



# Configurar backends

## Astra Trident

NetApp  
February 05, 2025

# Índice

- Configurar backends ..... 1
  - Configurar um back-end do Azure NetApp Files ..... 1
  - Configurar um back-end do CVS para GCP ..... 9
  - Configurar um back-end NetApp HCI ou SolidFire ..... 18
  - Configure um back-end com drivers SAN ONTAP ou Cloud Volumes ONTAP ..... 25
  - Configurar um back-end com drivers nas ONTAP ..... 44
  - Use o Astra Trident com o Amazon FSX para NetApp ONTAP ..... 65

# Configurar backends

Um back-end define a relação entre o Astra Trident e um sistema de storage. Ele diz ao Astra Trident como se comunicar com esse sistema de storage e como o Astra Trident deve provisionar volumes a partir dele. O Astra Trident oferecerá automaticamente pools de storage de back-ends que atendem aos requisitos definidos por uma classe de storage. Saiba mais sobre como configurar o back-end com base no tipo de sistema de armazenamento que você tem.

- ["Configurar um back-end do Azure NetApp Files"](#)
- ["Configure um back-end do Cloud Volumes Service para o Google Cloud Platform"](#)
- ["Configurar um back-end NetApp HCI ou SolidFire"](#)
- ["Configurar um back-end com drivers nas ONTAP ou Cloud Volumes ONTAP"](#)
- ["Configure um back-end com drivers SAN ONTAP ou Cloud Volumes ONTAP"](#)
- ["Use o Astra Trident com o Amazon FSX para NetApp ONTAP"](#)

## Configurar um back-end do Azure NetApp Files

Saiba mais sobre como configurar o Azure NetApp Files (ANF) como back-end para sua instalação do Astra Trident usando as configurações de amostra fornecidas.



O serviço Azure NetApp Files não suporta volumes inferiores a 100 GB. O Astra Trident cria automaticamente volumes de 100 GB se um volume menor for solicitado.

### O que você vai precisar

Para configurar e usar um ["Azure NetApp Files"](#) back-end, você precisa do seguinte:

- `subscriptionID` A partir de uma subscrição do Azure com o Azure NetApp Files ativado.
- `tenantID`, `clientID` E `clientSecret` de um ["Registo da aplicação"](#) no Azure active Directory com permissões suficientes para o serviço Azure NetApp Files. O Registo de aplicações deve utilizar a `Owner` função ou `Contributor` predefinida pelo Azure.



Para saber mais sobre as funções incorporadas do Azure, consulte o ["Documentação do Azure"](#).

- O Azure `location` que contém pelo menos um ["sub-rede delegada"](#). A partir do Trident 22,01, o `location` parâmetro é um campo obrigatório no nível superior do arquivo de configuração de back-end. Os valores de localização especificados em pools virtuais são ignorados.
- Se você estiver usando o Azure NetApp Files pela primeira vez ou em um novo local, alguma configuração inicial será necessária. Consulte ["guia quickstart"](#) .

### Sobre esta tarefa

Com base na configuração de back-end (sub-rede, rede virtual, nível de serviço e local), o Trident cria volumes do ANF em pools de capacidade disponíveis no local solicitado e correspondem ao nível de serviço e à sub-rede solicitados.



**OBSERVAÇÃO:** O Astra Trident não é compatível com pools de capacidade de QoS manual.

## Opções de configuração de back-end

Consulte a tabela a seguir para obter as opções de configuração de back-end:

Parâmetro	Descrição	Padrão
version		Sempre 1
storageDriverName	Nome do controlador de armazenamento	"ficheiros azure-NetApp"
backendName	Nome personalizado ou back-end de storage	Nome do condutor e caracteres aleatórios
subscriptionID	O ID da assinatura da sua assinatura do Azure	
tenantID	O ID do locatário de um Registro de aplicativo	
clientID	A ID do cliente de um registo de aplicação	
clientSecret	O segredo do cliente de um Registro de aplicativo	
serviceLevel	Um de Standard, Premium, ou Ultra	"" (aleatório)
location	Nome do local do Azure onde os novos volumes serão criados	
serviceLevel	Um de Standard, Premium, ou Ultra	"" (aleatório)
resourceGroups	Lista de grupos de recursos para filtragem de recursos descobertos	[] (sem filtro)
netappAccounts	Lista de contas do NetApp para filtragem de recursos descobertos	[] (sem filtro)
capacityPools	Lista de pools de capacidade para filtrar recursos descobertos	[] (sem filtro, aleatório)
virtualNetwork	Nome de uma rede virtual com uma sub-rede delegada	""
subnet	Nome de uma sub-rede delegada Microsoft.Netapp/volumes	""
nfsMountOptions	Controle refinado das opções de montagem NFS.	"3"
limitVolumeSize	Falha no provisionamento se o tamanho do volume solicitado estiver acima desse valor	"" (não aplicado por padrão)

Parâmetro	Descrição	Padrão
debugTraceFlags	Debug flags para usar ao solucionar problemas. Exemplo, <code>\{"api": false, "method": true, "discovery": true\}</code> . Não use isso a menos que você esteja solucionando problemas e exija um despejo de log detalhado.	nulo



Se você encontrar um erro "sem pools de capacidade encontrados" ao tentar criar um PVC, é provável que o Registro do aplicativo não tenha as permissões e recursos necessários (sub-rede, rede virtual, pool de capacidade) associados. O Astra Trident registrará os recursos do Azure descobertos quando o back-end for criado quando o debug estiver habilitado. Certifique-se de verificar se está a ser utilizada uma função adequada.



Se você quiser montar os volumes usando o NFS versão 4,1, você pode incluir `nfsvers=4` na lista de opções de montagem delimitadas por vírgulas para escolher NFS v4,1. Todas as opções de montagem definidas em uma classe de armazenamento substituem as opções de montagem definidas em um arquivo de configuração de back-end.

Os valores para `resourceGroups`, `netappAccounts`, `capacityPools`, `virtualNetwork` e `subnet` podem ser especificados usando nomes curtos ou totalmente qualificados. Nomes curtos podem corresponder vários recursos com o mesmo nome, portanto, o uso de nomes totalmente qualificados é recomendado na maioria das situações. Os `resourceGroups` valores `netappAccounts`, e `capacityPools` são filtros que restringem o conjunto de recursos descobertos aos disponíveis para esse back-end de armazenamento e podem ser especificados em qualquer combinação. Os nomes totalmente qualificados são do seguinte formato:

Tipo	Formato
Grupo de recursos	<code>&lt;resource group&gt;</code>
Conta NetApp	<code>&lt;resource group&gt;/ cliente NetApp account&gt;</code>
Pool de capacidade	<code>&lt;resource group&gt;/ cliente NetApp account&gt;/&lt;capacity pool&gt;</code>
Rede virtual	<code>&lt;resource group&gt;/&lt;virtual network&gt;</code>
Sub-rede	<code>&lt;resource group&gt;/&lt;virtual network&gt;/&lt;subnet&gt;</code>

Você pode controlar como cada volume é provisionado por padrão, especificando as seguintes opções em uma seção especial do arquivo de configuração. Veja os exemplos de configuração abaixo.

Parâmetro	Descrição	Padrão
exportRule	As regras de exportação para novos volumes	"0,0.0,0/0"
snapshotDir	Controla a visibilidade do diretório <code>.snapshot</code>	"falso"
size	O tamanho padrão dos novos volumes	"100G"

Parâmetro	Descrição	Padrão
unixPermissions	As permissões unix de novos volumes (4 dígitos octal)	"" (recurso de pré-visualização, requer lista branca na assinatura)

O `exportRule` valor deve ser uma lista separada por vírgulas de qualquer combinação de endereços IPv4 ou sub-redes IPv4 na notação CIDR.



Para todos os volumes criados em um back-end do ANF, o Astra Trident copia todas as etiquetas presentes em um pool de storage para o volume de storage no momento em que ele é provisionado. Os administradores de storage podem definir rótulos por pool de storage e agrupar todos os volumes criados em um pool de storage. Isso fornece uma maneira conveniente de diferenciar volumes com base em um conjunto de rótulos personalizáveis que são fornecidos na configuração de back-end.

## Exemplo 1: Configuração mínima

Esta é a configuração mínima absoluta de back-end. Com essa configuração, o Astra Trident descobre todas as suas contas NetApp, pools de capacidade e sub-redes delegadas no ANF no local configurado e coloca novos volumes aleatoriamente em um desses pools e sub-redes.

Essa configuração é ideal quando você está apenas começando o ANF e experimentando as coisas, mas na prática você vai querer fornecer um escopo adicional para os volumes provisionados.

```
{
  "version": 1,
  "storageDriverName": "azure-netapp-files",
  "subscriptionID": "9f87c765-4774-fake-ae98-a721add45451",
  "tenantID": "68e4f836-edc1-fake-bff9-b2d865ee56cf",
  "clientID": "dd043f63-bf8e-fake-8076-8de91e5713aa",
  "clientSecret": "SECRET",
  "location": "eastus"
}
```

## Exemplo 2: Configuração específica de nível de serviço com filtros de pool de capacidade

Essa configuração de back-end coloca volumes no local do Azure `eastus` em um `Ultra` pool de capacidade. O Astra Trident descobre automaticamente todas as sub-redes delegadas no ANF nesse local e coloca um novo volume em uma delas aleatoriamente.

```
{
  "version": 1,
  "storageDriverName": "azure-netapp-files",
  "subscriptionID": "9f87c765-4774-fake-ae98-a721add45451",
  "tenantID": "68e4f836-edc1-fake-bff9-b2d865ee56cf",
  "clientID": "dd043f63-bf8e-fake-8076-8de91e5713aa",
  "clientSecret": "SECRET",
  "location": "eastus",
  "serviceLevel": "Ultra",
  "capacityPools": [
    "application-group-1/account-1/ultra-1",
    "application-group-1/account-1/ultra-2"
  ],
}
```

### Exemplo 3: Configuração avançada

Essa configuração de back-end reduz ainda mais o escopo do posicionamento de volume para uma única sub-rede e também modifica alguns padrões de provisionamento de volume.

```

{
  "version": 1,
  "storageDriverName": "azure-netapp-files",
  "subscriptionID": "9f87c765-4774-fake-ae98-a721add45451",
  "tenantID": "68e4f836-edc1-fake-bff9-b2d865ee56cf",
  "clientID": "dd043f63-bf8e-fake-8076-8de91e5713aa",
  "clientSecret": "SECRET",
  "location": "eastus",
  "serviceLevel": "Ultra",
  "capacityPools": [
    "application-group-1/account-1/ultra-1",
    "application-group-1/account-1/ultra-2"
  ],
  "virtualNetwork": "my-virtual-network",
  "subnet": "my-subnet",
  "nfsMountOptions": "vers=3,proto=tcp,timeo=600",
  "limitVolumeSize": "500Gi",
  "defaults": {
    "exportRule": "10.0.0.0/24,10.0.1.0/24,10.0.2.100",
    "snapshotDir": "true",
    "size": "200Gi",
    "unixPermissions": "0777"
  }
}
=====
}
}

```

#### Exemplo 4: Configuração do pool de armazenamento virtual

Essa configuração de back-end define vários pools de storage em um único arquivo. Isso é útil quando você tem vários pools de capacidade com suporte a diferentes níveis de serviço e deseja criar classes de storage no Kubernetes que os representem.



```

{
  "version": 1,
  "storageDriverName": "azure-netapp-files",
  "subscriptionID": "9f87c765-4774-fake-ae98-a721add45451",
  "tenantID": "68e4f836-edc1-fake-bff9-b2d865ee56cf",
  "clientID": "dd043f63-bf8e-fake-8076-8de91e5713aa",
  "clientSecret": "SECRET",
  "location": "eastus",
  "resourceGroups": ["application-group-1"],
  "nfsMountOptions": "vers=3,proto=tcp,timeo=600",
  "labels": {
    "cloud": "azure"
  },
  "location": "eastus",

  "storage": [
    {
      "labels": {
        "performance": "gold"
      },
      "serviceLevel": "Ultra",
      "capacityPools": ["ultra-1", "ultra-2"]
    },
    {
      "labels": {
        "performance": "silver"
      },
      "serviceLevel": "Premium",
      "capacityPools": ["premium-1"]
    },
    {
      "labels": {
        "performance": "bronze"
      },
      "serviceLevel": "Standard",
      "capacityPools": ["standard-1", "standard-2"]
    }
  ]
}

```

As definições a seguir StorageClass referem-se aos pools de armazenamento acima. Ao usar o `parameters.selector` campo, você pode especificar para cada StorageClass um o pool virtual que é usado para hospedar um volume. O volume terá os aspetos definidos no pool escolhido.

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: gold
provisioner: csi.trident.netapp.io
parameters:
  selector: "performance=gold"
allowVolumeExpansion: true
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: silver
provisioner: csi.trident.netapp.io
parameters:
  selector: "performance=silver"
allowVolumeExpansion: true
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: bronze
provisioner: csi.trident.netapp.io
parameters:
  selector: "performance=bronze"
allowVolumeExpansion: true
```

## O que se segue?

Depois de criar o arquivo de configuração de back-end, execute o seguinte comando:

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

Se a criação do backend falhar, algo está errado com a configuração do backend. Você pode exibir os logs para determinar a causa executando o seguinte comando:

```
tridentctl logs
```

Depois de identificar e corrigir o problema com o arquivo de configuração, você pode executar o comando `create` novamente.

# Configurar um back-end do CVS para GCP

Saiba como configurar o NetApp Cloud Volumes Service (CVS) para o Google Cloud Platform (GCP) como back-end para a instalação do Astra Trident usando as configurações de exemplo fornecidas.



O NetApp Cloud Volumes Service não é compatível com volumes CVS-performance com tamanho inferior a 100 GiB ou volumes CVS com tamanho inferior a 300 GiB. O Astra Trident cria automaticamente volumes do tamanho mínimo se a o volume solicitado for menor que o tamanho mínimo.

## O que você vai precisar

Para configurar e usar o "[Cloud Volumes Service para Google Cloud](#)" back-end, você precisa do seguinte:

- Uma conta do Google Cloud configurada com o NetApp CVS
- Número do projeto da sua conta do Google Cloud
- Conta de serviço do Google Cloud com a `netappcloudvolumes.admin` função
- Arquivo de chave de API para sua conta de serviço CVS

O Astra Trident agora inclui suporte a volumes menores com o padrão "[Tipo de serviço CVS no GCP](#)". Para backends criados com `storageClass=software`, os volumes agora terão um tamanho mínimo de provisionamento de 300 GiB. O CVS atualmente fornece esse recurso sob disponibilidade controlada e não fornece suporte técnico. Os usuários devem se inscrever para acesso a volumes menores de 1TiB "[aqui](#)" TB. A NetApp recomenda que os clientes consumam volumes inferiores a 1TiB TB para **cargas de trabalho que não sejam de produção**.



Ao implantar backends usando o tipo de serviço CVS padrão (`storageClass=software`), os usuários devem obter acesso ao recurso volumes sub-1TiB no GCP para o(s) número(s) de Projeto e ID(s) de Projeto em questão. Isso é necessário para que o Astra Trident provisione volumes inferiores a 1TiB TB. Caso contrário, as criações de volume falharão para PVCs menores que 600 GiB. Obter acesso a volumes inferiores a 1TiB com "[este formulário](#)"o .

Os volumes criados pelo Astra Trident para o nível de serviço CVS padrão serão provisionados da seguinte forma:

- PVCs menores que 300 GiB resultarão em Astra Trident criando um volume CVS de 300 GiB.
- Os PVCs que estão entre 300 GiB e 600 GiB resultarão na criação do Astra Trident de um volume CVS do tamanho solicitado.
- Os PVCs que estão entre 600 GiB e 1 TIB resultarão na criação de um volume CVS de 1TiB TB do Astra Trident.
- PVCs maiores que 1 TIB resultarão na criação do Astra Trident de um volume CVS do tamanho solicitado.

## Opções de configuração de back-end

Consulte a tabela a seguir para obter as opções de configuração de back-end:

Parâmetro	Descrição	Padrão
<code>version</code>		Sempre 1

<b>Parâmetro</b>	<b>Descrição</b>	<b>Padrão</b>
<code>storageDriverName</code>	Nome do controlador de armazenamento	"gcp-cvs"
<code>backendName</code>	Nome personalizado ou back-end de storage	Nome do driver e parte da chave da API
<code>storageClass</code>	Tipo de armazenamento. Escolha entre <code>hardware</code> (otimizado para performance) ou <code>software</code> (tipo de serviço CVS)	
<code>projectNumber</code>	Número do projeto da conta Google Cloud. O valor é encontrado na página inicial do portal do Google Cloud.	
<code>apiRegion</code>	Região da conta CVS. É a região onde o backend provisionará os volumes.	
<code>apiKey</code>	Chave de API para a conta de serviço do Google Cloud com a <code>netappcloudvolumes.admin</code> função. Ele inclui o conteúdo formatado em JSON do arquivo de chave privada de uma conta de serviço do Google Cloud (copiado literalmente no arquivo de configuração de back-end).	
<code>proxyURL</code>	URL do proxy se o servidor proxy for necessário para se conectar à conta CVS. O servidor proxy pode ser um proxy HTTP ou um proxy HTTPS. Para um proxy HTTPS, a validação do certificado é ignorada para permitir o uso de certificados autoassinados no servidor proxy. Os servidores proxy com autenticação ativada não são suportados.	
<code>nfsMountOptions</code>	Controle refinado das opções de montagem NFS.	"3"
<code>limitVolumeSize</code>	Falha no provisionamento se o tamanho do volume solicitado estiver acima desse valor	"" (não aplicado por padrão)
<code>serviceLevel</code>	O nível de serviço CVS para novos volumes. Os valores são "padrão", "premium" e "extremo".	"standard" (padrão)
<code>network</code>	Rede GCP usada para volumes CVS	"padrão"

Parâmetro	Descrição	Padrão
debugTraceFlags	Debug flags para usar ao solucionar problemas. Exemplo, <code>\{"api":false, "method":true\}</code> . Não use isso a menos que você esteja solucionando problemas e exija um despejo de log detalhado.	nulo

Se estiver usando uma rede VPC compartilhada, ambos `projectNumber` e `hostProjectNumber` devem ser especificados. Nesse caso, `projectNumber` é o projeto de serviço, e `hostProjectNumber` é o projeto host.

O `apiRegion` representa a região do GCP em que o Astra Trident cria volumes CVS. Ao criar clusters de Kubernetes entre regiões, os volumes CVS criados em um `apiRegion` podem ser usados em workloads programados em nós em várias regiões do GCP. Esteja ciente de que o tráfego entre regiões incorre em um custo adicional.

- Para habilitar o acesso entre regiões, a definição do `StorageClass` para `allowedTopologies` deve incluir todas as regiões. Por exemplo:

```
- key: topology.kubernetes.io/region
  values:
  - us-east1
  - europe-west1
```



- `storageClass` é um parâmetro opcional que você pode usar para selecionar o desejado "[Tipo de serviço CVS](#)". Você pode escolher entre o tipo de serviço CVS básico (`storageClass=software`) ou o tipo de serviço CVS-Performance (`storageClass=hardware`), que o Trident usa por padrão. Certifique-se de especificar um `apiRegion` que forneça o CVS respectivo `storageClass` na definição de back-end.



A integração do Astra Trident com o tipo de serviço CVS básico no Google Cloud é um recurso **beta**, não destinado a cargas de trabalho de produção. O Trident é **totalmente suportado** com o tipo de serviço CVS-Performance e o usa por padrão.

Cada back-end provisiona volumes em uma única região do Google Cloud. Para criar volumes em outras regiões, você pode definir backends adicionais.

Você pode controlar como cada volume é provisionado por padrão, especificando as seguintes opções em uma seção especial do arquivo de configuração. Veja os exemplos de configuração abaixo.

Parâmetro	Descrição	Padrão
exportRule	As regras de exportação para novos volumes	"0,0.0,0/0"
snapshotDir	Acesso ao <code>.snapshot</code> diretório	"falso"

Parâmetro	Descrição	Padrão
snapshotReserve	Porcentagem de volume reservado para snapshots	"" (aceitar o padrão CVS de 0)
size	O tamanho dos novos volumes	"100Gi"

O `exportRule` valor deve ser uma lista separada por vírgulas de qualquer combinação de endereços IPv4 ou sub-redes IPv4 na notação CIDR.



Para todos os volumes criados em um back-end do Google Cloud do CVS, o Trident copia todas as etiquetas presentes em um pool de storage para o volume de storage no momento em que ele é provisionado. Os administradores de storage podem definir rótulos por pool de storage e agrupar todos os volumes criados em um pool de storage. Isso fornece uma maneira conveniente de diferenciar volumes com base em um conjunto de rótulos personalizáveis que são fornecidos na configuração de back-end.

## Exemplo 1: Configuração mínima

Esta é a configuração mínima absoluta de back-end.

```
{
  "version": 1,
  "storageDriverName": "gcp-cvs",
  "projectNumber": "012345678901",
  "apiRegion": "us-west2",
  "apiKey": {
    "type": "service_account",
    "project_id": "my-gcp-project",
    "private_key_id": "12345678901234567890123456789012345678901234567890",
    "private_key": "
    -----BEGIN PRIVATE KEY-----
    <key_value>
    -----END PRIVATE KEY-----\n",
    "client_email": "cloudvolumes-admin-sa@my-gcp-
    project.iam.gserviceaccount.com",
    "client_id": "123456789012345678901",
    "auth_uri": "https://accounts.google.com/o/oauth2/auth",
    "token_uri": "https://oauth2.googleapis.com/token",
    "auth_provider_x509_cert_url":
    "https://www.googleapis.com/oauth2/v1/certs",
    "client_x509_cert_url":
    "https://www.googleapis.com/robot/v1/metadata/x509/cloudvolumes-admin-
    sa%40my-gcp-project.iam.gserviceaccount.com"
  }
}
```

## Exemplo 2: Configuração do tipo de serviço CVS básico

Este exemplo mostra uma definição de back-end que usa o tipo de serviço CVS básico, destinado a cargas de trabalho de uso geral e fornece desempenho leve/moderado, juntamente com alta disponibilidade por zona.

```
{
  "version": 1,
  "storageDriverName": "gcp-cvs",
  "projectNumber": "012345678901",
  "storageClass": "software",
  "apiRegion": "us-east4",
  "apiKey": {
    "type": "service_account",
    "project_id": "my-gcp-project",
    "private_key_id": "<id_value>",
    "private_key": "
    -----BEGIN PRIVATE KEY-----
    <key_value>
    -----END PRIVATE KEY-----\n",
    "client_email": "cloudvolumes-admin-sa@my-gcp-
project.iam.gserviceaccount.com",
    "client_id": "123456789012345678901",
    "auth_uri": "https://accounts.google.com/o/oauth2/auth",
    "token_uri": "https://oauth2.googleapis.com/token",
    "auth_provider_x509_cert_url":
"https://www.googleapis.com/oauth2/v1/certs",
    "client_x509_cert_url":
"https://www.googleapis.com/robot/v1/metadata/x509/cloudvolumes-admin-
sa%40my-gcp-project.iam.gserviceaccount.com"
  }
}
```

## Exemplo 3: Configuração de nível de serviço único

Este exemplo mostra um arquivo de back-end que aplica os mesmos aspectos a todo o storage criado pelo Astra Trident na região Google Cloud US-west2. Este exemplo também mostra o uso do `proxyURL` no arquivo de configuração de back-end.

```

{
  "version": 1,
  "storageDriverName": "gcp-cvs",
  "projectNumber": "012345678901",
  "apiRegion": "us-west2",
  "apiKey": {
    "type": "service_account",
    "project_id": "my-gcp-project",
    "private_key_id": "<id_value>",
    "private_key": "
    -----BEGIN PRIVATE KEY-----
    <key_value>
    -----END PRIVATE KEY-----\n",
    "client_email": "cloudvolumes-admin-sa@my-gcp-
project.iam.gserviceaccount.com",
    "client_id": "123456789012345678901",
    "auth_uri": "https://accounts.google.com/o/oauth2/auth",
    "token_uri": "https://oauth2.googleapis.com/token",
    "auth_provider_x509_cert_url":
"https://www.googleapis.com/oauth2/v1/certs",
    "client_x509_cert_url":
"https://www.googleapis.com/robot/v1/metadata/x509/cloudvolumes-admin-
sa%40my-gcp-project.iam.gserviceaccount.com"
  },
  "proxyURL": "http://proxy-server-hostname/",
  "nfsMountOptions": "vers=3,proto=tcp,timeo=600",
  "limitVolumeSize": "10Ti",
  "serviceLevel": "premium",
  "defaults": {
    "snapshotDir": "true",
    "snapshotReserve": "5",
    "exportRule": "10.0.0.0/24,10.0.1.0/24,10.0.2.100",
    "size": "5Ti"
  }
}

```

## Exemplo 4: Configuração do pool de armazenamento virtual

Este exemplo mostra o arquivo de definição de back-end configurado com pools de armazenamento virtual juntamente com `StorageClasses` isso se referem a eles.

No arquivo de definição de back-end de exemplo mostrado abaixo, padrões específicos são definidos para todos os pools de armazenamento, que definem o `snapshotReserve` em 5% e o `exportRule` para 0,0,0,0/0. Os pools de armazenamento virtual são definidos na `storage` seção. Neste exemplo, cada pool de armazenamento individual define seu próprio `serviceLevel`, e alguns pools substituem os valores padrão.



```

{
  "version": 1,
  "storageDriverName": "gcp-cvs",
  "projectNumber": "012345678901",
  "apiRegion": "us-west2",
  "apiKey": {
    "type": "service_account",
    "project_id": "my-gcp-project",
    "private_key_id": "<id_value>",
    "private_key": "
    -----BEGIN PRIVATE KEY-----
    <key_value>
    -----END PRIVATE KEY-----\n",
    "client_email": "cloudvolumes-admin-sa@my-gcp-
project.iam.gserviceaccount.com",
    "client_id": "123456789012345678901",
    "auth_uri": "https://accounts.google.com/o/oauth2/auth",
    "token_uri": "https://oauth2.googleapis.com/token",
    "auth_provider_x509_cert_url":
"https://www.googleapis.com/oauth2/v1/certs",
    "client_x509_cert_url":
"https://www.googleapis.com/robot/v1/metadata/x509/cloudvolumes-admin-
sa%40my-gcp-project.iam.gserviceaccount.com"
  },
  "nfsMountOptions": "vers=3,proto=tcp,timeo=600",

  "defaults": {
    "snapshotReserve": "5",
    "exportRule": "0.0.0.0/0"
  },

  "labels": {
    "cloud": "gcp"
  },
  "region": "us-west2",

  "storage": [
    {
      "labels": {
        "performance": "extreme",
        "protection": "extra"
      },
      "serviceLevel": "extreme",
      "defaults": {
        "snapshotDir": "true",
        "snapshotReserve": "10",

```

```

        "exportRule": "10.0.0.0/24"
    }
},
{
    "labels": {
        "performance": "extreme",
        "protection": "standard"
    },
    "serviceLevel": "extreme"
},
{
    "labels": {
        "performance": "premium",
        "protection": "extra"
    },
    "serviceLevel": "premium",
    "defaults": {
        "snapshotDir": "true",
        "snapshotReserve": "10"
    }
},
{
    "labels": {
        "performance": "premium",
        "protection": "standard"
    },
    "serviceLevel": "premium"
},
{
    "labels": {
        "performance": "standard"
    },
    "serviceLevel": "standard"
}
]
}

```

As seguintes definições do StorageClass referem-se aos pools de armazenamento acima. Usando o `parameters.selector` campo, você pode especificar para cada StorageClass o pool virtual usado para hospedar um volume. O volume terá os aspectos definidos no pool escolhido.

O primeiro StorageClass (`'cvs-extreme-extra-protection'`) mapeia para o primeiro pool de armazenamento virtual. Esse é o único pool que oferece desempenho extremo com uma reserva de snapshot de 10%. O último StorageClass (`'cvs-extra-protection'`) chama qualquer pool de armazenamento que forneça uma reserva de snapshot de 10%. O Astra Trident decide qual pool de storage virtual está selecionado e garante

que o requisito de reserva de snapshot seja atendido.

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: cvs-extreme-extra-protection
provisioner: netapp.io/trident
parameters:
  selector: "performance=extreme; protection=extra"
allowVolumeExpansion: true
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: cvs-extreme-standard-protection
provisioner: netapp.io/trident
parameters:
  selector: "performance=premium; protection=standard"
allowVolumeExpansion: true
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: cvs-premium-extra-protection
provisioner: netapp.io/trident
parameters:
  selector: "performance=premium; protection=extra"
allowVolumeExpansion: true
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: cvs-premium
provisioner: netapp.io/trident
parameters:
  selector: "performance=premium; protection=standard"
allowVolumeExpansion: true
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: cvs-standard
provisioner: netapp.io/trident
parameters:
  selector: "performance=standard"
allowVolumeExpansion: true
```

```
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: cvs-extra-protection
provisioner: netapp.io/trident
parameters:
  selector: "protection=extra"
allowVolumeExpansion: true
```

## O que se segue?

Depois de criar o arquivo de configuração de back-end, execute o seguinte comando:

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

Se a criação do backend falhar, algo está errado com a configuração do backend. Você pode exibir os logs para determinar a causa executando o seguinte comando:

```
tridentctl logs
```

Depois de identificar e corrigir o problema com o arquivo de configuração, você pode executar o comando `create` novamente.

## Configurar um back-end NetApp HCI ou SolidFire

Saiba mais sobre como criar e usar um back-end Element com sua instalação do Astra Trident.

### O que você vai precisar

- Um sistema de storage compatível que executa o software Element.
- Credenciais para um usuário de administrador ou locatário de cluster do NetApp HCI/SolidFire que possa gerenciar volumes.
- Todos os seus nós de trabalho do Kubernetes devem ter as ferramentas iSCSI apropriadas instaladas. ["informações sobre a preparação do nó de trabalho"](#)Consulte .

### O que você precisa saber

O `solidfire-san` driver de armazenamento suporta ambos os modos de volume: Arquivo e bloco. Para o `Filesystem` volumeMode, o Astra Trident cria um volume e cria um sistema de arquivos. O tipo de sistema de arquivos é especificado pelo StorageClass.

Condutor	Protocolo	Modo de volume	Modos de acesso suportados	Sistemas de arquivos suportados
solidfire-san	ISCSI	Bloco	RWO, ROX, RWX	Sem sistema de ficheiros. Dispositivo de bloco bruto.
solidfire-san	ISCSI	Bloco	RWO, ROX, RWX	Sem sistema de ficheiros. Dispositivo de bloco bruto.
solidfire-san	ISCSI	Sistema de ficheiros	RWO, ROX	xf <sub>s</sub> ext3, , ext4
solidfire-san	ISCSI	Sistema de ficheiros	RWO, ROX	xf <sub>s</sub> ext3, , ext4



O Astra Trident usa o CHAP quando funciona como um supervisor de CSI aprimorado. Se você estiver usando CHAP (que é o padrão para CSI), nenhuma preparação adicional é necessária. Recomenda-se definir explicitamente a `UseCHAP` opção para usar CHAP com Trident não-CSI. Caso contrário, ["aqui"](#) consulte .



Os grupos de acesso a volume só são compatíveis com a estrutura convencional não CSI para Astra Trident. Quando configurado para funcionar no modo CSI, o Astra Trident usa CHAP.

Se nenhuma `AccessGroups` ou `UseCHAP` for definida, uma das seguintes regras será aplicada:

- Se o grupo de acesso padrão `trident` for detetado, os grupos de acesso serão usados.
- Se nenhum grupo de acesso for detetado e a versão do Kubernetes for 1,7 ou posterior, o CHAP será usado.

