Seguridad

Astra Trident

NetApp
January 14, 2026

Tabla de contenidos

Seguridad

Seguridad

Utilice las recomendaciones que se enumeran aquí para asegurarse de que su instalación de Astra Trident es segura.

Ejecute Astra Trident en su propio espacio de nombres

Es importante evitar que las aplicaciones, los administradores de aplicaciones, los usuarios y las aplicaciones de gestión accedan a las definiciones de objetos de Astra Trident o a los pods para garantizar un almacenamiento fiable y bloquear la potencial actividad maliciosa.

Para separar el resto de aplicaciones y usuarios de Astra Trident, instale siempre Astra Trident en su propio espacio de nombres de Kubernetes (`trident`). Si coloca Astra Trident en su propio espacio de nombres, solo el personal administrativo de Kubernetes tiene acceso al pod de la Astra Trident y los artefactos (como los secretos CHAP y de back-end, si corresponde) almacenados en los objetos de CRD named. Debes asegurarte de permitir solo el acceso de los administradores al espacio de nombres de Astra Trident y, por lo tanto, el acceso a `tridentctl` la aplicación.

Utilice la autenticación CHAP con los back-ends DE SAN de ONTAP

Astra Trident admite la autenticación basada en CHAP para cargas de trabajo SAN de ONTAP (mediante `ontap-san` y `ontap-san-economy` controladores). NetApp recomienda utilizar CHAP bidireccional con Astra Trident para la autenticación entre un host y el back-end de almacenamiento.

Para los back-ends ONTAP que utilizan controladores de almacenamiento SAN, Astra Trident puede configurar CHAP bidireccional y gestionar nombres de usuario y secretos CHAP a través de `tridentctl`. Consulte ""para comprender cómo Astra Trident configura CHAP en back-ends de ONTAP.

Utilice la autenticación CHAP con NetApp HCI y back-ends de SolidFire

NetApp recomienda poner en marcha CHAP bidireccional para garantizar la autenticación entre un host y los back-ends de NetApp HCI y SolidFire. Astra Trident utiliza un objeto secreto que incluye dos contraseñas CHAP por inquilino. Cuando se instala Astra Trident, gestiona los secretos CHAP y los almacena en `tridentvolume` un objeto CR para el VP correspondiente. Al crear un VP, Astra Trident utiliza los secretos de CHAP para iniciar una sesión iSCSI y comunicarse con el sistema NetApp HCI y SolidFire a través de CHAP.

> Los volúmenes que crea Astra Trident no están asociados con ningún grupo de acceso de volumen.

Utilice Astra Trident con NVE y NAE

ONTAP de NetApp proporciona cifrado de datos en reposo para proteger los datos confidenciales en el caso de robo, devolución o reasignación de un disco. Para obtener más información, consulte "Configure la información general de cifrado de volúmenes de NetApp".

- Si NAE está habilitado en el back-end, cualquier volumen aprovisionado en Astra Trident se habilitará para NAE.

- Si NAE no está habilitado en el back-end, cualquier volumen aprovisionado en Astra Trident se habilitará NVE a menos que establezca la marca de cifrado de NVE en `false` la configuración de back-end.

> Los volúmenes que se crean en Astra Trident en un back-end con la NAE habilitada deben ser NVE o NAE cifrados.
>
> - Puede establecer el indicador de cifrado de NVE `true` en la configuración de back-end de Trident para anular el cifrado NAE y usar una clave de cifrado específica por volumen.
> - Si se establece la marca de cifrado de NVE `false` en un back-end con NAE habilitado, se creará un volumen con la función NAE habilitada. No se puede deshabilitar el cifrado NAE mediante la marca de cifrado de NVE en `false`.

- Si desea crear manualmente un volumen de NVE en Astra Trident, debe establecer explícitamente la marca de cifrado de NVE en `true`.

