Segurança

Trident

NetApp
January 15, 2026

Índice

Segurança

Segurança

Utilize as recomendações listadas aqui para garantir a segurança da sua instalação do Trident .

Execute o Trident em seu próprio espaço de nomes.

É importante impedir que aplicativos, administradores de aplicativos, usuários e aplicativos de gerenciamento acessem as definições de objetos do Trident ou os pods para garantir um armazenamento confiável e bloquear possíveis atividades maliciosas.

Para separar os outros aplicativos e usuários do Trident, sempre instale o Trident em seu próprio namespace do Kubernetes.(`trident`). Ao colocar o Trident em seu próprio namespace, garante-se que apenas a equipe administrativa do Kubernetes tenha acesso ao pod do Trident e aos artefatos (como segredos de backend e CHAP, se aplicável) armazenados nos objetos CRD do namespace. Você deve garantir que apenas administradores tenham acesso ao namespace Trident e, portanto, acesso ao... `tridentctl` aplicativo.

Utilize a autenticação CHAP com backends SAN do ONTAP.

O Trident oferece suporte à autenticação baseada em CHAP para cargas de trabalho ONTAP SAN (usando o `ontap-san` e `ontap-san-economy` motoristas). A NetApp recomenda o uso de CHAP bidirecional com Trident para autenticação entre um host e o backend de armazenamento.

Para backends ONTAP que utilizam drivers de armazenamento SAN, o Trident pode configurar CHAP bidirecional e gerenciar nomes de usuário e segredos CHAP através de `tridentctl` . Consulte"Prepare-se para configurar o backend com os drivers ONTAP SAN." Para entender como o Trident configura o CHAP em backends ONTAP .

Utilize a autenticação CHAP com os backends NetApp HCI e SolidFire.

A NetApp recomenda a implementação do CHAP bidirecional para garantir a autenticação entre um host e os backends do NetApp HCI e SolidFire . O Trident utiliza um objeto secreto que inclui duas senhas CHAP por locatário. Quando o Trident é instalado, ele gerencia os segredos CHAP e os armazena em um `tridentvolume` Objeto CR para o respectivo PV. Ao criar um PV, o Trident usa os segredos CHAP para iniciar uma sessão iSCSI e se comunicar com o sistema NetApp HCI e SolidFire via CHAP.

Os volumes criados pelo Trident não estão associados a nenhum Grupo de Acesso a Volumes.

Use Trident com NVE e NAE

O NetApp ONTAP oferece criptografia de dados em repouso para proteger dados confidenciais caso um disco seja roubado, devolvido ou reutilizado. Para mais detalhes, consulte"Visão geral da configuração da criptografia de volume do NetApp" .

- Se o NAE estiver habilitado no backend, qualquer volume provisionado no Trident terá o NAE habilitado.
  - Você pode definir o sinalizador de criptografia NVE para `""` Para criar volumes habilitados para NAE.
- Se o NAE não estiver habilitado no backend, qualquer volume provisionado no Trident terá o NVE habilitado, a menos que o sinalizador de criptografia NVE esteja definido como `false` (o valor padrão) na

configuração do backend.

> Os volumes criados no Trident em um backend habilitado para NAE devem ser criptografados com NVE ou NAE.
>
> - Você pode definir o sinalizador de criptografia NVE para `true` Na configuração do backend do Trident , é possível substituir a criptografia NAE e usar uma chave de criptografia específica para cada volume.
> - Definir o sinalizador de criptografia NVE para `false` Em um backend habilitado para NAE, cria-se um volume habilitado para NAE. Não é possível desativar a criptografia NAE definindo o sinalizador de criptografia NVE como `false` .

- Você pode criar manualmente um volume NVE no Trident definindo explicitamente o sinalizador de criptografia NVE para `true` .

Para obter mais informações sobre as opções de configuração do backend, consulte:

- "Opções de configuração do ONTAP SAN"
- "Opções de configuração do ONTAP NAS"

Configuração Unificada de Chaves do Linux (LUKS)

Você pode habilitar o Linux Unified Key Setup (LUKS) para criptografar volumes ONTAP SAN e ONTAP SAN ECONOMY no Trident. O Trident suporta rotação de senha e expansão de volume para volumes criptografados com LUKS.

No Trident, os volumes criptografados com LUKS usam a cifra e o modo aes-xts-plain64, conforme recomendado por"NIST" .

> A criptografia LUKS não é compatível com sistemas ASA r2. Para obter informações sobre sistemas ASA r2, consulte"Saiba mais sobre os sistemas de armazenamento ASA r2" .