## Opções de configuração de back-end

Consulte a tabela a seguir para obter as opções de configuração de back-end:

Parâmetro	Descrição	Padrão
<code>version</code>		Sempre 1
<code>storageDriverName</code>	Nome do controlador de armazenamento	Sempre "SolidFire-san"
<code>backendName</code>	Nome personalizado ou back-end de storage	Endereço IP "SolidFire_" e armazenamento (iSCSI)
<code>Endpoint</code>	MVIP para o cluster SolidFire com credenciais de locatário	
<code>SVIP</code>	Porta e endereço IP de armazenamento (iSCSI)	

Parâmetro	Descrição	Padrão
labels	Conjunto de rótulos arbitrários formatados em JSON para aplicar em volumes.	""
TenantName	Nome do locatário a utilizar (criado se não for encontrado)	
InitiatorIFace	Restringir o tráfego iSCSI a uma interface de host específica	"padrão"
UseCHAP	Use CHAP para autenticar iSCSI	verdadeiro
AccessGroups	Lista de IDs de Grupo de Acesso a utilizar	Encontra a ID de um grupo de acesso chamado "Trident"
Types	Especificações de QoS	
limitVolumeSize	Falha no provisionamento se o tamanho do volume solicitado estiver acima desse valor	"" (não aplicado por padrão)
debugTraceFlags	Debug flags para usar ao solucionar problemas. Por exemplo, "api":false, "método":true"	nulo



Não use `debugTraceFlags` a menos que você esteja solucionando problemas e exija um despejo de log detalhado.



Para todos os volumes criados, o Astra Trident copiará todas as etiquetas presentes em um pool de storage para a LUN de storage de backup no momento em que ela for provisionada. Os administradores de storage podem definir rótulos por pool de storage e agrupar todos os volumes criados em um pool de storage. Isso fornece uma maneira conveniente de diferenciar volumes com base em um conjunto de rótulos personalizáveis que são fornecidos na configuração de back-end.

## Exemplo 1: Configuração de back-end para `solidfire-san` driver com três tipos de volume

Este exemplo mostra um arquivo de back-end usando autenticação CHAP e modelagem de três tipos de volume com garantias de QoS específicas. Provavelmente você definiria classes de armazenamento para consumir cada uma delas usando o `IOPS` parâmetro de classe de armazenamento.

```

{
  "version": 1,
  "storageDriverName": "solidfire-san",
  "Endpoint": "https://<user>:<password>@<mvip>/json-rpc/8.0",
  "SVIP": "<svip>:3260",
  "TenantName": "<tenant>",
  "labels": {"k8scluster": "dev1", "backend": "dev1-element-cluster"},
  "UseCHAP": true,
  "Types": [{"Type": "Bronze", "Qos": {"minIOPS": 1000, "maxIOPS": 2000,
"burstIOPS": 4000}},
            {"Type": "Silver", "Qos": {"minIOPS": 4000, "maxIOPS": 6000,
"burstIOPS": 8000}},
            {"Type": "Gold", "Qos": {"minIOPS": 6000, "maxIOPS": 8000,
"burstIOPS": 10000}}]
}

```

## Exemplo 2: Configuração de classe de back-end e armazenamento para solidfire-san driver com pools de armazenamento virtual

Este exemplo mostra o arquivo de definição de back-end configurado com pools de armazenamento virtual junto com o StorageClasses que se referem a eles.

No arquivo de definição de back-end de exemplo mostrado abaixo, padrões específicos são definidos para todos os pools de armazenamento, que definem o `type` em Prata. Os pools de armazenamento virtual são definidos na `storage` seção. Neste exemplo, alguns conjuntos de armazenamento definem seu próprio tipo e alguns conjuntos substituem os valores padrão definidos acima.

```

{
  "version": 1,
  "storageDriverName": "solidfire-san",
  "Endpoint": "https://<user>:<password>@<mvip>/json-rpc/8.0",
  "SVIP": "<svip>:3260",
  "TenantName": "<tenant>",
  "UseCHAP": true,
  "Types": [{"Type": "Bronze", "Qos": {"minIOPS": 1000, "maxIOPS": 2000,
"burstIOPS": 4000}},
            {"Type": "Silver", "Qos": {"minIOPS": 4000, "maxIOPS": 6000,
"burstIOPS": 8000}},
            {"Type": "Gold", "Qos": {"minIOPS": 6000, "maxIOPS": 8000,
"burstIOPS": 10000}}],

  "type": "Silver",
  "labels":{"store":"solidfire", "k8scluster": "dev-1-cluster"},
  "region": "us-east-1",

  "storage": [
    {
      "labels":{"performance":"gold", "cost":"4"},
      "zone":"us-east-1a",
      "type":"Gold"
    },
    {
      "labels":{"performance":"silver", "cost":"3"},
      "zone":"us-east-1b",
      "type":"Silver"
    },
    {
      "labels":{"performance":"bronze", "cost":"2"},
      "zone":"us-east-1c",
      "type":"Bronze"
    },
    {
      "labels":{"performance":"silver", "cost":"1"},
      "zone":"us-east-1d"
    }
  ]
}

```

As seguintes definições do StorageClass referem-se aos pools de armazenamento virtual acima. Usando o `parameters.selector` campo, cada StorageClass chama qual(s) pool(s) virtual(s) pode(m) ser(ão) usado(s) para hospedar um volume. O volume terá os aspetos definidos no pool virtual escolhido.

O primeiro StorageClass (`solidfire-gold-four`) será mapeado para o primeiro pool de



armazenamento virtual. Este é o único pool que oferece desempenho de ouro com um `Volume Type QoS de ouro. O último StorageClass ) (`solidfire-silver`chama qualquer pool de armazenamento que ofereça um desempenho prateado. O Astra Trident decidirá qual pool de storage virtual está selecionado e garantirá que o requisito de storage seja atendido.

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: solidfire-gold-four
provisioner: csi.trident.netapp.io
parameters:
  selector: "performance=gold; cost=4"
  fsType: "ext4"
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: solidfire-silver-three
provisioner: csi.trident.netapp.io
parameters:
  selector: "performance=silver; cost=3"
  fsType: "ext4"
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: solidfire-bronze-two
provisioner: csi.trident.netapp.io
parameters:
  selector: "performance=bronze; cost=2"
  fsType: "ext4"
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: solidfire-silver-one
provisioner: csi.trident.netapp.io
parameters:
  selector: "performance=silver; cost=1"
  fsType: "ext4"
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: solidfire-silver
provisioner: csi.trident.netapp.io
parameters:
  selector: "performance=silver"
  fsType: "ext4"
```

## Encontre mais informações

- ["Grupos de acesso de volume"](#)

# Configure um back-end com drivers SAN ONTAP ou Cloud Volumes ONTAP

Saiba mais sobre como configurar um back-end ONTAP com drivers SAN ONTAP e Cloud Volumes ONTAP.

- ["Preparação"](#)
- ["Configuração e exemplos"](#)

## Permissões do usuário

O Astra Trident espera ser executado como administrador da ONTAP ou SVM, normalmente usando o `admin` usuário do cluster ou um `vsadmin` usuário SVM, ou um usuário com um nome diferente que tenha a mesma função. Para implantações do Amazon FSX for NetApp ONTAP, o Astra Trident espera ser executado como administrador do ONTAP ou SVM, usando o usuário do cluster `fsxadmin` ou um `vsadmin` usuário SVM, ou um usuário com um nome diferente que tenha a mesma função. O `fsxadmin` usuário é um substituto limitado para o usuário administrador do cluster.



Se você usar o `limitAggregateUsage` parâmetro, as permissões de administrador do cluster serão necessárias. Ao usar o Amazon FSX for NetApp ONTAP com Astra Trident, o `limitAggregateUsage` parâmetro não funcionará com as `vsadmin` contas de usuário e `fsxadmin`. A operação de configuração falhará se você especificar este parâmetro.

Embora seja possível criar uma função mais restritiva no ONTAP que um driver Trident pode usar, não recomendamos. A maioria das novas versões do Trident chamarão APIs adicionais que teriam que ser contabilizadas, tornando as atualizações difíceis e suscetíveis a erros.

## Preparação

Saiba mais sobre como se preparar para configurar um back-end ONTAP com drivers SAN ONTAP. Para todos os back-ends ONTAP, o Astra Trident requer pelo menos um agregado atribuído ao SVM.

Lembre-se de que você também pode executar mais de um driver e criar classes de armazenamento que apontam para um ou outro. Por exemplo, você pode configurar uma `san-dev` classe que usa o `ontap-san` driver e uma `san-default` classe que usa a `ontap-san-economy` mesma.

Todos os seus nós de trabalho do Kubernetes precisam ter as ferramentas iSCSI apropriadas instaladas. ["aqui"](#) Consulte para obter mais detalhes.

## Autenticação

O Astra Trident oferece dois modos de autenticação no back-end do ONTAP.

- Baseado em credenciais: O nome de usuário e senha para um usuário do ONTAP com as permissões necessárias. Recomenda-se a utilização de uma função de início de sessão de segurança predefinida, como `admin` ou `vsadmin` para garantir a máxima compatibilidade com as versões do ONTAP.
- Baseado em certificado: O Astra Trident também pode se comunicar com um cluster ONTAP usando um certificado instalado no back-end. Aqui, a definição de back-end deve conter valores codificados em

Base64 do certificado de cliente, chave e certificado de CA confiável, se usado (recomendado).

Os usuários também podem optar por atualizar os backends existentes, optando por mover-se de credenciais para baseadas em certificados e vice-versa. Se **as credenciais e os certificados forem fornecidos**, o Astra Trident usará os certificados por padrão ao emitir um aviso para remover as credenciais da definição de backend.

#### Ative a autenticação baseada em credenciais

O Astra Trident requer as credenciais para um administrador com escopo SVM/cluster para se comunicar com o back-end do ONTAP. Recomenda-se a utilização de funções padrão predefinidas, como `admin` ou `vsadmin`. Isso garante compatibilidade direta com futuras versões do ONTAP que podem expor APIs de recursos a serem usadas por futuras versões do Astra Trident. Uma função de login de segurança personalizada pode ser criada e usada com o Astra Trident, mas não é recomendada.

Uma definição de backend de exemplo será assim:

```
{
  "version": 1,
  "backendName": "ExampleBackend",
  "storageDriverName": "ontap-san",
  "managementLIF": "10.0.0.1",
  "dataLIF": "10.0.0.2",
  "svm": "svm_nfs",
  "username": "vsadmin",
  "password": "secret",
}
```

Tenha em mente que a definição de back-end é o único lugar onde as credenciais são armazenadas em texto simples. Depois que o back-end é criado, os nomes de usuário/senhas são codificados com Base64 e armazenados como segredos do Kubernetes. A criação/updatation de um backend é a única etapa que requer conhecimento das credenciais. Como tal, é uma operação somente de administrador, a ser realizada pelo administrador do Kubernetes/storage.

#### Ativar autenticação baseada em certificado

Backends novos e existentes podem usar um certificado e se comunicar com o back-end do ONTAP. Três parâmetros são necessários na definição de backend.

- `ClientCertificate`: Valor codificado base64 do certificado do cliente.
- `ClientPrivateKey`: Valor codificado em base64 da chave privada associada.
- `TrustedCACertificate`: Valor codificado base64 do certificado CA confiável. Se estiver usando uma CA confiável, esse parâmetro deve ser fornecido. Isso pode ser ignorado se nenhuma CA confiável for usada.

Um fluxo de trabalho típico envolve as etapas a seguir.

#### Passos

1. Gerar um certificado e chave de cliente. Ao gerar, defina Nome Comum (CN) para o usuário ONTAP para autenticar como.

```
openssl req -x509 -nodes -days 1095 -newkey rsa:2048 -keyout k8senv.key
-out k8senv.pem -subj "/C=US/ST=NC/L=RTP/O=NetApp/CN=admin"
```

2. Adicionar certificado de CA confiável ao cluster do ONTAP. Isso pode já ser Tratado pelo administrador do armazenamento. Ignore se nenhuma CA confiável for usada.

```
security certificate install -type server -cert-name <trusted-ca-cert-
name> -vserver <vserver-name>
ssl modify -vserver <vserver-name> -server-enabled true -client-enabled
true -common-name <common-name> -serial <SN-from-trusted-CA-cert> -ca
<cert-authority>
```

3. Instale o certificado e a chave do cliente (a partir do passo 1) no cluster do ONTAP.

```
security certificate install -type client-ca -cert-name <certificate-
name> -vserver <vserver-name>
security ssl modify -vserver <vserver-name> -client-enabled true
```

4. Confirme se a função de login de segurança do ONTAP suporta cert o método de autenticação.

```
security login create -user-or-group-name admin -application ontapi
-authentication-method cert
security login create -user-or-group-name admin -application http
-authentication-method cert
```

5. Teste a autenticação usando certificado gerado. Substitua o ONTAP Management LIF> e o <vserver name> por IP de LIF de gerenciamento e nome da SVM.

```
curl -X POST -Lk https://<ONTAP-Management-
LIF>/servlets/netapp.servlets.admin.XMLrequest_filer --key k8senv.key
--cert ~/k8senv.pem -d '<?xml version="1.0" encoding="UTF-8"?><netapp
xmlns="http://www.netapp.com/filer/admin" version="1.21"
vfiler="<vserver-name>"><vserver-get></vserver-get></netapp>'
```

6. Codificar certificado, chave e certificado CA confiável com Base64.

```
base64 -w 0 k8senv.pem >> cert_base64
base64 -w 0 k8senv.key >> key_base64
base64 -w 0 trustedca.pem >> trustedca_base64
```

7. Crie backend usando os valores obtidos na etapa anterior.

```

$ cat cert-backend.json
{
  "version": 1,
  "storageDriverName": "ontap-san",
  "backendName": "SanBackend",
  "managementLIF": "1.2.3.4",
  "dataLIF": "1.2.3.8",
  "svm": "vserver_test",
  "clientCertificate": "Faaaakkkkeeee...Vaaalllluuueeee",
  "clientPrivateKey": "LS0tFaKE...0VaLuES0tLS0K",
  "trustedCACertificate": "QNFinfO...SiqOyN",
  "storagePrefix": "myPrefix_"
}

$ tridentctl create backend -f cert-backend.json -n trident
+-----+-----+-----+
+-----+-----+
|   NAME   | STORAGE DRIVER |          UUID          |
STATE | VOLUMES |
+-----+-----+-----+
+-----+-----+
| SanBackend | ontap-san      | 586b1cd5-8cf8-428d-a76c-2872713612c1 |
online |         0 |
+-----+-----+-----+
+-----+-----+

```

### Atualizar métodos de autenticação ou girar credenciais

Você pode atualizar um back-end existente para fazer uso de um método de autenticação diferente ou para girar suas credenciais. Isso funciona de ambas as maneiras: Backends que fazem uso de nome de usuário / senha podem ser atualizados para usar certificados; backends que utilizam certificados podem ser atualizados para nome de usuário / senha com base. Para fazer isso, use um arquivo atualizado `backend.json` contendo os parâmetros necessários para executar `tridentctl backend update`o` .`

```

$ cat cert-backend-updated.json
{
"version": 1,
"storageDriverName": "ontap-san",
"backendName": "SanBackend",
"managementLIF": "1.2.3.4",
"dataLIF": "1.2.3.8",
"svm": "vserver_test",
"username": "vsadmin",
"password": "secret",
"storagePrefix": "myPrefix_"
}

