



Konfiguration und Management von Back-Ends

Trident

NetApp
January 14, 2026

Inhalt

Konfiguration und Management von Back-Ends	1
Back-Ends konfigurieren	1
Azure NetApp Dateien	1
Konfigurieren Sie ein Azure NetApp Files-Backend	1
Konfiguration eines Azure NetApp Files-Backends wird vorbereitet	5
Azure NetApp Files Back-End-Konfigurationsoptionen und -Beispiele	8
Google Cloud NetApp Volumes	19
Google Cloud NetApp Volumes-Back-End konfigurieren	19
Bereiten Sie sich auf die Konfiguration eines Google Cloud NetApp Volumes-Back-End vor	22
Konfigurationsoptionen und Beispiele für die Backend-Konfiguration von Google Cloud NetApp Volumes	22
Cloud Volumes Service für Google Cloud-Back-End konfigurieren	35
Treiberdetails zu Google Cloud	35
Erfahren Sie mehr über den Trident Support für Cloud Volumes Service für Google Cloud	35
Back-End-Konfigurationsoptionen	36
Optionen zur Volume-Bereitstellung	37
Beispiele für CVS-Performance-Diensttypen	38
Beispiele für CVS-Diensttypen	43
Was kommt als Nächstes?	45
Konfigurieren Sie ein NetApp HCI- oder SolidFire-Backend	46
Details zum Elementtreiber	46
Bevor Sie beginnen	46
Back-End-Konfigurationsoptionen	46
Beispiel 1: Backend-Konfiguration für <code>solidfire-san</code> Treiber mit drei Volume-Typen	47
Beispiel 2: Back-End- und Storage-Klassenkonfiguration für <code>solidfire-san</code> Treiber mit virtuellen Pools	48
Weitere Informationen	52
ONTAP SAN-Treiber	52
Übersicht über ONTAP SAN-Treiber	52
Vorbereiten der Back-End-Konfiguration mit ONTAP-SAN-Treibern	54
ONTAP-SAN-Konfigurationsoptionen und Beispiele	61
ONTAP NAS-Treiber	78
Übersicht über ONTAP NAS-Treiber	78
Bereiten Sie sich auf die Konfiguration eines Backend mit ONTAP-NAS-Treibern vor	80
ONTAP-NAS-Konfigurationsoptionen und Beispiele	90
Amazon FSX für NetApp ONTAP	109
Verwenden Sie Trident mit Amazon FSX für NetApp ONTAP	109
IAM-Rolle und AWS Secret erstellen	111
Installation Von Trident	115
Konfigurieren Sie das Speicher-Back-End	122
Konfigurieren Sie eine Storage-Klasse und PVC	133
Beispielanwendung bereitstellen	137
Konfigurieren Sie das Trident EKS-Add-on auf einem EKS-Cluster	138

Back-Ends mit kubectrl erstellen	140
TridentBackendConfig	140
Schritte im Überblick	142
Schritt: Ein Kubernetes Secret erstellen	142
Schritt 2: Erstellen Sie den TridentBackendConfig CR	144
Schritt 3: Überprüfen Sie den Status des TridentBackendConfig CR	144
(Optional) Schritt 4: Weitere Informationen	145
Back-Ends managen	147
Führen Sie das Back-End-Management mit kubectrl durch	147
Back-End-Management mit tridentctl	148
Wechseln Sie zwischen den Back-End-Managementoptionen	150

Konfiguration und Management von Back-Ends

Back-Ends konfigurieren

Ein Backend definiert die Beziehung zwischen Trident und einem Storage-System. Er erzählt Trident, wie man mit diesem Storage-System kommuniziert und wie Trident Volumes daraus bereitstellen sollte.

Trident bietet automatisch Back-Ends-Storage-Pools an, die den von einer Storage-Klasse definierten Anforderungen entsprechen. Erfahren Sie, wie Sie das Backend für Ihr Storage-System konfigurieren.

- ["Konfigurieren Sie ein Azure NetApp Files-Backend"](#)
- ["Google Cloud NetApp Volumes-Back-End konfigurieren"](#)
- ["Konfigurieren Sie ein Back-End für Cloud Volumes Service für Google Cloud Platform"](#)
- ["Konfigurieren Sie ein NetApp HCI- oder SolidFire-Backend"](#)
- ["Konfigurieren Sie ein Backend mit ONTAP- oder Cloud Volumes ONTAP-NAS-Treibern"](#)
- ["Konfigurieren Sie ein Backend mit ONTAP- oder Cloud Volumes ONTAP-SAN-Treibern"](#)
- ["Verwenden Sie Trident mit Amazon FSX für NetApp ONTAP"](#)

Azure NetApp Dateien

Konfigurieren Sie ein Azure NetApp Files-Backend

Sie können Azure NetApp Files als Backend für Trident konfigurieren. Sie können NFS- und SMB-Volumes über ein Azure NetApp Files-Back-End einbinden. Trident unterstützt außerdem das Anmeldeinformationsmanagement unter Verwendung von Managed Identities für AKS-Cluster (Azure Kubernetes Services).

Azure NetApp Files-Treiberdetails

Trident stellt die folgenden Azure NetApp Files-Speichertreiber für die Kommunikation mit dem Cluster bereit. Unterstützte Zugriffsmodi sind: *ReadWriteOnce* (RWO), *ReadOnly Many* (ROX), *ReadWriteMany* (RWX), *ReadWriteOncePod* (RWOP).

Treiber	Protokoll	VolumeModus	Unterstützte Zugriffsmodi	Unterstützte Filesysteme
azure-netapp-files	NFS SMB	Dateisystem	RWO, ROX, RWX, RWOP	nfs, smb

Überlegungen

- Der Azure NetApp Files-Service unterstützt keine Volumes, die kleiner als 50 gib sind. Trident erstellt automatisch 50-gib-Volumes, wenn ein kleineres Volume angefordert wird.
- Trident unterstützt nur SMB Volumes, die in Pods gemountet sind, die nur auf Windows Nodes ausgeführt werden.

Verwaltete Identitäten für AKS

Trident unterstützt "[Verwaltete Identitäten](#)" Cluster mit Azure Kubernetes Services. Um die Vorteile einer optimierten Verwaltung von Anmeldeinformationen zu nutzen, die von verwalteten Identitäten angeboten wird, müssen Sie über Folgendes verfügen:

- Implementierung eines Kubernetes Clusters mit AKS
- Verwaltete Identitäten, die auf dem AKS kubernetes-Cluster konfiguriert sind
- Trident installiert, die die zu spezifizieren "Azure" enthält `cloudProvider`.

Betreiber von Trident

Um Trident mit dem Trident-Operator zu installieren, bearbeiten Sie, `tridentorchestrator_cr.yaml` um auf "Azure" einzustellen `cloudProvider`. Beispiel:

```
apiVersion: trident.netapp.io/v1
kind: TridentOrchestrator
metadata:
  name: trident
spec:
  debug: true
  namespace: trident
  imagePullPolicy: IfNotPresent
  cloudProvider: "Azure"
```

Helm

Im folgenden Beispiel werden Trident-Sets mit der Umgebungsvariable auf Azure `$CP` installiert `cloudProvider`:

```
helm install trident trident-operator-100.2410.0.tgz --create
--namespace --namespace <trident-namespace> --set cloudProvider=$CP
```

`<code>-Datei findet </code>`

Das folgende Beispiel installiert Trident und setzt das `cloudProvider` Flag auf Azure:

```
tridentctl install --cloud-provider="Azure" -n trident
```

Cloud-Identität für AKS

Die Cloud-Identität ermöglicht Kubernetes-Pods den Zugriff auf Azure-Ressourcen durch Authentifizierung als Workload-Identität anstatt durch Angabe explizite Azure-Anmeldedaten.

Um die Vorteile der Cloud-Identität in Azure zu nutzen, müssen Sie über folgende Voraussetzungen verfügen:

- Implementierung eines Kubernetes Clusters mit AKS
- Workload-Identität und oidc-Issuer, die auf dem AKS Kubernetes-Cluster konfiguriert sind
- Trident wurde installiert, das die zum Angeben "Azure" und `cloudIdentity` Angeben der Workload-Identität enthält `cloudProvider`

Betreiber von Trident

Um Trident mithilfe des Trident-Operators zu installieren, bearbeiten Sie die `tridentorchestrator_cr.yaml` **Einstellung `cloudProvider` auf und setzen Sie `cloudIdentity` auf `azure.workload.identity/client-id: xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx` sie auf "Azure" .**

Beispiel:

```
apiVersion: trident.netapp.io/v1
kind: TridentOrchestrator
metadata:
  name: trident
spec:
  debug: true
  namespace: trident
  imagePullPolicy: IfNotPresent
  cloudProvider: "Azure"
  *cloudIdentity: 'azure.workload.identity/client-id: xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx' *
```

Helm

Legen Sie die Werte für **Cloud-Provider (CP)** und **Cloud-Identity (CI)** unter Verwendung der folgenden Umgebungsvariablen fest:

```
export CP="Azure"
export CI="'azure.workload.identity/client-id: xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx' "
```

Das folgende Beispiel installiert Trident und setzt `cloudProvider` auf `Azure` unter Verwendung der Umgebungsvariable `$CP` und setzt die `cloudIdentity` unter Verwendung der Umgebungsvariable `$CI`:

```
helm install trident trident-operator-100.2410.0.tgz --set
cloudProvider=$CP --set cloudIdentity="$CI"
```

<code>-Datei findet </code>

Legen Sie die Werte für **Cloud Provider** und **Cloud Identity** unter Verwendung der folgenden Umgebungsvariablen fest:

```
export CP="Azure"
export CI="azure.workload.identity/client-id: xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx"
```

Das folgende Beispiel installiert Trident und setzt das `cloud-provider` Flag auf `$CP`, und `cloud-identity` auf `$CI`:

```
tridentctl install --cloud-provider=$CP --cloud-identity="$CI" -n
trident
```

Konfiguration eines Azure NetApp Files-Backends wird vorbereitet

Bevor Sie Ihr Azure NetApp Files-Backend konfigurieren können, müssen Sie sicherstellen, dass die folgenden Anforderungen erfüllt sind.

Voraussetzungen für NFS und SMB Volumes

Wenn Sie Azure NetApp Files zum ersten Mal oder an einem neuen Standort verwenden, ist eine Erstkonfiguration erforderlich, um Azure NetApp Files einzurichten und ein NFS-Volume zu erstellen. Siehe ["Azure: Azure NetApp Files einrichten und ein NFS Volume erstellen"](#).

Um ein Backend zu konfigurieren und zu verwenden ["Azure NetApp Dateien"](#), benötigen Sie Folgendes:



- `subscriptionID`, `tenantID`, `clientID`, `location` Und `clientSecret` sind optional, wenn verwaltete Identitäten auf einem AKS-Cluster verwendet werden.
- `tenantID`, `clientID` Und `clientSecret` sind optional, wenn eine Cloud-Identität auf einem AKS-Cluster verwendet wird.

- Ein Kapazitäts-Pool. Siehe ["Microsoft: Erstellen Sie einen Kapazitäts-Pool für Azure NetApp Files"](#).
- Ein an Azure NetApp Files delegiertes Subnetz. Siehe ["Microsoft: Delegieren Sie ein Subnetz an Azure NetApp Files"](#).
- `subscriptionID` Von einem Azure-Abonnement mit aktiviertem Azure NetApp Files
- `tenantID`, `clientID` Und `clientSecret` von einem ["App-Registrierung"](#) in Azure Active Directory mit ausreichenden Berechtigungen für den Azure NetApp Files-Dienst. Die App-Registrierung sollte Folgendes verwenden:
 - Der Eigentümer oder die Rolle des Beitragenden ["Vordefiniert von Azure"](#).
 - A ["Benutzerdefinierte Beitragsrolle"](#) auf Abonnementebene (`assignableScopes`) mit den folgenden Berechtigungen, die auf das beschränkt sind, was Trident benötigt. Nach dem Erstellen der benutzerdefinierten Rolle, ["Weisen Sie die Rolle über das Azure-Portal zu"](#).

```
{
  "id": "/subscriptions/<subscription-id>/providers/Microsoft.Authorization/roleDefinitions/<role-definition-id>",
  "properties": {
    "roleName": "custom-role-with-limited-perms",
    "description": "custom role providing limited permissions",
    "assignableScopes": [
      "/subscriptions/<subscription-id>"
    ],
    "permissions": [
      {
        "actions": [

"Microsoft.NetApp/netAppAccounts/capacityPools/read",

"Microsoft.NetApp/netAppAccounts/capacityPools/write",

"Microsoft.NetApp/netAppAccounts/capacityPools/volumes/read",

"Microsoft.NetApp/netAppAccounts/capacityPools/volumes/write",

"Microsoft.NetApp/netAppAccounts/capacityPools/volumes/delete",

"Microsoft.NetApp/netAppAccounts/capacityPools/volumes/snapshots/read",

"Microsoft.NetApp/netAppAccounts/capacityPools/volumes/snapshots/write",

"Microsoft.NetApp/netAppAccounts/capacityPools/volumes/snapshots/delete",

"Microsoft.NetApp/netAppAccounts/capacityPools/volumes/MountTargets/read",

          "Microsoft.Network/virtualNetworks/read",

"Microsoft.Network/virtualNetworks/subnets/read",

"Microsoft.Features/featureProviders/subscriptionFeatureRegistrations/read",

"Microsoft.Features/featureProviders/subscriptionFeatureRegistrat
```

```

ions/write",

"Microsoft.Features/featureProviders/subscriptionFeatureRegistrat
ions/delete",

        "Microsoft.Features/features/read",
        "Microsoft.Features/operations/read",
        "Microsoft.Features/providers/features/read",

"Microsoft.Features/providers/features/register/action",

"Microsoft.Features/providers/features/unregister/action",

"Microsoft.Features/subscriptionFeatureRegistrations/read"
    ],
    "notActions": [],
    "dataActions": [],
    "notDataActions": []
  }
]
}
}

```

- Der Azure location, der mindestens einen enthält ["Delegiertes Subnetz"](#). Ab Trident 22.01 ist der location Parameter ein Pflichtfeld auf der obersten Ebene der Backend-Konfigurationsdatei. In virtuellen Pools angegebene Standortwerte werden ignoriert.
- Um zu verwenden Cloud Identity, erhalten Sie die client ID von A ["Vom Benutzer zugewiesene verwaltete Identität"](#) und geben Sie diese ID in an azure.workload.identity/client-id: xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx.

