	Sí nota al pie: 5[]	Sí	NO [3]	Sí	NO [3]
`ontap-nas-flexgroup`	Sí nota al pie: 1[]	NO	Sí nota al pie: 5[]	Sí	Sí nota al pie: 1[]	Sí	Sí nota al pie: 1[]

Trident ofrece 2 controladores SAN para ONTAP, cuyas capacidades se muestran a continuación.

Controladores SAN de ONTAP	Instantáneas	Clones	Conexión múltiple	CHAP bidireccional	QoS	Cambiar tamaño	Replicación
`ontap-san`	Sí	Sí	Sí nota al pie: 4[]	Sí	Sí nota al pie: 1[]	Sí	Sí nota al pie: 1[]
`ontap-san-economy`	Sí	Sí	Sí nota al pie: 4[]	Sí	NO [3]	Sí	NO [3]

> Nota al pie para las tablas anteriores: Sí [1]: No administrado por Trident Sí [2]: Administrado por Trident, pero no granular de PV NO [3]: No administrado por Trident y no granular de PV Sí [4]: Compatible con volúmenes raw-block Sí [5]: Compatible con Trident

Las características que no son granulares de PV se aplican a todo el FlexVolume y todos los PV (es decir, qtrees o LUNs en FlexVols compartidos) compartirán una programación común.

Como podemos ver en las tablas anteriores, gran parte de la funcionalidad entre el `ontap-nas` y el `ontap-nas-economy` es la misma. Sin embargo, porque el `ontap-nas-economy` driver limita la capacidad de controlar la programación con granularidad por PV, esto puede afectar tu planificación de recuperación ante desastres y copias de seguridad en particular. Para los equipos de desarrollo que quieren aprovechar la funcionalidad de clonado de PVC en almacenamiento ONTAP, esto solo es posible cuando usas los `ontap-nas`, `ontap-san` o `ontap-san-economy` drivers.

> El `solidfire-san`controlador también es capaz de clonar PVCs.

Controladores backend de Cloud Volumes ONTAP

Cloud Volumes ONTAP proporciona control de datos junto con funciones de almacenamiento para la gran empresa para varios casos de uso, incluyendo recursos compartidos de archivos y almacenamiento a nivel de bloque que sirven protocolos NAS y SAN (NFS, SMB / CIFS e iSCSI). Los controladores compatibles para Cloud Volume ONTAP son `ontap-nas`, `ontap-nas-economy`, `ontap-san` y `ontap-san-economy`. Estos son aplicables para Cloud Volume ONTAP para Azure, Cloud Volume ONTAP para GCP.

Controladores de backend de Amazon FSx for ONTAP

Amazon FSx for NetApp ONTAP te permite aprovechar las características, el rendimiento y las capacidades administrativas de NetApp que ya conoces, mientras aprovechas la simplicidad, agilidad, seguridad y escalabilidad de almacenar datos en AWS. FSx for ONTAP es compatible con muchas características del sistema de archivos ONTAP y APIs de administración. Los controladores compatibles para Cloud Volume ONTAP son `ontap-nas`, `ontap-nas-economy`, `ontap-nas-flexgroup`, `ontap-san` y `ontap-san-economy`.

NetApp HCI/SolidFire controladores backend

El `solidfire-san` controlador utilizado con las plataformas NetApp HCI/SolidFire ayuda al admin a configurar un backend de Element para Trident según los límites de QoS. Si quieres diseñar tu backend para establecer límites de QoS específicos en los volúmenes aprovisionados por Trident, usa el `type` parámetro en el archivo del backend. El admin también puede restringir el tamaño del volumen que se puede crear en el almacenamiento usando el `limitVolumeSize` parámetro. Actualmente, las funciones de almacenamiento de Element como el redimensionamiento y la replicación de volúmenes no están soportadas a través del

`solidfire-san` controlador. Estas operaciones deben hacerse manualmente desde la interfaz web de Element Software.

Controlador de SolidFire	Instantáneas	Clones	Conexión múltiple	CHAP	QoS	Cambiar tamaño	Replicación
`solidfire-san`	Sí	Sí	Sí	Sí	Sí	Sí	Sí nota al pie: 1[]

> Nota al pie: Sínota al pie:1[]: No administrado por Trident SÍnota al pie:2[]: Compatible con volúmenes de bloques sin procesar

Controladores de backend de Azure NetApp Files

Trident utiliza el `azure-netapp-files` driver para gestionar el "Azure NetApp Files" service.

Más información sobre este controlador y cómo configurarlo la puedes encontrar en "Configuración del backend de Trident para Azure NetApp Files".

Controlador de archivos de Azure NetApp Files	Instantáneas	Clones	Conexión múltiple	QoS	Expandir	Replicación
`azure-netapp-files`	Sí	Sí	Sí	Sí	Sí	Sí nota al pie: 1[]

> Nota al pie: Sínota al pie:1[]: No gestionado por Trident

Diseño de storage class

Es necesario configurar y aplicar clases de almacenamiento individuales para crear un objeto de clase de almacenamiento de Kubernetes. Esta sección explica cómo diseñar una clase de almacenamiento para tu aplicación.

Utilización específica del backend

El filtrado se puede usar dentro de un objeto de clase de almacenamiento específico para determinar qué grupo o conjunto de grupos de almacenamiento se usará con esa clase de almacenamiento específica. Se pueden configurar tres conjuntos de filtros en la clase de almacenamiento: `storagePools`, `additionalStoragePools`, y/o `excludeStoragePools`.

El `storagePools` parámetro ayuda a restringir el almacenamiento al conjunto de pools que coinciden con cualquier atributo especificado. El `additionalStoragePools` parámetro se usa para ampliar el conjunto de pools que Trident utiliza para el aprovisionamiento junto con el conjunto de pools seleccionados por los atributos y los parámetros `storagePools`. Puedes usar cualquiera de los parámetros por separado o ambos juntos para asegurarte de que se seleccione el conjunto adecuado de pools de almacenamiento.

El `excludeStoragePools`parámetro se usa para excluir específicamente el conjunto de pools que coinciden con los atributos.

Emular políticas de QoS

Si quieres diseñar Storage Classes para emular políticas de Quality of Service, crea una Storage Class con el `media` atributo como `hdd` o `ssd`. Según el `media` atributo mencionado en la storage class, Trident

seleccionará el backend adecuado que sirve `hdd` o `ssd` aggregates para que coincida con el atributo media y luego dirigirá el aprovisionamiento de los volúmenes al agregado específico. Por eso, podemos crear una storage class PREMIUM que tenga el `media` atributo establecido como `ssd`, lo que podría clasificarse como la política de QoS PREMIUM. Podemos crear otra storage class STANDARD que tenga el atributo media establecido como `hdd`, lo que podría clasificarse como la política de QoS STANDARD. También podemos usar el atributo ``IOPS'' en la storage class para redirigir el aprovisionamiento a un Element appliance, que puede definirse como una política de QoS.

Utiliza el backend en función de características específicas

Las clases de almacenamiento se pueden diseñar para dirigir el aprovisionamiento de volúmenes en un backend específico donde se habilitan funciones como aprovisionamiento fino y grueso, instantáneas, clones y cifrado. Para especificar qué almacenamiento usar, crea clases de almacenamiento que especifiquen el backend adecuado con la función requerida habilitada.

Pools virtuales

Los grupos virtuales están disponibles para todos los backends de Trident. Puedes definir grupos virtuales para cualquier backend usando cualquier controlador que Trident proporciona.

Los pools virtuales permiten a un administrador crear un nivel de abstracción sobre los backends que se puede referenciar mediante Storage Classes, para mayor flexibilidad y una colocación eficiente de volúmenes en los backends. Se pueden definir diferentes backends con la misma clase de servicio. Además, se pueden crear múltiples storage pools en el mismo backend pero con diferentes características. Cuando una Storage Class se configura con un selector con etiquetas específicas, Trident elige un backend que coincida con todas las etiquetas del selector para colocar el volumen. Si las etiquetas del selector de la Storage Class coinciden con varios storage pools, Trident elegirá uno de ellos para aprovisionar el volumen.

Diseño de pool virtual

Al crear un backend, generalmente puedes especificar un conjunto de parámetros. Era imposible para el administrador crear otro backend con las mismas credenciales de almacenamiento y con un conjunto diferente de parámetros. Con la introducción de los pools virtuales, este problema se ha aliviado. Un pool virtual es una abstracción de nivel introducida entre el backend y la Kubernetes Storage Class para que el administrador pueda definir parámetros junto con etiquetas que pueden ser referenciadas a través de las Kubernetes Storage Classes como un selector, de una manera independiente del backend. Se pueden definir pools virtuales para todos los backends compatibles de NetApp con Trident. Esa lista incluye SolidFire/NetApp HCI, ONTAP, así como Azure NetApp Files.