#Update backend with tridentctl
$ tridentctl update backend SanBackend -f cert-backend-updated.json -n
trident
+-----+-----+-----+-----+
+-----+-----+
|   NAME   | STORAGE DRIVER |           UUID           |
STATE | VOLUMES |
+-----+-----+-----+-----+
+-----+-----+
| SanBackend | ontap-san      | 586b1cd5-8cf8-428d-a76c-2872713612c1 |
online |          9 |
+-----+-----+-----+-----+
+-----+-----+

```



Ao girar senhas, o administrador de armazenamento deve primeiro atualizar a senha do usuário no ONTAP. Isso é seguido por uma atualização de back-end. Ao girar certificados, vários certificados podem ser adicionados ao usuário. O back-end é então atualizado para usar o novo certificado, seguindo o qual o certificado antigo pode ser excluído do cluster do ONTAP.

A atualização de um back-end não interrompe o acesso a volumes que já foram criados, nem afeta as conexões de volume feitas depois. Uma atualização de back-end bem-sucedida indica que o Astra Trident pode se comunicar com o back-end do ONTAP e lidar com operações de volume futuras.

### Especifique grupos

O Astra Trident usa os grupos para controlar o acesso aos volumes (LUNs) provisionados. Os administradores têm duas opções quando se trata de especificar grupos para backends:

- O Astra Trident pode criar e gerenciar automaticamente um grupo por back-end. Se `igroupName` não estiver incluído na definição de back-end, o Astra Trident criará um grupo nomeado `trident-<backend-UUID>` no SVM. Isso garantirá que cada back-end tenha um iggroup dedicado e tratará da adição/exclusão automatizada de IQNs do nó Kubernetes.
- Alternativamente, os grupos pré-criados também podem ser fornecidos em uma definição de back-end. Isso pode ser feito usando o `igroupName` parâmetro config. O Astra Trident adicionará/excluirá IQNs de

nós do Kubernetes ao grupo pré-existente.

Para backends que `igroupName` tenham definido, o `igroupName` pode ser excluído com um `tridentctl backend update` para ter os grupos de auto-manipulação Astra Trident. Isso não interromperá o acesso a volumes que já estão anexados a cargas de trabalho. Conexões futuras serão tratadas usando o `igroup` Astra Trident criado.



Dedicar um grupo para cada instância única do Astra Trident é uma prática recomendada que é benéfica para o administrador do Kubernetes, bem como para o administrador de storage. O CSI Trident automatiza a adição e remoção de IQNs de nó de cluster ao `igroup`, simplificando muito seu gerenciamento. Ao usar o mesmo SVM em ambientes Kubernetes (e instalações Astra Trident), o uso de um grupo dedicado garante que as alterações feitas em um cluster do Kubernetes não influenciem os grupos associados a outro. Além disso, também é importante garantir que cada nó no cluster do Kubernetes tenha uma IQN exclusiva. Como mencionado acima, o Astra Trident lida automaticamente com a adição e remoção de IQNs. A reutilização de IQNs entre hosts pode levar a cenários indesejáveis nos quais os hosts se confundem uns com os outros e o acesso a LUNs é negado.

Se o Astra Trident estiver configurado para funcionar como um supervisor do CSI, os IQNs do nó do Kubernetes serão automaticamente adicionados/removidos do grupo. Quando os nós são adicionados a um cluster Kubernetes, `trident-csi` o DaemonSet implanta um pod (`trident-csi-xxxxx`) nos nós recém-adicionados e registra os novos nós aos quais pode anexar volumes. Os IQNs de nó também são adicionados ao `igroup` do back-end. Um conjunto semelhante de etapas manipula a remoção de IQNs quando os nós são cordonados, drenados e excluídos do Kubernetes.

Se o Astra Trident não for executado como um supervisor de CSI, o grupo deve ser atualizado manualmente para conter os IQNs iSCSI de cada nó de trabalho no cluster do Kubernetes. As IQNs de nós que ingressam no cluster do Kubernetes precisarão ser adicionadas ao grupo. Da mesma forma, as IQNs de nós removidos do cluster do Kubernetes devem ser removidas do grupo.

### Autentique conexões com CHAP bidirecional

O Astra Trident pode autenticar sessões iSCSI com CHAP bidirecional para os `ontap-san drivers` e `ontap-san-economy`. Isso requer a ativação da `useCHAP` opção na definição de backend. Quando definido como `true`, o Astra Trident configura a segurança do iniciador padrão do SVM para CHAP bidirecional e define o nome de usuário e os segredos do arquivo de back-end. O NetApp recomenda o uso de CHAP bidirecional para autenticar conexões. Veja a seguinte configuração de exemplo:



```

{
  "version": 1,
  "storageDriverName": "ontap-san",
  "backendName": "ontap_san_chap",
  "managementLIF": "192.168.0.135",
  "svm": "ontap_iscsi_svm",
  "useCHAP": true,
  "username": "vsadmin",
  "password": "FaKePaSsWoRd",
  "igroupName": "trident",
  "chapInitiatorSecret": "cl9qxIm36DKyawxy",
  "chapTargetInitiatorSecret": "rqxigXgkesIpwxyz",
  "chapTargetUsername": "iJF4heBRT0TCwxyz",
  "chapUsername": "uh2aNCLsd6cNwxyz",
}

```



O `useCHAP` parâmetro é uma opção booleana que pode ser configurada apenas uma vez. Ele é definido como `false` por padrão. Depois de configurá-lo como verdadeiro, você não pode configurá-lo como falso.

Além `useCHAP=true` do, os `chapInitiatorSecret` campos, `chapTargetInitiatorSecret`, `chapTargetUsername`, e `chapUsername` devem ser incluídos na definição de back-end. Os segredos podem ser alterados depois que um backend é criado executando `tridentctl update`.

### Como funciona

Ao definir `useCHAP` como verdadeiro, o administrador de storage instrui o Astra Trident a configurar o CHAP no back-end de storage. Isso inclui o seguinte:

- Configuração do CHAP no SVM:
  - Se o tipo de segurança do iniciador padrão da SVM for nenhum (definido por padrão) e não houver LUNs pré-existentes no volume, o Astra Trident definirá o tipo de segurança padrão CHAP e continuará configurando o iniciador CHAP e o nome de usuário e os segredos de destino.
  - Se o SVM contiver LUNs, o Astra Trident não ativará o CHAP no SVM. Isso garante que o acesso a LUNs que já estão presentes no SVM não seja restrito.
- Configurando o iniciador CHAP e o nome de usuário e os segredos de destino; essas opções devem ser especificadas na configuração de back-end (como mostrado acima).
- Gerenciando a adição de iniciadores ao `igroupName` dado no back-end. Se não for especificado, o padrão é `trident`.

Depois que o back-end é criado, o Astra Trident cria um CRD correspondente `tridentbackend` e armazena os segredos e nomes de usuário do CHAP como segredos do Kubernetes. Todos os PVS criados pelo Astra Trident neste back-end serão montados e anexados através do CHAP.

### Gire credenciais e atualize os backends

Você pode atualizar as credenciais CHAP atualizando os parâmetros CHAP no `backend.json` arquivo. Isso

exigirá a atualização dos segredos CHAP e o uso do `tridentctl update` comando para refletir essas alterações.



Ao atualizar os segredos CHAP para um backend, você deve usar `tridentctl` para atualizar o backend. Não atualize as credenciais no cluster de storage por meio da IU da CLI/ONTAP, pois o Astra Trident não conseguirá aceitar essas alterações.

```
$ cat backend-san.json
{
  "version": 1,
  "storageDriverName": "ontap-san",
  "backendName": "ontap_san_chap",
  "managementLIF": "192.168.0.135",
  "svm": "ontap_iscsi_svm",
  "useCHAP": true,
  "username": "vsadmin",
  "password": "FaKePaSsWoRd",
  "igroupName": "trident",
  "chapInitiatorSecret": "cl9qxUpDaTeD",
  "chapTargetInitiatorSecret": "rqxigXgkeUpDaTeD",
  "chapTargetUsername": "iJF4heBRT0TCwxyz",
  "chapUsername": "uh2aNCLSD6cNwxyz",
}

$ ./tridentctl update backend ontap_san_chap -f backend-san.json -n
trident
+-----+-----+-----+
+-----+-----+
|  NAME          | STORAGE DRIVER |          UUID                               |
STATE  | VOLUMES  |
+-----+-----+-----+
+-----+-----+
| ontap_san_chap | ontap-san      | aa458f3b-ad2d-4378-8a33-1a472ffbeb5c |
online |      7  |
+-----+-----+-----+
+-----+-----+

```

As conexões existentes não serão afetadas. Elas continuarão ativas se as credenciais forem atualizadas pelo Astra Trident no SVM. As novas conexões usarão as credenciais atualizadas e as conexões existentes continuam ativas. Desconectar e reconectar PVS antigos resultará em eles usando as credenciais atualizadas.

## Opções de configuração e exemplos

Saiba mais sobre como criar e usar drivers SAN ONTAP com sua instalação do Astra Trident. Esta seção fornece exemplos de configuração de back-end e detalhes sobre como mapear backends para `StorageClasses`.

## Opções de configuração de back-end

Consulte a tabela a seguir para obter as opções de configuração de back-end:

Parâmetro	Descrição	Padrão
version		Sempre 1
storageDriverName	Nome do controlador de armazenamento	"ONTAP-nas", "ONTAP-nas-economy", "ONTAP-nas-FlexGroup", "ONTAP-san", "ONTAP-san-economy"
backendName	Nome personalizado ou back-end de storage	Nome do driver
managementLIF	Endereço IP de um cluster ou LIF de gerenciamento de SVM	"10,0,0,1", "[2001:1234:abcd::fefe]"
dataLIF	Endereço IP do protocolo LIF. Use suportes quadrados para IPv6. Não pode ser atualizado depois de configurá-lo	Derivado do SVM, a menos que especificado
useCHAP	Usar CHAP para autenticar iSCSI para drivers SAN ONTAP [Boolean]	falso
chapInitiatorSecret	Segredo do iniciador CHAP. Necessário se useCHAP=true	""
labels	Conjunto de rótulos arbitrários formatados em JSON para aplicar em volumes	""
chapTargetInitiatorSecret	Segredo do iniciador de destino CHAP. Necessário se useCHAP=true	""
chapUsername	Nome de utilizador de entrada. Necessário se useCHAP=true	""
chapTargetUsername	Nome de utilizador alvo. Necessário se useCHAP=true	""
clientCertificate	Valor codificado em base64 do certificado do cliente. Usado para autenticação baseada em certificado	""
clientPrivateKey	Valor codificado em base64 da chave privada do cliente. Usado para autenticação baseada em certificado	""
trustedCACertificate	Valor codificado em base64 do certificado CA confiável. Opcional. Usado para autenticação baseada em certificado	""

Parâmetro	Descrição	Padrão
username	Nome de usuário para se conectar ao cluster/SVM. Usado para autenticação baseada em credenciais	""
password	Senha para se conectar ao cluster/SVM. Usado para autenticação baseada em credenciais	""
svm	Máquina virtual de armazenamento para usar	Derivado se uma SVM managementLIF for especificada
igroupName	Nome do grupo para volumes SAN a serem usados	"Trident-<backend-UUID>"
storagePrefix	Prefixo usado ao provisionar novos volumes na SVM. Não pode ser atualizado depois de configurá-lo	"Trident"
limitAggregateUsage	Falha no provisionamento se o uso estiver acima dessa porcentagem. <b>Não se aplica ao Amazon FSX for ONTAP</b>	"" (não aplicado por padrão)
limitVolumeSize	Falha no provisionamento se o tamanho do volume solicitado estiver acima desse valor.	"" (não aplicado por padrão)
lunsPerFlexvol	Máximo de LUNs por FlexVol, tem de estar no intervalo [50, 200]	"100"
debugTraceFlags	Debug flags para usar ao solucionar problemas. Por exemplo, "api":false, "método":true"	nulo
useREST	Parâmetro booleano para usar APIs REST do ONTAP. <b>Pré-visualização técnica</b>	falso



useREST é fornecido como uma **prévia técnica** recomendada para ambientes de teste e não para cargas de trabalho de produção. Quando definido como true, o Astra Trident usará as APIs REST do ONTAP para se comunicar com o back-end. Esse recurso requer o ONTAP 9.9 e posterior. Além disso, a função de login do ONTAP usada deve ter acesso ao ontap aplicativo. Isso é satisfeito com as funções e cluster-admin predefinidas vsadmin.

Para se comunicar com o cluster ONTAP, você deve fornecer os parâmetros de autenticação. Esse pode ser o nome de usuário/senha para um login de segurança ou um certificado instalado.



Se você estiver usando um back-end do Amazon FSX for NetApp ONTAP, não especifique o limitAggregateUsage parâmetro. fsxadmin`As funções e `vsadmin fornecidas pelo Amazon FSX para NetApp ONTAP não contêm as permissões de acesso necessárias para recuperar o uso agregado e limitá-lo por meio do Astra Trident.



Não use `debugTraceFlags` a menos que você esteja solucionando problemas e exija um despejo de log detalhado.

Para os `ontap-san` drivers, o padrão é usar todos os IPs de LIF de dados da SVM e usar `multipath iSCSI`. Especificar um endereço IP para o `dataLIF` para os `ontap-san` drivers obriga-os a desabilitar o `multipath` e usar apenas o endereço especificado.



Ao criar um backend, lembre-se disso `dataLIF` e `storagePrefix` não pode ser modificado após a criação. Para atualizar esses parâmetros, você precisará criar um novo backend.

`igroupName` Pode ser definido como um grupo que já está criado no cluster ONTAP. Se não for especificado, o Astra Trident cria automaticamente um grupo chamado `Trident-<backend-UUID>`. Se estiver fornecendo um nome de grupo predefinido, o NetApp recomenda o uso de um grupo por cluster do Kubernetes, se o SVM for compartilhado entre ambientes. Isso é necessário para que o Astra Trident mantenha automaticamente adições/exclusões ao IQN.

Os backends também podem ter grupos atualizados após a criação:

- O `igroup Name` pode ser atualizado para apontar para um novo `igroup` que é criado e gerenciado no SVM fora do Astra Trident.
- O `igroupName` pode ser omitido. Nesse caso, o Astra Trident criará e gerenciará um grupo `Trident-<backend-UUID>` automaticamente.

Em ambos os casos, os anexos de volume continuarão a ser acessíveis. Futuros anexos de volume usarão o `igroup` atualizado. Esta atualização não interrompe o acesso aos volumes presentes no back-end.

Um nome de domínio totalmente qualificado (FQDN) pode ser especificado para a `managementLIF` opção.

```
`managementLIF` Para todos os drivers ONTAP também pode ser definido como endereços IPv6. Certifique-se de que instala o Trident com o `--use-ipv6` sinalizador. Deve-se ter cuidado para definir `managementLIF` o endereço IPv6 entre parênteses retos.
```



Ao usar endereços IPv6, certifique-se de `managementLIF` que e `dataLIF` (se incluídos na definição do backend) estejam definidos entre colchetes, como `[28e8:d9fb:a825:b7bf:69a8:d02f:9e7b:3555]`. Se `dataLIF` não for fornecido, o Astra Trident irá buscar os LIFs de dados do IPv6 do SVM.

Para habilitar os drivers ONTAP-san para usar o CHAP, defina o `useCHAP` parâmetro como `true` em sua definição de back-end. Em seguida, o Astra Trident configurará e usará CHAP bidirecional como a autenticação padrão para a SVM fornecida no back-end. ["aqui"](#) Consulte para saber como funciona.

Para `ontap-san-economy` o driver, a `limitVolumeSize` opção também restringirá o tamanho máximo dos volumes que gerencia para `qtrees` e LUNs.



O Astra Trident define rótulos de provisionamento no campo "Comentários" de todos os volumes criados usando `ontap-san` o driver. Para cada volume criado, o campo "Comentários" no FlexVol será preenchido com todas as etiquetas presentes no pool de armazenamento em que ele é colocado. Os administradores de armazenamento podem definir rótulos por pool de armazenamento e agrupar todos os volumes criados em um pool de armazenamento. Isso fornece uma maneira conveniente de diferenciar volumes com base em um conjunto de rótulos personalizáveis que são fornecidos na configuração de back-end.

### Opções de configuração de back-end para volumes de provisionamento

Você pode controlar como cada volume é provisionado por padrão usando essas opções em uma seção especial da configuração. Para obter um exemplo, consulte os exemplos de configuração abaixo.

Parâmetro	Descrição	Padrão
<code>spaceAllocation</code>	Alocação de espaço para LUNs	"verdadeiro"
<code>spaceReserve</code>	Modo de reserva de espaço; "nenhum" (fino) ou "volume" (grosso)	"nenhum"
<code>snapshotPolicy</code>	Política de instantâneos a utilizar	"nenhum"
<code>qosPolicy</code>	Grupo de políticas de QoS a atribuir aos volumes criados. Escolha uma das <code>qosPolicy</code> ou <code>adaptiveQosPolicy</code> por pool de armazenamento/backend	""
<code>adaptiveQosPolicy</code>	Grupo de políticas de QoS adaptável a atribuir para volumes criados. Escolha uma das <code>qosPolicy</code> ou <code>adaptiveQosPolicy</code> por pool de armazenamento/backend	""
<code>snapshotReserve</code>	Porcentagem de volume reservado para snapshots "0"	Se <code>snapshotPolicy</code> é "nenhum", então ""
<code>splitOnClone</code>	Divida um clone de seu pai na criação	"falso"
<code>splitOnClone</code>	Divida um clone de seu pai na criação	"falso"
<code>encryption</code>	Ative a criptografia de volume do NetApp	"falso"
<code>securityStyle</code>	Estilo de segurança para novos volumes	"unix"
<code>tieringPolicy</code>	Política de disposição em camadas para usar "nenhuma"	"Somente snapshot" para configuração pré-ONTAP 9.5 SVM-DR



O uso de grupos de política de QoS com o Astra Trident requer o ONTAP 9.8 ou posterior. Recomenda-se usar um grupo de políticas QoS não compartilhado e garantir que o grupo de políticas seja aplicado individualmente a cada componente. Um grupo de política de QoS compartilhado aplicará o limite máximo da taxa de transferência total de todos os workloads.

Aqui está um exemplo com padrões definidos:

```
{
  "version": 1,
  "storageDriverName": "ontap-san",
  "managementLIF": "10.0.0.1",
  "dataLIF": "10.0.0.2",
  "svm": "trident_svm",
  "username": "admin",
  "password": "password",
  "labels": {"k8scluster": "dev2", "backend": "dev2-sanbackend"},
  "storagePrefix": "alternate-trident",
  "igroupName": "custom",
  "debugTraceFlags": {"api":false, "method":true},
  "defaults": {
    "spaceReserve": "volume",
    "qosPolicy": "standard",
    "spaceAllocation": "false",
    "snapshotPolicy": "default",
    "snapshotReserve": "10"
  }
}
```



Para todos os volumes criados com `ontap-san` o driver, o Astra Trident adiciona uma capacidade extra de 10% ao FlexVol para acomodar os metadados do LUN. O LUN será provisionado com o tamanho exato que o usuário solicita no PVC. O Astra Trident adiciona 10% ao FlexVol (mostra como tamanho disponível no ONTAP). Os usuários agora terão a capacidade utilizável que solicitaram. Essa alteração também impede que LUNs fiquem somente leitura, a menos que o espaço disponível seja totalmente utilizado. Isto não se aplica à ONTAP-san-economia.

Para backends que definem `snapshotReserve`, o Astra Trident calcula o tamanho dos volumes da seguinte forma:

```
Total volume size = [(PVC requested size) / (1 - (snapshotReserve
percentage) / 100)] * 1.1
```

O 1,1 é o 10% adicional que o Astra Trident adiciona ao FlexVol para acomodar os metadados do LUN. Para `snapshotReserve` 5%, e o pedido de PVC é de 5GiB, o tamanho total do volume é de 5,79GiB e o tamanho disponível é de 5,5GiB. O `volume show` comando deve mostrar resultados semelhantes a este exemplo:

Vserver	Volume	Aggregate	State	Type	Size	Available	Used%
		_pvc_89f1c156_3801_4de4_9f9d_034d54c395f4	online	RW	10GB	5.00GB	0%
		_pvc_e42ec6fe_3baa_4af6_996d_134adbbb8e6d	online	RW	5.79GB	5.50GB	0%
		_pvc_e8372153_9ad9_474a_951a_08ae15e1c0ba	online	RW	1GB	511.8MB	0%

3 entries were displayed.

Atualmente, o redimensionamento é a única maneira de usar o novo cálculo para um volume existente.

### Exemplos mínimos de configuração

Os exemplos a seguir mostram configurações básicas que deixam a maioria dos parâmetros padrão. Esta é a maneira mais fácil de definir um backend.



Se você estiver usando o Amazon FSX no NetApp ONTAP com Astra Trident, a recomendação é especificar nomes DNS para LIFs em vez de endereços IP.

#### ontap-san driver com autenticação baseada em certificado

Este é um exemplo de configuração de back-end mínimo. `clientCertificate`, `clientPrivateKey` e `trustedCACertificate` (opcional, se estiver usando CA confiável) são preenchidos `backend.json` e recebem os valores codificados em base64 do certificado do cliente, da chave privada e do certificado de CA confiável, respectivamente.

```
{
  "version": 1,
  "storageDriverName": "ontap-san",
  "backendName": "DefaultSANBackend",
  "managementLIF": "10.0.0.1",
  "dataLIF": "10.0.0.3",
  "svm": "svm_iscsi",
  "useCHAP": true,
  "chapInitiatorSecret": "cl9qxIm36DKyawxy",
  "chapTargetInitiatorSecret": "rqxigXgkesIpwxyz",
  "chapTargetUsername": "iJF4heBRT0TCwxyz",
  "chapUsername": "uh2aNCLSD6cNwxyz",
  "igroupName": "trident",
  "clientCertificate": "ZXR0ZXJwYXB...ICMgJ3BhcGVyc2",
  "clientPrivateKey": "vciwKIYAgZG...0cnksIGRlc2NyaX",
  "trustedCACertificate": "zcyBbaG...b3Igb3duIGNsYXNz"
}
```

#### ontap-san Driver com CHAP bidirecional

Este é um exemplo de configuração de back-end mínimo. Essa configuração básica cria um `ontap-san` back-end com `useCHAP` definido como `true`.



```

{
  "version": 1,
  "storageDriverName": "ontap-san",
  "managementLIF": "10.0.0.1",
  "dataLIF": "10.0.0.3",
  "svm": "svm_iscsi",
  "labels": {"k8scluster": "test-cluster-1", "backend": "testcluster1-
sanbackend"},
  "useCHAP": true,
  "chapInitiatorSecret": "cl9qxIm36DKyawxy",
  "chapTargetInitiatorSecret": "rqxigXgkesIpwxyz",
  "chapTargetUsername": "iJF4heBRT0TCwxyz",
  "chapUsername": "uh2aNCLSD6cNwxyz",
  "igroupName": "trident",
  "username": "vsadmin",
  "password": "secret"
}

```

ontap-san-economy **condutor**

```

{
  "version": 1,
  "storageDriverName": "ontap-san-economy",
  "managementLIF": "10.0.0.1",
  "svm": "svm_iscsi_eco",
  "useCHAP": true,
  "chapInitiatorSecret": "cl9qxIm36DKyawxy",
  "chapTargetInitiatorSecret": "rqxigXgkesIpwxyz",
  "chapTargetUsername": "iJF4heBRT0TCwxyz",
  "chapUsername": "uh2aNCLSD6cNwxyz",
  "igroupName": "trident",
  "username": "vsadmin",
  "password": "secret"
}

```

## Exemplos de backends com pools de armazenamento virtual

No arquivo de definição de back-end de exemplo mostrado abaixo, padrões específicos são definidos para todos os pools de armazenamento, como `spaceReserve` em `nenhum`, `spaceAllocation` em `falso` e `encryption` em `falso`. Os pools de armazenamento virtual são definidos na seção `armazenamento`.

Neste exemplo, alguns dos conjuntos de armazenamento definem os seus próprios `spaceReserve`, `spaceAllocation` valores, e `encryption`, e alguns conjuntos substituem os valores predefinidos acima.

```

{
  "version": 1,
  "storageDriverName": "ontap-san",
  "managementLIF": "10.0.0.1",
  "dataLIF": "10.0.0.3",
  "svm": "svm_iscsi",
  "useCHAP": true,
  "chapInitiatorSecret": "cl9qxIm36DKyawxy",
  "chapTargetInitiatorSecret": "rqxigXgkesIpwxyz",
  "chapTargetUsername": "iJF4heBRT0TCwxyz",
  "chapUsername": "uh2aNCLSd6cNwxyz",
  "igroupName": "trident",
  "username": "vsadmin",
  "password": "secret",

  "defaults": {
    "spaceAllocation": "false",
    "encryption": "false",
    "qosPolicy": "standard"
  },
  "labels":{"store": "san_store", "kubernetes-cluster": "prod-cluster-1"},
  "region": "us_east_1",
  "storage": [
    {
      "labels":{"protection":"gold", "creditpoints":"40000"},
      "zone":"us_east_1a",
      "defaults": {
        "spaceAllocation": "true",
        "encryption": "true",
        "adaptiveQosPolicy": "adaptive-extreme"
      }
    },
    {
      "labels":{"protection":"silver", "creditpoints":"20000"},
      "zone":"us_east_1b",
      "defaults": {
        "spaceAllocation": "false",
        "encryption": "true",
        "qosPolicy": "premium"
      }
    },
    {
      "labels":{"protection":"bronze", "creditpoints":"5000"},
      "zone":"us_east_1c",
      "defaults": {

```

```

        "spaceAllocation": "true",
        "encryption": "false"
    }
}
]
}

```

Aqui está um exemplo iSCSI para ontap-san-economy o driver:

```

{
  "version": 1,
  "storageDriverName": "ontap-san-economy",
  "managementLIF": "10.0.0.1",
  "svm": "svm_iscsi_eco",
  "useCHAP": true,
  "chapInitiatorSecret": "cl9qxIm36DKyawxy",
  "chapTargetInitiatorSecret": "rqxigXgkesIpwxyz",
  "chapTargetUsername": "iJF4heBRT0TCwxyz",
  "chapUsername": "uh2aNCLSD6cNwxyz",
  "igroupName": "trident",
  "username": "vsadmin",
  "password": "secret",

  "defaults": {
    "spaceAllocation": "false",
    "encryption": "false"
  },
  "labels": {"store": "san_economy_store"},
  "region": "us_east_1",
  "storage": [
    {
      "labels": {"app": "oracledb", "cost": "30"},
      "zone": "us_east_1a",
      "defaults": {
        "spaceAllocation": "true",
        "encryption": "true"
      }
    },
    {
      "labels": {"app": "postgresdb", "cost": "20"},
      "zone": "us_east_1b",
      "defaults": {
        "spaceAllocation": "false",
        "encryption": "true"
      }
    }
  ]
}

```

```

    },
    {
      "labels":{"app":"mysqldb", "cost":"10"},
      "zone":"us_east_1c",
      "defaults": {
        "spaceAllocation": "true",
        "encryption": "false"
      }
    }
  ]
}

```

## Mapeie os backends para StorageClasses

As seguintes definições do StorageClass referem-se aos pools de armazenamento virtual acima. Usando o `parameters.selector` campo, cada StorageClass chama qual(s) pool(s) virtual(s) pode(m) ser(ão) usado(s) para hospedar um volume. O volume terá os aspetos definidos no pool virtual escolhido.

- O primeiro StorageClass (`protection-gold`) será mapeado para o primeiro e segundo pool de armazenamento virtual `ontap-nas-flexgroup` no back-end e o primeiro pool de armazenamento virtual `ontap-san` no back-end. Estas são as únicas piscinas que oferecem proteção de nível de ouro.
- O segundo StorageClass (`protection-not-gold`) será mapeado para o terceiro, quarto pool de armazenamento virtual no `ontap-nas-flexgroup` back-end e o segundo, terceiro pool de armazenamento virtual `ontap-san` no back-end. Estas são as únicas piscinas que oferecem um nível de proteção diferente do ouro.
- O terceiro StorageClass (`app-mysqldb`) será mapeado para o quarto pool de armazenamento virtual no `ontap-nas` back-end e o terceiro pool de armazenamento virtual `ontap-san-economy` no back-end. Estes são os únicos pools que oferecem configuração de pool de armazenamento para o aplicativo do tipo `mysqldb`.
- O quarto StorageClass (`protection-silver-creditpoints-20k`) será mapeado para o terceiro pool de armazenamento virtual no `ontap-nas-flexgroup` back-end e o segundo pool de armazenamento virtual `ontap-san` no back-end. Estas são as únicas piscinas que oferecem proteção de nível dourado em 20000 pontos de crédito.
- O quinto StorageClass (`creditpoints-5k`) será mapeado para o segundo pool de armazenamento virtual `ontap-nas-economy` no back-end e o terceiro pool de armazenamento virtual `ontap-san` no back-end. Estas são as únicas ofertas de pool em 5000 pontos de crédito.

O Astra Trident decidirá qual pool de storage virtual está selecionado e garantirá que o requisito de storage seja atendido.

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: protection-gold
provisioner: netapp.io/trident
parameters:
  selector: "protection=gold"
  fsType: "ext4"
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: protection-not-gold
provisioner: netapp.io/trident
parameters:
  selector: "protection!=gold"
  fsType: "ext4"
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: app-mysqldb
provisioner: netapp.io/trident
parameters:
  selector: "app=mysqldb"
  fsType: "ext4"
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: protection-silver-creditpoints-20k
provisioner: netapp.io/trident
parameters:
  selector: "protection=silver; creditpoints=20000"
  fsType: "ext4"
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: creditpoints-5k
provisioner: netapp.io/trident
parameters:
  selector: "creditpoints=5000"
  fsType: "ext4"
```

# Configurar um back-end com drivers nas ONTAP

Saiba mais sobre como configurar um back-end ONTAP com drivers nas ONTAP e Cloud Volumes ONTAP.

- ["Preparação"](#)
- ["Configuração e exemplos"](#)

## Permissões do usuário

O Astra Trident espera ser executado como administrador da ONTAP ou SVM, normalmente usando o `admin` usuário do cluster ou um `vsadmin` usuário SVM, ou um usuário com um nome diferente que tenha a mesma função. Para implantações do Amazon FSX for NetApp ONTAP, o Astra Trident espera ser executado como administrador do ONTAP ou SVM, usando o usuário do cluster `fsxadmin` ou um `vsadmin` usuário SVM, ou um usuário com um nome diferente que tenha a mesma função. O `fsxadmin` usuário é um substituto limitado para o usuário administrador do cluster.



Se você usar o `limitAggregateUsage` parâmetro, as permissões de administrador do cluster serão necessárias. Ao usar o Amazon FSX for NetApp ONTAP com Astra Trident, o `limitAggregateUsage` parâmetro não funcionará com as `vsadmin` contas de usuário e `fsxadmin`. A operação de configuração falhará se você especificar este parâmetro.

Embora seja possível criar uma função mais restritiva no ONTAP que um driver Trident pode usar, não recomendamos. A maioria das novas versões do Trident chamarão APIs adicionais que teriam que ser contabilizadas, tornando as atualizações difíceis e suscetíveis a erros.

## Preparação

Saiba mais sobre como se preparar para configurar um back-end ONTAP com drivers NAS ONTAP. Para todos os back-ends ONTAP, o Astra Trident requer pelo menos um agregado atribuído ao SVM.

Para todos os back-ends ONTAP, o Astra Trident requer pelo menos um agregado atribuído ao SVM.

Lembre-se de que você também pode executar mais de um driver e criar classes de armazenamento que apontam para um ou outro. Por exemplo, você pode configurar uma classe Gold que usa o `ontap-nas` driver e uma classe Bronze que usa o `ontap-nas-economy` um.

Todos os seus nós de trabalho do Kubernetes precisam ter as ferramentas NFS apropriadas instaladas. ["aqui"](#) Consulte para obter mais detalhes.

## Autenticação

O Astra Trident oferece dois modos de autenticação no back-end do ONTAP.

- Baseado em credenciais: O nome de usuário e senha para um usuário do ONTAP com as permissões necessárias. Recomenda-se a utilização de uma função de início de sessão de segurança predefinida, como `admin` ou `vsadmin` para garantir a máxima compatibilidade com as versões do ONTAP.
- Baseado em certificado: O Astra Trident também pode se comunicar com um cluster ONTAP usando um certificado instalado no back-end. Aqui, a definição de back-end deve conter valores codificados em Base64 do certificado de cliente, chave e certificado de CA confiável, se usado (recomendado).

Os usuários também podem optar por atualizar os backends existentes, optando por mover-se de credenciais para baseadas em certificados e vice-versa. **Se as credenciais e os certificados forem fornecidos**, o Astra

Trident usará os certificados por padrão ao emitir um aviso para remover as credenciais da definição de back-end.

### Ative a autenticação baseada em credenciais

O Astra Trident requer as credenciais para um administrador com escopo SVM/cluster para se comunicar com o back-end do ONTAP. Recomenda-se a utilização de funções padrão predefinidas, como `admin` ou `vsadmin`. Isso garante compatibilidade direta com futuras versões do ONTAP que podem expor APIs de recursos a serem usadas por futuras versões do Astra Trident. Uma função de login de segurança personalizada pode ser criada e usada com o Astra Trident, mas não é recomendada.

Uma definição de backend de exemplo será assim:

```
{
  "version": 1,
  "backendName": "ExampleBackend",
  "storageDriverName": "ontap-nas",
  "managementLIF": "10.0.0.1",
  "dataLIF": "10.0.0.2",
  "svm": "svm_nfs",
  "username": "vsadmin",
  "password": "secret"
}
```

Tenha em mente que a definição de back-end é o único lugar onde as credenciais são armazenadas em texto simples. Depois que o back-end é criado, os nomes de usuário/senhas são codificados com Base64 e armazenados como segredos do Kubernetes. A criação/updatation de um backend é a única etapa que requer conhecimento das credenciais. Como tal, é uma operação somente de administrador, a ser realizada pelo administrador do Kubernetes/storage.