Antes de começar

- Os nós de trabalho devem ter o cryptsetup 2.1 ou superior (mas inferior a 3.0) instalado. Para mais informações, visite"Gitlab: cryptsetup" .
- Por motivos de desempenho, a NetApp recomenda que os nós de trabalho suportem o padrão de criptografia avançada New Instructions (AES-NI). Para verificar a compatibilidade com AES-NI, execute o seguinte comando:

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

Se nada for retornado, seu processador não suporta AES-NI. Para obter mais informações sobre AES-NI, visite:"Intel: Instruções do Padrão de Criptografia Avançada (AES-NI)" .

Ativar criptografia LUKS

Você pode habilitar a criptografia por volume no lado do host usando o Linux Unified Key Setup (LUKS) para volumes ONTAP SAN e ONTAP SAN ECONOMY.

Passos

1. Defina os atributos de criptografia LUKS na configuração do backend. Para obter mais informações sobre as opções de configuração de backend para ONTAP SAN, consulte"Opções de configuração do ONTAP SAN" .

```
{
  "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 os pools de armazenamento usando criptografia LUKS. Por exemplo:

```
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. Crie um segredo que contenha a senha LUKS. Por exemplo:

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

Limitações

Volumes criptografados com LUKS não podem aproveitar a deduplicação e a compressão do ONTAP .

Configuração de backend para importação de volumes LUKS

Para importar um volume LUKS, você deve configurar `luksEncryption` para(`true` nos bastidores. O `luksEncryption` Essa opção informa ao Trident se o volume é compatível com LUKS.(`true`) ou não compatível com LUKS(`false`) conforme mostrado no exemplo a seguir.

```
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'
```

Configuração de PVC para importação de volumes LUKS

Para importar volumes LUKS dinamicamente, defina a anotação. `trident.netapp.io/luksEncryption` para `true` e inclua uma classe de armazenamento habilitada para LUKS no PVC, conforme mostrado neste exemplo.

```
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
```

Rotacionar uma senha LUKS

Você pode alternar a senha LUKS e confirmar a alternância.

> Não se esqueça de uma senha até verificar se ela não está mais sendo referenciada por nenhum volume, snapshot ou segredo. Caso a senha referenciada seja perdida, você poderá não conseguir montar o volume e os dados permanecerão criptografados e inacessíveis.

Sobre esta tarefa

A rotação da senha LUKS ocorre quando um pod que monta o volume é criado após a especificação de uma nova senha LUKS. Quando um novo pod é criado, o Trident compara a senha LUKS no volume com a senha ativa no segredo.

- Se a senha no volume não corresponder à senha ativa no segredo, ocorrerá a rotação.
- Se a senha no volume corresponder à senha ativa no segredo, o `previous-luks-passphrase` O parâmetro é ignorado.

Passos

1. Adicione o `node-publish-secret-name` e `node-publish-secret-namespace` Parâmetros da classe de armazenamento. Por exemplo:

```
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 as senhas existentes no volume ou no snapshot.

Volume

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

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

Instantâneo

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

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

3. Atualize o segredo LUKS do volume para especificar as senhas novas e anteriores. Garantir `previous-luke-passphrase-name` e `previous-luks-passphrase` A senha corresponde à senha 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. Crie um novo pod para montar o volume. Isso é necessário para iniciar a rotação.
5. Verifique se a senha foi rotacionada.

Volume

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

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

Instantâneo

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

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

Resultados

A senha foi rotacionada quando apenas a nova senha foi retornada no volume e no snapshot.

Se duas senhas forem retornadas, por exemplo `luksPassphraseNames: ["B", "A"]`, a rotação está incompleta. Você pode acionar um novo módulo para tentar completar a rotação.

Ativar expansão de volume

Você pode habilitar a expansão de volume em um volume criptografado com LUKS.

Passos

1. Ative o `CSINodeExpandSecret` recurso gate (beta 1.25+). Consulte "Kubernetes 1.25: Use segredos para expansão de volumes CSI orientada a nós" para mais detalhes.
2. Adicione o `node-expand-secret-name` e `node-expand-secret-namespace` Parâmetros da classe de armazenamento. Por exemplo:

```
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

Ao iniciar a expansão do armazenamento online, o kubelet passa as credenciais apropriadas para o driver.

Criptografia Kerberos em voo

Ao usar a criptografia Kerberos em trânsito, você pode melhorar a segurança do acesso aos dados, habilitando a criptografia para o tráfego entre seu cluster gerenciado e o backend de armazenamento.

O Trident oferece suporte à criptografia Kerberos para ONTAP como backend de armazenamento:

- * ONTAP local* - O Trident oferece suporte à criptografia Kerberos em conexões NFSv3 e NFSv4 de clusters Red Hat OpenShift e Kubernetes upstream para volumes ONTAP locais.