> Al definir grupos virtuales, se recomienda no reorganizar el orden de los grupos virtuales existentes en una definición de backend. También es recomendable no editar ni modificar los atributos de un grupo virtual existente y definir un nuevo grupo virtual en su lugar.

Emulando diferentes niveles de servicio/QoS

Es posible diseñar grupos virtuales para emular clases de servicio. Usando la implementación de grupo virtual de Cloud Volume Service para Azure NetApp Files, veamos cómo podemos configurar diferentes clases de servicio. Configura el backend de Azure NetApp Files con múltiples etiquetas que representen diferentes niveles de rendimiento. Establece el aspecto `servicelevel` en el nivel de rendimiento adecuado y agrega otros aspectos necesarios bajo cada etiqueta. Ahora crea diferentes clases de almacenamiento de Kubernetes que se asignarán a diferentes grupos virtuales. Usando el campo `parameters.selector`, cada StorageClass indica qué grupos virtuales se pueden usar para alojar un volumen.

Asignar un conjunto específico de aspectos

Se pueden diseñar varios pools virtuales con un conjunto específico de aspectos desde un único backend de almacenamiento. Para hacerlo, configura el backend con varias etiquetas y establece los aspectos necesarios bajo cada etiqueta. Ahora crea diferentes clases de almacenamiento de Kubernetes usando el campo `parameters.selector` que se asignará a distintos pools virtuales. Los volúmenes que se aprovisionen en el backend tendrán los aspectos definidos en el pool virtual elegido.

Características del PVC que afectan el aprovisionamiento de almacenamiento

Algunos parámetros más allá de la clase de almacenamiento solicitada pueden afectar el proceso de decisión de aprovisionamiento de Trident al crear un PVC.

Modo de acceso

Cuando solicitas almacenamiento a través de un PVC, uno de los campos obligatorios es el modo de acceso. El modo que elijas puede afectar el backend seleccionado para alojar la solicitud de almacenamiento.

Trident intentará hacer coincidir el protocolo de almacenamiento utilizado con el método de acceso especificado según la siguiente matriz. Esto es independiente de la plataforma de almacenamiento subyacente.

	ReadWriteOnce	ReadOnlyMany	ReadWriteMany
iSCSI	Sí	Sí	Sí (Raw block)
NFS	Sí	Sí	Sí

Una solicitud de un PVC de ReadWriteMany enviada a una implementación de Trident sin un backend NFS configurado dará como resultado que no se aprovisione ningún volumen. Por esta razón, el solicitante debe usar el modo de acceso que sea apropiado para su aplicación.

Operaciones de volumen

Modificar volúmenes persistentes

Los volúmenes persistentes son, con dos excepciones, objetos inmutables en Kubernetes. Una vez creados, la política de recuperación y el tamaño se pueden modificar. Sin embargo, esto no impide que algunos aspectos del volumen se modifiquen fuera de Kubernetes. Esto puede ser deseable para personalizar el volumen para aplicaciones específicas, para asegurarte de que la capacidad no se consuma accidentalmente o simplemente para mover el volumen a un controlador de almacenamiento diferente por cualquier motivo.

> Por el momento, los provisionadores en árbol de Kubernetes no admiten operaciones de redimensionamiento de volúmenes para NFS, iSCSI o FC PV. Trident admite la expansión de volúmenes NFS, iSCSI y FC.

Los detalles de conexión del PV no se pueden modificar después de la creación.

Crea instantáneas de volumen bajo demanda

Trident admite la creación de instantáneas de volumen bajo demanda y la creación de PVCs a partir de instantáneas mediante el framework CSI. Las instantáneas proporcionan un método práctico para mantener copias de un momento específico de los datos y tienen un ciclo de vida independiente del PV de origen en Kubernetes. Estas instantáneas se pueden usar para clonar PVCs.

Crear volúmenes a partir de instantáneas

Trident también admite la creación de PersistentVolumes a partir de instantáneas de volúmenes. Para lograr esto, solo tienes que crear un PersistentVolumeClaim y mencionar el `datasource` como la instantánea requerida desde la cual se debe crear el volumen. Trident gestionará este PVC creando un volumen con los datos presentes en la instantánea. Con esta función, puedes duplicar datos entre regiones, crear entornos de prueba, reemplazar un volumen de producción dañado o corrupto por completo, o recuperar archivos y directorios específicos y transferirlos a otro volumen adjunto.

Mover volúmenes en el clúster

Los administradores de almacenamiento tienen la capacidad de mover volúmenes entre agregados y controladores en el clúster ONTAP de forma no disruptiva para el consumidor de almacenamiento. Esta operación no afecta a Trident ni al clúster Kubernetes, siempre y cuando el agregado de destino sea uno al que la SVM que usa Trident tenga acceso. Lo importante es que, si el agregado se ha añadido recientemente a la SVM, será necesario actualizar el backend volviéndolo a agregar a Trident. Esto hará que Trident vuelva a inventariar la SVM para que se reconozca el nuevo agregado.

Sin embargo, mover volúmenes entre backends no es compatible automáticamente por Trident. Esto incluye entre SVMs en el mismo clúster, entre clústeres o en una plataforma de almacenamiento diferente (incluso si ese sistema de almacenamiento está conectado a Trident).

Si se copia un volumen a otra ubicación, se puede usar la función de importación de volúmenes para importar los volúmenes actuales en Trident.

Ampliar volúmenes

Trident admite el cambio de tamaño de NFS, iSCSI y FC PVs. Esto permite a los usuarios cambiar el tamaño de sus volúmenes directamente a través de la capa de Kubernetes. La expansión del volumen es posible para todas las principales plataformas de almacenamiento NetApp, incluyendo ONTAP y backends SolidFire/NetApp HCI. Para permitir una posible expansión después, establece `allowVolumeExpansion` en `true` en tu StorageClass asociado con el volumen. Cuando sea necesario cambiar el tamaño del volumen persistente, edita la anotación `spec.resources.requests.storage` en la Persistent Volume Claim al tamaño de volumen requerido. Trident se encargará automáticamente de cambiar el tamaño del volumen en el clúster de almacenamiento.

Importa un volumen existente en Kubernetes

La importación de volúmenes ofrece la posibilidad de importar un volumen de almacenamiento existente en un entorno Kubernetes. Esto es compatible actualmente con los controladores `ontap-nas`, `ontap-nas-flexgroup`, `solidfire-san` y `azure-netapp-files`. Esta función es útil cuando portas una aplicación existente a Kubernetes o durante escenarios de recuperación ante desastres.

Cuando uses los controladores ONTAP y `solidfire-san` utiliza el comando `tridentctl import volume <backend-name> <volume-name> -f /path/pvc.yaml` para importar un volumen existente en Kubernetes para que sea gestionado por Trident. El archivo YAML o JSON de PVC usado en el comando de importación de volumen apunta a una clase de almacenamiento que identifica a Trident como el provisioner. Cuando uses un backend NetApp HCI/SolidFire, asegúrate de que los nombres de los volúmenes sean únicos. Si los nombres de los volúmenes están duplicados, clona el volumen con un nombre único para que la función de importación de volumen pueda distinguir entre ellos.

Si se utiliza el controlador `azure-netapp-files`, usa el comando `tridentctl import volume <backend-name> <volume path> -f /path/pvc.yaml` para importar el volumen en Kubernetes y que sea gestionado por Trident. Esto garantiza una referencia de volumen única.

Cuando se ejecute el comando anterior, Trident encontrará el volumen en el backend y leerá su tamaño. Añadirá automáticamente (y sobrescribirá si es necesario) el tamaño de volumen del PVC configurado. Luego, Trident crea el nuevo PV y Kubernetes vincula el PVC al PV.

Si un contenedor fue desplegado de tal manera que requería el PVC importado específico, permanecería en estado pendiente hasta que el par PVC/PV se vincule mediante el proceso de importación de volumen. Después de que el par PVC/PV se vincule, el contenedor debería iniciarse, siempre que no haya otros problemas.

Servicio de registro

El despliegue y la gestión del almacenamiento para el registro se han documentado en "netapp.io" en el "blog".