### Ativar autenticação baseada em certificado

Backends novos e existentes podem usar um certificado e se comunicar com o back-end do ONTAP. Três parâmetros são necessários na definição de backend.

- `ClientCertificate`: Valor codificado base64 do certificado do cliente.
- `ClientPrivateKey`: Valor codificado em base64 da chave privada associada.
- `TrustedCACertificate`: Valor codificado base64 do certificado CA confiável. Se estiver usando uma CA confiável, esse parâmetro deve ser fornecido. Isso pode ser ignorado se nenhuma CA confiável for usada.

Um fluxo de trabalho típico envolve as etapas a seguir.

### Passos

1. Gerar um certificado e chave de cliente. Ao gerar, defina Nome Comum (CN) para o usuário ONTAP para autenticar como.

```
openssl req -x509 -nodes -days 1095 -newkey rsa:2048 -keyout k8senv.key
-out k8senv.pem -subj "/C=US/ST=NC/L=RTP/O=NetApp/CN=vsadmin"
```

- Adicionar certificado de CA confiável ao cluster do ONTAP. Isso pode já ser Tratado pelo administrador do armazenamento. Ignore se nenhuma CA confiável for usada.

```
security certificate install -type server -cert-name <trusted-ca-cert-name> -vserver <vserver-name>
ssl modify -vserver <vserver-name> -server-enabled true -client-enabled true -common-name <common-name> -serial <SN-from-trusted-CA-cert> -ca <cert-authority>
```

- Instale o certificado e a chave do cliente (a partir do passo 1) no cluster do ONTAP.

```
security certificate install -type client-ca -cert-name <certificate-name> -vserver <vserver-name>
security ssl modify -vserver <vserver-name> -client-enabled true
```

- Confirme se a função de login de segurança do ONTAP suporta cert o método de autenticação.

```
security login create -user-or-group-name vsadmin -application ontapi -authentication-method cert -vserver <vserver-name>
security login create -user-or-group-name vsadmin -application http -authentication-method cert -vserver <vserver-name>
```

- Teste a autenticação usando certificado gerado. Substitua o ONTAP Management LIF> e o <vserver name> por IP de LIF de gerenciamento e nome da SVM. Você deve garantir que o LIF tenha sua política de serviço definida como default-data-management.

```
curl -X POST -Lk https://<ONTAP-Management-LIF>/servlets/netapp.servlets.admin.XMLrequest_filer --key k8senv.key --cert ~/k8senv.pem -d '<?xml version="1.0" encoding="UTF-8"?><netapp xmlns="http://www.netapp.com/filer/admin" version="1.21" vfiler="<vserver-name>"><vserver-get></vserver-get></netapp>'
```

- Codificar certificado, chave e certificado CA confiável com Base64.

```
base64 -w 0 k8senv.pem >> cert_base64
base64 -w 0 k8senv.key >> key_base64
base64 -w 0 trustedca.pem >> trustedca_base64
```

- Crie backend usando os valores obtidos na etapa anterior.



```

$ cat cert-backend-updated.json
{
  "version": 1,
  "storageDriverName": "ontap-nas",
  "backendName": "NasBackend",
  "managementLIF": "1.2.3.4",
  "dataLIF": "1.2.3.8",
  "svm": "vserver_test",
  "clientCertificate": "Faaaakkkkeeee...Vaaalllluuueeee",
  "clientPrivateKey": "LS0tFaKE...0VaLuES0tLS0K",
  "storagePrefix": "myPrefix_"
}

#Update backend with tridentctl
$ tridentctl update backend NasBackend -f cert-backend-updated.json -n
trident
+-----+-----+-----+-----+
+-----+-----+
|      NAME      | STORAGE DRIVER |                UUID                |
STATE | VOLUMES |
+-----+-----+-----+-----+
+-----+-----+
| NasBackend | ontap-nas      | 98e19b74-aec7-4a3d-8dcf-128e5033b214 |
online |          9 |
+-----+-----+-----+-----+
+-----+-----+

```

### Atualizar métodos de autenticação ou girar credenciais

Você pode atualizar um back-end existente para fazer uso de um método de autenticação diferente ou para girar suas credenciais. Isso funciona de ambas as maneiras: Backends que fazem uso de nome de usuário / senha podem ser atualizados para usar certificados; backends que utilizam certificados podem ser atualizados para nome de usuário / senha com base. Para fazer isso, use um arquivo atualizado `backend.json` contendo os parâmetros necessários para executar ``tridentctl backend update``o .

```

$ cat cert-backend-updated.json
{
"version": 1,
"storageDriverName": "ontap-nas",
"backendName": "NasBackend",
"managementLIF": "1.2.3.4",
"dataLIF": "1.2.3.8",
"svm": "vserver_test",
"username": "vsadmin",
"password": "secret",
"storagePrefix": "myPrefix_"
}

#Update backend with tridentctl
$ tridentctl update backend NasBackend -f cert-backend-updated.json -n
trident
+-----+-----+-----+-----+
+-----+-----+
|   NAME   | STORAGE DRIVER |           UUID           |
STATE | VOLUMES |
+-----+-----+-----+-----+
+-----+-----+
| NasBackend | ontap-nas      | 98e19b74-aec7-4a3d-8dcf-128e5033b214 |
online |      9 |
+-----+-----+-----+-----+
+-----+-----+

```



Ao girar senhas, o administrador de armazenamento deve primeiro atualizar a senha do usuário no ONTAP. Isso é seguido por uma atualização de back-end. Ao girar certificados, vários certificados podem ser adicionados ao usuário. O back-end é então atualizado para usar o novo certificado, seguindo o qual o certificado antigo pode ser excluído do cluster do ONTAP.

A atualização de um back-end não interrompe o acesso a volumes que já foram criados, nem afeta as conexões de volume feitas depois. Uma atualização de back-end bem-sucedida indica que o Astra Trident pode se comunicar com o back-end do ONTAP e lidar com operações de volume futuras.

### Gerenciar políticas de exportação de NFS

O Astra Trident usa políticas de exportação de NFS para controlar o acesso aos volumes provisionados.

O Astra Trident oferece duas opções ao trabalhar com políticas de exportação:

- O Astra Trident pode gerenciar dinamicamente a própria política de exportação; nesse modo de operação, o administrador de armazenamento especifica uma lista de blocos CIDR que representam endereços IP admissíveis. O Astra Trident adiciona IPs de nós que se enquadram nesses intervalos à política de exportação automaticamente. Como alternativa, quando nenhum CIDR é especificado, qualquer IP unicast de escopo global encontrado nos nós será adicionado à política de exportação.

- Os administradores de storage podem criar uma política de exportação e adicionar regras manualmente. O Astra Trident usa a política de exportação padrão, a menos que um nome de política de exportação diferente seja especificado na configuração.

### Gerencie dinamicamente políticas de exportação

A versão 20,04 do CSI Trident oferece a capacidade de gerenciar dinamicamente políticas de exportação para backends ONTAP. Isso fornece ao administrador de armazenamento a capacidade de especificar um espaço de endereço permitido para IPs de nó de trabalho, em vez de definir regras explícitas manualmente. Ele simplifica muito o gerenciamento de políticas de exportação. As modificações na política de exportação não exigem mais intervenção manual no cluster de storage. Além disso, isso ajuda a restringir o acesso ao cluster de armazenamento somente aos nós de trabalho que têm IPs no intervalo especificado, suportando um gerenciamento automatizado e refinado.



O gerenciamento dinâmico das políticas de exportação está disponível apenas para o CSI Trident. É importante garantir que os nós de trabalho não estejam sendo repartidos.

### Exemplo

Há duas opções de configuração que devem ser usadas. Aqui está um exemplo de definição de back-end:

```
{
  "version": 1,
  "storageDriverName": "ontap-nas",
  "backendName": "ontap_nas_auto_export",
  "managementLIF": "192.168.0.135",
  "svm": "svm1",
  "username": "vsadmin",
  "password": "FaKePaSsWoRd",
  "autoExportCIDRs": ["192.168.0.0/24"],
  "autoExportPolicy": true
}
```



Ao usar esse recurso, você deve garantir que a junção raiz do SVM tenha uma política de exportação pré-ajustada com uma regra de exportação que permita o bloco CIDR do nó (como a política de exportação padrão). Siga sempre as práticas recomendadas pela NetApp para dedicar um SVM ao Astra Trident.

Aqui está uma explicação de como esse recurso funciona usando o exemplo acima:

- `autoExportPolicy` está definido como `true`. Isso indica que o Astra Trident criará uma política de exportação para `svm1` o SVM e tratará da adição e exclusão de regras usando `autoExportCIDRs` blocos de endereço. Por exemplo, um back-end com UUID `403b5326-8482-40dB-96d0-d83fb3f4daec` e `autoExportPolicy` definido como `true` cria uma política de exportação nomeada `trident-403b5326-8482-40db-96d0-d83fb3f4daec` no SVM.
- `autoExportCIDRs` contém uma lista de blocos de endereços. Este campo é opcional e o padrão é `["0.0.0.0/0", "::/0"]`. Se não estiver definido, o Astra Trident adiciona todos os endereços unicast de escopo global encontrados nos nós de trabalho.

Neste exemplo, o 192.168.0.0/24 espaço de endereço é fornecido. Isso indica que os IPs de nós do Kubernetes que se enquadram nesse intervalo de endereços serão adicionados à política de exportação criada pelo Astra Trident. Quando o Astra Trident registra um nó em que ele é executado, ele recupera os endereços IP do nó e os verifica em relação aos blocos de endereço fornecidos no `autoExportCIDRs`. Depois de filtrar os IPs, o Astra Trident cria regras de política de exportação para os IPs de cliente que ele descobre, com uma regra para cada nó que identifica.

Você pode atualizar `autoExportPolicy` e `autoExportCIDRs` para backends depois de criá-los. Você pode anexar novos CIDR para um back-end que é gerenciado automaticamente ou excluir CIDR existentes. Tenha cuidado ao excluir CIDR para garantir que as conexões existentes não sejam descartadas. Você também pode optar por desativar `autoExportPolicy` um back-end e retornar a uma política de exportação criada manualmente. Isso exigirá a configuração do `exportPolicy` parâmetro em sua configuração de backend.

Depois que o Astra Trident criar ou atualizar um back-end, você pode verificar o back-end usando `tridentctl` ou o CRD correspondente `tridentbackend`:

```
$ ./tridentctl get backends ontap_nas_auto_export -n trident -o yaml
items:
- backendUUID: 403b5326-8482-40db-96d0-d83fb3f4daec
  config:
    aggregate: ""
    autoExportCIDRs:
    - 192.168.0.0/24
    autoExportPolicy: true
    backendName: ontap_nas_auto_export
    chapInitiatorSecret: ""
    chapTargetInitiatorSecret: ""
    chapTargetUsername: ""
    chapUsername: ""
    dataLIF: 192.168.0.135
    debug: false
    debugTraceFlags: null
    defaults:
      encryption: "false"
      exportPolicy: <automatic>
      fileType: ext4
```

Conforme os nós são adicionados a um cluster do Kubernetes e registrados na controladora Astra Trident, as políticas de exportação dos back-ends existentes são atualizadas (desde que elas estejam no intervalo de endereços especificado `autoExportCIDRs` no back-end).

Quando um nó é removido, o Astra Trident verifica todos os back-ends on-line para remover a regra de acesso do nó. Ao remover esse IP de nó das políticas de exportação de backends gerenciados, o Astra Trident impede montagens fraudulentas, a menos que esse IP seja reutilizado por um novo nó no cluster.

Para backends existentes anteriormente, a atualização do back-end com `tridentctl update backend` garantirá que o Astra Trident gerencie as políticas de exportação automaticamente. Isso criará uma nova política de exportação nomeada após o UUID do back-end e os volumes presentes no back-end usarão a política de exportação recém-criada quando forem montados novamente.



A exclusão de um back-end com políticas de exportação gerenciadas automaticamente excluirá a política de exportação criada dinamicamente. Se o backend for recriado, ele será tratado como um novo backend e resultará na criação de uma nova política de exportação.

Se o endereço IP de um nó ativo for atualizado, será necessário reiniciar o pod Astra Trident no nó. Em seguida, o Astra Trident atualizará a política de exportação para backends que ele conseguir refletir essa alteração de IP.

## Opções de configuração e exemplos

Saiba mais sobre como criar e usar drivers NAS ONTAP com sua instalação do Astra Trident. Esta seção fornece exemplos de configuração de back-end e detalhes sobre como mapear backends para StorageClasses.

### Opções de configuração de back-end

Consulte a tabela a seguir para obter as opções de configuração de back-end:

Parâmetro	Descrição	Padrão
version		Sempre 1
storageDriverName	Nome do controlador de armazenamento	"ONTAP-nas", "ONTAP-nas-economy", "ONTAP-nas-FlexGroup", "ONTAP-san", "ONTAP-san-economy"
backendName	Nome personalizado ou back-end de storage	Nome do driver
managementLIF	Endereço IP de um cluster ou LIF de gerenciamento de SVM	"10,0,0,1", "[2001:1234:abcd::fefe]"
dataLIF	Endereço IP do protocolo LIF. Use suportes quadrados para IPv6. Não pode ser atualizado depois de configurá-lo	Derivado do SVM, a menos que especificado
autoExportPolicy	Ativar criação e atualização automática de políticas de exportação [Boolean]	falso
autoExportCIDRs	Lista de CIDR para filtrar IPs de nós do Kubernetes em relação ao autoExportPolicy quando o está ativado	["0,0,0,0/0", ":::0"]»
labels	Conjunto de rótulos arbitrários formatados em JSON para aplicar em volumes	""
clientCertificate	Valor codificado em base64 do certificado do cliente. Usado para autenticação baseada em certificado	""

Parâmetro	Descrição	Padrão
clientPrivateKey	Valor codificado em base64 da chave privada do cliente. Usado para autenticação baseada em certificado	""
trustedCACertificate	Valor codificado em base64 do certificado CA confiável. Opcional. Usado para autenticação baseada em certificado	""
username	Nome de usuário para se conectar ao cluster/SVM. Usado para autenticação baseada em credenciais	
password	Senha para se conectar ao cluster/SVM. Usado para autenticação baseada em credenciais	
svm	Máquina virtual de armazenamento para usar	Derivado se uma SVM managementLIF for especificada
igroupName	Nome do grupo para volumes SAN a serem usados	"Trident-<backend-UUID>"
storagePrefix	Prefixo usado ao provisionar novos volumes na SVM. Não pode ser atualizado depois de configurá-lo	"Trident"
limitAggregateUsage	Falha no provisionamento se o uso estiver acima dessa porcentagem. <b>Não se aplica ao Amazon FSX for ONTAP</b>	"" (não aplicado por padrão)
limitVolumeSize	Falha no provisionamento se o tamanho do volume solicitado estiver acima desse valor.	"" (não aplicado por padrão)
lunsPerFlexvol	Máximo de LUNs por FlexVol, tem de estar no intervalo [50, 200]	"100"
debugTraceFlags	Debug flags para usar ao solucionar problemas. Por exemplo, "api":false, "método":true"	nulo
nfsMountOptions	Lista separada por vírgulas de opções de montagem NFS	""
qtreesPerFlexvol	Qtrees máximos por FlexVol, têm de estar no intervalo [50, 300]	"200"
useREST	Parâmetro booleano para usar APIs REST do ONTAP. <b>Pré-visualização técnica</b>	falso



`useREST` é fornecido como uma **prévia técnica** recomendada para ambientes de teste e não para cargas de trabalho de produção. Quando definido como `true`, o Astra Trident usará as APIs REST do ONTAP para se comunicar com o back-end. Esse recurso requer o ONTAP 9.9 e posterior. Além disso, a função de login do ONTAP usada deve ter acesso ao `ontap` aplicativo. Isso é satisfeito com as funções e `cluster-admin` predefinidas `vsadmin`.

Para se comunicar com o cluster ONTAP, você deve fornecer os parâmetros de autenticação. Esse pode ser o nome de usuário/senha para um login de segurança ou um certificado instalado.



Se você estiver usando um back-end do Amazon FSX for NetApp ONTAP, não especifique o `limitAggregateUsage` parâmetro. `fsxadmin``As funções e ``vsadmin` fornecidas pelo Amazon FSX para NetApp ONTAP não contêm as permissões de acesso necessárias para recuperar o uso agregado e limitá-lo por meio do Astra Trident.