Zusätzliche Anforderungen für SMB Volumes

Zur Erstellung eines SMB-Volumes müssen folgende Voraussetzungen erfüllt sein:

- Active Directory konfiguriert und mit Azure NetApp Files verbunden. Siehe ["Microsoft: Erstellen und Verwalten von Active Directory-Verbindungen für Azure NetApp Files"](#).
- Kubernetes-Cluster mit einem Linux-Controller-Knoten und mindestens einem Windows-Worker-Node, auf dem Windows Server 2022 ausgeführt wird. Trident unterstützt nur SMB Volumes, die in Pods gemountet sind, die nur auf Windows Nodes ausgeführt werden.
- Mindestens ein Trident-Schlüssel, der Ihre Active Directory-Anmeldeinformationen enthält, damit Azure NetApp Files sich bei Active Directory authentifizieren kann. So generieren Sie ein Geheimnis smbcreds:

```

kubectl create secret generic smbcreds --from-literal username=user
--from-literal password='password'

```

- Ein CSI-Proxy, der als Windows-Dienst konfiguriert ist. Informationen zum Konfigurieren csi-proxy von

finden Sie unter "[GitHub: CSI-Proxy](#)" oder "[GitHub: CSI Proxy für Windows](#)" für Kubernetes-Nodes, die unter Windows ausgeführt werden.

Azure NetApp Files Back-End-Konfigurationsoptionen und -Beispiele

Informieren Sie sich über die Backend-Konfigurationsoptionen NFS und SMB für Azure NetApp Files und sehen Sie sich Konfigurationsbeispiele an.

Back-End-Konfigurationsoptionen

Trident erstellt mithilfe Ihrer Backend-Konfiguration (Subnetz, virtuelles Netzwerk, Service Level und Standort) Azure NetApp Files Volumes in Kapazitätspools, die am angeforderten Standort verfügbar sind und mit dem angeforderten Service-Level und Subnetz übereinstimmen.



Trident unterstützt keine manuellen QoS-Kapazitätspools.

Azure NetApp Files Back-Ends bieten diese Konfigurationsoptionen.

Parameter	Beschreibung	Standard
version		Immer 1
storageDriverName	Name des Speichertreibers	„azure-netapp-files“
backendName	Benutzerdefinierter Name oder das Storage-Backend	Treibername + „_“ + zufällige Zeichen
subscriptionID	Die Abonnement-ID Ihres Azure-Abonnements Optional, wenn verwaltete Identitäten auf einem AKS-Cluster aktiviert sind.	
tenantID	Die Mandanten-ID einer App-Registrierung Optional, wenn verwaltete Identitäten oder Cloud-Identität auf einem AKS-Cluster verwendet wird.	
clientID	Die Client-ID einer App-Registrierung Optional, wenn verwaltete Identitäten oder Cloud-Identität auf einem AKS-Cluster verwendet wird.	
clientSecret	Der Client-Schlüssel aus einer App-Registrierung Optional, wenn verwaltete Identitäten oder Cloud-Identität auf einem AKS-Cluster verwendet wird.	
serviceLevel	Einer von Standard, Premium oder Ultra	„“ (zufällig)

Parameter	Beschreibung	Standard
location	Name des Azure-Standorts, an dem die neuen Volumes erstellt werden Optional, wenn verwaltete Identitäten auf einem AKS-Cluster aktiviert sind	
resourceGroups	Liste der Ressourcengruppen zum Filtern ermittelter Ressourcen	„[]“ (kein Filter)
netappAccounts	Liste von NetApp Accounts zur Filterung erkannter Ressourcen	„[]“ (kein Filter)
capacityPools	Liste der Kapazitäts-Pools zur Filterung erkannter Ressourcen	„[]“ (kein Filter, zufällig)
virtualNetwork	Name eines virtuellen Netzwerks mit einem delegierten Subnetz	“
subnet	Name eines Subnetzes, an das delegiert wurde Microsoft.Netapp/volumes	“
networkFeatures	Satz von vnet-Features für ein Volume, kann oder Standard sein Basic. Netzwerkfunktionen sind nicht in allen Regionen verfügbar und müssen möglicherweise in einem Abonnement aktiviert werden. Wenn die `networkFeatures` Funktion nicht aktiviert ist, schlägt die Volume-Bereitstellung fehl.	“
nfsMountOptions	Engmaschige Kontrolle der NFS-Mount-Optionen Für SMB Volumes ignoriert. Um Volumes mit NFS-Version 4.1 zu mounten, fügen Sie in die Liste mit kommagetrennten Mount-Optionen ein nfsvers=4, um NFS v4.1 auszuwählen. Mount-Optionen, die in einer Storage-Klassen-Definition festgelegt sind, überschreiben Mount-Optionen, die in der Backend-Konfiguration festgelegt sind.	„Nfsvers=3“
limitVolumeSize	Bereitstellung fehlgeschlagen, wenn die angeforderte Volume-Größe über diesem Wert liegt	“ (nicht standardmäßig durchgesetzt)

Parameter	Beschreibung	Standard
debugTraceFlags	Fehler-Flags bei der Fehlerbehebung beheben. Beispiel, <code>\{"api": false, "method": true, "discovery": true\}</code> . Verwenden Sie dies nur, wenn Sie Fehler beheben und einen detaillierten Log Dump benötigen.	Null
nasType	Konfiguration der Erstellung von NFS- oder SMB-Volumes Optionen sind <code>nfs</code> , <code>smb</code> oder Null. Einstellung auf null setzt standardmäßig auf NFS-Volumes.	<code>nfs</code>
supportedTopologies	Stellt eine Liste von Regionen und Zonen dar, die von diesem Backend unterstützt werden. Weitere Informationen finden Sie unter " Verwenden Sie die CSI-Topologie ".	



Weitere Informationen zu Netzwerkfunktionen finden Sie unter "[Konfigurieren Sie Netzwerkfunktionen für ein Azure NetApp Files Volume](#)".

Erforderliche Berechtigungen und Ressourcen

Wenn Sie beim Erstellen einer PVC den Fehler „Keine Kapazitätspools gefunden“ erhalten, sind Ihre App-Registrierung wahrscheinlich nicht über die erforderlichen Berechtigungen und Ressourcen (Subnetz, virtuelles Netzwerk, Kapazitäts-Pool) verbunden. Wenn Debug aktiviert ist, protokolliert Trident die beim Erstellen des Backends erkannten Azure-Ressourcen. Überprüfen Sie, ob eine geeignete Rolle verwendet wird.

Die Werte für `resourceGroups`, `netappAccounts`, `capacityPools`, `virtualNetwork` und `subnet` können mit kurzen oder vollqualifizierten Namen angegeben werden. In den meisten Fällen werden vollqualifizierte Namen empfohlen, da kurze Namen mehrere Ressourcen mit demselben Namen entsprechen können.

Die `resourceGroups` Werte, `netappAccounts` und `capacityPools` sind Filter, die die ermittelten Ressourcen auf die Ressourcen beschränken, die für dieses Speicher-Backend verfügbar sind und in jeder Kombination angegeben werden können. Vollqualifizierte Namen folgen diesem Format:

Typ	Formatieren
Ressourcengruppe	<code><Ressourcengruppe></code>
NetApp Konto	<code><Resource Group>/<netapp Account></code>
Kapazitäts-Pool	<code><Resource Group>/<netapp Account>/<Capacity Pool></code>
Virtuelles Netzwerk	<code><Ressourcengruppe>/<virtuelles Netzwerk></code>
Subnetz	<code><Ressourcengruppe>/<virtuelles Netzwerk>/<Subnetz></code>

Volume-Provisionierung

Sie können die standardmäßige Volume-Bereitstellung steuern, indem Sie die folgenden Optionen in einem speziellen Abschnitt der Konfigurationsdatei angeben. Weitere Informationen finden Sie unter [Beispielkonfigurationen](#).

Parameter	Beschreibung	Standard
exportRule	Exportregeln für neue Volumes exportRule Muss eine kommagetrennte Liste einer beliebigen Kombination von IPv4-Adressen oder IPv4-Subnetzen in CIDR-Notation sein. Für SMB Volumes ignoriert.	„0.0.0.0/0“
snapshotDir	Steuert die Sichtbarkeit des .Snapshot-Verzeichnisses	„Wahr“ für NFSv4 „falsch“ für NFSv3
size	Die Standardgröße der neuen Volumes	„100 GB“
unixPermissions	die unix-Berechtigungen neuer Volumes (4 Oktal-Ziffern). Für SMB Volumes ignoriert.	„“ (Vorschau-Funktion, erfordert Whitelisting im Abonnement)

Beispielkonfigurationen

Die folgenden Beispiele zeigen grundlegende Konfigurationen, bei denen die meisten Parameter standardmäßig belassen werden. Dies ist der einfachste Weg, ein Backend zu definieren.

Minimalkonfiguration

Dies ist die absolute minimale Backend-Konfiguration. Mit dieser Konfiguration erkennt Trident alle NetApp-Konten, Kapazitätspools und an Azure NetApp Files delegierte Subnetze am konfigurierten Standort und platziert neue Volumes zufällig in einem dieser Pools und Subnetze. Da `nasType` nicht angegeben ist, gilt der `nfs` Standard und das Backend wird für NFS Volumes bereitgestellt.

Diese Konfiguration ist ideal, wenn Sie gerade erst mit Azure NetApp Files beginnen und Dinge ausprobieren möchten, aber in der Praxis möchten Sie einen zusätzlichen Umfang für die bereitgestellten Volumes angeben.

```
---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-tbc-anf-1
  namespace: trident
spec:
  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
```

Verwaltete Identitäten für AKS

Diese Backend-Konfiguration unterlässt `subscriptionID`, `tenantID`, `clientID` und `clientSecret`, die bei der Verwendung von verwalteten Identitäten optional sind.

```
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-tbc-anf-1
  namespace: trident
spec:
  version: 1
  storageDriverName: azure-netapp-files
  capacityPools: ["ultra-pool"]
  resourceGroups: ["aks-ami-eastus-rg"]
  netappAccounts: ["smb-na"]
  virtualNetwork: eastus-prod-vnet
  subnet: eastus-anf-subnet
```

Cloud-Identität für AKS

Diese Backend-Konfiguration unterlässt `tenantID`, `clientID` und `clientSecret`, die optional sind, wenn Sie eine Cloud-Identität verwenden.

```
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-tbc-anf-1
  namespace: trident
spec:
  version: 1
  storageDriverName: azure-netapp-files
  capacityPools: ["ultra-pool"]
  resourceGroups: ["aks-ami-eastus-rg"]
  netappAccounts: ["smb-na"]
  virtualNetwork: eastus-prod-vnet
  subnet: eastus-anf-subnet
  location: eastus
  subscriptionID: 9f87c765-4774-fake-ae98-a721add45451
```

Spezifische Service-Level-Konfiguration mit Filtern nach Kapazitäts-Pools

Diese Backend-Konfiguration platziert Volumes an Azure `eastus` in einem `Ultra` Kapazitäts-Pool. Trident erkennt automatisch alle an Azure NetApp Files delegierten Subnetze an diesem Standort und platziert ein neues Volume zufällig in einem davon.

```
---
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
```

Erweiterte Konfiguration

Diese Back-End-Konfiguration reduziert den Umfang der Volume-Platzierung auf ein einzelnes Subnetz und ändert auch einige Standardwerte für die Volume-Bereitstellung.

```
---
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
networkFeatures: Standard
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'
```

Konfiguration des virtuellen Pools

Diese Back-End-Konfiguration definiert mehrere Storage-Pools in einer einzelnen Datei. Dies ist nützlich, wenn Sie über mehrere Kapazitäts-Pools verfügen, die unterschiedliche Service-Level unterstützen, und Sie Storage-Klassen in Kubernetes erstellen möchten, die diese unterstützen. Virtuelle Pool-Etiketten wurden verwendet, um die Pools anhandzu differenzieren *performance*.

```
---
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
networkFeatures: Basic
nfsMountOptions: vers=3,proto=tcp,timeo=600
labels:
  cloud: azure
storage:
- labels:
    performance: gold
    serviceLevel: Ultra
    capacityPools:
    - ultra-1
    - ultra-2
    networkFeatures: Standard
- labels:
    performance: silver
    serviceLevel: Premium
    capacityPools:
    - premium-1
- labels:
    performance: bronze
    serviceLevel: Standard
    capacityPools:
    - standard-1
    - standard-2
```

Konfiguration unterstützter Topologien

Trident erleichtert die Bereitstellung von Volumes für Workloads, basierend auf Regionen und Verfügbarkeitszonen. Der `supportedTopologies` Block in dieser Backend-Konfiguration dient zur Bereitstellung einer Liste von Regionen und Zonen pro Backend. Die hier angegebenen Region- und Zonenwerte müssen mit den Region- und Zonenwerten der Beschriftungen auf jedem Kubernetes-Cluster-Node übereinstimmen. Diese Regionen und Zonen stellen die Liste der zulässigen Werte dar, die in einer Lagerklasse bereitgestellt werden können. Für Storage-Klassen, die eine Teilmenge der Regionen und Zonen enthalten, die in einem Back-End bereitgestellt werden, erstellt Trident Volumes in der genannten Region und Zone. Weitere Informationen finden Sie unter ["Verwenden Sie die CSI-Topologie"](#).

```
---
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
supportedTopologies:
- topology.kubernetes.io/region: eastus
  topology.kubernetes.io/zone: eastus-1
- topology.kubernetes.io/region: eastus
  topology.kubernetes.io/zone: eastus-2
```

Definitionen der Storage-Klassen

Die folgenden `StorageClass` Definitionen beziehen sich auf die Speicherpools oben.

Beispieldefinitionen mit `parameter.selector` Feld

Mit `parameter.selector` können Sie für jeden virtuellen Pool angeben `StorageClass`, der zum Hosten eines Volumes verwendet wird. Im Volume werden die Aspekte definiert, die im ausgewählten Pool definiert sind.

```

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

```

Beispieldefinitionen für SMB Volumes

Mit `nasType`, `node-stage-secret-name` und `node-stage-secret-namespace` können Sie ein SMB-Volume angeben und die erforderlichen Active Directory-Anmeldeinformationen eingeben.