Servicio de registro

Al igual que otros servicios de OpenShift, el servicio de registro se implementa usando Ansible con parámetros de configuración suministrados por el archivo de inventario, o sea, hosts, proporcionado al playbook. Hay dos métodos de instalación que se van a cubrir: implementar el registro durante la instalación inicial de OpenShift e implementar el registro después de que OpenShift ya esté instalado.

A partir de la versión 3.9 de Red Hat OpenShift, la documentación oficial desaconseja NFS para el servicio de registro debido a preocupaciones sobre la corrupción de datos. Esto se basa en pruebas de Red Hat con sus productos. El servidor NFS de ONTAP no tiene estos problemas y puede respaldar fácilmente un despliegue de registro. Al final, la elección del protocolo para el servicio de registro depende de ti, solo ten en cuenta que ambos funcionarán genial cuando uses plataformas NetApp y no hay razón para evitar NFS si esa es tu preferencia.

Si decides usar NFS con el servicio de registro, tendrás que establecer la variable de Ansible `openshift_enable_unsupported_configurations` en `true` para evitar que el instalador falle.

Empezar

El servicio de registro puede, opcionalmente, desplegarse tanto para aplicaciones como para las operaciones principales del clúster OpenShift en sí. Si decides desplegar el registro de operaciones, especificando la variable `openshift_logging_use_ops` como `true`, se crearán dos instancias del servicio. Las variables que controlan la instancia de registro para operaciones contienen "ops", mientras que la instancia para aplicaciones no.

Configurar las variables de Ansible según el método de despliegue es importante para garantizar que los servicios subyacentes utilicen el almacenamiento correcto. Veamos las opciones para cada uno de los métodos de despliegue.

Las tablas siguientes contienen únicamente las variables relevantes para la configuración del almacenamiento en lo que respecta al servicio de registro. Puedes encontrar otras opciones en "Documentación de registro de Red Hat OpenShift" que deberías revisar, configurar y usar según tu implementación.

Las variables de la tabla siguiente harán que el libro de jugadas de Ansible cree un PV y un PVC para el servicio de registro usando los detalles proporcionados. Este método es significativamente menos flexible que usar el libro de jugadas de instalación de componentes después de la instalación de OpenShift, pero si tienes volúmenes existentes disponibles, es una opción.

Variable	Detalles
`openshift_logging_storage_kind`	Establécelo en `nfs` para que el instalador cree un NFS PV para el servicio de registro.
`openshift_logging_storage_host`	El nombre de host o la dirección IP del host NFS. Esto debe configurarse en el dataLIF de tu máquina virtual.
`openshift_logging_storage_nfs_directory`	La ruta de montaje para la exportación NFS. Por ejemplo, si el volumen está unido como `/openshift_logging`, usarías esa ruta para esta variable.
`openshift_logging_storage_volume_name`	El nombre, por ejemplo `pv_ose_logs`, del PV a crear.
`openshift_logging_storage_volume_size`	El tamaño de la exportación NFS, por ejemplo `100Gi`.

Si tu clúster de OpenShift ya está funcionando y, por lo tanto, Trident ha sido desplegado y configurado, el instalador puede usar el aprovisionamiento dinámico para crear los volúmenes. Las siguientes variables deberán configurarse.

Variable	Detalles
`openshift_logging_es_pvc_dynamic`	Establécelo en true para usar volúmenes aprovisionados dinámicamente.
`openshift_logging_es_pvc_storage_class_name`	El nombre de la clase de almacenamiento que se utilizará en el PVC.
`openshift_logging_es_pvc_size`	El tamaño del volumen solicitado en el PVC.
`openshift_logging_es_pvc_prefix`	Un prefijo para los PVC usados por el servicio de registro.
`openshift_logging_es_ops_pvc_dynamic`	Establécelo en `true` para usar volúmenes aprovisionados dinámicamente para la instancia de registro de operaciones.
`openshift_logging_es_ops_pvc_storage_class_name`	El nombre de la clase de almacenamiento para la instancia de registro ops.
`openshift_logging_es_ops_pvc_size`	El tamaño de la solicitud de volumen para la instancia ops.
`openshift_logging_es_ops_pvc_prefix`	Un prefijo para los PVC de la instancia ops.

Despliega la pila de registro

Si estás desplegando el registro como parte del proceso de instalación inicial de OpenShift, solo necesitas seguir el proceso de despliegue estándar. Ansible configurará y desplegará los servicios necesarios y los objetos de OpenShift para que el servicio esté disponible en cuanto Ansible termine.

Sin embargo, si estás haciendo el despliegue después de la instalación inicial, Ansible deberá usar el playbook del componente. Este proceso puede cambiar ligeramente con diferentes versiones de OpenShift, así que asegúrate de leer y seguir "Red Hat OpenShift Container Platform 3.11 documentación" para tu versión.

Servicio de métricas

El servicio de métricas proporciona información valiosa al administrador sobre el estado, la utilización de recursos y la disponibilidad del clúster OpenShift. También es necesario para la funcionalidad de autoescalado de pods y muchas organizaciones usan datos del servicio de métricas para sus aplicaciones de charge back y/o show back.

Al igual que con el servicio de registro, y OpenShift en su conjunto, Ansible se usa para desplegar el servicio de métricas. También, como el servicio de registro, el servicio de métricas se puede desplegar durante una configuración inicial del clúster o después de que esté en funcionamiento usando el método de instalación de componentes. Las siguientes tablas contienen las variables que son importantes al configurar el almacenamiento persistente para el servicio de métricas.

Las tablas siguientes solo contienen las variables relevantes para la configuración del almacenamiento en relación con el servicio de métricas. Hay muchas otras opciones en la documentación que deberías revisar, configurar y usar según tu puesta en marcha.

Variable	Detalles
`openshift_metrics_storage_kind`	Establécelo en `nfs` para que el instalador cree un NFS PV para el servicio de registro.
`openshift_metrics_storage_host`	El nombre de host o la dirección IP del host NFS. Esto debe establecerse en el dataLIF de tu SVM.
`openshift_metrics_storage_nfs_directory`	La ruta de montaje para la exportación NFS. Por ejemplo, si el volumen está unido como `/openshift_metrics`, usarías esa ruta para esta variable.
`openshift_metrics_storage_volume_name`	El nombre, por ejemplo `pv_ose_metrics`, del PV a crear.
`openshift_metrics_storage_volume_size`	El tamaño de la exportación NFS, por ejemplo `100Gi`.

Si tu clúster de OpenShift ya está funcionando y, por lo tanto, Trident ha sido desplegado y configurado, el instalador puede usar el aprovisionamiento dinámico para crear los volúmenes. Las siguientes variables deberán configurarse.

Variable	Detalles
`openshift_metrics_cassandra_pvc_prefix`	Un prefijo que se usará para los PVC de métricas.
`openshift_metrics_cassandra_pvc_size`	El tamaño de los volúmenes a solicitar.
`openshift_metrics_cassandra_storage_type`	El tipo de almacenamiento que se debe usar para las métricas, esto debe estar configurado como dinámico para que Ansible cree PVC con la clase de almacenamiento adecuada.
`openshift_metrics_cassanda_pvc_storage_class_name`	El nombre de la clase de almacenamiento que se va a usar.

Implementa el servicio de métricas

Con las variables Ansible apropiadas definidas en tu archivo hosts/inventory, despliega el servicio usando

Ansible. Si estás desplegando en el momento de instalación de OpenShift, entonces el PV se creará y usará automáticamente. Si estás desplegando usando los playbooks de componentes, después de la instalación de OpenShift, entonces Ansible crea cualquier PVC que sea necesario y, después de que Trident haya aprovisionado almacenamiento para ellos, despliega el servicio.

Las variables anteriores y el proceso de despliegue pueden cambiar con cada versión de OpenShift. Asegúrate de revisar y seguir "Guía de puesta en marcha de OpenShift de Red Hat" para tu versión, así estará configurado para tu entorno.

Protección de datos y recuperación de desastres

Conoce las opciones de protección y recuperación para Trident y los volúmenes creados usando Trident. Deberías tener una estrategia de protección y recuperación de datos para cada aplicación con un requisito de persistencia.

Replicación y recuperación de Trident

Puedes crear una copia de seguridad para restaurar Trident en caso de desastre.

Replicación de Trident

Trident utiliza los CRD de Kubernetes para almacenar y gestionar su propio estado y el etcd del clúster de Kubernetes para almacenar sus metadatos.