Para obtener más información sobre las opciones de configuración del back-end, consulte:

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

Configuración de clave unificada de Linux (LUKS)

Puede habilitar Unified Key Setup (LUKS) de Linux para cifrar los volúmenes DE ECONOMÍA SAN de ONTAP y SAN DE ONTAP en Astra Trident. Astra Trident admite la rotación de claves de acceso y la expansión de volumen para volúmenes cifrados con LUKS.

En Astra Trident, los volúmenes cifrados con LUKS utilizan el cifrado y el modo aes-xts-plain64, como recomienda "NIST".

Antes de empezar

- Los nodos de trabajo deben tener instalado cryptsetup 2.1 o superior (pero inferior a 3.0). Para obtener más información, visite "Gitlab: Cryptsetup".
- Por motivos de rendimiento, recomendamos que los nodos de trabajo admitan las nuevas instrucciones estándar de cifrado avanzado (AES-ni). Para verificar el soporte de AES-ni, ejecute el siguiente comando:

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

Si no se devuelve nada, su procesador no admite AES-ni. Para obtener más información sobre AES-NI, visite: "Intel: Instrucciones estándar de cifrado avanzado (AES-ni)".

Active el cifrado LUKS

Puede habilitar el cifrado por volumen en el lado del host usando la configuración de clave unificada de Linux (LUKS) para SAN de ONTAP y volúmenes DE ECONOMÍA SAN de ONTAP.

Pasos

1. Defina los atributos de cifrado LUKS en la configuración de backend. Para obtener más información sobre

las opciones de configuración de backend para SAN de ONTAP, consulte "Opciones de configuración de SAN de ONTAP".

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

2. Se utiliza `parameters.selector` para definir los pools de almacenamiento mediante el 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. Cree un secreto que contenga la frase de paso 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 LUKS no pueden aprovechar la deduplicación y la compresión de ONTAP.