Grundkonfiguration im Standard-Namespace

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: anf-sc-smb
provisioner: csi.trident.netapp.io
parameters:
  backendType: "azure-netapp-files"
  trident.netapp.io/nasType: "smb"
  csi.storage.k8s.io/node-stage-secret-name: "smbcreds"
  csi.storage.k8s.io/node-stage-secret-namespace: "default"
```

Verschiedene Schlüssel pro Namespace verwenden

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: anf-sc-smb
provisioner: csi.trident.netapp.io
parameters:
  backendType: "azure-netapp-files"
  trident.netapp.io/nasType: "smb"
  csi.storage.k8s.io/node-stage-secret-name: "smbcreds"
  csi.storage.k8s.io/node-stage-secret-namespace: ${pvc.namespace}
```

Verschiedene Geheimnisse pro Band verwenden

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: anf-sc-smb
provisioner: csi.trident.netapp.io
parameters:
  backendType: "azure-netapp-files"
  trident.netapp.io/nasType: "smb"
  csi.storage.k8s.io/node-stage-secret-name: ${pvc.name}
  csi.storage.k8s.io/node-stage-secret-namespace: ${pvc.namespace}
```



`nasType: smb` Filter für Pools, die SMB Volumes unterstützen. `nasType: nfs` Oder `nasType: null` Filter für NFS-Pools.

Erstellen Sie das Backend

Führen Sie nach dem Erstellen der Back-End-Konfigurationsdatei den folgenden Befehl aus:

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

Wenn die Backend-Erstellung fehlschlägt, ist mit der Back-End-Konfiguration ein Fehler aufgetreten. Sie können die Protokolle zur Bestimmung der Ursache anzeigen, indem Sie den folgenden Befehl ausführen:

```
tridentctl logs
```

Nachdem Sie das Problem mit der Konfigurationsdatei identifiziert und korrigiert haben, können Sie den Befehl „Erstellen“ erneut ausführen.

Google Cloud NetApp Volumes

Google Cloud NetApp Volumes-Back-End konfigurieren

Sie können jetzt Google Cloud NetApp Volumes als Backend für Trident konfigurieren. Sie können NFS-Volumes über ein Google Cloud NetApp Volumes-Back-End einbinden.

Treiberdetails zu Google Cloud NetApp Volumes

Trident stellt den `google-cloud-netapp-volumes` Treiber für die Kommunikation mit dem Cluster bereit. Unterstützte Zugriffsmodi sind: *ReadWriteOnce* (RWO), *ReadOnly Many* (ROX), *ReadWriteMany* (RWX), *ReadWriteOncePod* (RWOP).

Treiber	Protokoll	VolumeModus	Unterstützte Zugriffsmodi	Unterstützte Filesysteme
<code>google-cloud-netapp-volumes</code>	NFS	Dateisystem	RWO, ROX, RWX, RWOP	<code>nfs</code>

Cloud-Identität für GKE

Die Cloud-Identität ermöglicht Kubernetes-Pods den Zugriff auf Google Cloud-Ressourcen durch Authentifizierung als Workload-Identität anstatt durch Angabe explizite Google Cloud-Anmeldedaten.

Um die Vorteile der Cloud-Identität in Google Cloud zu nutzen, müssen Sie über folgende Voraussetzungen verfügen:

- Ein mit GKE implementierter Kubernetes-Cluster.
- Auf dem GKE-Cluster konfigurierte Workload-Identität und auf den Node-Pools konfigurierten GKE-Metadatenserver.

- Ein GCP-Service-Konto mit der Google Cloud NetApp Volumes Admin-Rolle (Rollen/NetApp.admin) oder einer benutzerdefinierten Rolle.
- Trident installiert, das den CloudProvider enthält, um „GCP“ und CloudIdentity anzugeben, die das neue GCP-Dienstkonto angeben. Ein Beispiel ist unten angegeben.

Betreiber von Trident

Um Trident mithilfe des Trident-Operators zu installieren, bearbeiten Sie die `tridentorchestrator_cr.yaml` **Einstellung** `cloudProvider` auf und setzen Sie `cloudIdentity` `iam.gke.io/gcp-service-account: cloudvolumes-admin-sa@mygcpproject.iam.gserviceaccount.com` **sie** auf `"GCP"` .

Beispiel:

```
apiVersion: trident.netapp.io/v1
kind: TridentOrchestrator
metadata:
  name: trident
spec:
  debug: true
  namespace: trident
  imagePullPolicy: IfNotPresent
  cloudProvider: "GCP"
  cloudIdentity: 'iam.gke.io/gcp-service-account: cloudvolumes-admin-sa@mygcpproject.iam.gserviceaccount.com'
```

Helm

Legen Sie die Werte für **Cloud-Provider (CP)** und **Cloud-Identity (CI)** unter Verwendung der folgenden Umgebungsvariablen fest:

```
export CP="GCP"
export ANNOTATION="iam.gke.io/gcp-service-account: cloudvolumes-admin-sa@mygcpproject.iam.gserviceaccount.com"
```

Das folgende Beispiel installiert Trident und setzt `cloudProvider` auf `GCP` unter Verwendung der Umgebungsvariable `$CP` und setzt die `cloudIdentity` unter Verwendung der Umgebungsvariable `$ANNOTATION`:

```
helm install trident trident-operator-100.2406.0.tgz --set
cloudProvider=$CP --set cloudIdentity="$ANNOTATION"
```

`<code>-Datei findet </code>`

Legen Sie die Werte für **Cloud Provider** und **Cloud Identity** unter Verwendung der folgenden Umgebungsvariablen fest:

```
export CP="GCP"
export ANNOTATION="iam.gke.io/gcp-service-account: cloudvolumes-admin-sa@mygcpproject.iam.gserviceaccount.com"
```

Das folgende Beispiel installiert Trident und setzt das `cloud-provider` Flag auf `$CP`, und `cloud-identity` auf `$ANNOTATION`:

```
tridentctl install --cloud-provider=$CP --cloud
-identity="$ANNOTATION" -n trident
```

Bereiten Sie sich auf die Konfiguration eines Google Cloud NetApp Volumes-Back-End vor

Bevor Sie Ihr Google Cloud NetApp Volumes-Backend konfigurieren können, müssen Sie sicherstellen, dass die folgenden Anforderungen erfüllt sind.

Voraussetzungen für NFS Volumes

Wenn Sie Google Cloud NetApp Volumes zum ersten Mal oder an einem neuen Speicherort verwenden, ist eine Erstkonfiguration erforderlich, um Google Cloud NetApp Volumes einzurichten und ein NFS-Volume zu erstellen. Siehe ["Bevor Sie beginnen"](#).

Stellen Sie vor der Konfiguration des Google Cloud NetApp Volumes-Back-End sicher, dass folgende Voraussetzungen bestehen:

- Ein Google Cloud Konto, das mit dem Google Cloud NetApp Volumes Service konfiguriert ist. Siehe ["Google Cloud NetApp Volumes"](#).
- Projektnummer Ihres Google Cloud-Kontos. Siehe ["Projekte identifizieren"](#).
- Ein Google Cloud-Service-Konto mit der Rolle NetApp Volumes Admin (`roles/netapp.admin`). Siehe ["Rollen und Berechtigungen für Identitäts- und Zugriffsmanagement"](#).
- API-Schlüsseldatei für Ihr GCNV-Konto. Siehe ["Erstellen eines Service-Kontokonschlüssels"](#)
- Ein Speicherpool. Siehe ["Überblick über Speicherpools"](#).

Weitere Informationen zum Einrichten des Zugriffs auf Google Cloud NetApp Volumes finden Sie unter ["Zugriff auf Google Cloud NetApp Volumes einrichten"](#).

Konfigurationsoptionen und Beispiele für die Backend-Konfiguration von Google Cloud NetApp Volumes

Informieren Sie sich über die NFS-Back-End-Konfigurationsoptionen für Google Cloud NetApp Volumes und sehen Sie sich Konfigurationsbeispiele an.

Back-End-Konfigurationsoptionen

Jedes Back-End stellt Volumes in einer einzigen Google Cloud-Region bereit. Um Volumes in anderen Regionen zu erstellen, können Sie zusätzliche Back-Ends definieren.

Parameter	Beschreibung	Standard
version		Immer 1

Parameter	Beschreibung	Standard
storageDriverName	Name des Speichertreibers	Der Wert von <code>storageDriverName</code> muss als „google-Cloud-netapp-Volumes“ angegeben werden.
backendName	(Optional) Benutzerdefinierter Name des Speicher-Backends	Treibernamen + „_“ + Teil des API-Schlüssels
storagePools	Optionaler Parameter zur Angabe von Speicherpools für die Volume-Erstellung.	
projectNumber	Google Cloud Account Projektnummer. Der Wert ist auf der Startseite des Google Cloud Portals zu finden.	
location	Die Google Cloud-Umgebung, an der Trident GCNV Volumes erstellt. Bei der Erstellung regionsübergreifender Kubernetes-Cluster können in A erstellte Volumes <code>location</code> für Workloads verwendet werden, die auf Nodes in mehreren Google Cloud-Regionen geplant sind. Der regionale Verkehr verursacht zusätzliche Kosten.	
apiKey	API-Schlüssel für das Google Cloud-Servicekonto mit der <code>netapp.admin</code> Rolle. Er enthält den JSON-formatierten Inhalt der privaten Schlüsseldatei eines Google Cloud-Dienstkontos (wortgetreu in die Back-End-Konfigurationsdatei kopiert). Das <code>apiKey</code> muss Schlüssel-Wert-Paare für die folgenden Schlüssel enthalten: <code>type</code> , <code>project_id</code> , <code>client_email</code> , <code>client_id</code> , <code>auth_uri</code> , <code>token_uri</code> , <code>auth_provider_x509_cert_url</code> , und <code>client_x509_cert_url</code> .	
nfsMountOptions	Engmaschige Kontrolle der NFS-Mount-Optionen	„Nfsvers=3“
limitVolumeSize	Bereitstellung fehlgeschlagen, wenn die angeforderte Volume-Größe über diesem Wert liegt.	„“ (nicht standardmäßig durchgesetzt)
serviceLevel	Service-Level eines Storage-Pools und seiner Volumes. Die Werte sind <code>flex</code> , <code>standard</code> , <code>premium</code> oder <code>extreme</code> .	
network	Für GCNV-Volumes verwendetes Google Cloud-Netzwerk	
debugTraceFlags	Fehler-Flags bei der Fehlerbehebung beheben. Beispiel, <code>{"api":false, "method":true}</code> . Verwenden Sie dies nur, wenn Sie Fehler beheben und einen detaillierten Log Dump benötigen.	Null

Parameter	Beschreibung	Standard
supportedTopologies	Stellt eine Liste von Regionen und Zonen dar, die von diesem Backend unterstützt werden. Weitere Informationen finden Sie unter " Verwenden Sie die CSI-Topologie ". Beispiel: supportedTopologies: - topology.kubernetes.io/region: asia-east1 topology.kubernetes.io/zone: asia-east1-a	

Optionen zur Volume-Bereitstellung

Sie können die standardmäßige Volume-Bereitstellung im Abschnitt der Konfigurationsdatei steuern defaults.

Parameter	Beschreibung	Standard
exportRule	Die Exportregeln für neue Volumes. Muss eine kommasetrennte Liste einer beliebigen Kombination von IPv4-Adressen sein.	„0.0.0.0/0“
snapshotDir	Zugriff auf das .snapshot Verzeichnis	„Wahr“ für NFSv4 „falsch“ für NFSv3
snapshotReserve	Prozentsatz des für Snapshots reservierten Volumes	„“ (Standardeinstellung 0 akzeptieren)
unixPermissions	die unix-Berechtigungen neuer Volumes (4 Oktal-Ziffern).	„“

Beispielkonfigurationen

Die folgenden Beispiele zeigen grundlegende Konfigurationen, bei denen die meisten Parameter standardmäßig belassen werden. Dies ist der einfachste Weg, ein Backend zu definieren.

Minimalkonfiguration

Dies ist die absolute minimale Backend-Konfiguration. Mit dieser Konfiguration erkennt Trident alle an Google Cloud NetApp Volumes delegierten Storage-Pools am konfigurierten Standort und platziert neue Volumes zufällig in einem dieser Pools. Da `nasType` nicht angegeben ist, gilt der `nfs` Standard und das Backend wird für NFS Volumes bereitgestellt.

Diese Konfiguration ist ideal, wenn Sie gerade erst mit Google Cloud NetApp Volumes beginnen und alles ausprobieren möchten, aber in der Praxis müssen Sie höchstwahrscheinlich einen zusätzlichen Umfang für die bereitgestellten Volumes angeben.

[illegible]

```
XsYg6gyxy4zq7OlwWgLwGa==\n
-----END PRIVATE KEY-----\n
```

```
---
```

```
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-tbc-gcnv
spec:
  version: 1
  storageDriverName: google-cloud-netapp-volumes
  projectNumber: '123455380079'
  location: europe-west6
  serviceLevel: premium
  apiKey:
    type: service_account
    project_id: my-gcnv-project
    client_email: myproject-prod@my-gcnv-
project.iam.gserviceaccount.com
    client_id: '103346282737811234567'
    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/myproject-prod%40my-
gcnv-project.iam.gserviceaccount.com
  credentials:
    name: backend-tbc-gcnv-secret
```

[illegible]

```
version: 1
storageDriverName: google-cloud-netapp-volumes
projectNumber: '123455380079'
location: europe-west6
serviceLevel: premium
storagePools:
- premium-pool1-europe-west6
- premium-pool2-europe-west6
apiKey:
  type: service_account
  project_id: my-gcnv-project
  client_email: myproject-prod@my-gcnv-
project.iam.gserviceaccount.com
  client_id: '103346282737811234567'
  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/myproject-prod%40my-
gcnv-project.iam.gserviceaccount.com
credentials:
  name: backend-tbc-gcnv-secret
```

Konfiguration des virtuellen Pools

Diese Backend-Konfiguration definiert mehrere virtuelle Pools in einer einzelnen Datei. Virtuelle Pools werden im Abschnitt definiert `storage`. Sie sind nützlich, wenn Sie mehrere Storage-Pools haben, die unterschiedliche Service-Level unterstützen, und Sie Storage-Klassen in Kubernetes erstellen möchten, die diese repräsentieren. Zur Unterscheidung der Pools werden Bezeichnungen für virtuelle Pools verwendet. Im Beispiel unten werden beispielsweise `performance` Label und `serviceLevel` type zur Unterscheidung virtueller Pools verwendet.

Sie können auch einige Standardwerte für alle virtuellen Pools festlegen und die Standardwerte für einzelne virtuelle Pools überschreiben. Im folgenden Beispiel `snapshotReserve` und `exportRule` dienen als Standard für alle virtuellen Pools.

Weitere Informationen finden Sie unter ["Virtuelle Pools"](#).

[illegible]