Pasos

1. Haz una copia de seguridad del clúster Kubernetes etcd usando "Kubernetes: copia de seguridad de un clúster etcd".
2. Coloca los artefactos de copia de seguridad en un volumen FlexVol

> NetApp recomienda que protejas la SVM donde reside el FlexVol con una relación de SnapMirror hacia otra SVM.

Recuperación de Trident

Usando los CRD de Kubernetes y la instantánea etcd del clúster de Kubernetes, puedes recuperar Trident.

Pasos

1. Desde la SVM de destino, monta el volumen que contiene los archivos de datos etcd de Kubernetes y los certificados en el host que se configurará como nodo maestro.
2. Copia todos los certificados necesarios relacionados con el clúster de Kubernetes en `/etc/kubernetes/pki` y los archivos de miembros de etcd en `/var/lib/etcd`.
3. Restaura el clúster de Kubernetes desde la copia de seguridad de etcd usando "Kubernetes: restaurar un clúster etcd".
4. Ejecuta `kubectl get crd` para verificar que todos los recursos personalizados de Trident han aparecido y recupera los objetos Trident para comprobar que todos los datos están disponibles.

Replicación y recuperación de SVM

Trident no puede configurar las relaciones de replicación, sin embargo, el administrador de almacenamiento

puede usar "ONTAP SnapMirror" para replicar una SVM.

En caso de desastre, puedes activar la SVM de destino SnapMirror para empezar a servir datos. Puedes volver al primario cuando se restauren los sistemas.

Acerca de esta tarea

Ten en cuenta lo siguiente cuando uses la función SnapMirror SVM Replication:

- Deberías crear un backend distinto para cada SVM con SVM-DR activado.
- Configura las clases de almacenamiento para seleccionar los backends replicados solo cuando sea necesario, para evitar tener volúmenes que no necesitan replicación aprovisionados en los backends que soportan SVM-DR.
- Los administradores de aplicaciones deben comprender el coste y la complejidad adicionales asociados con la replicación y considerar detenidamente su plan de recuperación antes de comenzar este proceso.

Replicación SVM

Puedes usar "ONTAP: SnapMirror replicación SVM" para crear la relación de replicación SVM.

SnapMirror te permite establecer opciones para controlar qué se va a replicar. Vas a necesitar saber qué opciones seleccionaste al realizar Recuperación de SVM usando Trident.

- "-identity-preserve true" replica toda la configuración del SVM.
- "-discard-configs red" excluye las LIF y la configuración de red relacionada.
- "-identity-preserve false" replica solo los volúmenes y la configuración de seguridad.

Recuperación de SVM usando Trident

Trident no detecta automáticamente los fallos de la SVM. En caso de desastre, el administrador puede iniciar manualmente la conmutación por error de Trident a la nueva SVM.

Pasos

1. Cancela las transferencias programadas y en curso de SnapMirror, rompe la relación de replicación, detén la SVM de origen y luego activa la SVM de destino de SnapMirror.
2. Si especificaste `-identity-preserve false` o `-discard-config network` al configurar tu replicación de SVM, actualiza el `managementLIF` y el `dataLIF` en el archivo de definición del backend de Trident.
3. Confirma que `storagePrefix` está presente en el archivo de definición del backend de Trident. Este parámetro no se puede cambiar. Omitir `storagePrefix` hará que la actualización del backend falle.
4. Actualiza todos los backends necesarios para reflejar el nuevo nombre de SVM de destino usando:

```
./tridentctl update backend <backend-name> -f <backend-json-file> -n
<namespace>
```

5. Si especificaste `-identity-preserve false` o `discard-config network`, tienes que reiniciar todos los pods de la aplicación.

Si especificaste `-identity-preserve true`, todos los volúmenes aprovisionados por Trident comienzan a servir datos cuando se activa la SVM de destino.

Replicación y recuperación de volúmenes

Trident no puede configurar las relaciones de replicación de SnapMirror, sin embargo, el administrador de almacenamiento puede usar "Replicación y recuperación de ONTAP SnapMirror" para replicar volúmenes creados por Trident.

Luego, puedes importar los volúmenes recuperados en Trident usando "tridentctl volume import".

La importación no es compatible en `ontap-nas-economy`, `ontap-san-economy` o en `ontap-flexgroup-economy` drivers.

Protección de datos Snapshot

Puedes proteger y restaurar datos usando:

- Un controlador de instantáneas externo y CRDs para crear instantáneas de volúmenes de Kubernetes de Volúmenes Persistentes (PVs).

  "Instantáneas de volumen"

- ONTAP Snapshots para restaurar todo el contenido de un volumen o recuperar archivos individuales o LUNs.

  "Instantáneas de ONTAP"

Automatizando la conmutación por error de aplicaciones con estado con Trident

La función de separación forzada de Trident te permite separar automáticamente volúmenes de nodos insalubres en un clúster de Kubernetes, evitando la corrupción de datos y asegurando la disponibilidad de las aplicaciones. Esta función es especialmente útil en escenarios donde los nodos dejan de responder o se desconectan para mantenimiento.

Detalles sobre la desconexión forzada

Force detach está disponible para `ontap-san`, `ontap-san-economy`, `ontap-nas` y `ontap-nas-economy` solamente. Antes de activar force detach, debe estar habilitado el apagado no pacífico de nodos (NGNS) en el clúster de Kubernetes. NGNS está habilitado por defecto para Kubernetes 1.28 y versiones superiores. Para obtener más información, consulta "Kubernetes: apagado no controlado del nodo".

Cuando usas el controlador `ontap-nas` o `ontap-nas-economy`, necesitas establecer el parámetro `autoExportPolicy` en la configuración del backend a `true` para que Trident pueda restringir el acceso desde el nodo de Kubernetes con el taint aplicado usando políticas de exportación gestionadas.

Debido a que Trident depende de Kubernetes NGNS, no elimines `out-of-service` taints de un nodo no saludable hasta que todas las cargas de trabajo no tolerables se hayan reprogramado. Aplicar o eliminar el taint de forma imprudente puede poner en riesgo la protección de datos del backend.

Cuando el administrador del clúster de Kubernetes ha aplicado la `node.kubernetes.io/out-of-service=nodeshutdown:NoExecute` mancha al nodo y `enableForceDetach` está configurado en `true`, Trident determinará el estado del nodo y:

1. Detén el acceso de E/S del backend para los volúmenes montados en ese nodo.
2. Marca el objeto nodo Trident como `dirty` (no seguro para nuevas publicaciones).

El controlador Trident rechazará nuevas solicitudes de publicación de volumen hasta que el nodo sea recalificado (después de haber sido marcado como `dirty`) por el pod de nodo Trident. Cualquier carga de trabajo programada con un PVC montado (incluso después de que el nodo del clúster esté en buen estado y listo) no se aceptará hasta que Trident pueda verificar el nodo `clean` (seguro para nuevas publicaciones).

Cuando se restablezca la salud del nodo y se elimine la taint, Trident hará lo siguiente:

1. Identifica y limpia las rutas publicadas obsoletas en el nodo.
2. Si el nodo está en un `cleanable` estado (se ha eliminado la mancha fuera de servicio y el nodo está en `Ready` estado) y todas las rutas publicadas obsoletas están limpias, Trident readmitirá el nodo como `clean` y permitirá nuevos volúmenes publicados en el nodo.

Detalles sobre la conmutación automática al respaldo

Puedes automatizar el proceso de desconexión forzada mediante la integración con "operador de verificación de estado del nodo (NHC)". Cuando ocurre un fallo en un nodo, NHC activa la remediación de nodos Trident (TNR) y la desconexión forzada automáticamente creando un CR TridentNodeRemediation en el espacio de nombres de Trident que define el nodo fallido. TNR se crea solo cuando falla un nodo y NHC la elimina una vez que el nodo vuelve a estar en línea o se elimina.

Proceso de eliminación de pod de nodo con fallo

La conmutación automática al respaldo selecciona las cargas de trabajo que se eliminarán del nodo fallido. Cuando se crea un TNR, el controlador del TNR marca el nodo como sucio, impide la publicación de nuevos volúmenes y comienza a eliminar los pods compatibles con la desconexión forzada y sus conexiones de volumen.

Todos los volúmenes/PVC compatibles con force-detach son compatibles con la conmutación automática al respaldo:

- Volúmenes NAS y NAS-economy que utilizan políticas de exportación automática (SMB aún no es compatible).
- Volúmenes SAN y SAN-economy.

Consulta Detalles sobre la desconexión forzada.

