Seguridad

Trident

NetApp
January 15, 2026

Tabla de contenidos

Seguridad

Seguridad

Siga las recomendaciones que se enumeran aquí para garantizar la seguridad de su instalación de Trident .

Ejecutar Trident en su propio espacio de nombres.

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

Para separar las demás aplicaciones y usuarios de Trident, instale siempre Trident en su propio espacio de nombres de Kubernetes.(`trident`). Al colocar Trident en su propio espacio de nombres, se garantiza 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 corresponde) almacenados en los objetos CRD con espacio de nombres. Debe asegurarse de permitir únicamente a los administradores el acceso al espacio de nombres de Trident y, por lo tanto, el acceso a `tridentctl` solicitud.

Utilice la autenticación CHAP con los backends SAN de ONTAP.

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

Para los backends ONTAP que utilizan los controladores de almacenamiento SAN, Trident puede configurar CHAP bidireccional y administrar los nombres de usuario y secretos de CHAP a través de `tridentctl` . Referirse a"Prepárese para configurar el backend con los controladores SAN de ONTAP." Para comprender cómo Trident configura CHAP en los sistemas backend de ONTAP .

Utilice la autenticación CHAP con los sistemas backend NetApp HCI y SolidFire.

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

Los volúmenes creados por Trident no están asociados a ningún grupo de acceso a volúmenes.

Utilice Trident con NVE y NAE

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

- Si NAE está habilitado en el backend, cualquier volumen aprovisionado en Trident estará habilitado para NAE.
  - Puedes configurar la bandera de cifrado NVE a `""` para crear volúmenes compatibles con NAE.

- Si NAE no está habilitado en el backend, cualquier volumen aprovisionado en Trident tendrá NVE habilitado a menos que la bandera de cifrado NVE esté configurada en `false` (el valor predeterminado) en la configuración del backend.

> Los volúmenes creados en Trident en un backend habilitado para NAE deben estar cifrados con NVE o NAE.
>
> - Puedes configurar la bandera 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.
> - Configurar la bandera de cifrado NVE a `false` En un backend habilitado para NAE se crea un volumen habilitado para NAE. No se puede deshabilitar el cifrado NAE configurando la bandera de cifrado NVE en `false` .

- Puede crear manualmente un volumen NVE en Trident configurando explícitamente la bandera de cifrado NVE a `true` .

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

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

Configuración de clave unificada de Linux (LUKS)

Puede habilitar Linux Unified Key Setup (LUKS) para cifrar los volúmenes ONTAP SAN y ONTAP SAN ECONOMY en Trident. Trident admite la rotación de contraseñas y la expansión de volumen para volúmenes cifrados con LUKS.

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

> El cifrado LUKS no es compatible con los sistemas ASA r2. Para obtener información sobre los sistemas ASA r2, consulte"Conozca 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 obtener más información, visite"Gitlab: configuración de criptografía" .
- Por motivos de rendimiento, NetApp recomienda que los nodos de trabajo sean compatibles con las nuevas instrucciones del estándar de cifrado avanzado (AES-NI). Para verificar la compatibilidad con 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 del estándar de cifrado avanzado (AES-NI)" .

Habilitar el cifrado LUKS

Puede habilitar el cifrado por volumen en el host mediante Linux Unified Key Setup (LUKS) para volúmenes ONTAP SAN y ONTAP SAN ECONOMY.

Pasos

1. Define los atributos de cifrado LUKS en la configuración del backend. Para obtener más información sobre las opciones de configuración de backend para ONTAP SAN, consulte"Opciones de configuración 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. Usar `parameters.selector` para definir los grupos de almacenamiento utilizando 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. 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 compresión de ONTAP .

Configuración del backend para importar volúmenes LUKS

Para importar un volumen LUKS, debe configurar `luksEncryption` a(`true` en el backend. El `luksEncryption` Esta opción le indica a Trident si el volumen es compatible con LUKS.(`true`) o no cumple 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, configure la anotación `trident.netapp.io/luksEncryption` a `true` e incluir una clase de almacenamiento habilitada para LUKS en el 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 cambiar la contraseña LUKS y confirmar el cambio.