```
znHczZsrtrHisIsAbOguSaPIKeyAZNchRAGz1zZE4jK3bl/qp8B4Kws8zX5ojY9m
znHczZsrtrHisIsAbOguSaPIKeyAZNchRAGz1zZE4jK3bl/qp8B4Kws8zX5ojY9m
znHczZsrtrHisIsAbOguSaPIKeyAZNchRAGz1zZE4jK3bl/qp8B4Kws8zX5ojY9m
XsYg6gyxy4zq7OlwWgLwGa==
-----END PRIVATE KEY-----
```

```
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-tbc-gcnv
spec:
  version: 1
  storageDriverName: google-cloud-netapp-volumes
  projectNumber: '123455380079'
  location: europe-west6
  apiKey:
    type: service_account
    project_id: my-gcnv-project
    client_email: myproject-prod@my-gcnv-
project.iam.gserviceaccount.com
    client_id: '103346282737811234567'
    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/myproject-prod%40my-
gcnv-project.iam.gserviceaccount.com
  credentials:
    name: backend-tbc-gcnv-secret
  defaults:
    snapshotReserve: '10'
    exportRule: 10.0.0.0/24
  storage:
    - labels:
        performance: extreme
        serviceLevel: extreme
      defaults:
        snapshotReserve: '5'
        exportRule: 0.0.0.0/0
    - labels:
        performance: premium
        serviceLevel: premium
    - labels:
```

```
performance: standard
serviceLevel: standard
```

Cloud-Identität für GKE

```
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-tbc-gcp-gcnv
spec:
  version: 1
  storageDriverName: google-cloud-netapp-volumes
  projectNumber: '012345678901'
  network: gcnv-network
  location: us-west2
  serviceLevel: Premium
  storagePool: pool-premium1
```

Konfiguration unterstützter Topologien

Trident erleichtert die Bereitstellung von Volumes für Workloads, basierend auf Regionen und Verfügbarkeitszonen. Der `supportedTopologies` Block in dieser Backend-Konfiguration dient zur Bereitstellung einer Liste von Regionen und Zonen pro Backend. Die hier angegebenen Region- und Zonenwerte müssen mit den Region- und Zonenwerten der Beschriftungen auf jedem Kubernetes-Cluster-Node übereinstimmen. Diese Regionen und Zonen stellen die Liste der zulässigen Werte dar, die in einer Lagerklasse bereitgestellt werden können. Für Storage-Klassen, die eine Teilmenge der Regionen und Zonen enthalten, die in einem Back-End bereitgestellt werden, erstellt Trident Volumes in der genannten Region und Zone. Weitere Informationen finden Sie unter ["Verwenden Sie die CSI-Topologie"](#).

```
---
version: 1
storageDriverName: google-cloud-netapp-volumes
subscriptionID: 9f87c765-4774-fake-ae98-a721add45451
tenantID: 68e4f836-edc1-fake-bff9-b2d865ee56cf
clientID: dd043f63-bf8e-fake-8076-8de91e5713aa
clientSecret: SECRET
location: asia-east1
serviceLevel: flex
supportedTopologies:
- topology.kubernetes.io/region: asia-east1
  topology.kubernetes.io/zone: asia-east1-a
- topology.kubernetes.io/region: asia-east1
  topology.kubernetes.io/zone: asia-east1-b
```

Was kommt als Nächstes?

Führen Sie nach dem Erstellen der Back-End-Konfigurationsdatei den folgenden Befehl aus:

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

Führen Sie den folgenden Befehl aus, um zu überprüfen, ob das Backend erfolgreich erstellt wurde:

```
kubectl get tridentbackendconfig
```

NAME	BACKEND NAME	BACKEND UUID
backend-tbc-gcnv	backend-tbc-gcnv	b2fd1ff9-b234-477e-88fd-713913294f65
Bound	Success	

Wenn die Backend-Erstellung fehlschlägt, ist mit der Back-End-Konfiguration ein Fehler aufgetreten. Sie können das Backend mit dem Befehl `kubectl get tridentbackendconfig <backend-name>` oder die Protokolle anzeigen, um die Ursache zu ermitteln, indem Sie den folgenden Befehl ausführen:

```
tridentctl logs
```

Nachdem Sie das Problem mit der Konfigurationsdatei identifiziert und behoben haben, können Sie das Backend löschen und den Befehl `create` erneut ausführen.

Weitere Beispiele

Beispiele für Definitionen von Storage-Klassen

Im Folgenden finden Sie eine grundlegende `StorageClass` Definition, die sich auf das Backend oben bezieht.

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: gcnv-nfs-sc
provisioner: csi.trident.netapp.io
parameters:
  backendType: "google-cloud-netapp-volumes"
```

Beispieldefinitionen mit dem `parameter.selector` Feld:

Mit `parameter.selector` können Sie für jeden angeben `StorageClass` "Virtueller Pool", der zum Hosten eines Volumes verwendet wird. Im Volume werden die Aspekte definiert, die im ausgewählten Pool definiert sind.

```

apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: extreme-sc
provisioner: csi.trident.netapp.io
parameters:
  selector: "performance=extreme"
  backendType: "google-cloud-netapp-volumes"
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: premium-sc
provisioner: csi.trident.netapp.io
parameters:
  selector: "performance=premium"
  backendType: "google-cloud-netapp-volumes"
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: standard-sc
provisioner: csi.trident.netapp.io
parameters:
  selector: "performance=standard"
  backendType: "google-cloud-netapp-volumes"

```

Weitere Informationen zu Speicherklassen finden Sie unter ["Erstellen Sie eine Speicherklasse"](#).

Beispiel für eine PVC-Definition

```

kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: gcnv-nfs-pvc
spec:
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 100Gi
  storageClassName: gcnv-nfs-sc

```

Um zu überprüfen, ob die PVC gebunden ist, führen Sie den folgenden Befehl aus:

```
kubectl get pvc gcnv-nfs-pvc
```

NAME	STATUS	VOLUME	CAPACITY
ACCESS MODES	STORAGECLASS	AGE	
gcnv-nfs-pvc	Bound	pvc-b00f2414-e229-40e6-9b16-ee03eb79a213	100Gi
RWX	gcnv-nfs-sc	1m	

Cloud Volumes Service für Google Cloud-Back-End konfigurieren

Erfahren Sie, wie Sie NetApp Cloud Volumes Service für Google Cloud mithilfe der bereitgestellten Beispielkonfigurationen als Back-End für Ihre Trident-Installation konfigurieren.

Treiberdetails zu Google Cloud

Trident stellt den `gcp-cvs` Treiber für die Kommunikation mit dem Cluster bereit. Unterstützte Zugriffsmodi sind: *ReadWriteOnce* (RWO), *ReadOnlyMany* (ROX), *ReadWriteMany* (RWX), *ReadWriteOncePod* (RWOP).

Treiber	Protokoll	VolumeModu s	Unterstützte Zugriffsmodi	Unterstützte Filesysteme
gcp-cvs	NFS	Dateisystem	RWO, ROX, RWX, RWOP	nfs

Erfahren Sie mehr über den Trident Support für Cloud Volumes Service für Google Cloud

Trident kann Cloud Volumes Service Volumes in einer von zwei erstellen "[Servicetypen](#)":

- **CVS-Performance:** Der Standard-Trident-Diensttyp. Dieser Performance-optimierte Service-Typ ist ideal für Produktions-Workloads, die Performance schätzen. Der CVS-Performance-Servicetyp ist eine Hardwareoption, die Volumes mit einer Größe von mindestens 100 gib unterstützt. Sie können eine der "[Drei Service-Level](#)" folgenden Optionen wählen:
 - `standard`
 - `premium`
 - `extreme`
- **CVS:** Der CVS-Servicetyp bietet eine hohe zonale Verfügbarkeit bei begrenzten bis moderaten Leistungsstufen. Der CVS-Servicetyp ist eine Software-Option, die Storage Pools zur Unterstützung von Volumes mit einer Größe von 1 gib verwendet. Der Speicherpool kann bis zu 50 Volumes enthalten, in denen sich alle Volumes die Kapazität und Performance des Pools teilen. Sie können eine der "[Zwei Service-Level](#)" folgenden Optionen wählen:
 - `standardsw`
 - `zoneredundantstandardsw`

Was Sie benötigen

Um das Backend zu konfigurieren und zu verwenden "[Cloud Volumes Service für Google Cloud](#)", benötigen Sie Folgendes:

- Ein Google Cloud Konto, das mit NetApp Cloud Volumes Service konfiguriert ist
- Projektnummer Ihres Google Cloud-Kontos
- Google Cloud Service-Konto mit der `netappcloudvolumes.admin` Rolle
- API-Schlüsseldatei für Ihr Cloud Volumes Service-Konto

Back-End-Konfigurationsoptionen

Jedes Back-End stellt Volumes in einer einzigen Google Cloud-Region bereit. Um Volumes in anderen Regionen zu erstellen, können Sie zusätzliche Back-Ends definieren.

Parameter	Beschreibung	Standard
<code>version</code>		Immer 1
<code>storageDriverName</code>	Name des Speichertreibers	„gcp-cvs“
<code>backendName</code>	Benutzerdefinierter Name oder das Storage-Backend	Treibername + „_“ + Teil des API-Schlüssels
<code>storageClass</code>	Optionaler Parameter zur Angabe des CVS-Servicetyps. Verwenden Sie <code>software</code> , um den CVS-Diensttyp auszuwählen. Andernfalls übernimmt Trident den CVS-Performance Servicetyp (<code>hardware</code>).	
<code>storagePools</code>	CVS-Diensttyp nur. Optionaler Parameter zur Angabe von Speicherpools für die Volume-Erstellung.	
<code>projectNumber</code>	Google Cloud Account Projektnummer. Der Wert ist auf der Startseite des Google Cloud Portals zu finden.	
<code>hostProjectNumber</code>	Erforderlich bei Verwendung eines gemeinsamen VPC-Netzwerks. In diesem Szenario <code>projectNumber</code> handelt es sich um das Service-Projekt und <code>hostProjectNumber</code> das Host-Projekt.	
<code>apiRegion</code>	Die Google Cloud-Region, in der Trident Cloud Volumes Service Volumes erstellt. Bei der Erstellung regionsübergreifender Kubernetes-Cluster können in einem erstellten Volumes <code>apiRegion</code> für Workloads verwendet werden, die auf Nodes in mehreren Google Cloud-Regionen geplant sind. Der regionale Verkehr verursacht zusätzliche Kosten.	
<code>apiKey</code>	API-Schlüssel für das Google Cloud-Servicekonto mit der <code>netappcloudvolumes.admin</code> Rolle. Er enthält den JSON-formatierten Inhalt der privaten Schlüsseldatei eines Google Cloud-Dienstkontos (wortgetreu in die Back-End-Konfigurationsdatei kopiert).	

Parameter	Beschreibung	Standard
proxyURL	Proxy-URL, wenn Proxyserver für die Verbindung mit dem CVS-Konto benötigt wird. Der Proxy-Server kann entweder ein HTTP-Proxy oder ein HTTPS-Proxy sein. Bei einem HTTPS-Proxy wird die Zertifikatvalidierung übersprungen, um die Verwendung von selbstsignierten Zertifikaten im Proxyserver zu ermöglichen. Proxy-Server mit aktivierter Authentifizierung werden nicht unterstützt.	
nfsMountOptions	Engmaschige Kontrolle der NFS-Mount-Optionen	„Nfsvers=3“
limitVolumeSize	Bereitstellung fehlgeschlagen, wenn die angeforderte Volume-Größe über diesem Wert liegt.	„“ (nicht standardmäßig durchgesetzt)
serviceLevel	Das CVS-Performance oder CVS Service-Level für neue Volumes. CVS-Leistungswerte sind <code>standard</code> , <code>premium</code> oder <code>extreme</code> . CVS-Werte sind <code>standardsw</code> oder <code>zoneredundantstandardsw</code> .	CVS-Performance ist der Standard. Der CVS-Standardwert ist „standardsw“.
network	Für Cloud Volumes Service Volumes verwendetes Google Cloud Netzwerk	„Standard“
debugTraceFlags	Fehler-Flags bei der Fehlerbehebung beheben. Beispiel, <code>\{"api":false, "method":true\}</code> . Verwenden Sie dies nur, wenn Sie Fehler beheben und einen detaillierten Log Dump benötigen.	Null
allowedTopologies	Um den regionsübergreifenden Zugriff zu ermöglichen, muss die StorageClass-Definition für <code>allowedTopologies</code> alle Regionen umfassen. Beispiel: - <code>key: topology.kubernetes.io/region</code> <code>values:</code> - <code>us-east1</code> - <code>europa-west1</code>	

Optionen zur Volume-Bereitstellung

Sie können die standardmäßige Volume-Bereitstellung im Abschnitt der Konfigurationsdatei steuern `defaults`.

Parameter	Beschreibung	Standard
exportRule	Die Exportregeln für neue Volumes. Muss eine kommasetrennte Liste beliebiger Kombinationen von IPv4-Adressen oder IPv4-Subnetzen in CIDR-Notation sein.	„0.0.0.0/0“
snapshotDir	Zugriff auf das <code>.snapshot</code> Verzeichnis	„Falsch“
snapshotReserve	Prozentsatz des für Snapshots reservierten Volumes	"" (CVS Standard 0 akzeptieren)

Parameter	Beschreibung	Standard
size	Die Größe neuer Volumes. Die Mindestmenge von CVS-Performance beträgt 100 gib. CVS mindestens 1 gib.	Der Servicetyp CVS-Performance ist standardmäßig auf „100 gib“ eingestellt. CVS-Diensttyp setzt keine Standardeinstellung, erfordert jedoch mindestens 1 gib.

Beispiele für CVS-Performance-Diensttypen

Die folgenden Beispiele enthalten Beispielkonfigurationen für den CVS-Performance-Servicetyp.