Comportamiento predeterminado:

- Los pods que utilizan volúmenes compatibles con la desconexión forzada se eliminan del nodo fallido. Kubernetes volverá a programarlos en un nodo en buen estado.
- Los pods que usan un volumen no compatible con la desconexión forzada, incluidos los volúmenes que no son Trident, no se eliminan del nodo fallido.
- Los pods sin estado (no PVC) no se eliminan del nodo fallido, a menos que la anotación del pod `trident.netapp.io/podRemediationPolicy: delete` esté configurada.

Anulación del comportamiento de eliminación de pod:

El comportamiento de eliminación de pods se puede personalizar usando una anotación de pod: `trident.netapp.io/podRemediationPolicy[retain, delete]`. Estas anotaciones se examinan y se usan cuando ocurre una conmutación al respaldo. Aplica anotaciones a la especificación del pod en el deployment o replicaset de Kubernetes para evitar que la anotación desaparezca después de una conmutación al respaldo:

- `retain` - El pod NO se eliminará del nodo fallido durante una conmutación automática al respaldo.
- `delete` - El pod se eliminará del nodo fallido durante una conmutación automática al respaldo.

Estas anotaciones se pueden aplicar a cualquier pod.

- Las operaciones de E/S solo se bloquearán en los nodos fallidos para los volúmenes que admiten force-detach.
- Para los volúmenes que no admiten la desconexión forzada, existe el riesgo de corrupción de datos y problemas de multi-attach.

CR de TridentNodeRemediation

El CR TridentNodeRemediation (TNR) define un nodo fallido. El nombre del TNR es el nombre del nodo fallido.

Ejemplo TNR:

```
apiVersion: trident.netapp.io/v1
kind: TridentNodeRemediation
metadata:
  name: <K8s-node-name>
spec: {}
```

Estados de TNR: usa los siguientes comandos para ver el estado de los TNR:
`kubectl get tnr <name> -n <trident-namespace>`

Los TNR pueden estar en uno de los siguientes estados:

- *Remediando*:
  - Detén el acceso de E/S de backend para los volúmenes compatibles con force-detach que están montados en ese nodo.
  - El objeto de nodo Trident está marcado como sucio (no es seguro para nuevas publicaciones).
  - Eliminar pods y adjuntos de volumen del nodo

- *Recuperación de nodo pendiente*:
  - El controlador está esperando que el nodo vuelva a estar en línea.
  - Una vez que el nodo esté en línea, publish-enforcement se asegurará de que el nodo esté limpio y listo para nuevas publicaciones de volúmenes.
- Si se elimina el nodo de K8s, el controlador TNR eliminará el TNR y dejará de conciliar.
- *Correcto*:
  - Todos los pasos de remediación y recuperación del nodo se completaron correctamente. El nodo está limpio y listo para nuevas publicaciones de volúmenes.
- *Fallido*:
  - Error irrecuperable. Las causas del error se especifican en el campo status.message del CR.

Habilitar la conmutación automática al respaldo

Prerrequisitos:

- Asegúrate de que la desconexión forzada esté habilitada antes de habilitar la conmutación automática al respaldo. Para obtener más información, consulta Detalles sobre la desconexión forzada.
- Instala la comprobación del estado del nodo (NHC) en el clúster de Kubernetes.
  - "Instala operator-sdk".
  - Instala Operator Lifecycle Manager (OLM) en el clúster si aún no está instalado: `operator-sdk olm install`.
  - Instala Node Health check Operator: `kubectl create -f https://operatorhub.io/install/node-healthcheck-operator.yaml`.

> También puedes usar formas alternativas para detectar fallas de nodos como se especifica en la sección de [Integrating Custom Node Health Check Solutions] abajo.

Consulta "Operador de comprobación de estado del nodo" para más información.

Pasos

1. Crea un NodeHealthCheck (NHC) en el espacio de nombres Trident para monitorear los nodos de trabajo en el clúster. Ejemplo:

```
apiVersion: remediation.medik8s.io/v1alpha1
kind: NodeHealthCheck
metadata:
  name: <CR name>
spec:
  selector:
    matchExpressions:
      - key: node-role.kubernetes.io/control-plane
        operator: DoesNotExist
      - key: node-role.kubernetes.io/master
        operator: DoesNotExist
  remediationTemplate:
    apiVersion: trident.netapp.io/v1
    kind: TridentNodeRemediationTemplate
    namespace: <Trident installation namespace>
    name: trident-node-remediation-template
  minHealthy: 0 # Trigger force-detach upon one or more node failures
  unhealthyConditions:
    - type: Ready
      status: "False"
      duration: 0s
    - type: Ready
      status: Unknown
      duration: 0s
```

2. Aplica la verificación de estado del nodo CR en el `trident` namespace.

```
kubectl apply -f <nhc-cr-file>.yaml -n <trident-namespace>
```

El CR anterior está configurado para supervisar los nodos de trabajo de K8s en busca de las condiciones Ready: false y Unknown. La conmutación automática al respaldo se activará cuando un nodo entre en estado Ready: false o Ready: Unknown.

El `unhealthyConditions` en el CR usa un periodo de gracia de 0 segundos. Esto hace que la conmutación automática al respaldo se active de inmediato cuando K8s establece la condición del nodo Ready: false, que se configura después de que K8s pierde el latido de un nodo. K8s tiene un valor predeterminado de 40 segundos de espera después del último latido antes de establecer Ready: false. Este periodo de gracia se puede personalizar en las opciones de despliegue de K8s.

Para obtener opciones de configuración adicionales, consulta "Documentación de Node-Healthcheck-Operator".

Información adicional de configuración

Cuando se instala Trident con force-detach habilitado, se crean automáticamente dos recursos adicionales en el espacio de nombres de Trident para facilitar la integración con NHC: TridentNodeRemediationTemplate (TNRT) y ClusterRole.

TridentNodeRemediationTemplate (TNRT):

El TNRT sirve como plantilla para el controlador del NHC, que usa TNRT para generar recursos TNR según sea necesario.

```
apiVersion: trident.netapp.io/v1
kind: TridentNodeRemediationTemplate
metadata:
  name: trident-node-remediation-template
  namespace: trident
spec:
  template:
    spec: {}
```

ClusterRole:

También se agrega un rol de clúster durante la instalación cuando se habilita la desconexión forzada. Esto le da a NHC permisos para los TNR en el espacio de nombres Trident.

```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  labels:
    rbac.ext-remediation/aggregate-to-ext-remediation: "true"
  name: tridentnoderemediation-access
rules:
- apiGroups:
  - trident.netapp.io
  resources:
  - tridentnoderemediationtemplates
  - tridentnoderemediations
  verbs:
  - get
  - list
  - watch
  - create
  - update
  - patch
  - delete
```

Actualizaciones y mantenimiento del clúster K8s

Para evitar cualquier conmutación automática al respaldo, pausa la conmutación automática al respaldo durante el mantenimiento o las actualizaciones de K8s, cuando se espera que los nodos se apaguen o se reinicien. Puedes pausar el CR de NHC (descrito arriba) parchando su CR:

```
kubectl patch NodeHealthCheck <cr-name> --patch
'{"spec":{"pauseRequests":["<description-for-reason-of-pause>"]}}' --type=merge
```

Esto pausa la conmutación automática al respaldo. Para volver a habilitar la conmutación automática al respaldo, elimina pauseRequests de la especificación después de que termine el mantenimiento.

Limitaciones

- Las operaciones de E/S solo se impiden en los nodos fallidos para los volúmenes compatibles con force-detach. Solo los pods que usan volúmenes/PVCs compatibles con force-detach se eliminan automáticamente.
- La conmutación automática al respaldo y la desconexión forzada se ejecutan dentro del pod trident-controller. Si el nodo que aloja trident-controller falla, la conmutación automática al respaldo se retrasará hasta que K8s mueva el pod a un nodo saludable.

Integración de soluciones personalizadas de comprobación del estado del nodo

Puedes reemplazar Node Healthcheck Operator con herramientas alternativas de detección de fallos de nodo para activar la conmutación automática al respaldo. Para garantizar la compatibilidad con el mecanismo de conmutación automática al respaldo, tu solución personalizada debe:

- Crea un TNR cuando se detecte una falla en un nodo, usando el nombre del nodo fallido como nombre CR del TNR.
- Elimina el TNR cuando el nodo se haya recuperado y el TNR esté en el estado Succeeded.

Seguridad

Seguridad

Usa las recomendaciones que se enumeran aquí para asegurarte de que tu instalación de Trident sea segura.