Não use `debugTraceFlags` a menos que você esteja solucionando problemas e exija um despejo de log detalhado.



Ao criar um backend, lembre-se de que o `dataLIF` e `storagePrefix` não pode ser modificado após a criação. Para atualizar esses parâmetros, você precisará criar um novo backend.

Um nome de domínio totalmente qualificado (FQDN) pode ser especificado para a `managementLIF` opção. Um FQDN também pode ser especificado para a `dataLIF` opção, caso em que o FQDN será usado para as operações de montagem NFS. Dessa forma, você pode criar um DNS de round-robin para balanceamento de carga em vários LIFs de dados.

```
`managementLIF` Para todos os drivers ONTAP também pode ser definido como endereços IPv6. Certifique-se de instalar o Astra Trident com o `--use-ipv6` sinalizador. Deve-se ter cuidado para definir o `managementLIF` endereço IPv6 entre parênteses retos.
```



Ao usar endereços IPv6, certifique-se de `managementLIF` que e `dataLIF` (se incluídos na definição do backend) estejam definidos entre colchetes, como `[28e8:d9fb:a825:b7bf:69a8:d02f:9e7b:3555]`. Se `dataLIF` não for fornecido, o Astra Trident irá buscar os LIFs de dados do IPv6 do SVM.

Usando as `autoExportPolicy` opções e `autoExportCIDRs`, o CSI Trident pode gerenciar políticas de exportação automaticamente. Isso é compatível com todos os drivers ONTAP-nas-\*

Para o `ontap-nas-economy` driver, a `limitVolumeSize` opção também restringirá o tamanho máximo dos volumes que gerencia para `qtrees` e LUNs, e a `qtreesPerFlexvol` opção permite personalizar o número máximo de `qtrees` por FlexVol.

O `nfsMountOptions` parâmetro pode ser usado para especificar opções de montagem. As opções de montagem para volumes persistentes do Kubernetes normalmente são especificadas em classes de storage, mas se nenhuma opção de montagem for especificada em uma classe de storage, o Astra Trident voltará a usar as opções de montagem especificadas no arquivo de configuração do back-end de storage. Se nenhuma opção de montagem for especificada na classe de storage ou no arquivo de configuração, o Astra Trident não definirá nenhuma opção de montagem em um volume persistente associado.



O Astra Trident define rótulos de provisionamento no campo "Comentários" de todos os volumes criados usando `(ontap-nas)` e `(ontap-nas-flexgroup)`. Com base no driver usado, os comentários são definidos no FlexVol (`ontap-nas`) ou no FlexGroup (`ontap-nas-flexgroup`). O Astra Trident copiará todas as etiquetas presentes em um pool de storage para o volume de storage no momento em que ele for provisionado. Os administradores de storage podem definir rótulos por pool de storage e agrupar todos os volumes criados em um pool de storage. Isso fornece uma maneira conveniente de diferenciar volumes com base em um conjunto de rótulos personalizáveis que são fornecidos na configuração de back-end.

### Opções de configuração de back-end para volumes de provisionamento

Você pode controlar como cada volume é provisionado por padrão usando essas opções em uma seção especial da configuração. Para obter um exemplo, consulte os exemplos de configuração abaixo.

Parâmetro	Descrição	Padrão
<code>spaceAllocation</code>	Alocação de espaço para LUNs	"verdadeiro"
<code>spaceReserve</code>	Modo de reserva de espaço; "nenhum" (fino) ou "volume" (grosso)	"nenhum"
<code>snapshotPolicy</code>	Política de instantâneos a utilizar	"nenhum"
<code>qosPolicy</code>	Grupo de políticas de QoS a atribuir aos volumes criados. Escolha uma das <code>qosPolicy</code> ou <code>adaptiveQosPolicy</code> por pool de armazenamento/backend	""
<code>adaptiveQosPolicy</code>	Grupo de políticas de QoS adaptável a atribuir para volumes criados. Escolha uma das <code>qosPolicy</code> ou <code>adaptiveQosPolicy</code> por pool de armazenamento/backend. Não suportado pela ONTAP-nas-Economy.	""
<code>snapshotReserve</code>	Porcentagem de volume reservado para snapshots "0"	Se <code>snapshotPolicy</code> é "nenhum", então ""
<code>splitOnClone</code>	Divida um clone de seu pai na criação	"falso"
<code>encryption</code>	Ative a criptografia de volume do NetApp	"falso"
<code>securityStyle</code>	Estilo de segurança para novos volumes	"unix"
<code>tieringPolicy</code>	Política de disposição em camadas para usar "nenhuma"	"Somente snapshot" para configuração pré-ONTAP 9.5 SVM-DR
<code>UnixPermissions</code>	Modo para novos volumes	"777"



Parâmetro	Descrição	Padrão
Snapshotdir	Controla a visibilidade <code>.snapshot</code> do diretório	"falso"
Política de exportação	Política de exportação a utilizar	"padrão"
Estilo de segurança	Estilo de segurança para novos volumes	"unix"



O uso de grupos de política de QoS com o Astra Trident requer o ONTAP 9.8 ou posterior. Recomenda-se usar um grupo de políticas QoS não compartilhado e garantir que o grupo de políticas seja aplicado individualmente a cada componente. Um grupo de política de QoS compartilhado aplicará o limite máximo da taxa de transferência total de todos os workloads.

Aqui está um exemplo com padrões definidos:

```
{
  "version": 1,
  "storageDriverName": "ontap-nas",
  "backendName": "customBackendName",
  "managementLIF": "10.0.0.1",
  "dataLIF": "10.0.0.2",
  "labels": {"k8scluster": "dev1", "backend": "dev1-nasbackend"},
  "svm": "trident_svm",
  "username": "cluster-admin",
  "password": "password",
  "limitAggregateUsage": "80%",
  "limitVolumeSize": "50Gi",
  "nfsMountOptions": "nfsvers=4",
  "debugTraceFlags": {"api":false, "method":true},
  "defaults": {
    "spaceReserve": "volume",
    "qosPolicy": "premium",
    "exportPolicy": "myk8scluster",
    "snapshotPolicy": "default",
    "snapshotReserve": "10"
  }
}
```

Para `ontap-nas` e `ontap-nas-flexgroups`, o Astra Trident agora usa um novo cálculo para garantir que o FlexVol seja dimensionado corretamente com a porcentagem de `snapshotServe` e PVC. Quando o usuário solicita um PVC, o Astra Trident cria o FlexVol original com mais espaço usando o novo cálculo. Esse cálculo garante que o usuário receba o espaço gravável que solicitou no PVC, e não menor espaço do que o que solicitou. Antes de v21,07, quando o usuário solicita um PVC (por exemplo, 5GiB), com o `snapshotServe` a 50 por cento, eles recebem apenas 2,5GiBMB de espaço gravável. Isso ocorre porque o que o usuário solicitou é todo o volume e `snapshotReserve` é uma porcentagem disso. Com o Trident 21,07, o que o usuário solicita é o espaço gravável e o Astra Trident define o `snapshotReserve` número como a porcentagem de todo o volume. Isto não se aplica `ontap-nas-economy` ao `.` Veja o exemplo a seguir para ver como isso funciona:

O cálculo é o seguinte:

```
Total volume size = (PVC requested size) / (1 - (snapshotReserve percentage) / 100)
```

Para snapshotServe de 50%, e a solicitação de PVC de 5GiB, o volume total é de 2/5 10GiB e o tamanho disponível é de 5GiB, o que o usuário solicitou na solicitação de PVC. O `volume show` comando deve mostrar resultados semelhantes a este exemplo:

```
vserver Volume Aggregate State Type Size Available Used%
-----
_pvc_89f1c156_3801_4de4_9f9d_034d54c395f4
online RW 10GB 5.00GB 0%
_pvc_e8372153_9ad9_474a_951a_08ae15e1c0ba
online RW 1GB 511.8MB 0%
2 entries were displayed.
```

Os back-ends existentes de instalações anteriores provisionarão volumes conforme explicado acima ao atualizar o Astra Trident. Para volumes que você criou antes da atualização, você deve redimensionar seus volumes para que a alteração seja observada. Por exemplo, um PVC de 2GiB mm com `snapshotReserve=50` anterior resultou em um volume que fornece 1GiB GB de espaço gravável. Redimensionar o volume para 3GiB, por exemplo, fornece ao aplicativo 3GiBMB de espaço gravável em um volume de 6 GiB.

### Exemplos mínimos de configuração

Os exemplos a seguir mostram configurações básicas que deixam a maioria dos parâmetros padrão. Esta é a maneira mais fácil de definir um backend.



Se você estiver usando o Amazon FSX no NetApp ONTAP com Trident, a recomendação é especificar nomes DNS para LIFs em vez de endereços IP.

#### `ontap-nas` driver com autenticação baseada em certificado

Este é um exemplo de configuração de back-end mínimo. `clientCertificate`, `clientPrivateKey` E `trustedCACertificate` (opcional, se estiver usando CA confiável) são preenchidos `backend.json` e recebem os valores codificados em base64 do certificado do cliente, da chave privada e do certificado de CA confiável, respectivamente.

```

{
  "version": 1,
  "backendName": "DefaultNASBackend",
  "storageDriverName": "ontap-nas",
  "managementLIF": "10.0.0.1",
  "dataLIF": "10.0.0.15",
  "svm": "nfs_svm",
  "clientCertificate": "ZXR0ZXJwYXB...ICMgJ3BhcGVyc2",
  "clientPrivateKey": "vcIwKIyAgZG...0cnksIGRlc2NyaX",
  "trustedCACertificate": "zcyBbaG...b3Igb3duIGNsYXNz",
  "storagePrefix": "myPrefix_"
}

```

ontap-nas **driver com política de exportação automática**

Este exemplo mostra como você pode instruir o Astra Trident a usar políticas de exportação dinâmicas para criar e gerenciar a política de exportação automaticamente. Isso funciona da mesma forma para os ontap-nas-economy drivers e ontap-nas-flexgroup.

```

{
  "version": 1,
  "storageDriverName": "ontap-nas",
  "managementLIF": "10.0.0.1",
  "dataLIF": "10.0.0.2",
  "svm": "svm_nfs",
  "labels": {"k8scluster": "test-cluster-east-1a", "backend": "test1-
nasbackend"},
  "autoExportPolicy": true,
  "autoExportCIDRs": ["10.0.0.0/24"],
  "username": "admin",
  "password": "secret",
  "nfsMountOptions": "nfsvers=4",
}

```

ontap-nas-flexgroup **condutor**

```

{
  "version": 1,
  "storageDriverName": "ontap-nas-flexgroup",
  "managementLIF": "10.0.0.1",
  "dataLIF": "10.0.0.2",
  "labels": {"k8scluster": "test-cluster-east-1b", "backend": "test1-ontap-cluster"},
  "svm": "svm_nfs",
  "username": "vsadmin",
  "password": "secret",
}

```

ontap-nas **Motorista com IPv6**

```

{
  "version": 1,
  "storageDriverName": "ontap-nas",
  "backendName": "nas_ipv6_backend",
  "managementLIF": "[5c5d:5edf:8f:7657:bef8:109b:1b41:d491]",
  "labels": {"k8scluster": "test-cluster-east-1a", "backend": "test1-ontap-ipv6"},
  "svm": "nas_ipv6_svm",
  "username": "vsadmin",
  "password": "netapp123"
}

```

ontap-nas-economy **condutor**

```

{
  "version": 1,
  "storageDriverName": "ontap-nas-economy",
  "managementLIF": "10.0.0.1",
  "dataLIF": "10.0.0.2",
  "svm": "svm_nfs",
  "username": "vsadmin",
  "password": "secret"
}

```

## Exemplos de backends com pools de armazenamento virtual

No arquivo de definição de back-end de exemplo mostrado abaixo, padrões específicos são definidos para todos os pools de armazenamento, como `spaceReserve` em `nenhum`, `spaceAllocation` em `falso` e `encryption` em `falso`. Os pools de armazenamento virtual são definidos na seção `armazenamento`.

Neste exemplo, alguns dos conjuntos de armazenamento definem os seus próprios `spaceReserve`, `spaceAllocation` valores, e `encryption`, e alguns conjuntos substituem os valores predefinidos acima.

ontap-nas **condutor**

```
{
  {
    "version": 1,
    "storageDriverName": "ontap-nas",
    "managementLIF": "10.0.0.1",
    "dataLIF": "10.0.0.2",
    "svm": "svm_nfs",
    "username": "admin",
    "password": "secret",
    "nfsMountOptions": "nfsvers=4",

    "defaults": {
      "spaceReserve": "none",
      "encryption": "false",
      "qosPolicy": "standard"
    },
    "labels": {"store": "nas_store", "k8scluster": "prod-cluster-1"},
    "region": "us_east_1",
    "storage": [
      {
        "labels": {"app": "msoffice", "cost": "100"},
        "zone": "us_east_1a",
        "defaults": {
          "spaceReserve": "volume",
          "encryption": "true",
          "unixPermissions": "0755",
          "adaptiveQosPolicy": "adaptive-premium"
        }
      },
      {
        "labels": {"app": "slack", "cost": "75"},
        "zone": "us_east_1b",
        "defaults": {
          "spaceReserve": "none",
          "encryption": "true",
          "unixPermissions": "0755"
        }
      },
      {
        "labels": {"app": "wordpress", "cost": "50"},
        "zone": "us_east_1c",
```

```

    "defaults": {
      "spaceReserve": "none",
      "encryption": "true",
      "unixPermissions": "0775"
    }
  },
  {
    "labels":{"app":"mysqldb", "cost":"25"},
    "zone":"us_east_1d",
    "defaults": {
      "spaceReserve": "volume",
      "encryption": "false",
      "unixPermissions": "0775"
    }
  }
]
}

```

ontap-nas-flexgroup **condutor**

```

{
  "version": 1,
  "storageDriverName": "ontap-nas-flexgroup",
  "managementLIF": "10.0.0.1",
  "dataLIF": "10.0.0.2",
  "svm": "svm_nfs",
  "username": "vsadmin",
  "password": "secret",

  "defaults": {
    "spaceReserve": "none",
    "encryption": "false"
  },
  "labels":{"store":"flexgroup_store", "k8scluster": "prod-cluster-1"},
  "region": "us_east_1",
  "storage": [
    {
      "labels":{"protection":"gold", "creditpoints":"50000"},
      "zone":"us_east_1a",
      "defaults": {
        "spaceReserve": "volume",
        "encryption": "true",
        "unixPermissions": "0755"
      }
    }
  ],
}

```

```

    {
      "labels":{"protection":"gold", "creditpoints":"30000"},
      "zone":"us_east_1b",
      "defaults": {
        "spaceReserve": "none",
        "encryption": "true",
        "unixPermissions": "0755"
      }
    },
    {
      "labels":{"protection":"silver", "creditpoints":"20000"},
      "zone":"us_east_1c",
      "defaults": {
        "spaceReserve": "none",
        "encryption": "true",
        "unixPermissions": "0775"
      }
    },
    {
      "labels":{"protection":"bronze", "creditpoints":"10000"},
      "zone":"us_east_1d",
      "defaults": {
        "spaceReserve": "volume",
        "encryption": "false",
        "unixPermissions": "0775"
      }
    }
  ]
}

```

ontap-nas-economy **condutor**

```

{
  "version": 1,
  "storageDriverName": "ontap-nas-economy",
  "managementLIF": "10.0.0.1",
  "dataLIF": "10.0.0.2",
  "svm": "svm_nfs",
  "username": "vsadmin",
  "password": "secret",

  "defaults": {
    "spaceReserve": "none",
    "encryption": "false"
  },

```

```

"labels":{"store":"nas_economy_store"},
"region": "us_east_1",
"storage": [
  {
    "labels":{"department":"finance", "creditpoints":"6000"},
    "zone":"us_east_1a",
    "defaults": {
      "spaceReserve": "volume",
      "encryption": "true",
      "unixPermissions": "0755"
    }
  },
  {
    "labels":{"department":"legal", "creditpoints":"5000"},
    "zone":"us_east_1b",
    "defaults": {
      "spaceReserve": "none",
      "encryption": "true",
      "unixPermissions": "0755"
    }
  },
  {
    "labels":{"department":"engineering", "creditpoints":"3000"},
    "zone":"us_east_1c",
    "defaults": {
      "spaceReserve": "none",
      "encryption": "true",
      "unixPermissions": "0775"
    }
  },
  {
    "labels":{"department":"humanresource",
"creditpoints":"2000"},
    "zone":"us_east_1d",
    "defaults": {
      "spaceReserve": "volume",
      "encryption": "false",
      "unixPermissions": "0775"
    }
  }
]
}

```

## Mapeie os backends para StorageClasses

As seguintes definições do StorageClass referem-se aos pools de armazenamento virtual acima. Usando o



parameters.selector campo, cada StorageClass chama qual(s) pool(s) virtual(s) pode(m) ser(ão) usado(s) para hospedar um volume. O volume terá os aspetos definidos no pool virtual escolhido.