Configuración de backend para importar volúmenes LUKS

Para importar un volumen LUKS, debe establecer `luksEncryption` en(`true` en el backend. `luksEncryption`La opción indica a Astra Trident si el volumen es compatible con LUKS (`true`) o no 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 dinámicamente, establezca la anotación `trident.netapp.io/luksEncryption` en `true` e incluya una clase de almacenamiento habilitada para LUKS en la RVP 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
```

Gire una frase de paso LUKS

Puede girar la frase de paso de LUKS y confirmar la rotación.

> No olvide una clave de acceso hasta que haya verificado que ya no hace referencia a ningún volumen, snapshot o secreto. Si se pierde una clave de acceso de referencia, es posible que no se pueda montar el volumen y los datos seguirán estando cifrados e inaccesibles.

Acerca de esta tarea

LA rotación DE la frase de paso LUKS se produce cuando se crea un pod que monta el volumen después de especificar una nueva frase de paso LUKS. Cuando se crea un nuevo pod, Astra Trident compara la frase de paso de LUKS del volumen con la frase de paso activa en el secreto.

- Si la clave de acceso del volumen no coincide con la clave de acceso activa en el secreto, se produce la rotación.
- Si la clave de acceso del volumen coincide con la clave de acceso activa en el secreto, `previous-luks-passphrase` se omite el parámetro.

Pasos

1. Añada `node-publish-secret-name` los parámetros y `node-publish-secret-namespace` 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. Identifique las bases de datos passhrases 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. Actualice el secreto LUKS del volumen para especificar las passphrases nuevas y anteriores. Asegúrese de que `previous-luke-passphrase-name` `previous-luks-passphrase`coincide 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. Cree un nuevo pod montando el volumen. Esto es necesario para iniciar la rotación.
5. Compruebe que se ha girado la frase de paso.

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 giró 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. Puede activar un nuevo pod para intentar completar la rotación.

Habilite la expansión de volumen

Es posible habilitar la ampliación de volumen en un volumen cifrado LUKS.

Pasos

1. Active la `CSINodeExpandSecret` puerta de función (beta 1,25+). Consulte "Kubernetes 1.25: Use Secrets for Node-Driven Expansion of CSI Volumes" para obtener más información.
2. Añada `node-expand-secret-name` los parámetros y `node-expand-secret-namespace` 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

Al iniciar la ampliación de almacenamiento en línea, el kubelet pasa las credenciales adecuadas al controlador.

Configurar el cifrado de Kerberos en tránsito

Con Astra Control Provisioning, puede mejorar la seguridad de acceso a los datos al habilitar el cifrado del tráfico entre su clúster gestionado y el back-end de almacenamiento.

El aprovisionador de control de Astra admite el cifrado de Kerberos en conexiones NFSv3 y NFSv4 desde Red Hat OpenShift y clústeres de Kubernetes ascendentes a volúmenes de ONTAP en las instalaciones.

Puede crear, eliminar, cambiar el tamaño, copiar, clonar, Clone de solo lectura e importe volúmenes que usen cifrado NFS.

Configure el cifrado de Kerberos en tránsito con volúmenes de ONTAP en las instalaciones

Puede habilitar el cifrado de Kerberos en el tráfico de almacenamiento entre su clúster gestionado y un back-end de almacenamiento de ONTAP en las instalaciones.

> El cifrado de Kerberos para el tráfico NFS con back-ends de almacenamiento de ONTAP en las instalaciones solo se admite mediante `ontap-nas` el controlador de almacenamiento.

Antes de empezar

- Asegúrese de "Habilitado Astra Control Provisioning" tener en el clúster gestionado.
- Asegúrese de tener acceso a la `tridentctl` utilidad.
- Asegúrese de tener acceso de administrador al back-end de almacenamiento de ONTAP.
- Asegúrese de conocer el nombre del volumen o los volúmenes que compartirá desde el back-end de almacenamiento ONTAP.
- Asegúrese de haber preparado la máquina virtual de almacenamiento de ONTAP para admitir el cifrado de Kerberos para los volúmenes de NFS. Consulte "Habilite Kerberos en una LIF de datos" para obtener instrucciones.
- Asegúrese de que los volúmenes de NFSv4 GB que utilice con el cifrado de Kerberos se hayan configurado correctamente. Consulte la sección Configuración del dominio de NetApp NFSv4 (página 13) de "Guía de mejoras y prácticas recomendadas de NetApp NFSv4".

Añada o modifique las políticas de exportación de ONTAP

Tiene que agregar reglas a políticas de exportación de ONTAP existentes o crear nuevas políticas de exportación que sean compatibles con el cifrado de Kerberos para el volumen raíz de la máquina virtual de almacenamiento de ONTAP, así como para cualquier volumen de ONTAP compartido con el clúster de Kubernetes ascendente. Las reglas de políticas de exportación que añada, o las nuevas políticas de exportación que cree, deben admitir los siguientes protocolos de acceso y permisos de acceso:

Protocolos de acceso

Configure la directiva de exportación con los protocolos de acceso NFS, NFSv3 y NFSv4.

Detalles de acceso

Puede configurar una de tres versiones diferentes de cifrado de Kerberos, según las necesidades del volumen:

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

Configure la regla de política de exportación de ONTAP con los permisos de acceso adecuados. Por ejemplo, si los clústeres montarán los volúmenes NFS con una combinación de Kerberos 5i y cifrado Kerberos 5p, utilice los siguientes ajustes de acceso:

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

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