Você pode criar, excluir, redimensionar, criar snapshots, clonar, clonar em modo somente leitura e importar volumes que utilizam criptografia NFS.

Configure a criptografia Kerberos em trânsito com volumes ONTAP locais.

Você pode habilitar a criptografia Kerberos no tráfego de armazenamento entre seu cluster gerenciado e um backend de armazenamento ONTAP local.

> A criptografia Kerberos para tráfego NFS com backends de armazenamento ONTAP locais só é compatível com o uso do `ontap-nas` driver de armazenamento.

Antes de começar

- Certifique-se de ter acesso ao `tridentctl` utilidade.
- Certifique-se de ter acesso de administrador ao backend de armazenamento do ONTAP .
- Certifique-se de saber o nome do(s) volume(s) que você compartilhará do backend de armazenamento ONTAP .
- Certifique-se de ter preparado a máquina virtual de armazenamento ONTAP para suportar a criptografia Kerberos para volumes NFS. Consulte "Habilitar Kerberos em um dataLIF" para obter instruções.
- Certifique-se de que todos os volumes NFSv4 que você usa com criptografia Kerberos estejam configurados corretamente. Consulte a seção Configuração de Domínio NFSv4 da NetApp (página 13) do manual. "Guia de Aprimoramentos e Melhores Práticas do NetApp NFSv4" .

Adicionar ou modificar políticas de exportação do ONTAP

Você precisa adicionar regras às políticas de exportação ONTAP existentes ou criar novas políticas de exportação que suportem criptografia Kerberos para o volume raiz da VM de armazenamento ONTAP , bem como para quaisquer volumes ONTAP compartilhados com o cluster Kubernetes upstream. As regras de política de exportação que você adicionar, ou as novas políticas de exportação que você criar, precisam ser compatíveis com os seguintes protocolos de acesso e permissões de acesso:

Protocolos de acesso

Configure a política de exportação com os protocolos de acesso NFS, NFSv3 e NFSv4.

Detalhes de acesso

Você pode configurar uma das três versões diferentes de criptografia Kerberos, dependendo das suas necessidades para o volume:

- **Kerberos 5** - (autenticação e criptografia)
- **Kerberos 5i** - (autenticação e criptografia com proteção de identidade)
- **Kerberos 5p** - (autenticação e criptografia com proteção de identidade e privacidade)

Configure a regra de política de exportação do ONTAP com as permissões de acesso apropriadas. Por exemplo, se os clusters forem montar os volumes NFS com uma combinação de criptografia Kerberos 5i e Kerberos 5p, use as seguintes configurações de acesso:

Tipo	Acesso somente leitura	Acesso de leitura/gravação	Acesso de superusuário
UNIX	Habilitado	Habilitado	Habilitado
Kerberos 5i	Habilitado	Habilitado	Habilitado
Kerberos 5p	Habilitado	Habilitado	Habilitado

Consulte a seguinte documentação para obter informações sobre como criar políticas de exportação e regras de política de exportação do ONTAP :

- "Crie uma política de exportação"
- "Adicionar uma regra a uma política de exportação"

Crie um backend de armazenamento

Você pode criar uma configuração de backend de armazenamento Trident que inclua a capacidade de criptografia Kerberos.

Sobre esta tarefa

Ao criar um arquivo de configuração de backend de armazenamento que configura a criptografia Kerberos, você pode especificar uma das três versões diferentes de criptografia Kerberos usando o `spec.nfsMountOptions` parâmetro:

- `spec.nfsMountOptions: sec=krb5`(autenticação e criptografia)
- `spec.nfsMountOptions: sec=krb5i`(autenticação e criptografia com proteção de identidade)
- `spec.nfsMountOptions: sec=krb5p`(autenticação e criptografia com proteção de identidade e privacidade)

Especifique apenas um nível Kerberos. Se você especificar mais de um nível de criptografia Kerberos na lista de parâmetros, somente a primeira opção será utilizada.