Beispiel 1: Minimale Konfiguration

Dies ist die minimale Backend-Konfiguration, die den standardmäßigen CVS-Performance-Servicetyp mit dem Standard-Service Level verwendet.

```
---
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-----
  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
```

Beispiel 2: Service Level-Konfiguration

Dieses Beispiel stellt die Back-End-Konfigurationsoptionen dar, einschließlich Service Level und Volume-Standard Einstellungen.

```
---
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-----
  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
```

Beispiel 3: Konfiguration des virtuellen Pools

Dieses Beispiel verwendet `storage`, um virtuelle Pools und die zu konfigurieren `StorageClasses`, die auf sie verweisen. Siehe [Definitionen der Storage-Klassen](#), um zu sehen, wie die Speicherklassen definiert wurden.

Hier werden spezifische Standardwerte für alle virtuellen Pools festgelegt, die den auf 5 % und den auf `exportRule 0.0.0.0/0` setzen `snapshotReserve`. Die virtuellen Pools werden im Abschnitt definiert `storage`. Jeder einzelne virtuelle Pool definiert seinen eigenen `serviceLevel`, und einige Pools überschreiben die Standardwerte. Virtuelle Pool-Etiketten wurden verwendet, um die Pools basierend auf `und protection` zu unterscheiden `performance`.

```
---
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-----
  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

```

Definitionen der Storage-Klassen

Die folgenden StorageClass-Definitionen gelten für das Beispiel der virtuellen Pool-Konfiguration. Mit `parameters.selector` können Sie für jede StorageClass den virtuellen Pool angeben, der zum Hosten eines Volumes verwendet wird. Im Volume werden die Aspekte definiert, die im ausgewählten Pool definiert sind.

Beispiel für Storage-Klasse

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

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

- Die erste StorageClass (`cvs-extreme-extra-protection`) wird dem ersten virtuellen Pool zugeordnet. Dies ist der einzige Pool, der eine extreme Performance mit einer Snapshot-Reserve von 10 % bietet.
- Die letzte StorageClass (`cvs-extra-protection`) ruft jeden Speicherpool auf, der eine Snapshot-Reserve von 10% bietet. Trident entscheidet, welcher virtuelle Pool ausgewählt wird, und stellt sicher, dass die Anforderung der Snapshot-Reserve erfüllt wird.

Beispiele für CVS-Diensttypen

Die folgenden Beispiele enthalten Beispielkonfigurationen für den CVS-Servicetyp.

Beispiel 1: Minimalkonfiguration

Dies ist die minimale Backend-Konfiguration `storageClass` zur Angabe des CVS-Diensttyps und des Standard-`standardsw` Service-Levels.

```
---
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-----
  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
serviceLevel: standardsw
```

Beispiel 2: Konfiguration des Storage Pools

Diese Beispiel-Backend-Konfiguration verwendet `storagePools`, um einen Speicherpool zu konfigurieren.

```
---
version: 1
storageDriverName: gcp-cvs
backendName: gcp-std-so-with-pool
projectNumber: '531265380079'
apiRegion: europe-west1
apiKey:
  type: service_account
  project_id: cloud-native-data
  private_key_id: "<id_value>"
  private_key: |-
    -----BEGIN PRIVATE KEY-----
    <key_value>
    -----END PRIVATE KEY-----
  client_email: cloudvolumes-admin-sa@cloud-native-
data.iam.gserviceaccount.com
  client_id: '107071413297115343396'
  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%40cloud-native-data.iam.gserviceaccount.com
storageClass: software
zone: europe-west1-b
network: default
storagePools:
- 1bc7f380-3314-6005-45e9-c7dc8c2d7509
serviceLevel: Standardsw
```

Was kommt als Nächstes?

Führen Sie nach dem Erstellen der Back-End-Konfigurationsdatei den folgenden Befehl aus:

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

Wenn die Backend-Erstellung fehlschlägt, ist mit der Back-End-Konfiguration ein Fehler aufgetreten. Sie können die Protokolle zur Bestimmung der Ursache anzeigen, indem Sie den folgenden Befehl ausführen:

```
tridentctl logs
```

Nachdem Sie das Problem mit der Konfigurationsdatei identifiziert und korrigiert haben, können Sie den Befehl „Erstellen“ erneut ausführen.

Konfigurieren Sie ein NetApp HCI- oder SolidFire-Backend

Erfahren Sie, wie Sie mit Ihrer Trident Installation ein Element Backend erstellen und verwenden.

Details zum Elementtreiber

Trident stellt den `solidfire-san` Speichertreiber für die Kommunikation mit dem Cluster bereit. Unterstützte Zugriffsmodi sind: *ReadWriteOnce* (RWO), *ReadOnly Many* (ROX), *ReadWriteMany* (RWX), *ReadWriteOncePod* (RWOP).

Der `solidfire-san` Speichertreiber unterstützt die Volume-Modi *File* und *Block*. Für den `Filesystem` Volumemodus erstellt Trident ein Volume und ein Dateisystem. Der Dateisystem-Typ wird von `StorageClass` angegeben.

Treiber	Protokoll	VolumeMode	Unterstützte Zugriffsmodi	Unterstützte Filesysteme
<code>solidfire-san</code>	ISCSI	Block-Storage	RWO, ROX, RWX, RWOP	Kein Dateisystem. Rohes Blockgerät.
<code>solidfire-san</code>	ISCSI	Dateisystem	RWO, RWOP	<code>xfs ext3, , ext4</code>

Bevor Sie beginnen

Sie benötigen Folgendes, bevor Sie ein Element-Backend erstellen.

- Ein unterstütztes Storage-System, auf dem die Element Software ausgeführt wird.
- Anmeldedaten für einen NetApp HCI/SolidFire Cluster-Administrator oder einen Mandantenbenutzer, der Volumes managen kann
- Alle Kubernetes-Worker-Nodes sollten die entsprechenden iSCSI-Tools installiert haben. Siehe ["Informationen zur Vorbereitung auf den Worker-Node"](#).

Back-End-Konfigurationsoptionen

Die Back-End-Konfigurationsoptionen finden Sie in der folgenden Tabelle:

Parameter	Beschreibung	Standard
<code>version</code>		Immer 1
<code>storageDriverName</code>	Name des Speichertreibers	Immer „solidfire-san“

Parameter	Beschreibung	Standard
backendName	Benutzerdefinierter Name oder das Storage-Backend	IP-Adresse „SolidFire_“ + Storage (iSCSI)
Endpoint	MVIP für den SolidFire-Cluster mit Mandanten-Anmeldedaten	
SVIP	Speicher-IP-Adresse und -Port	
labels	Satz willkürlicher JSON-formatierter Etiketten für Volumes.	„
TenantName	Zu verwendende Mandantenbezeichnung (wird erstellt, wenn sie nicht gefunden wurde)	
InitiatorIFace	Beschränken Sie den iSCSI-Datenverkehr auf eine bestimmte Host-Schnittstelle	„Standard“
UseCHAP	Verwenden Sie CHAP zur Authentifizierung von iSCSI. Trident verwendet CHAP.	Richtig
AccessGroups	Liste der zu verwendenden Zugriffsgruppen-IDs	Findet die ID einer Zugriffsgruppe namens „Dreizack“
Types	QoS-Spezifikationen	
limitVolumeSize	Bereitstellung fehlgeschlagen, wenn die angeforderte Volume-Größe über diesem Wert liegt	„ (nicht standardmäßig durchgesetzt)
debugTraceFlags	Fehler-Flags bei der Fehlerbehebung beheben. Beispiel: { „API“:false, „Methode“:true}	Null



Verwenden Sie diese Funktion `debugTraceFlags` nur, wenn Sie eine Fehlerbehebung durchführen und einen detaillierten Protokollauszug benötigen.

Beispiel 1: Backend-Konfiguration für `solidfire-san` Treiber mit drei Volume-Typen

Dieses Beispiel zeigt eine Backend-Datei mit CHAP-Authentifizierung und Modellierung von drei Volume-Typen mit spezifischen QoS-Garantien. Sehr wahrscheinlich würden Sie dann Storage-Klassen definieren, um diese mit dem Storage-Klassen-Parameter zu nutzen `IOPS`.

```

---
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

```

Beispiel 2: Back-End- und Storage-Klassenkonfiguration für solidfire-san Treiber mit virtuellen Pools

Dieses Beispiel zeigt die mit virtuellen Pools zusammen mit StorageClasses konfigurierte Back-End-Definitionsdatei.

Trident kopiert bei der Bereitstellung Labels, die sich in einem Storage-Pool befinden, auf die Back-End-Storage-LUN. Storage-Administratoren können Labels je virtuellen Pool definieren und Volumes nach Label gruppieren.

In der unten abgebildeten Beispieldefinitionsdatei für das Backend werden spezifische Standardwerte für alle Speicherpools festgelegt, die die auf „Silber“ setzen `type`. Die virtuellen Pools werden im Abschnitt `storage` definiert. In diesem Beispiel legen einige Speicherpools ihren eigenen Typ fest, und einige Pools überschreiben die oben festgelegten Standardwerte.

```

---
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

```

Die folgenden StorageClass-Definitionen beziehen sich auf die oben genannten virtuellen Pools. Mit dem

`parameters.selector` Feld ruft jede `StorageClass` ab, welche virtuellen Pools zum Hosten eines Volumes verwendet werden können. Auf dem Volume werden die Aspekte im ausgewählten virtuellen Pool definiert.

Die erste `StorageClass` (`solidfire-gold-four`) wird dem ersten virtuellen Pool zugeordnet. Dies ist der einzige Pool, der eine Goldleistung mit einem Gold bietet `Volume Type QoS`. Die letzte `StorageClass` (`solidfire-silver`) ruft jeden Speicherpool auf, der eine silberne Performance bietet. Trident entscheidet, welcher virtuelle Pool ausgewählt wird, und stellt sicher, dass die Speicheranforderungen erfüllt werden.

```

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"

```

Weitere Informationen

- ["Volume-Zugriffsgruppen"](#)

ONTAP SAN-Treiber

Übersicht über ONTAP SAN-Treiber

Erfahren Sie mehr über die Konfiguration eines ONTAP-Backend mit ONTAP- und Cloud Volumes ONTAP-SAN-Treibern.

Details zum ONTAP-SAN-Treiber

Trident stellt die folgenden SAN-Speichertreiber für die Kommunikation mit dem ONTAP-Cluster bereit. Unterstützte Zugriffsmodi sind: *ReadWriteOnce* (RWO), *ReadOnly Many* (ROX), *ReadWriteMany* (RWX), *ReadWriteOncePod* (RWOP).

Treiber	Protokoll	VolumeModus	Unterstützte Zugriffsmodi	Unterstützte Filesysteme
ontap-san	ISCSI SCSI über FC (technische Vorschau in Trident 24.10)	Block-Storage	RWO, ROX, RWX, RWOP	Kein Filesystem, rohes Block-Gerät
ontap-san	ISCSI SCSI über FC (technische Vorschau in Trident 24.10)	Dateisystem	RWO, RWOP ROX und RWX sind im Filesystem-Volume-Modus nicht verfügbar.	xfs ext3, , ext4
ontap-san	NVMe/TCP Siehe Weitere Überlegungen zu NVMe/TCP .	Block-Storage	RWO, ROX, RWX, RWOP	Kein Filesystem, rohes Block-Gerät
ontap-san	NVMe/TCP Siehe Weitere Überlegungen zu NVMe/TCP .	Dateisystem	RWO, RWOP ROX und RWX sind im Filesystem-Volume-Modus nicht verfügbar.	xfs ext3, , ext4

Treiber	Protokoll	VolumeModus	Unterstützte Zugriffsmodi	Unterstützte Filesysteme
ontap-san-economy	ISCSI	Block-Storage	RWO, ROX, RWX, RWOP	Kein Filesystem, rohes Block-Gerät
ontap-san-economy	ISCSI	Dateisystem	RWO, RWOP ROX und RWX sind im Filesystem-Volume-Modus nicht verfügbar.	xfs ext3, , ext4



- Verwenden Sie `ontap-san-economy` diese Option nur, wenn die Anzahl der persistenten Volumes voraussichtlich höher ist als "[Unterstützte ONTAP-Volume-Größen](#)".
- Verwenden Sie `ontap-nas-economy` diese Option nur, wenn die Anzahl der persistenten Volumes voraussichtlich höher ist als "[Unterstützte ONTAP-Volume-Größen](#)" und der `ontap-san-economy` Treiber nicht verwendet werden kann.
- Verwenden Sie diese Option nicht `ontap-nas-economy`, wenn Sie voraussehen, dass Datensicherung, Disaster Recovery oder Mobilität erforderlich sind.

Benutzerberechtigungen

Trident geht davon aus, dass es entweder als ONTAP- oder SVM-Administrator ausgeführt wird, wobei der Cluster-Benutzer oder ein `vsadmin` SVM-Benutzer oder ein Benutzer mit einem anderen Namen und derselben Rolle verwendet `admin` wird. Bei Implementierungen von Amazon FSX for NetApp ONTAP rechnet Trident damit, als ONTAP- oder SVM-Administrator ausgeführt zu werden. Dabei verwendet er den Cluster-`fsxadmin` Benutzer, einen `vsadmin` SVM-Benutzer oder einen Benutzer mit einem anderen Namen mit derselben Rolle. Der `fsxadmin` Benutzer ist ein eingeschränkter Ersatz für den Cluster-Admin-Benutzer.



Wenn Sie den Parameter verwenden `limitAggregateUsage`, sind Administratorberechtigungen für den Cluster erforderlich. Wenn Amazon FSX for NetApp ONTAP mit Trident verwendet wird, funktioniert der `limitAggregateUsage` Parameter nicht mit den `vsadmin` Benutzerkonten und `fsxadmin`. Der Konfigurationsvorgang schlägt fehl, wenn Sie diesen Parameter angeben.

Es ist zwar möglich, eine restriktivere Rolle in ONTAP zu erstellen, die ein Trident-Treiber verwenden kann, wir empfehlen sie jedoch nicht. Bei den meisten neuen Versionen von Trident sind zusätzliche APIs erforderlich, die berücksichtigt werden müssten, was Upgrades schwierig und fehleranfällig macht.