- "Cree una política de exportación"
- "Añada una regla a una política de exportación"

Cree un back-end de almacenamiento

Puede crear una configuración de back-end de almacenamiento de Astra Control Provisioner que incluya la funcionalidad de cifrado Kerberos.

Acerca de esta tarea

Al crear un archivo de configuración de backend de almacenamiento que configure el cifrado Kerberos, puede especificar una de las tres versiones diferentes del cifrado Kerberos mediante el `spec.nfsMountOptions` parámetro:

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

Especifique solo un nivel de Kerberos. Si especifica más de un nivel de cifrado de Kerberos en la lista de parámetros, sólo se utilizará la primera opción.

Pasos

1. En el clúster gestionado, cree un archivo de configuración de back-end de almacenamiento utilizando el ejemplo siguiente. Sustituya los valores entre paréntesis <> por información de su 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. Utilice el archivo de configuración que creó en el paso anterior para crear el backend:

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

Si la creación del back-end falla, algo está mal con la configuración del back-end. Puede 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, puede ejecutar de nuevo el comando create.

Cree una clase de almacenamiento

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

Acerca de esta tarea

Al crear un objeto de clase de almacenamiento, puede especificar una de las tres versiones diferentes del cifrado de Kerberos mediante el `mountOptions` parámetro:

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

Especifique solo un nivel de Kerberos. Si especifica más de un nivel de cifrado de Kerberos en la lista de parámetros, sólo se utilizará la primera opción. Si el nivel de cifrado especificado en la configuración de backend de almacenamiento es diferente al nivel especificado en el objeto de clase de almacenamiento, el objeto de clase de almacenamiento tiene prioridad.

Pasos

1. Cree un objeto de Kubernetes StorageClass, mediante 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. Cree la clase de almacenamiento:

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

3. Asegúrese de que se ha creado la clase de almacenamiento:

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

   Debería ver una salida similar a la siguiente:

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

Aprovisione los volúmenes

Después de crear un back-end de almacenamiento y una clase de almacenamiento, ahora puede aprovisionar

un volumen. Consulte estas instrucciones para "aprovisionamiento de un volumen".

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

Puede habilitar el cifrado de Kerberos en el tráfico de almacenamiento entre su clúster gestionado y un solo back-end de almacenamiento de Azure NetApp Files o un pool virtual de back-ends de almacenamiento de Azure NetApp Files.

Antes de empezar

- Asegúrese de haber habilitado el aprovisionador de Astra Control en el clúster Red Hat OpenShift gestionado. Consulte "Habilita el aprovisionador de Astra Control" para obtener instrucciones.
- Asegúrese de tener acceso a la `tridentctl` utilidad.
- Asegúrese de haber preparado el back-end de almacenamiento de Azure NetApp Files para el cifrado Kerberos siguiendo los requisitos y siguiendo las instrucciones de "Documentación de Azure NetApp Files".
- Asegúrese de que los volúmenes de NFSv4 GB que utilice con el cifrado de Kerberos se hayan configurado correctamente. Consulte la sección Configuración del dominio de NetApp NFSv4 (página 13) de "Guía de mejoras y prácticas recomendadas de NetApp NFSv4".

Cree un back-end de almacenamiento

Puede crear una configuración de back-end de almacenamiento de Azure NetApp Files que incluya la funcionalidad de cifrado de Kerberos.

Acerca de esta tarea

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

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

Cuando se define la configuración en el nivel del pool virtual, el pool se selecciona con la etiqueta de la clase de almacenamiento.

En cualquier nivel, puede 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 encriptación con protección de identidad y privacidad)

Pasos

1. En el clúster gestionado, cree un archivo de configuración de back-end de almacenamiento mediante uno de los siguientes ejemplos, en función del lugar donde necesite definir el back-end de almacenamiento (nivel de back-end de almacenamiento o nivel de pool virtual). Sustituya los valores entre paréntesis <> por información de su entorno:

Ejemplo de nivel de back-end 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 pool virtual

```
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. Utilice el archivo de configuración que creó en el paso anterior para crear el backend:

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

Si la creación del back-end falla, algo está mal con la configuración del back-end. Puede 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, puede ejecutar de nuevo el comando create.

Cree una clase de almacenamiento

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

Pasos

1. Cree un objeto de Kubernetes StorageClass, mediante 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. Cree la clase de almacenamiento:

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

3. Asegúrese de que se ha creado la clase de almacenamiento:

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

Debería ver una salida similar a la siguiente:

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

Aprovisione los volúmenes

Después de crear un back-end de almacenamiento y una clase de almacenamiento, ahora puede aprovisionar un volumen. Consulte estas instrucciones para "aprovisionamiento de un volumen".