> No olvide una frase de contraseña hasta que haya verificado que ya no está siendo referenciada por ningún volumen, instantánea o secreto. Si se pierde la contraseña a la que se hace referencia, es posible que no pueda montar el volumen y los datos permanecerán cifrados e inaccesibles.

Acerca de esta tarea

La rotación de la contraseña LUKS se produce cuando se crea un pod que monta el volumen después de que se haya especificado una nueva contraseña LUKS. Cuando se crea un nuevo pod, Trident compara la contraseña LUKS del volumen con la contraseña activa del secreto.

- Si la contraseña del volumen no coincide con la contraseña activa del secreto, se produce una rotación.
- Si la contraseña del volumen coincide con la contraseña activa del secreto, `previous-luks-passphrase` El parámetro se ignora.

Pasos

1. Añade el `node-publish-secret-name` y `node-publish-secret-namespace` Parámetros de 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. Identificar las contraseñas existentes en el volumen o instantánea.

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 la clave secreta LUKS del volumen para especificar las contraseñas nueva y anterior. Asegurar `previous-luke-passphrase-name` y `previous-luks-passphrase` Coincide con la 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 para montar el volumen. Esto es necesario para iniciar la rotación.
5. Verifique que la contraseña se haya rotado.

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 contraseña se rotó cuando solo se devolvió la nueva contraseña en el volumen y la instantánea.

> Si se devuelven dos frases de contraseña, por ejemplo `luksPassphraseNames: ["B", "A"]` , la rotación está incompleta. Puedes activar una nueva cápsula para intentar completar la rotación.

Habilitar la expansión de volumen

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

Pasos

1. Habilitar el `CSINodeExpandSecret` Puerta de características (beta 1.25+). Referirse a "Kubernetes 1.25: Uso de secretos para la expansión de volúmenes CSI basada en nodos" Para más detalles.
2. Añade el `node-expand-secret-name` y `node-expand-secret-namespace` Parámetros de 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 se inicia la expansión del almacenamiento en línea, el kubelet pasa las credenciales apropiadas al controlador.

Cifrado en vuelo Kerberos

Mediante el cifrado en tránsito de Kerberos, puede mejorar la seguridad del acceso a los datos habilitando el cifrado para el tráfico entre su clúster administrado y el backend de almacenamiento.

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

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

Puede crear, eliminar, cambiar el tamaño, crear instantáneas, clonar, clonar en modo de solo lectura e importar volúmenes que utilicen cifrado NFS.

Configurar el cifrado Kerberos en vuelo con volúmenes ONTAP locales

Puede habilitar el cifrado Kerberos en el tráfico de almacenamiento entre su clúster administrado y un backend de almacenamiento ONTAP local.

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

Antes de empezar

- Asegúrese de tener acceso a `tridentctl` utilidad.
- Asegúrese de tener acceso de administrador al backend de almacenamiento de ONTAP .
- Asegúrese de conocer el nombre del volumen o volúmenes que compartirá desde el backend de almacenamiento de ONTAP .
- Asegúrese de haber preparado la máquina virtual de almacenamiento ONTAP para que admita el cifrado Kerberos para volúmenes NFS. Referirse a "Habilitar Kerberos en un dataLIF" para obtener instrucciones.
- Asegúrese de que todos los volúmenes NFSv4 que utilice con cifrado Kerberos estén configurados correctamente. Consulte la sección Configuración del dominio NFSv4 de NetApp (página 13) del "Guía de mejoras y prácticas recomendadas de NetApp NFSv4" .