Ejecuta Trident en su propio espacio de nombres

Es importante evitar que las aplicaciones, los administradores de aplicaciones, los usuarios y las aplicaciones de administración accedan a las definiciones de objetos de Trident o a los pods para garantizar un almacenamiento confiable y bloquear posibles actividades maliciosas.

Para separar las demás aplicaciones y usuarios de Trident, siempre instala Trident en su propio espacio de nombres de Kubernetes (`trident`). Poner Trident en su propio espacio de nombres asegura que solo el personal administrativo de Kubernetes tenga acceso al pod de Trident y a los artefactos (como los secretos de backend y CHAP, si aplica) almacenados en los objetos CRD con espacio de nombres. Debes asegurarte de permitir que solo los administradores tengan acceso al espacio de nombres de Trident y así acceso a la `tridentctl` aplicación.

Usa la autenticación CHAP con backends SAN de ONTAP

Trident admite la autenticación basada en CHAP para cargas de trabajo ONTAP SAN (usando los `ontap-san` y `ontap-san-economy` controladores). NetApp recomienda usar CHAP bidireccional con Trident para la autenticación entre un host y el backend de almacenamiento.

Para los backends de ONTAP que usan los controladores de almacenamiento SAN, Trident puede configurar

CHAP bidireccional y administrar los nombres de usuario y secretos de CHAP a través de `tridentctl`. Consulta "Prepárate para configurar el backend con controladores SAN de ONTAP" para entender cómo Trident configura CHAP en los backends de ONTAP.

Usa la autenticación CHAP con NetApp HCI y SolidFire backends

NetApp recomienda implementar CHAP bidireccional para garantizar la autenticación entre un host y los backends NetApp HCI y SolidFire. Trident utiliza un objeto secreto que incluye dos contraseñas CHAP por inquilino. Cuando se instala Trident, administra los secretos CHAP y los almacena en un objeto CR `tridentvolume` para el PV correspondiente. Cuando creas un PV, Trident utiliza los secretos CHAP para iniciar una sesión iSCSI y comunicarse con el sistema NetApp HCI y SolidFire mediante CHAP.

Los volúmenes que crea Trident no están asociados con ningún Volume Access Group.

Usa Trident con NVE y NAE

NetApp ONTAP proporciona cifrado de datos en reposo para proteger la información confidencial en caso de que un disco sea robado, devuelto o reutilizado. Para más información, consulta "Descripción general de la configuración de NetApp Volume Encryption".

- Si NAE está habilitado en el backend, cualquier volumen aprovisionado en Trident tendrá NAE habilitado.
  - Puedes configurar el indicador de cifrado NVE en `""` para crear volúmenes habilitados para NAE.
- Si NAE no está habilitado en el backend, cualquier volumen aprovisionado en Trident tendrá NVE habilitado a menos que el indicador de cifrado NVE esté configurado en `false` (el valor predeterminado) en la configuración del backend.

Los volúmenes creados en Trident en un backend habilitado para NAE deben ser cifrados con NVE o NAE.

- Puedes configurar el indicador de cifrado NVE a `true` en la configuración del backend de Trident para anular el cifrado NAE y usar una clave de cifrado específica por volumen.
- Al configurar el indicador de cifrado NVE en `false` un backend con NAE habilitado, se crea un volumen con NAE habilitado. No puedes deshabilitar el cifrado NAE configurando el indicador de cifrado NVE en `false`.

- Puedes crear manualmente un volumen NVE en Trident configurando explícitamente el indicador de cifrado NVE en `true`.

Para obtener más información sobre las opciones de configuración del backend, consulta:

- "Opciones de configuración de ONTAP SAN"
- "Opciones de configuración de ONTAP NAS"

Linux Unified Key Setup (LUKS)

Puedes habilitar Linux Unified Key Setup (LUKS) para cifrar volúmenes ONTAP SAN y ONTAP SAN ECONOMY en Trident. Trident admite la rotación de frases de contraseña y la expansión de volúmenes cifrados con LUKS.

En Trident, los volúmenes cifrados con LUKS utilizan el cifrado y modo aes-xts-plain64, como se recomienda

en "NIST".

> El cifrado LUKS no es compatible con sistemas ASA r2. Para información sobre los sistemas ASA r2, consulta "Conoce los sistemas de almacenamiento ASA r2".

Antes de empezar

- Los nodos de trabajo deben tener instalado cryptsetup 2.1 o superior (pero inferior a 3.0). Para más información, visita "Gitlab: cryptsetup".
- Por razones de rendimiento, NetApp recomienda que los nodos de trabajo admitan Advanced Encryption Standard New Instructions (AES-NI). Para verificar la compatibilidad con AES-NI, ejecuta el siguiente comando:

```
grep "aes" /proc/cpuinfo
```

Si no se devuelve nada, tu procesador no es compatible con AES-NI. Para más información sobre AES-NI, visita: "Intel: Instrucciones de cifrado avanzado estándar (AES-NI)".

Habilita el cifrado LUKS

Puedes habilitar el cifrado por volumen del lado del host usando Linux Unified Key Setup (LUKS) para los volúmenes ONTAP SAN y ONTAP SAN ECONOMY.

Pasos

1. Define los atributos de cifrado LUKS en la configuración del backend. Para más información sobre las opciones de configuración del backend para ONTAP SAN, consulta "Opciones de configuración de ONTAP SAN".

```
{
  "storage": [
    {
      "labels": {
        "luks": "true"
      },
      "zone": "us_east_1a",
      "defaults": {
        "luksEncryption": "true"
      }
    },
    {
      "labels": {
        "luks": "false"
      },
      "zone": "us_east_1a",
      "defaults": {
        "luksEncryption": "false"
      }
    }
  ]
}
```

2. Usa `parameters.selector` para definir los grupos de almacenamiento usando cifrado LUKS. Por ejemplo:

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: luks
provisioner: csi.trident.netapp.io
parameters:
  selector: "luks=true"
  csi.storage.k8s.io/node-stage-secret-name: luks-${pvc.name}
  csi.storage.k8s.io/node-stage-secret-namespace: ${pvc.namespace}
```

3. Crea un secreto que contenga la contraseña LUKS. Por ejemplo:

```
kubectl -n trident create -f luks-pvc1.yaml
apiVersion: v1
kind: Secret
metadata:
  name: luks-pvc1
stringData:
  luks-passphrase-name: A
  luks-passphrase: secretA
```

Limitaciones

Los volúmenes cifrados con LUKS no pueden aprovechar la deduplicación y la compresión de ONTAP.

Configuración de backend para importar volúmenes LUKS

Para importar un volumen LUKS, debes establecer `luksEncryption` en (`true` en el backend. La opción `luksEncryption` le dice a Trident si el volumen es compatible con LUKS (`true` o no es compatible con LUKS (`false`, como se muestra en el siguiente ejemplo.

```
version: 1
storageDriverName: ontap-san
managementLIF: 10.0.0.1
dataLIF: 10.0.0.2
svm: trident_svm
username: admin
password: password
defaults:
  luksEncryption: 'true'
  spaceAllocation: 'false'
  snapshotPolicy: default
  snapshotReserve: '10'
```

Configuración de PVC para importar volúmenes LUKS

Para importar volúmenes LUKS de forma dinámica, configura la anotación `trident.netapp.io/luksEncryption` a `true` e incluye una clase de almacenamiento habilitada para LUKS en la PVC como se muestra en este ejemplo.

