Seguridad

Trident

NetApp
July 01, 2026

Tabla de contenidos

Seguridad . . . 1
- Seguridad . . . 1
  - Ejecuta Trident en su propio espacio de nombres . . . 1
  - Usa la autenticación CHAP con backends SAN de ONTAP . . . 1
  - Usa la autenticación CHAP con NetApp HCI y SolidFire backends . . . 1
  - Usa Trident con NVE y NAE . . . 1
- Linux Unified Key Setup (LUKS) . . . 2
  - Habilita el cifrado LUKS . . . 2
  - Configuración de backend para importar volúmenes LUKS . . . 4
  - Configuración de PVC para importar volúmenes LUKS . . . 4
  - Rotar una frase de contraseña LUKS . . . 5
  - Habilita la expansión de volumen . . . 7
- Cifrado Kerberos en vuelo . . . 8
  - Configura el cifrado Kerberos en vuelo con volúmenes ONTAP locales . . . 8
  - Configura el cifrado Kerberos en tránsito con volúmenes de Azure NetApp Files . . . 12

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".