Passos

1. No cluster gerenciado, crie um arquivo de configuração de backend de armazenamento usando o seguinte exemplo. Substitua os valores entre colchetes <> por informações do seu ambiente:

```
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. Utilize o arquivo de configuração que você criou na etapa anterior para criar o backend:

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

Se a criação do backend falhar, há algo errado com a configuração do backend. Você pode visualizar os registros para determinar a causa executando o seguinte comando:

```
tridentctl logs
```

Após identificar e corrigir o problema com o arquivo de configuração, você poderá executar o comando de criação novamente.

Criar uma classe de armazenamento

Você pode criar uma classe de armazenamento para provisionar volumes com criptografia Kerberos.

Sobre esta tarefa

Ao criar um objeto de classe de armazenamento, você pode especificar uma das três versões diferentes de criptografia Kerberos usando o `mountOptions` parâmetro:

- `mountOptions: sec=krb5`(autenticação e criptografia)
- `mountOptions: sec=krb5i`(autenticação e criptografia com proteção de identidade)
- `mountOptions: sec=krb5p`(autenticação e criptografia com proteção de identidade e privacidade)

Especifique apenas um nível Kerberos. Se você especificar mais de um nível de criptografia Kerberos na lista de parâmetros, somente a primeira opção será utilizada. Se o nível de criptografia especificado na configuração do backend de armazenamento for diferente do nível especificado no objeto da classe de armazenamento, o objeto da classe de armazenamento terá precedência.

Passos

1. Crie um objeto Kubernetes do tipo StorageClass, utilizando o seguinte exemplo:

```
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. Crie a classe de armazenamento:

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

3. Certifique-se de que a classe de armazenamento foi criada:

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

   Você deverá ver uma saída semelhante à seguinte:

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

volumes de provisão

Após criar um backend de armazenamento e uma classe de armazenamento, você poderá provisionar um volume. Para obter instruções, consulte "Forneça um volume" .

Configure a criptografia Kerberos em trânsito com volumes do Azure NetApp Files.

Você pode habilitar a criptografia Kerberos no tráfego de armazenamento entre seu cluster gerenciado e um único backend de armazenamento do Azure NetApp Files ou um pool virtual de backends de armazenamento do Azure NetApp Files .

Antes de começar

- Certifique-se de ter habilitado o Trident no cluster Red Hat OpenShift gerenciado.
- Certifique-se de ter acesso ao `tridentctl` utilidade.
- Certifique-se de ter preparado o backend de armazenamento do Azure NetApp Files para criptografia Kerberos, observando os requisitos e seguindo as instruções em "Documentação do Azure NetApp Files" .
- Certifique-se de que todos os volumes NFSv4 que você usa com criptografia Kerberos estejam configurados corretamente. Consulte a seção Configuração de Domínio NFSv4 da NetApp (página 13) do manual. "Guia de Aprimoramentos e Melhores Práticas do NetApp NFSv4" .

Crie um backend de armazenamento

Você pode criar uma configuração de back-end de armazenamento do Azure NetApp Files que inclua a capacidade de criptografia Kerberos.

Sobre esta tarefa

Ao criar um arquivo de configuração de backend de armazenamento que configura a criptografia Kerberos, você pode defini-lo para ser aplicado em um dos dois níveis possíveis:

- O **nível de backend de armazenamento** usando o `spec.kerberos` campo
- O **nível da piscina virtual** usando o `spec.storage.kerberos` campo

Ao definir a configuração no nível do pool virtual, o pool é selecionado usando o rótulo na classe de armazenamento.

Em qualquer um dos níveis, você pode especificar uma das três versões diferentes de criptografia Kerberos:

- `kerberos: sec=krb5`(autenticação e criptografia)
- `kerberos: sec=krb5i`(autenticação e criptografia com proteção de identidade)
- `kerberos: sec=krb5p`(autenticação e criptografia com proteção de identidade e privacidade)

Passos

1. No cluster gerenciado, crie um arquivo de configuração de backend de armazenamento usando um dos exemplos a seguir, dependendo de onde você precisa definir o backend de armazenamento (nível de backend de armazenamento ou nível de pool virtual). Substitua os valores entre colchetes <> por informações do seu ambiente:

Exemplo de nível de backend de armazenamento

```
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
```

Exemplo de nível 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. Utilize o arquivo de configuração que você criou na etapa anterior para criar o backend:

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

Se a criação do backend falhar, há algo errado com a configuração do backend. Você pode visualizar os registros para determinar a causa executando o seguinte comando:

```
tridentctl logs
```

Após identificar e corrigir o problema com o arquivo de configuração, você poderá executar o comando de criação novamente.

Criar uma classe de armazenamento

Você pode criar uma classe de armazenamento para provisionar volumes com criptografia Kerberos.

Passos

1. Crie um objeto Kubernetes do tipo StorageClass, utilizando o seguinte exemplo:

```
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. Crie a classe de armazenamento:

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

3. Certifique-se de que a classe de armazenamento foi criada:

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

Você deverá ver uma saída semelhante à seguinte:

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

volumes de provisão

Após criar um backend de armazenamento e uma classe de armazenamento, você poderá provisionar um volume. Para obter instruções, consulte "Forneça um volume" .

Informações sobre direitos autorais

Informações sobre marcas comerciais