```
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: luks-pvc
  namespace: trident
  annotations:
    trident.netapp.io/luksEncryption: "true"
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 1Gi
  storageClassName: luks-sc
```

Rotar una frase de contraseña LUKS

Puedes rotar la frase de contraseña LUKS y confirmar la rotación.

> No olvides una contraseña hasta que hayas verificado que ya no está referenciada por ningún volumen, instantánea o secreto. Si se pierde una contraseña referenciada, puede que no puedas montar el volumen y los datos permanecerán cifrados e inaccesibles.

Acerca de esta tarea

La rotación de la frase de contraseña LUKS ocurre cuando se crea un pod que monta el volumen después de especificar una nueva frase de contraseña LUKS. Cuando se crea un nuevo pod, Trident compara la frase de contraseña LUKS en el volumen con la frase de contraseña activa en el secreto.

- Si la frase de contraseña del volumen no coincide con la frase de contraseña activa en el secreto, ocurre una rotación.
- Si la frase de contraseña del volumen coincide con la frase de contraseña activa en el secreto, el `previous-luks-passphrase` parámetro se ignora.

Pasos

1. Añade los `node-publish-secret-name` y `node-publish-secret-namespace` parámetros StorageClass. Por ejemplo:

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: csi-san
provisioner: csi.trident.netapp.io
parameters:
  trident.netapp.io/backendType: "ontap-san"
  csi.storage.k8s.io/node-stage-secret-name: luks
  csi.storage.k8s.io/node-stage-secret-namespace: ${pvc.namespace}
  csi.storage.k8s.io/node-publish-secret-name: luks
  csi.storage.k8s.io/node-publish-secret-namespace: ${pvc.namespace}
```

2. Identifica las frases de contraseña existentes en el volumen o la snapshot.

Volumen

```
tridentctl -d get volume luks-pvc1
GET http://127.0.0.1:8000/trident/v1/volume/<volumeID>

...luksPassphraseNames:["A"]
```

Snapshot

```
tridentctl -d get snapshot luks-pvc1
GET http://127.0.0.1:8000/trident/v1/volume/<volumeID>/<snapshotID>

...luksPassphraseNames:["A"]
```

3. Actualiza el secreto LUKS del volumen para especificar las frases de contraseña nueva y anterior. Asegúrate de que `previous-luke-passphrase-name` y `previous-luks-passphrase` coincidan con la frase de contraseña anterior.

```
apiVersion: v1
kind: Secret
metadata:
  name: luks-pvc1
stringData:
  luks-passphrase-name: B
  luks-passphrase: secretB
  previous-luks-passphrase-name: A
  previous-luks-passphrase: secretA
```

4. Crea un nuevo pod que monte el volumen. Esto es necesario para iniciar la rotación.
5. Verifica que la frase de contraseña haya sido rotada.

Volumen

```
tridentctl -d get volume luks-pvc1
GET http://127.0.0.1:8000/trident/v1/volume/<volumeID>

...luksPassphraseNames:["B"]
```

Snapshot

```
tridentctl -d get snapshot luks-pvc1
GET http://127.0.0.1:8000/trident/v1/volume/<volumeID>/<snapshotID>

...luksPassphraseNames:["B"]
```

Resultados

La frase de contraseña se rotó cuando solo se devuelve la nueva frase de contraseña en el volumen y la instantánea.

Si se devuelven dos contraseñas, por ejemplo `luksPassphraseNames: ["B", "A"]`, la rotación está incompleta. Puedes activar un nuevo pod para intentar completar la rotación.

Habilita la expansión de volumen

Puedes habilitar la expansión de volumen en un volumen cifrado con LUKS.

Pasos

1. Habilita la `CSINodeExpandSecret` puerta de funciones (beta 1.25+). Consulta "Kubernetes 1.25: usa secretos para la expansión de volúmenes CSI basada en nodos" para más detalles.
2. Añade los `node-expand-secret-name` y `node-expand-secret-namespace` parámetros StorageClass. Por ejemplo:

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: luks
provisioner: csi.trident.netapp.io
parameters:
  selector: "luks=true"
  csi.storage.k8s.io/node-stage-secret-name: luks-${pvc.name}
  csi.storage.k8s.io/node-stage-secret-namespace: ${pvc.namespace}
  csi.storage.k8s.io/node-expand-secret-name: luks-${pvc.name}
  csi.storage.k8s.io/node-expand-secret-namespace: ${pvc.namespace}
allowVolumeExpansion: true
```

Resultados

Cuando inicias la expansión del almacenamiento en línea, el kubelet pasa las credenciales apropiadas al driver.

Cifrado Kerberos en vuelo

Mediante el cifrado Kerberos en vuelo, puedes mejorar la seguridad de acceso a los datos habilitando el cifrado para el tráfico entre tu clúster gestionado y el backend de almacenamiento.

Trident admite el cifrado Kerberos para ONTAP como backend de almacenamiento:

- **ONTAP local** - Trident admite el cifrado Kerberos sobre conexiones NFSv3 y NFSv4 desde Red Hat OpenShift y clústeres Kubernetes upstream a volúmenes ONTAP locales.

Puedes crear, borrar, redimensionar, crear snapshots, clonar, clonar de solo lectura e importar volúmenes que usan cifrado NFS.

Configura el cifrado Kerberos en vuelo con volúmenes ONTAP locales

Puedes activar el cifrado Kerberos en el tráfico de almacenamiento entre tu clúster gestionado y un backend de almacenamiento ONTAP local.

> El cifrado Kerberos para el tráfico NFS con backends de almacenamiento ONTAP locales solo es compatible usando el controlador de almacenamiento `ontap-nas`.

Antes de empezar

- Asegúrate de que tienes acceso a la utilidad `tridentctl`.
- Asegúrate de que tienes acceso de administrador al backend de almacenamiento de ONTAP.
- Asegúrate de que sabes el nombre del volumen o los volúmenes que vas a compartir desde el backend de almacenamiento de ONTAP.
- Asegúrate de que has preparado la VM de almacenamiento ONTAP para admitir el cifrado Kerberos para volúmenes NFS. Consulta "Habilita Kerberos en un dataLIF" para ver las instrucciones.
- Asegúrate de que cualquier volumen NFSv4 que uses con cifrado Kerberos esté configurado correctamente. Consulta la sección NetApp NFSv4 Domain Configuration (página 13) de "NetApp NFSv4 mejoras y guía de mejores prácticas".

Añade o modifica las políticas de exportación de ONTAP

Necesitas añadir reglas a las políticas de exportación de ONTAP existentes o crear nuevas políticas de exportación que admitan el cifrado Kerberos para el volumen raíz de la máquina virtual de almacenamiento ONTAP, así como para cualquier volumen ONTAP compartido con el clúster de Kubernetes ascendente. Las reglas de las políticas de exportación que añadas, o las nuevas políticas de exportación que crees, deben admitir los siguientes protocolos de acceso y permisos de acceso:

Protocolos de acceso

Configura la política de exportación con los protocolos de acceso NFS, NFSv3 y NFSv4.

Datos de acceso

Puedes configurar una de las tres versiones diferentes de cifrado Kerberos, según lo que necesites para el volumen:

- **Kerberos 5** - (autenticación y cifrado)
- **Kerberos 5i** - (autenticación y cifrado con protección de identidad)
- **Kerberos 5p** - (autenticación y cifrado con protección de identidad y privacidad)

Configura la regla de exportación de ONTAP con los permisos de acceso adecuados. Por ejemplo, si los clusters van a montar los volúmenes NFS con una mezcla de cifrado Kerberos 5i y Kerberos 5p, usa la siguiente configuración de acceso:

Tipo	Acceso de solo lectura	Acceso de lectura/escritura	Acceso de superusuario
UNIX	Habilitado	Habilitado	Habilitado
Kerberos 5i	Habilitado	Habilitado	Habilitado
Kerberos 5p	Habilitado	Habilitado	Habilitado

Consulta la siguiente documentación para saber cómo crear políticas de exportación de ONTAP y reglas de políticas de exportación:

- "Crea una política de exportación"
- "Añade una regla a una política de exportación"

Crea un backend de almacenamiento

Puedes crear una configuración de backend de almacenamiento Trident que incluya la capacidad de cifrado Kerberos.

Acerca de esta tarea

Cuando creas un archivo de configuración de backend de almacenamiento que configura el cifrado Kerberos, puedes especificar una de las tres versiones diferentes de cifrado Kerberos usando el parámetro `spec.nfsMountOptions`:

- `spec.nfsMountOptions: sec=krb5` (autenticación y cifrado)
- `spec.nfsMountOptions: sec=krb5i` (autenticación y cifrado con protección de identidad)
- `spec.nfsMountOptions: sec=krb5p` (autenticación y cifrado con protección de identidad y privacidad)

Especifica solo un nivel de Kerberos. Si especificas más de un nivel de cifrado Kerberos en la lista de parámetros, solo se usa la primera opción.

Pasos

1. En el clúster gestionado, crea un archivo de configuración de backend de almacenamiento usando el siguiente ejemplo. Reemplaza los valores entre corchetes <> con la información de tu entorno:

```
apiVersion: v1
kind: Secret
metadata:
  name: backend-ontap-nas-secret
type: Opaque
stringData:
  clientID: <CLIENT_ID>
  clientSecret: <CLIENT_SECRET>
---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-ontap-nas
spec:
  version: 1
  storageDriverName: "ontap-nas"
  managementLIF: <STORAGE_VM_MGMT_LIF_IP_ADDRESS>
  dataLIF: <PROTOCOL_LIF_FQDN_OR_IP_ADDRESS>
  svm: <STORAGE_VM_NAME>
  username: <STORAGE_VM_USERNAME_CREDENTIAL>
  password: <STORAGE_VM_PASSWORD_CREDENTIAL>
  nasType: nfs
  nfsMountOptions: ["sec=krb5i"] #can be krb5, krb5i, or krb5p
  qtreesPerFlexvol:
  credentials:
    name: backend-ontap-nas-secret
```

2. Usa el archivo de configuración que creaste en el paso anterior para crear el backend:

```
tridentctl create backend -f <backend-configuration-file>
```

Si falla la creación del backend, algo anda mal con la configuración del backend. Puedes ver los registros para determinar la causa ejecutando el siguiente comando:

```
tridentctl logs
```

Después de identificar y corregir el problema con el archivo de configuración, puedes volver a ejecutar el comando create.

Crear una clase de almacenamiento

Puedes crear una clase de almacenamiento para aprovisionar volúmenes con cifrado Kerberos.

Acerca de esta tarea

Cuando creas un objeto de clase de almacenamiento, puedes especificar una de las tres versiones diferentes de cifrado Kerberos usando el parámetro `mountOptions`:

- `mountOptions: sec=krb5` (autenticación y cifrado)
- `mountOptions: sec=krb5i` (autenticación y cifrado con protección de identidad)
- `mountOptions: sec=krb5p` (autenticación y cifrado con protección de identidad y privacidad)

Especifica solo un nivel de Kerberos. Si especificas más de un nivel de cifrado Kerberos en la lista de parámetros, solo se usa la primera opción. Si el nivel de cifrado que especificaste en la configuración del backend de almacenamiento es diferente al nivel que especificas en el objeto de clase de almacenamiento, el objeto de clase de almacenamiento tiene prioridad.

Pasos

1. Crea un objeto de Kubernetes StorageClass usando el siguiente ejemplo:

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: ontap-nas-sc
provisioner: csi.trident.netapp.io
mountOptions:
  - sec=krb5i #can be krb5, krb5i, or krb5p
parameters:
  backendType: ontap-nas
  storagePools: ontapnas_pool
  trident.netapp.io/nasType: nfs
allowVolumeExpansion: true
```

2. Crea la clase de almacenamiento:

```
kubectl create -f sample-input/storage-class-ontap-nas-sc.yaml
```

3. Asegúrate de que la clase de almacenamiento se haya creado:

```
kubectl get sc ontap-nas-sc
```

   Deberías ver una salida similar a la siguiente:

```
NAME          PROVISIONER             AGE
ontap-nas-sc    csi.trident.netapp.io   15h
```

Provisiona volúmenes

Después de crear un backend de almacenamiento y una clase de almacenamiento, ahora puedes aprovisionar un volumen. Para obtener instrucciones, consulta "Aprovisiona un volumen".

Configura el cifrado Kerberos en tránsito con volúmenes de Azure NetApp Files

Puedes habilitar el cifrado Kerberos en el tráfico de almacenamiento entre tu clúster administrado y un único backend de almacenamiento de Azure NetApp Files o un grupo virtual de backends de almacenamiento de Azure NetApp Files.

Antes de empezar

- Asegúrate de haber habilitado Trident en el clúster Red Hat OpenShift administrado.
- Asegúrate de que tienes acceso a la utilidad `tridentctl`.
- Asegúrate de haber preparado el backend de almacenamiento Azure NetApp Files para el cifrado Kerberos, tomando en cuenta los requisitos y siguiendo las instrucciones en "Documentación de Azure NetApp Files".
- Asegúrate de que cualquier volumen NFSv4 que uses con cifrado Kerberos esté configurado correctamente. Consulta la sección NetApp NFSv4 Domain Configuration (página 13) de "NetApp NFSv4 mejoras y guía de mejores prácticas".

Crea un backend de almacenamiento

Puedes crear una configuración de backend de almacenamiento de Azure NetApp Files que incluya la capacidad de cifrado Kerberos.

Acerca de esta tarea

Cuando creas un archivo de configuración de backend de almacenamiento que configura el cifrado Kerberos, puedes definirlo para que se aplique en uno de dos niveles posibles:

- El **nivel de backend de almacenamiento** usando el campo `spec.kerberos`
- El **nivel de pool virtual** usando el `spec.storage.kerberos` campo

Cuando defines la configuración en el nivel de grupo virtual, el grupo se selecciona usando la etiqueta en la clase de almacenamiento.

En cualquier nivel, puedes especificar una de las tres versiones diferentes del cifrado Kerberos:

- `kerberos: sec=krb5` (autenticación y cifrado)
- `kerberos: sec=krb5i` (autenticación y cifrado con protección de identidad)
- `kerberos: sec=krb5p` (autenticación y cifrado con protección de identidad y privacidad)

Pasos

1. En el clúster administrado, crea un archivo de configuración de backend de almacenamiento usando uno de los siguientes ejemplos, según dónde necesites definir el backend de almacenamiento (a nivel de backend de almacenamiento o a nivel de pool virtual). Reemplaza los valores entre corchetes <> con la información de tu entorno:

Ejemplo de nivel de backend de almacenamiento

```
apiVersion: v1
kind: Secret
metadata:
  name: backend-tbc-secret
type: Opaque
stringData:
  clientID: <CLIENT_ID>
  clientSecret: <CLIENT_SECRET>

---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-tbc
spec:
  version: 1
  storageDriverName: azure-netapp-files
  subscriptionID: <SUBSCRIPTION_ID>
  tenantID: <TENANT_ID>
  location: <AZURE_REGION_LOCATION>
  serviceLevel: Standard
  networkFeatures: Standard
  capacityPools: <CAPACITY_POOL>
  resourceGroups: <RESOURCE_GROUP>
  netappAccounts: <NETAPP_ACCOUNT>
  virtualNetwork: <VIRTUAL_NETWORK>
  subnet: <SUBNET>
  nasType: nfs
  kerberos: sec=krb5i #can be krb5, krb5i, or krb5p
  credentials:
    name: backend-tbc-secret
```

Ejemplo de nivel de virtual pool

```
---
apiVersion: v1
kind: Secret
metadata:
  name: backend-tbc-secret
type: Opaque
stringData:
  clientID: <CLIENT_ID>
  clientSecret: <CLIENT_SECRET>

---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-tbc
spec:
  version: 1
  storageDriverName: azure-netapp-files
  subscriptionID: <SUBSCRIPTION_ID>
  tenantID: <TENANT_ID>
  location: <AZURE_REGION_LOCATION>
  serviceLevel: Standard
  networkFeatures: Standard
  capacityPools: <CAPACITY_POOL>
  resourceGroups: <RESOURCE_GROUP>
  netappAccounts: <NETAPP_ACCOUNT>
  virtualNetwork: <VIRTUAL_NETWORK>
  subnet: <SUBNET>
  nasType: nfs
  storage:
    - labels:
        type: encryption
      kerberos: sec=krb5i #can be krb5, krb5i, or krb5p
  credentials:
    name: backend-tbc-secret
```

2. Usa el archivo de configuración que creaste en el paso anterior para crear el backend:

```
tridentctl create backend -f <backend-configuration-file>
```

Si falla la creación del backend, algo anda mal con la configuración del backend. Puedes ver los registros para determinar la causa ejecutando el siguiente comando:

```
tridentctl logs
```

Después de identificar y corregir el problema con el archivo de configuración, puedes volver a ejecutar el comando create.

Crear una clase de almacenamiento

Puedes crear una clase de almacenamiento para aprovisionar volúmenes con cifrado Kerberos.

Pasos

1. Crea un objeto de Kubernetes StorageClass usando el siguiente ejemplo:

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: sc-nfs
provisioner: csi.trident.netapp.io
parameters:
  backendType: azure-netapp-files
  trident.netapp.io/nasType: nfs
  selector: type=encryption
```

2. Crea la clase de almacenamiento:

```
kubectl create -f sample-input/storage-class-sc-nfs.yaml
```

3. Asegúrate de que la clase de almacenamiento se haya creado:

```
kubectl get sc -sc-nfs
```

Deberías ver una salida similar a la siguiente:

```
NAME          PROVISIONER             AGE
sc-nfs        csi.trident.netapp.io   15h
```

Provisiona volúmenes

Después de crear un backend de almacenamiento y una clase de almacenamiento, ahora puedes aprovisionar un volumen. Para obtener instrucciones, consulta "Aprovisiona un volumen".