Weitere Überlegungen zu NVMe/TCP

Trident unterstützt das NVMe-Protokoll (Non-Volatile Memory Express) unter Verwendung des `ontap-san` Treibers, einschließlich:

- IPv6
- Snapshots und Klone von NVMe Volumes
- Größe eines NVMe Volumes ändern
- Importieren eines NVMe Volumes, das außerhalb von Trident erstellt wurde, damit sein Lebenszyklus durch Trident gemanagt werden kann

- NVMe-natives Multipathing
- Ordnungsgemäßes oder unzumutbar Herunterfahren der K8s-Nodes (24.06)

Trident unterstützt Folgendes nicht:

- Dh-HMAC-CHAP, das von nativ von NVMe unterstützt wird
- Multipathing für Device Mapper (DM)
- LUKS-Verschlüsselung

Vorbereiten der Back-End-Konfiguration mit ONTAP-SAN-Treibern

Verstehen Sie die Anforderungen und Authentifizierungsoptionen für die Konfiguration eines ONTAP-Backends mit ONTAP-SAN-Treibern.

Anforderungen

Für alle ONTAP-Back-Ends benötigt Trident der SVM mindestens ein Aggregat zugewiesen.

Denken Sie daran, dass Sie auch mehr als einen Treiber ausführen können und Speicherklassen erstellen können, die auf den einen oder anderen verweisen. Sie können beispielsweise eine Klasse konfigurieren `san-dev`, die den `ontap-san` Treiber und eine `san-default` Klasse verwendet, die diesen verwendet `ontap-san-economy`.

Alle Kubernetes-Worker-Nodes müssen über die entsprechenden iSCSI-Tools verfügen. Weitere Informationen finden Sie unter "[Bereiten Sie den Knoten „Worker“ vor](#)".

Authentifizieren Sie das ONTAP-Backend

Trident bietet zwei Arten der Authentifizierung eines ONTAP-Backends.

- Anmeldeinformationsbasiert: Benutzername und Passwort für einen ONTAP-Benutzer mit den erforderlichen Berechtigungen. Es wird empfohlen, eine vordefinierte Sicherheits-Login-Rolle zu verwenden, wie `admin` oder `vsadmin`, um maximale Kompatibilität mit ONTAP-Versionen zu gewährleisten.
- Zertifikat-basiert: Trident kann auch über ein auf dem Backend installiertes Zertifikat mit einem ONTAP-Cluster kommunizieren. Hier muss die Backend-Definition Base64-kodierte Werte des Client-Zertifikats, des Schlüssels und des vertrauenswürdigen CA-Zertifikats enthalten, sofern verwendet (empfohlen).

Sie können vorhandene Back-Ends aktualisieren, um zwischen auf Anmeldeinformationen basierenden und zertifikatbasierten Methoden zu verschieben. Es wird jedoch immer nur eine Authentifizierungsmethode unterstützt. Um zu einer anderen Authentifizierungsmethode zu wechseln, müssen Sie die vorhandene Methode von der Backend-Konfiguration entfernen.



Wenn Sie versuchen, **sowohl Anmeldeinformationen als auch Zertifikate** bereitzustellen, schlägt die Backend-Erstellung mit einem Fehler fehl, dass mehr als eine Authentifizierungsmethode in der Konfigurationsdatei angegeben wurde.

Aktivieren Sie die Anmeldeinformationsbasierte Authentifizierung

Für die Kommunikation mit dem ONTAP-Back-End ist die Zugangsdaten an einen Administrator mit SVM-Umfang/Cluster-Umfang erforderlich Trident. Es wird empfohlen, standardmäßige, vordefinierte Rollen wie

oder `vsadmin`` zu verwenden ``admin`. So wird die Kompatibilität mit zukünftigen ONTAP Versionen sichergestellt, die möglicherweise die FunktionAPIs für zukünftige Trident Versionen offenlegen. Eine benutzerdefinierte Sicherheits-Login-Rolle kann erstellt und mit Trident verwendet werden, wird aber nicht empfohlen.

Eine Beispiel-Back-End-Definition sieht folgendermaßen aus:

YAML

```
---
version: 1
backendName: ExampleBackend
storageDriverName: ontap-san
managementLIF: 10.0.0.1
svm: svm_nfs
username: vsadmin
password: password
```

JSON

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

Beachten Sie, dass die Backend-Definition der einzige Ort ist, an dem die Anmeldeinformationen im reinen Text gespeichert werden. Nach der Erstellung des Backend werden Benutzernamen/Passwörter mit Base64 codiert und als Kubernetes Secrets gespeichert. Die Erstellung oder Aktualisierung eines Backend ist der einzige Schritt, der Kenntnisse über die Anmeldeinformationen erfordert. Daher ist dieser Vorgang nur für Administratoren und wird vom Kubernetes-/Storage-Administrator ausgeführt.

Aktivieren Sie die zertifikatbasierte Authentifizierung

Neue und vorhandene Back-Ends können ein Zertifikat verwenden und mit dem ONTAP-Back-End kommunizieren. In der Backend-Definition sind drei Parameter erforderlich.

- **ClientCertificate:** Base64-codierter Wert des Clientzertifikats.
- **ClientPrivateKey:** Base64-kodierte Wert des zugeordneten privaten Schlüssels.
- **Trusted CACertificate:** Base64-codierter Wert des vertrauenswürdigen CA-Zertifikats. Bei Verwendung einer vertrauenswürdigen CA muss dieser Parameter angegeben werden. Dies kann ignoriert werden, wenn keine vertrauenswürdige CA verwendet wird.

Ein typischer Workflow umfasst die folgenden Schritte.

Schritte

1. Erzeugen eines Clientzertifikats und eines Schlüssels. Legen Sie beim Generieren den allgemeinen Namen (CN) für den ONTAP-Benutzer fest, der sich authentifizieren soll als.

```
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. Fügen Sie dem ONTAP-Cluster ein vertrauenswürdiges CA-Zertifikat hinzu. Dies kann möglicherweise bereits vom Storage-Administrator übernommen werden. Ignorieren, wenn keine vertrauenswürdige CA verwendet wird.

```
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. Installieren Sie das Client-Zertifikat und den Schlüssel (von Schritt 1) auf dem ONTAP-Cluster.

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

4. Vergewissern Sie sich, dass die ONTAP-Sicherheits-Anmeldungsrolle die Authentifizierungsmethode unterstützt cert.

```
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. Testen Sie die Authentifizierung mithilfe des generierten Zertifikats. <ONTAP Management LIF> und <vServer Name> durch Management-LIF-IP und SVM-Namen ersetzen.

```
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. Encodieren von Zertifikat, Schlüssel und vertrauenswürdigen CA-Zertifikat mit Base64.

```
base64 -w 0 k8serv.pem >> cert_base64
base64 -w 0 k8serv.key >> key_base64
base64 -w 0 trustedca.pem >> trustedca_base64
```

7. Erstellen Sie das Backend mit den Werten, die aus dem vorherigen Schritt ermittelt wurden.

```
cat cert-backend.json
{
  "version": 1,
  "storageDriverName": "ontap-san",
  "backendName": "SanBackend",
  "managementLIF": "1.2.3.4",
  "svm": "vserver_test",
  "clientCertificate": "Faaaakkkkeeee...Vaaalllluuuuueeee",
  "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 |
+-----+-----+-----+-----+
+-----+-----+
```

Aktualisieren Sie Authentifizierungsmethoden, oder drehen Sie die Anmeldedaten

Sie können ein vorhandenes Backend aktualisieren, um eine andere Authentifizierungsmethode zu verwenden oder ihre Anmeldedaten zu drehen. Das funktioniert auf beide Arten: Back-Ends, die einen Benutzernamen/ein Passwort verwenden, können aktualisiert werden, um Zertifikate zu verwenden; Back-Ends, die Zertifikate verwenden, können auf Benutzername/Passwort-basiert aktualisiert werden. Dazu müssen Sie die vorhandene Authentifizierungsmethode entfernen und die neue Authentifizierungsmethode hinzufügen. Verwenden Sie dann die aktualisierte Datei Backend.json, die die erforderlichen Parameter enthält `tridentctl backend update`.

```
cat cert-backend-updated.json
{
  "version": 1,
  "storageDriverName": "ontap-san",
  "backendName": "SanBackend",
  "managementLIF": "1.2.3.4",
  "svm": "vserver_test",
  "username": "vsadmin",
  "password": "password",
  "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 |
+-----+-----+-----+
+-----+-----+
```



Bei der Änderung von Passwörtern muss der Speicheradministrator das Kennwort für den Benutzer auf ONTAP aktualisieren. Auf diese Weise folgt ein Backend-Update. Beim Drehen von Zertifikaten können dem Benutzer mehrere Zertifikate hinzugefügt werden. Das Backend wird dann aktualisiert und verwendet das neue Zertifikat. Danach kann das alte Zertifikat aus dem ONTAP Cluster gelöscht werden.

Durch die Aktualisierung eines Backend wird der Zugriff auf Volumes, die bereits erstellt wurden, nicht unterbrochen, und auch die danach erstellten Volume-Verbindungen werden beeinträchtigt. Ein erfolgreiches Backend-Update zeigt an, dass Trident mit dem ONTAP Back-End kommunizieren und zukünftige Volume-Operationen verarbeiten kann.

Benutzerdefinierte ONTAP-Rolle für Trident erstellen

Sie können eine ONTAP-Cluster-Rolle mit minimaler Privileges erstellen, sodass Sie nicht die ONTAP-Administratorrolle verwenden müssen, um Vorgänge in Trident auszuführen. Wenn Sie den Benutzernamen in eine Trident-Back-End-Konfiguration aufnehmen, verwendet Trident die ONTAP-Cluster-Rolle, die Sie für die Durchführung der Vorgänge erstellt haben.

Weitere Informationen zum Erstellen benutzerdefinierter Trident-Rollen finden Sie unter ["Trident Custom-Role Generator"](#).

Verwenden der ONTAP CLI

1. Erstellen Sie eine neue Rolle mit dem folgenden Befehl:

```
security login role create <role_name\> -cmddirname "command" -access all  
-vserver <svm_name\>
```

2. Erstellen Sie einen Benutzernamen für den Trident-Benutzer:

```
security login create -username <user_name\> -application ontapi  
-authmethod <password\> -role <name_of_role_in_step_1\> -vserver  
<svm_name\> -comment "user_description"
```

3. Ordnen Sie die Rolle dem Benutzer zu:

```
security login modify username <user_name\> -vserver <svm_name\> -role  
<role_name\> -application ontapi -application console -authmethod  
<password\>
```

Verwenden Von System Manager

Führen Sie die folgenden Schritte im ONTAP System Manager durch:

1. Erstellen Sie eine benutzerdefinierte Rolle:

- a. Um eine benutzerdefinierte Rolle auf Cluster-Ebene zu erstellen, wählen Sie **Cluster > Einstellungen** aus.

(Oder) um eine benutzerdefinierte Rolle auf SVM-Ebene zu erstellen, wählen Sie **Storage > Storage VMs > > required svm Einstellungen > Benutzer und Rollen** aus.

- b. Wählen Sie das Pfeilsymbol (→) neben **Users and Roles**.
- c. Wählen Sie unter **Rollen +Hinzufügen** aus.
- d. Definieren Sie die Regeln für die Rolle und klicken Sie auf **Speichern**.

2. **Rolle dem Trident-Benutzer zuordnen:** + Führen Sie auf der Seite **Benutzer und Rollen** folgende Schritte aus:

- a. Wählen Sie unter **Benutzer** das Symbol Hinzufügen +.
- b. Wählen Sie den gewünschten Benutzernamen aus, und wählen Sie im Dropdown-Menü für **Rolle** eine Rolle aus.
- c. Klicken Sie Auf **Speichern**.

Weitere Informationen finden Sie auf den folgenden Seiten:

- ["Benutzerdefinierte Rollen für die Administration von ONTAP"](#) Oder ["Definieren benutzerdefinierter Rollen"](#)
- ["Arbeiten Sie mit Rollen und Benutzern"](#)

Verbindungen mit bidirektionalem CHAP authentifizieren

Trident kann iSCSI-Sitzungen mit bidirektionalem CHAP für den und `ontap-san-economy`-Treiber authentifizieren `ontap-san`. Dazu muss die Option in Ihrer Backend-Definition aktiviert `useCHAP` werden. Wenn auf festgelegt `true`, konfiguriert Trident die standardmäßige Initiatorsicherheit der SVM auf

bidirektionales CHAP und legt den Benutzernamen und die Schlüssel aus der Backend-Datei fest. NetApp empfiehlt die Verwendung von bidirektionalem CHAP zur Authentifizierung von Verbindungen. Die folgende Beispielkonfiguration ist verfügbar:

```
---
version: 1
storageDriverName: ontap-san
backendName: ontap_san_chap
managementLIF: 192.168.0.135
svm: ontap_iscsi_svm
useCHAP: true
username: vsadmin
password: password
chapInitiatorSecret: cl9qxIm36DKyawxy
chapTargetInitiatorSecret: rqxigXgkesIpwxyz
chapTargetUsername: iJF4heBRT0TCwxyz
chapUsername: uh2aNCLSD6cNwxyz
```



Der `useCHAP` Parameter ist eine Boolesche Option, die nur einmal konfiguriert werden kann. Die Standardeinstellung ist „false“. Nachdem Sie die Einstellung auf „true“ gesetzt haben, können Sie sie nicht auf „false“ setzen.

Zusätzlich zu `useCHAP=true` `chapTargetUsername` müssen die `chapInitiatorSecret` Felder `chapTargetInitiatorSecret` und `chapUsername` in die Backend-Definition einbezogen werden. Die Secrets können geändert werden, nachdem ein Backend durch Ausführen erstellt `tridentctl update` wurde.

So funktioniert es

Durch die Einstellung `useCHAP` auf `true` weist der Speicheradministrator Trident an, CHAP auf dem Speicher-Back-End zu konfigurieren. Dazu gehört Folgendes:

- Einrichten von CHAP auf der SVM:
 - Wenn der standardmäßige Sicherheitstyp des Initiators der SVM `none` (standardmäßig festgelegt) ist **und**, wenn keine bereits vorhandenen LUNs im Volume vorhanden sind, setzt Trident den Standardsicherheitstyp auf `CHAP` und fährt mit der Konfiguration des CHAP-Initiators und des Zielbenutzernamens und der -Schlüssel fort.
 - Wenn die SVM LUNs enthält, aktiviert Trident CHAP auf der SVM nicht. Dadurch wird sichergestellt, dass der Zugriff auf die LUNs, die bereits auf der SVM vorhanden sind, nicht eingeschränkt wird.
- Konfigurieren des CHAP-Initiators und des Ziel-Usernamens und der Schlüssel; diese Optionen müssen in der Back-End-Konfiguration angegeben werden (siehe oben).