- O primeiro StorageClass (protection-gold`será mapeado para o primeiro e segundo pool de armazenamento virtual `ontap-nas-flexgroup no back-end e o primeiro pool de armazenamento virtual ontap-san no back-end. Estas são as únicas piscinas que oferecem proteção de nível de ouro.
- O segundo StorageClass (protection-not-gold`será mapeado para o terceiro, quarto pool de armazenamento virtual no `ontap-nas-flexgroup back-end e o segundo, terceiro pool de armazenamento virtual ontap-san no back-end. Estas são as únicas piscinas que oferecem um nível de proteção diferente do ouro.
- O terceiro StorageClass (app-mysqldb`será mapeado para o quarto pool de armazenamento virtual no `ontap-nas back-end e o terceiro pool de armazenamento virtual ontap-san-economy no back-end. Estes são os únicos pools que oferecem configuração de pool de armazenamento para o aplicativo do tipo mysqldb.
- O quarto StorageClass (protection-silver-creditpoints-20k`será mapeado para o terceiro pool de armazenamento virtual no `ontap-nas-flexgroup back-end e o segundo pool de armazenamento virtual ontap-san no back-end. Estas são as únicas piscinas que oferecem proteção de nível dourado em 20000 pontos de crédito.
- O quinto StorageClass (creditpoints-5k`será mapeado para o segundo pool de armazenamento virtual `ontap-nas-economy no back-end e o terceiro pool de armazenamento virtual ontap-san no back-end. Estas são as únicas ofertas de pool em 5000 pontos de crédito.

O Astra Trident decidirá qual pool de storage virtual está selecionado e garantirá que o requisito de storage seja atendido.

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: protection-gold
provisioner: netapp.io/trident
parameters:
  selector: "protection=gold"
  fsType: "ext4"
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: protection-not-gold
provisioner: netapp.io/trident
parameters:
  selector: "protection!=gold"
  fsType: "ext4"
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: app-mysqldb
provisioner: netapp.io/trident
parameters:
  selector: "app=mysqldb"
  fsType: "ext4"
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: protection-silver-creditpoints-20k
provisioner: netapp.io/trident
parameters:
  selector: "protection=silver; creditpoints=20000"
  fsType: "ext4"
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: creditpoints-5k
provisioner: netapp.io/trident
parameters:
  selector: "creditpoints=5000"
  fsType: "ext4"
```

# Use o Astra Trident com o Amazon FSX para NetApp ONTAP

"Amazon FSX para NetApp ONTAP" é um serviço AWS totalmente gerenciado que permite que os clientes iniciem e executem sistemas de arquivos equipados com o sistema operacional de storage ONTAP da NetApp. O Amazon FSX for NetApp ONTAP permite que você aproveite os recursos, o desempenho e os recursos administrativos do NetApp com os quais você já conhece, ao mesmo tempo em que aproveita a simplicidade, a agilidade, a segurança e a escalabilidade do armazenamento de dados na AWS. O FSX suporta muitos dos recursos do sistema de arquivos e APIs de administração do ONTAP.

Um sistema de arquivos é o principal recurso do Amazon FSX, análogo a um cluster do ONTAP no local. Em cada SVM, você pode criar um ou vários volumes, que são contentores de dados que armazenam os arquivos e pastas em seu sistema de arquivos. Com o Amazon FSX for NetApp ONTAP, o Data ONTAP será fornecido como um sistema de arquivos gerenciado na nuvem. O novo tipo de sistema de arquivos é chamado de **NetApp ONTAP**.

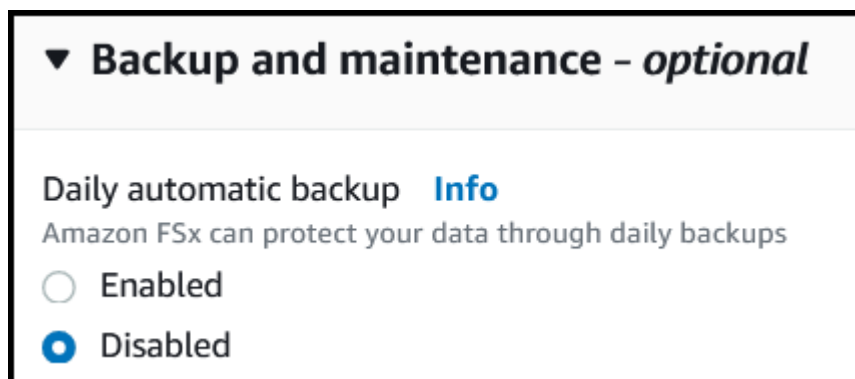
Usando o Astra Trident com o Amazon FSX for NetApp ONTAP, você pode garantir que os clusters do Kubernetes executados no Amazon Elastic Kubernetes Service (EKS) provisionem volumes persistentes de bloco e arquivo com o respaldo do do ONTAP.

## Criando seu sistema de arquivos do Amazon FSX for ONTAP

Os volumes criados nos sistemas de arquivos do Amazon FSX que têm backups automáticos ativados não podem ser excluídos pelo Trident. Para excluir PVCs, você precisa excluir manualmente o PV e o volume FSX for ONTAP.

Para evitar este problema:

- Não use **Quick Create** para criar o sistema de arquivos FSX for ONTAP. O fluxo de trabalho de criação rápida permite backups automáticos e não fornece uma opção de exclusão.
- Ao usar **Standard Create**, desative o backup automático. A desativação de backups automáticos permite que o Trident exclua com êxito um volume sem intervenção manual adicional.



## Saiba mais sobre o Astra Trident

Se você é novo no Astra Trident, familiarize-se usando os links fornecidos abaixo:

- ["FAQs"](#)
- ["Requisitos para uso do Astra Trident"](#)

- ["Implante o Astra Trident"](#)
- ["Práticas recomendadas para configurar o ONTAP, o Cloud Volumes ONTAP e o Amazon FSX for NetApp ONTAP"](#)
- ["Integre o Astra Trident"](#)
- ["Configuração de back-end SAN ONTAP"](#)
- ["Configuração de back-end do ONTAP nas"](#)

Saiba mais sobre os recursos do ["aqui"](#) driver .

O Amazon FSX para NetApp ONTAP usa ["FabricPool"](#) para gerenciar camadas de armazenamento. Ele permite armazenar dados em um nível, com base no acesso frequente aos dados.

O Astra Trident espera ser executado como um `vsadmin` usuário SVM ou como um usuário com um nome diferente que tenha a mesma função. O Amazon FSX for NetApp ONTAP tem um `fsxadmin` usuário que é uma substituição limitada do usuário do cluster do ONTAP `admin`. Não é recomendável usar o `fsxadmin` usuário, com o Trident, pois `vsadmin` o usuário do SVM tem acesso a mais funcionalidades do Astra Trident.

## Drivers

Você pode integrar o Astra Trident ao Amazon FSX for NetApp ONTAP usando os seguintes drivers:

- `ontap-san`: Cada PV provisionado é um LUN dentro de seu próprio volume do Amazon FSX for NetApp ONTAP.
- `ontap-san-economy`: Cada PV provisionado é um LUN com um número configurável de LUNs por volume do Amazon FSX for NetApp ONTAP.
- `ontap-nas`: Cada PV provisionado é um volume completo do Amazon FSX for NetApp ONTAP.
- `ontap-nas-economy`: Cada PV provisionado é uma `qtree`, com um número configurável de `qtrees` por volume do Amazon FSX for NetApp ONTAP.
- `ontap-nas-flexgroup`: Cada PV provisionado é um volume completo do Amazon FSX for NetApp ONTAP FlexGroup.

## Autenticação

O Astra Trident oferece dois modos de autenticação:

- Baseado em certificado: O Astra Trident se comunicará com o SVM em seu sistema de arquivos FSX usando um certificado instalado no seu SVM.
- Baseado em credenciais: Você pode usar o `fsxadmin` usuário para o sistema de arquivos ou o `vsadmin` usuário configurado para o SVM.



Recomendamos vivamente a utilização do `vsadmin` utilizador em vez do `fsxadmin` para configurar o back-end. O Astra Trident se comunicará com o sistema de arquivos FSX usando esse nome de usuário e senha.

Para saber mais sobre autenticação, consulte estes links:

- ["ONTAP nas"](#)
- ["San ONTAP"](#)

# Implante e configure o Astra Trident no EKS com o Amazon FSX for NetApp ONTAP

## O que você vai precisar

- Um cluster do Amazon EKS existente ou um cluster do Kubernetes autogerenciado com `kubectl` o instalado.
- Um sistema de arquivos e uma máquina virtual de armazenamento (SVM) do Amazon FSX for NetApp ONTAP que pode ser acessado a partir dos nós de trabalho do seu cluster.
- Nós de trabalho preparados para "NFS e/ou iSCSI".



Certifique-se de seguir as etapas de preparação de nós necessárias para o Amazon Linux e "Imagens de máquinas da Amazon" Ubuntu (AMIS), dependendo do seu tipo de AMI EKS.

Para outros requisitos do Astra Trident, ["aqui"](#) consulte .

## Passos

1. Implante o Astra Trident usando um dos métodos de implantação../Trident-get-started/kupere-deploy.html.
2. Configure o Astra Trident da seguinte forma:
  - a. Colete o nome DNS de LIF de gerenciamento do SVM. Por exemplo, usando a AWS CLI, localize a `DNSName` entrada em `Endpoints` → `Management` depois de executar o seguinte comando:

```
aws fsx describe-storage-virtual-machines --region <file system region>
```

3. Criar e instalar certificados para autenticação. Se você estiver usando um `ontap-san` backend, ["aqui"](#) consulte . Se você estiver usando um `ontap-nas` backend, ["aqui"](#) consulte .



Você pode fazer login no seu sistema de arquivos (por exemplo, para instalar certificados) usando SSH de qualquer lugar que possa chegar ao seu sistema de arquivos. Utilize o `fsxadmin` utilizador, a palavra-passe configurada quando criou o sistema de ficheiros e o nome DNS de gestão a partir ``aws fsx describe-file-systems`` do .

4. Crie um arquivo de back-end usando seus certificados e o nome DNS do seu LIF de gerenciamento, como mostrado na amostra abaixo:

```
{
  "version": 1,
  "storageDriverName": "ontap-san",
  "backendName": "customBackendName",
  "managementLIF": "svm-XXXXXXXXXXXXXXXXXXXX.fs-XXXXXXXXXXXXXXXXXXXX.fsx.us-east-2.aws.internal",
  "svm": "svm01",
  "clientCertificate": "ZXR0ZXJwYXB...ICMgJ3BhcGVyc2",
  "clientPrivateKey": "vciwKIyAgZG...0cnksIGRlc2NyaX",
  "trustedCACertificate": "zcyBbaG...b3Igb3duIGNsYXNz",
}
```

Para obter informações sobre como criar backends, consulte estes links:

- ["Configurar um back-end com drivers nas ONTAP"](#)
- ["Configure um back-end com drivers SAN ONTAP"](#)



Não especifique `dataLIF` para os `ontap-san drivers` e `ontap-san-economy` para permitir que o Astra Trident use `multipath`.



O `limitAggregateUsage` parâmetro não funcionará com as `vsadmin` contas de utilizador e `fsxadmin`. A operação de configuração falhará se você especificar este parâmetro.

Após a implantação, execute as etapas para criar um ["classe de storage, provisione um volume e monte o volume em um pod"](#).

## Encontre mais informações

- ["Documentação do Amazon FSX para NetApp ONTAP"](#)
- ["Blog post no Amazon FSX for NetApp ONTAP"](#)

## **Informações sobre direitos autorais**

Copyright © 2025 NetApp, Inc. Todos os direitos reservados. Impresso nos EUA. Nenhuma parte deste documento protegida por direitos autorais pode ser reproduzida de qualquer forma ou por qualquer meio — gráfico, eletrônico ou mecânico, incluindo fotocópia, gravação, gravação em fita ou storage em um sistema de recuperação eletrônica — sem permissão prévia, por escrito, do proprietário dos direitos autorais.

O software derivado do material da NetApp protegido por direitos autorais está sujeito à seguinte licença e isenção de responsabilidade:

ESTE SOFTWARE É FORNECIDO PELA NETAPP "NO PRESENTE ESTADO" E SEM QUAISQUER GARANTIAS EXPRESSAS OU IMPLÍCITAS, INCLUINDO, SEM LIMITAÇÕES, GARANTIAS IMPLÍCITAS DE COMERCIALIZAÇÃO E ADEQUAÇÃO A UM DETERMINADO PROPÓSITO, CONFORME A ISENÇÃO DE RESPONSABILIDADE DESTES DOCUMENTOS. EM HIPÓTESE ALGUMA A NETAPP SERÁ RESPONSÁVEL POR QUALQUER DANO DIRETO, INDIRETO, INCIDENTAL, ESPECIAL, EXEMPLAR OU CONSEQUENCIAL (INCLUINDO, SEM LIMITAÇÕES, AQUISIÇÃO DE PRODUTOS OU SERVIÇOS SOBRESSALIENTES; PERDA DE USO, DADOS OU LUCROS; OU INTERRUPTÃO DOS NEGÓCIOS), INDEPENDENTEMENTE DA CAUSA E DO PRINCÍPIO DE RESPONSABILIDADE, SEJA EM CONTRATO, POR RESPONSABILIDADE OBJETIVA OU PREJUÍZO (INCLUINDO NEGLIGÊNCIA OU DE OUTRO MODO), RESULTANTE DO USO DESTES SOFTWARES, MESMO SE ADVERTIDA DA RESPONSABILIDADE DE TAL DANO.

A NetApp reserva-se o direito de alterar quaisquer produtos descritos neste documento, a qualquer momento e sem aviso. A NetApp não assume nenhuma responsabilidade nem obrigação decorrentes do uso dos produtos descritos neste documento, exceto conforme expressamente acordado por escrito pela NetApp. O uso ou a compra deste produto não representam uma licença sob quaisquer direitos de patente, direitos de marca comercial ou quaisquer outros direitos de propriedade intelectual da NetApp.

O produto descrito neste manual pode estar protegido por uma ou mais patentes dos EUA, patentes estrangeiras ou pedidos pendentes.

LEGENDA DE DIREITOS LIMITADOS: o uso, a duplicação ou a divulgação pelo governo estão sujeitos a restrições conforme estabelecido no subparágrafo (b)(3) dos Direitos em Dados Técnicos - Itens Não Comerciais no DFARS 252.227-7013 (fevereiro de 2014) e no FAR 52.227- 19 (dezembro de 2007).

Os dados aqui contidos pertencem a um produto comercial e/ou serviço comercial (conforme definido no FAR 2.101) e são de propriedade da NetApp, Inc. Todos os dados técnicos e software de computador da NetApp fornecidos sob este Contrato são de natureza comercial e desenvolvidos exclusivamente com despesas privadas. O Governo dos EUA tem uma licença mundial limitada, irrevogável, não exclusiva, intransferível e não sublicenciável para usar os Dados que estão relacionados apenas com o suporte e para cumprir os contratos governamentais desse país que determinam o fornecimento de tais Dados. Salvo disposição em contrário no presente documento, não é permitido usar, divulgar, reproduzir, modificar, executar ou exibir os dados sem a aprovação prévia por escrito da NetApp, Inc. Os direitos de licença pertencentes ao governo dos Estados Unidos para o Departamento de Defesa estão limitados aos direitos identificados na cláusula 252.227-7015(b) (fevereiro de 2014) do DFARS.

## **Informações sobre marcas comerciais**

NETAPP, o logotipo NETAPP e as marcas listadas em <http://www.netapp.com/TM> são marcas comerciais da NetApp, Inc. Outros nomes de produtos e empresas podem ser marcas comerciais de seus respectivos proprietários.