



Configure um back-end com drivers SAN ONTAP

Astra Trident

NetApp
December 03, 2024

Índice

- Configure um back-end com drivers SAN ONTAP 1
 - Permissões do usuário 1
 - Prepare-se para configurar o back-end com drivers SAN ONTAP 1
 - Exemplos e opções de configuração de SAN ONTAP 8

Configure um back-end com drivers SAN 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.

Prepare-se para configurar o back-end com drivers SAN ONTAP

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 devem 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).

Você pode atualizar os backends existentes para mover entre métodos baseados em credenciais e baseados

em certificado. No entanto, apenas um método de autenticação é suportado por vez. Para alternar para um método de autenticação diferente, você deve remover o método existente da configuração de back-end.



Se você tentar fornecer **credenciais e certificados**, a criação de back-end falhará com um erro que mais de um método de autenticação foi fornecido no arquivo de configuração.

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/updation 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 usar 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, você deve remover o método de autenticação existente e adicionar o novo método de autenticação. Em seguida, use o arquivo backend.json atualizado contendo os parâmetros necessários para executar `tridentctl backend update`.

```

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 igroup 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.

Exemplos e opções de configuração de SAN ONTAP

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 para switchover MetroCluster otimizado, você precisa especificar um 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	""

Parâmetro	Descrição	Padrão
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. Não se aplica ao Amazon FSX for ONTAP	"" (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. Pré-visualização técnica não suportada com o MetroCluster.	falso

Considerações sobre o `useREST`



- `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 9.10 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`.
- `useREST` Não é suportado com MetroCluster.

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"

Parâmetro	Descrição	Padrão
encryption	Ative a criptografia de volume do NetApp (NVE) no novo volume; o padrão é <code>false</code> . O NVE deve ser licenciado e habilitado no cluster para usar essa opção. Se o NAE estiver ativado no back-end, qualquer volume provisionado no Astra Trident será o NAE ativado. Para obter mais informações, consulte: "Como o Astra Trident funciona com NVE e NAE" .	"falso"
luksEncryption	Ativar encriptação LUKS. "Usar a configuração de chave unificada do Linux (LUKS)" Consulte a .	""
securityStyle	Estilo de segurança para novos volumes	"unix"
tieringPolicy	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:

$$\text{Total volume size} = [(\text{PVC requested size}) / (1 - (\text{snapshotReserve percentage} / 100))] * 1.1$$

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": "c19qxIm36DKyawxy",
  "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"
```

Informações sobre direitos autorais

Copyright © 2024 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 INTERRUPÇÃO 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.