Nach der Erstellung des Backends erstellt Trident eine entsprechende `tridentbackend` CRD und speichert die CHAP-Geheimnisse und Benutzernamen als Kubernetes-Geheimnisse. Alle PVS, die von Trident auf diesem Backend erstellt werden, werden über CHAP gemountet und angehängt.

Anmeldedaten rotieren und Back-Ends aktualisieren

Sie können die CHAP-Anmeldeinformationen aktualisieren, indem Sie die CHAP-Parameter in der Datei aktualisieren `backend.json`. Dies erfordert die Aktualisierung der CHAP-Schlüssel und die Verwendung des `tridentctl update` Befehls, um diese Änderungen widerzuspiegeln.



Wenn Sie die CHAP-Schlüssel für ein Backend aktualisieren, müssen Sie `tridentctl` das Backend aktualisieren. Aktualisieren Sie die Zugangsdaten auf dem Storage-Cluster nicht über die CLI/ONTAP-Benutzeroberfläche, da Trident diese Änderungen nicht aufnehmen kann.

```
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": "password",
  "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 |
+-----+-----+-----+-----+
+-----+-----+
```

Bestehende Verbindungen bleiben nicht betroffen und bleiben weiterhin aktiv, wenn die Zugangsdaten von Trident auf der SVM aktualisiert werden. Für neue Verbindungen werden die aktualisierten Anmeldeinformationen verwendet, und bestehende Verbindungen bleiben weiterhin aktiv. Wenn Sie alte PVS trennen und neu verbinden, werden sie die aktualisierten Anmeldedaten verwenden.

ONTAP-SAN-Konfigurationsoptionen und Beispiele

Erfahren Sie, wie Sie ONTAP-SAN-Treiber mit Ihrer Trident-Installation erstellen und verwenden. Dieser Abschnitt enthält Beispiele und Details zur Back-End-Konfiguration für

die Zuordnung von Back-Ends zu StorageClasses.

Back-End-Konfigurationsoptionen

Die Back-End-Konfigurationsoptionen finden Sie in der folgenden Tabelle:

Parameter	Beschreibung	Standard
version		Immer 1
storageDriverName	Name des Speichertreibers	ontap-nas, ontap-nas-economy, ontap-nas-flexgroup, ontap-san, ontap-san-economy
backendName	Benutzerdefinierter Name oder das Storage-Backend	Treibername + „_“ + DatenLIF
managementLIF	Die IP-Adresse einer Cluster- oder SVM-Management-LIF. Es kann ein vollständig qualifizierter Domänenname (FQDN) angegeben werden. Kann so eingestellt werden, dass IPv6-Adressen verwendet werden, wenn Trident mit dem IPv6-Flag installiert wurde. IPv6-Adressen müssen in eckigen Klammern definiert werden, z. B. [28e8:d9fb:a825:b7bf:69a8:d02f:9e7b:3555]. Informationen über die nahtlose MetroCluster-Umschaltung finden Sie im [mcc-best] .	„10.0.0.1“, „[2001:1234:abcd::fefe]“
dataLIF	IP-Adresse des LIF-Protokolls. Kann so eingestellt werden, dass IPv6-Adressen verwendet werden, wenn Trident mit dem IPv6-Flag installiert wurde. IPv6-Adressen müssen in eckigen Klammern definiert werden, z. B. [28e8:d9fb:a825:b7bf:69a8:d02f:9e7b:3555]. Nicht für iSCSI angeben. Trident verwendet "ONTAP selektive LUN-Zuordnung", um die für die Einrichtung einer Multi-Path-Sitzung erforderlichen iSCSI LIFs zu ermitteln. Eine Warnung wird erzeugt, wenn dataLIF explizit definiert ist. Für MetroCluster weglassen. Siehe [mcc-best] .	Abgeleitet von SVM
svm	Zu verwendende virtuelle Speichermaschine omit für MetroCluster. Siehe [mcc-best] .	Abgeleitet, wenn eine SVM managementLIF angegeben wird
useCHAP	Verwenden Sie CHAP, um iSCSI für ONTAP-SAN-Treiber zu authentifizieren [Boolesch]. Legen Sie für Trident fest true, um bidirektionales CHAP als Standardauthentifizierung für die im Backend angegebene SVM zu konfigurieren und zu verwenden. Weitere Informationen finden Sie unter " Vorbereiten der Back-End-Konfiguration mit ONTAP-SAN-Treibern ".	false
chapInitiatorSecret	CHAP-Initiatorschlüssel. Erforderlich, wenn useCHAP=true	“ ”

Parameter	Beschreibung	Standard
labels	Satz willkürlicher JSON-formatierter Etiketten für Volumes	“ ”
chapTargetInitiatorSecret	Schlüssel für CHAP-Zielinitiator. Erforderlich, wenn useCHAP=true	“ ”
chapUsername	Eingehender Benutzername. Erforderlich, wenn useCHAP=true	“ ”
chapTargetUsername	Zielbenutzername. Erforderlich, wenn useCHAP=true	“ ”
clientCertificate	Base64-codierter Wert des Clientzertifikats. Wird für zertifikatbasierte Authentifizierung verwendet	“ ”
clientPrivateKey	Base64-kodierte Wert des privaten Client-Schlüssels. Wird für zertifikatbasierte Authentifizierung verwendet	“ ”
trustedCACertificate	Base64-kodierte Wert des vertrauenswürdigen CA-Zertifikats. Optional Wird für die zertifikatbasierte Authentifizierung verwendet.	“ ”
username	Benutzername für die Kommunikation mit dem ONTAP Cluster erforderlich. Wird für die Anmeldeinformationsbasierte Authentifizierung verwendet.	“ ”
password	Passwort, das für die Kommunikation mit dem ONTAP Cluster erforderlich ist. Wird für die Anmeldeinformationsbasierte Authentifizierung verwendet.	“ ”
svm	Zu verwendende Storage Virtual Machine	Abgeleitet, wenn eine SVM managementLIF angegeben wird
storagePrefix	Das Präfix wird beim Bereitstellen neuer Volumes in der SVM verwendet. Kann später nicht mehr geändert werden. Um diesen Parameter zu aktualisieren, müssen Sie ein neues Backend erstellen.	trident

Parameter	Beschreibung	Standard
aggregate	Aggregat für die Bereitstellung (optional, wenn eingestellt, muss der SVM zugewiesen werden) Für den <code>ontap-nas-flexgroup</code> Treiber wird diese Option ignoriert. Falls nicht, können alle verfügbaren Aggregate verwendet werden, um ein FlexGroup Volume bereitzustellen.  Wenn das Aggregat in einer SVM aktualisiert wird, wird es automatisch in Trident aktualisiert, indem es die SVM abfragt, ohne den Trident Controller neu starten zu müssen. Wenn Sie ein bestimmtes Aggregat in Trident für die Bereitstellung von Volumes konfiguriert haben, wird das Back-End Trident bei der Abfrage des SVM-Aggregats in den Status „Fehlgeschlagen“ verschoben. Sie müssen entweder das Aggregat zu einem auf der SVM vorhandenen Aggregat ändern oder es komplett entfernen, um das Back-End wieder online zu schalten.	“
limitAggregateUsage	Bereitstellung fehlgeschlagen, wenn die Nutzung über diesem Prozentsatz liegt. Wenn Sie ein Amazon FSX für NetApp ONTAP-Backend verwenden, geben Sie nicht an <code>limitAggregateUsage</code> . Die angegebenen <code>fsxadmin</code> und <code>vsadmin</code> enthalten nicht die erforderlichen Berechtigungen, um die aggregierte Nutzung abzurufen und sie mit Trident zu begrenzen.	„ (nicht standardmäßig durchgesetzt)
limitVolumeSize	Bereitstellung fehlgeschlagen, wenn die angeforderte Volume-Größe über diesem Wert liegt. Beschränkt außerdem die maximale Größe der Volumes, die es für LUNs managt.	„ (nicht standardmäßig durchgesetzt)
lunsPerFlexvol	Die maximale Anzahl an LUNs pro FlexVol muss im Bereich [50, 200] liegen.	100
debugTraceFlags	Fehler-Flags bei der Fehlerbehebung beheben. Beispiel, {„API“:false, „method“:true} nicht verwenden, es sei denn, Sie beheben die Fehlerbehebung und benötigen einen detaillierten Log Dump.	null

Parameter	Beschreibung	Standard
useREST	Boolescher Parameter zur Verwendung von ONTAP REST-APIs. useREST Bei Einstellung auf <code>true</code> verwendet Trident ONTAP REST APIs zur Kommunikation mit dem Backend; bei Einstellung auf <code>false</code> verwendet Trident ONTAP ZAPI Aufrufe zur Kommunikation mit dem Backend. Diese Funktion erfordert ONTAP 9.11.1 und höher. Darüber hinaus muss die verwendete ONTAP-Anmelderolle Zugriff auf die Anwendung haben <code>ontap</code>. Dies wird durch die vordefinierten <code>vsadmin</code> Rollen und <code>cluster-admin</code> erreicht. Ab Trident 24.06-Version und ONTAP 9.15.1 oder höher <code>useREST</code> ist standardmäßig auf eingestellt <code>true</code>; ändern Sie <code>useREST</code> zu <code>false</code> ONTAP-ZAPI-Aufrufe verwenden. <code>useREST</code> Ist vollständig für NVMe/TCP qualifiziert.	<code>true</code> Für ONTAP 9.15.1 oder höher, andernfalls <code>false</code> .
sanType	Verwenden Sie diese Option, um für iSCSI, <code>nvme</code> für NVMe/TCP oder <code>fcp</code> für SCSI über Fibre Channel (FC) auszuwählen <code>iscsi</code> . 'fcp' (SCSI over FC) ist ein Tech Preview Feature in der Trident 24.10 Version.	<code>iscsi</code> Falls leer
formatOptions	Verwenden Sie <code>formatOptions</code> zum Angeben von Befehlszeilenargumenten für den <code>mkfs</code> Befehl, die bei jedem Formatieren eines Volumes angewendet werden. Auf diese Weise können Sie die Lautstärke nach Ihren Wünschen formatieren. Stellen Sie sicher, dass Sie die Formatieroptionen ähnlich wie die der <code>mkfs</code>-Befehlsoptionen angeben, ohne den Gerätepfad. Beispiel: „-E nodiscard“ • <code>ontap-san`ontap-san-economy`</code>Nur für und Treiber unterstützt.*	
limitVolumePoolSize	Maximale anforderbare FlexVol-Größe bei Verwendung von LUNs im ONTAP-san-Economy-Backend.	„“ (nicht standardmäßig durchgesetzt)
denyNewVolumePools	Schränkt das Erstellen neuer FlexVol Volumes für LUNs ein <code>ontap-san-economy</code> Zur Bereitstellung neuer PVS werden nur vorbestehende FlexVols verwendet.	

Empfehlungen für die Verwendung von FormatOptions

Trident empfiehlt die folgende Option, um den Formatierungsprozess zu beschleunigen:

-E nodiscard:

- Beibehalten, versuchen Sie nicht, Blöcke zur `mkfs`-Zeit zu verwerfen (das Verwerfen von Blöcken ist zunächst auf Solid State-Geräten und selten/Thin Provisioning-Storage nützlich). Dies ersetzt die veraltete Option „-K“ und ist auf alle Dateisysteme anwendbar (xfs, ext3 und ext4).

Back-End-Konfigurationsoptionen für die Bereitstellung von Volumes

Mit diesen Optionen können Sie die Standardbereitstellung im Abschnitt der Konfiguration steuern `defaults`. Ein Beispiel finden Sie unten in den Konfigurationsbeispielen.

Parameter	Beschreibung	Standard
<code>spaceAllocation</code>	Speicherplatzzuweisung für LUNs	„Wahr“
<code>spaceReserve</code>	Modus für Speicherplatzreservierung; „none“ (Thin) oder „Volume“ (Thick)	„Keine“
<code>snapshotPolicy</code>	Die Snapshot-Richtlinie zu verwenden	„Keine“
<code>qosPolicy</code>	QoS-Richtliniengruppe zur Zuweisung für erstellte Volumes Wählen Sie eine der <code>qosPolicy</code> oder <code>adaptiveQosPolicy</code> pro Storage Pool/Backend. Für die Verwendung von QoS-Richtliniengruppen mit Trident ist ONTAP 9.8 oder höher erforderlich. Sie sollten eine nicht gemeinsam genutzte QoS-Richtliniengruppe verwenden und sicherstellen, dass die Richtliniengruppe auf jede Komponente einzeln angewendet wird. Eine Shared-QoS-Richtliniengruppe erzwingt die Obergrenze für den Gesamtdurchsatz aller Workloads.	“
<code>adaptiveQosPolicy</code>	Adaptive QoS-Richtliniengruppe mit Zuordnung für erstellte Volumes Wählen Sie eine der <code>qosPolicy</code> oder <code>adaptiveQosPolicy</code> pro Storage Pool/Backend	“
<code>snapshotReserve</code>	Prozentsatz des für Snapshots reservierten Volumes	„0“, wenn <code>snapshotPolicy</code> „keine“ ist, andernfalls „“
<code>splitOnClone</code>	Teilen Sie einen Klon bei der Erstellung von seinem übergeordneten Objekt auf	„Falsch“
<code>encryption</code>	Aktivieren Sie NetApp Volume Encryption (NVE) auf dem neuen Volume, Standardeinstellung ist <code>false</code> . NVE muss im Cluster lizenziert und aktiviert sein, damit diese Option verwendet werden kann. Wenn auf dem Backend NAE aktiviert ist, wird jedes in Trident bereitgestellte Volume NAE aktiviert. Weitere Informationen finden Sie unter "Funktionsweise von Trident mit NVE und NAE" .	„Falsch“
<code>luksEncryption</code>	Aktivieren Sie die LUKS-Verschlüsselung. Siehe "Linux Unified Key Setup (LUKS) verwenden" . LUKS-Verschlüsselung wird für NVMe/TCP nicht unterstützt.	“
<code>securityStyle</code>	Sicherheitstyp für neue Volumes	<code>unix</code>
<code>tieringPolicy</code>	Tiering-Richtlinie, die zu „keinen“ verwendet wird	„Nur snapshot“ für eine SVM-DR-Konfiguration vor ONTAP 9.5
<code>nameTemplate</code>	Vorlage zum Erstellen benutzerdefinierter Volume-Namen.	“

Beispiele für die Volume-Bereitstellung

Hier ein Beispiel mit definierten Standardwerten:

```
---
version: 1
storageDriverName: ontap-san
managementLIF: 10.0.0.1
svm: trident_svm
username: admin
password: <password>
labels:
  k8scluster: dev2
  backend: dev2-sanbackend
storagePrefix: alternate-trident
debugTraceFlags:
  api: false
  method: true
defaults:
  spaceReserve: volume
  qosPolicy: standard
  spaceAllocation: 'false'
  snapshotPolicy: default
  snapshotReserve: '10'
```



Für alle Volumes, die mit dem Treiber erstellt `ontap-san` wurden, fügt Trident der FlexVol zusätzliche Kapazität von 10 % hinzu, um die LUN-Metadaten aufzunehmen. Die LUN wird genau mit der Größe bereitgestellt, die der Benutzer in der PVC anfordert. Trident addiert 10 Prozent zum FlexVol (wird als verfügbare Größe in ONTAP angezeigt). Benutzer erhalten jetzt die Menge an nutzbarer Kapazität, die sie angefordert haben. Diese Änderung verhindert auch, dass LUNs schreibgeschützt werden, sofern der verfügbare Speicherplatz nicht vollständig genutzt wird. Dies gilt nicht für die Wirtschaft von `ontap-san`.