Agregar o modificar las políticas de exportación de ONTAP

Debe agregar 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 de ONTAP , así como para cualquier volumen de ONTAP compartido con el clúster de Kubernetes ascendente. Las reglas de política de exportación que agregue, o las nuevas políticas de exportación que cree, deben admitir los siguientes protocolos de acceso y permisos de acceso:

Protocolos de acceso

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

Detalles de acceso

Puede configurar una de las tres versiones diferentes de cifrado Kerberos, según sus necesidades de 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)

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

Consulte la siguiente documentación para obtener información sobre cómo crear políticas de exportación de ONTAP y reglas de políticas de exportación:

- "Crear una política de exportación"
- "Agregar una regla a una política de exportación"

Crea un backend de almacenamiento

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

Acerca de esta tarea

Cuando crea un archivo de configuración de backend de almacenamiento que configura el cifrado Kerberos, puede especificar una de las tres versiones diferentes de 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 cifrado con protección de identidad y privacidad)

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

Pasos

1. En el clúster administrado, cree un archivo de configuración de backend de almacenamiento utilizando el siguiente ejemplo. Reemplace los valores entre corchetes <> con 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 falla la creación del backend, algo falla en la configuración del backend. Puedes consultar 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 volver a ejecutar el comando de creación.

Crear una clase de almacenamiento

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

Acerca de esta tarea

Al crear un objeto de clase de almacenamiento, puede especificar una de las tres versiones diferentes de cifrado 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 cifrado con protección de identidad y privacidad)

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

Pasos

1. Crea un objeto StorageClass de Kubernetes, utilizando 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. Asegúrese de que se haya creado la clase de almacenamiento:

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

   Debería ver un resultado similar al siguiente:

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

volúmenes de provisiones

Después de crear un backend de almacenamiento y una clase de almacenamiento, ahora puede aprovisionar un volumen. Para obtener instrucciones, consulte "Provisión de un volumen" .

Configurar el cifrado Kerberos en vuelo con volúmenes de Azure NetApp Files

Puede habilitar el cifrado Kerberos en el tráfico de almacenamiento entre su 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

- Asegúrese de haber habilitado Trident en el clúster Red Hat OpenShift administrado.
- Asegúrese de tener acceso a `tridentctl` utilidad.
- Asegúrese de haber preparado el backend de almacenamiento de Azure NetApp Files para el cifrado Kerberos teniendo en cuenta los requisitos y siguiendo las instrucciones en "Documentación de Azure NetApp Files" .
- Asegúrese de que todos los volúmenes NFSv4 que utilice con cifrado Kerberos estén configurados correctamente. Consulte la sección Configuración del dominio NFSv4 de NetApp (página 13) del "Guía de mejoras y prácticas recomendadas de NetApp NFSv4" .

Crea un backend de almacenamiento

Puede crear una configuración de backend de almacenamiento de Azure NetApp Files que incluya la capacidad de cifrado 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 dos niveles posibles:

- El **nivel de backend de almacenamiento** que utiliza el `spec.kerberos` campo
- El **nivel de piscina virtual** que utiliza el `spec.storage.kerberos` campo

Cuando se define la configuración a nivel de grupo virtual, el grupo se selecciona utilizando la etiqueta en la clase de almacenamiento.

En cualquiera de los dos niveles, puede especificar una de las tres versiones diferentes de 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, cree un archivo de configuración de backend de almacenamiento utilizando uno de los siguientes ejemplos, según dónde necesite definir el backend de almacenamiento (nivel de backend de almacenamiento o nivel de grupo virtual). Reemplace los valores entre corchetes <> con información de su 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 piscina 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 falla la creación del backend, algo falla en la configuración del backend. Puedes consultar 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 volver a ejecutar el comando de creación.

Crear una clase de almacenamiento

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

Pasos

1. Crea un objeto StorageClass de Kubernetes, utilizando 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. Asegúrese de que se haya creado la clase de almacenamiento:

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

Debería ver un resultado similar al siguiente:

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

volúmenes de provisiones

Después de crear un backend de almacenamiento y una clase de almacenamiento, ahora puede aprovisionar un volumen. Para obtener instrucciones, consulte "Provisión de un volumen" .