Für Back-Ends, die definieren `snapshotReserve`, berechnet Trident die Größe der Volumes wie folgt:

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

Die 1.1 ist die zusätzliche 10 Prozent Trident fügt zu den FlexVol, um die LUN-Metadaten aufzunehmen. Für `snapshotReserve = 5 %` und die PVC-Anforderung = 5 gib beträgt die Gesamtgröße des Volumes 5,79 gib und die verfügbare Größe 5,5 gib. Der `volume show` Befehl sollte die Ergebnisse ähnlich wie in diesem Beispiel anzeigen:

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.

Die Größenanpassung ist derzeit die einzige Möglichkeit, die neue Berechnung für ein vorhandenes Volume zu verwenden.

Minimale Konfigurationsbeispiele

Die folgenden Beispiele zeigen grundlegende Konfigurationen, bei denen die meisten Parameter standardmäßig belassen werden. Dies ist der einfachste Weg, ein Backend zu definieren.



Wenn Sie Amazon FSX auf NetApp ONTAP mit Trident verwenden, empfehlen wir, DNS-Namen für LIFs anstelle von IP-Adressen anzugeben.

Beispiel: ONTAP SAN

Dies ist eine Grundkonfiguration mit dem `ontap-san` Treiber.

```
---
version: 1
storageDriverName: ontap-san
managementLIF: 10.0.0.1
svm: svm_iscsi
labels:
  k8scluster: test-cluster-1
  backend: testcluster1-sanbackend
username: vsadmin
password: <password>
```

Beispiel für die SAN-Ökonomie von ONTAP

```
---
version: 1
storageDriverName: ontap-san-economy
managementLIF: 10.0.0.1
svm: svm_iscsi_eco
username: vsadmin
password: <password>
```

1. Beispiel

Sie können das Backend konfigurieren, um zu vermeiden, dass die Backend-Definition nach Umschaltung und Switchback während manuell aktualisiert "[SVM-Replizierung und Recovery](#)" werden muss.

Geben Sie für ein nahtloses Switchover und Switchback die SVM mit an `managementLIF` und lassen Sie die Parameter und `svm` weg `dataLIF`. Beispiel:

```
---
version: 1
storageDriverName: ontap-san
managementLIF: 192.168.1.66
username: vsadmin
password: password
```

Beispiel für die zertifikatbasierte Authentifizierung

In diesem Beispiel der Grundkonfiguration `clientCertificate` werden , `clientPrivateKey` und `trustedCACertificate` (optional, wenn vertrauenswürdige CA verwendet wird) eingetragen `backend.json` und die base64-kodierten Werte des Clientzertifikats, des privaten Schlüssels und des vertrauenswürdigen CA-Zertifikats verwendet.

```
---
version: 1
storageDriverName: ontap-san
backendName: DefaultSANBackend
managementLIF: 10.0.0.1
svm: svm_iscsi
useCHAP: true
chapInitiatorSecret: cl9qxIm36DKyawxy
chapTargetInitiatorSecret: rqxigXgkesIpwxyz
chapTargetUsername: iJF4heBRT0TCwxyz
chapUsername: uh2aNCLSD6cNwxyz
clientCertificate: ZXR0ZXJwYXB...ICMgJ3BhcGVyc2
clientPrivateKey: vciwKIyAgZG...0cnksIGRlc2NyaX
trustedCACertificate: zcyBbaG...b3Igb3duIGNsYXNz
```

Beispiele für bidirektionales CHAP

Diese Beispiele erzeugen ein Backend mit `useCHAP` set to `true`.

Beispiel für ONTAP-SAN-CHAP

```
---
version: 1
storageDriverName: ontap-san
managementLIF: 10.0.0.1
svm: svm_iscsi
labels:
  k8scluster: test-cluster-1
  backend: testcluster1-sanbackend
useCHAP: true
chapInitiatorSecret: cl9qxIm36DKyawxy
chapTargetInitiatorSecret: rqxigXgkesIpwxyz
chapTargetUsername: iJF4heBRT0TCwxyz
chapUsername: uh2aNCLSD6cNwxyz
username: vsadmin
password: <password>
```

Beispiel für ONTAP SAN Economy CHAP

```
---
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
username: vsadmin
password: <password>
```

Beispiel für NVMe/TCP

Sie müssen eine SVM auf Ihrem ONTAP Back-End mit NVMe konfiguriert haben. Dies ist eine grundlegende Backend-Konfiguration für NVMe/TCP.

```
---
version: 1
backendName: NVMeBackend
storageDriverName: ontap-san
managementLIF: 10.0.0.1
svm: svm_nvme
username: vsadmin
password: password
sanType: nvme
useREST: true
```

Back-End-Konfigurationsbeispiel mit nameTemplate

```
---
version: 1
storageDriverName: ontap-san
backendName: ontap-san-backend
managementLIF: <ip address>
svm: svm0
username: <admin>
password: <password>
defaults: {
  "nameTemplate":
    "{{.volume.Name}}_{{.labels.cluster}}_{{.volume.Namespace}}_{{.volume.R
    equestName}}"
},
"labels": {"cluster": "ClusterA", "PVC":
  "{{.volume.Namespace}}_{{.volume.RequestName}}"}
}
```

FormatOptions-Beispiel für den `ONTAP-san-economy`-Treiber

```
version: 1
storageDriverName: ontap-san-economy
managementLIF: ''
svm: svm1
username: ''
password: "!"
storagePrefix: whelk_
debugTraceFlags:
  method: true
  api: true
defaults:
  formatOptions: "-E nodiscard"
```

Beispiele für Back-Ends mit virtuellen Pools

In diesen Beispiel-Back-End-Definitionsdateien werden spezifische Standardwerte für alle Speicherpools festgelegt, z. B. `spaceReserve` bei `none`, `spaceAllocation` bei `false` und `encryption` bei `false`. Die virtuellen Pools werden im Abschnitt `Speicher` definiert.

Trident legt die Bereitstellungsetiketten im Feld „Kommentare“ fest. Kommentare werden auf dem FlexVol gesetzt. Trident kopiert bei der Bereitstellung alle Labels, die sich in einem virtuellen Pool befinden, auf das Storage-Volume. Storage-Administratoren können Labels je virtuellen Pool definieren und Volumes nach Label gruppieren.

In diesen Beispielen legen einige Speicherpools eigene Werte für `spaceAllocation` und `spaceReserve` fest, und `encryption` überschreiben einige Pools die Standardwerte.



```

---
version: 1
storageDriverName: ontap-san
managementLIF: 10.0.0.1
svm: svm_iscsi
useCHAP: true
chapInitiatorSecret: cl9qxIm36DKyawxy
chapTargetInitiatorSecret: rqxigXgkesIpwxyz
chapTargetUsername: iJF4heBRT0TCwxyz
chapUsername: uh2aNCLSD6cNwxyz
username: vsadmin
password: <password>
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'

```

Beispiel für die SAN-Ökonomie von ONTAP

```
---
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
username: vsadmin
password: <password>
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'
- labels:
  department: legal
  creditpoints: '5000'
  zone: us_east_1c
```

```
defaults:
  spaceAllocation: 'true'
  encryption: 'false'
```

Beispiel für NVMe/TCP

```
---
version: 1
storageDriverName: ontap-san
sanType: nvme
managementLIF: 10.0.0.1
svm: nvme_svm
username: vsadmin
password: <password>
useREST: true
defaults:
  spaceAllocation: 'false'
  encryption: 'true'
storage:
- labels:
  app: testApp
  cost: '20'
  defaults:
    spaceAllocation: 'false'
    encryption: 'false'
```

Back-Ends StorageClasses zuordnen

Die folgenden StorageClass-Definitionen beziehen sich auf [Beispiele für Back-Ends mit virtuellen Pools](#). Mit dem `parameters.selector` Feld ruft jede StorageClass ab, welche virtuellen Pools zum Hosten eines Volumes verwendet werden können. Auf dem Volume werden die Aspekte im ausgewählten virtuellen Pool definiert.

- Die `protection-gold` StorageClass wird dem ersten virtuellen Pool im Backend zugeordnet `ontap-san`. Dies ist der einzige Pool mit Gold-Level-Schutz.

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: protection-gold
provisioner: csi.trident.netapp.io
parameters:
  selector: "protection=gold"
  fsType: "ext4"
```

- Die **protection-not-gold StorageClass** wird dem zweiten und dritten virtuellen Pool im Backend zugeordnet **ontap-san**. Dies sind die einzigen Pools, die ein anderes Schutzniveau als Gold bieten.

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: protection-not-gold
provisioner: csi.trident.netapp.io
parameters:
  selector: "protection!=gold"
  fsType: "ext4"
```

- Die **app-mysqldb StorageClass** wird dem dritten virtuellen Pool im Backend zugeordnet **ontap-san-economy**. Dies ist der einzige Pool, der Storage-Pool-Konfiguration für die **mysqldb-App** bietet.

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: app-mysqldb
provisioner: csi.trident.netapp.io
parameters:
  selector: "app=mysqldb"
  fsType: "ext4"
```

- Die **protection-silver-creditpoints-20k StorageClass** wird dem zweiten virtuellen Pool im Backend zugeordnet **ontap-san**. Dies ist der einzige Pool mit Silber-Level-Schutz und 20000 Kreditpunkte.

```

apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: protection-silver-creditpoints-20k
provisioner: csi.trident.netapp.io
parameters:
  selector: "protection=silver; creditpoints=20000"
  fsType: "ext4"

```

- Die creditpoints-5k StorageClass wird dem dritten virtuellen Pool im Backend und dem vierten virtuellen Pool im Backend ontap-san-economy zugeordnet ontap-san. Dies sind die einzigen Poolangebote mit 5000 Kreditpunkten.

```

apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: creditpoints-5k
provisioner: csi.trident.netapp.io
parameters:
  selector: "creditpoints=5000"
  fsType: "ext4"

```

- Die my-test-app-sc StorageClass wird dem virtuellen Pool im ontap-san Treiber mit sanType: nvme zugeordnet testAPP. Dies ist der einzige Pool, der angeboten `testApp` wird.

```

---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: my-test-app-sc
provisioner: csi.trident.netapp.io
parameters:
  selector: "app=testApp"
  fsType: "ext4"

```

Trident entscheidet, welcher virtuelle Pool ausgewählt wird, und stellt sicher, dass die Speicheranforderungen erfüllt werden.

ONTAP NAS-Treiber

Übersicht über ONTAP NAS-Treiber

Erfahren Sie mehr über die Konfiguration eines ONTAP-Backend mit ONTAP- und Cloud

Volumes ONTAP-NAS-Treibern.

Details zum ONTAP-NAS-Treiber

Trident stellt die folgenden NAS-Speichertreiber für die Kommunikation mit dem ONTAP-Cluster bereit. Unterstützte Zugriffsmodi sind: *ReadWriteOnce* (RWO), *ReadOnly Many* (ROX), *ReadWriteMany* (RWX), *ReadWriteOncePod* (RWOP).

Treiber	Protokoll	VolumeModus	Unterstützte Zugriffsmodi	Unterstützte Filesysteme
ontap-nas	NFS SMB	Dateisystem	RWO, ROX, RWX, RWOP	„“, nfs, smb
ontap-nas-economy	NFS SMB	Dateisystem	RWO, ROX, RWX, RWOP	„“, nfs, smb
ontap-nas-flexgroup	NFS SMB	Dateisystem	RWO, ROX, RWX, RWOP	„“, nfs, smb



- Verwenden Sie `ontap-nas-economy` diese Option nur, wenn die Anzahl der persistenten Volumes voraussichtlich höher ist als ["Unterstützte ONTAP-Volume-Größen"](#).
- Verwenden Sie `ontap-nas-economy` diese Option nur, wenn die Anzahl der persistenten Volumes voraussichtlich höher ist als ["Unterstützte ONTAP-Volume-Größen"](#) und der `ontap-nas-economy` Treiber nicht verwendet werden kann.
- Verwenden Sie diese Option nicht `ontap-nas-economy`, wenn Sie voraussehen, dass Datensicherung, Disaster Recovery oder Mobilität erforderlich sind.

Benutzerberechtigungen

Trident geht davon aus, dass es entweder als ONTAP- oder SVM-Administrator ausgeführt wird, wobei der Cluster-Benutzer oder ein `vsadmin` SVM-Benutzer oder ein Benutzer mit einem anderen Namen und derselben Rolle verwendet wird.

Bei Implementierungen von Amazon FSX for NetApp ONTAP rechnet Trident damit, als ONTAP- oder SVM-Administrator ausgeführt zu werden. Dabei verwendet er den Cluster- `fsxadmin` Benutzer, einen `vsadmin` SVM-Benutzer oder einen Benutzer mit einem anderen Namen mit derselben Rolle. Der `fsxadmin` Benutzer ist ein eingeschränkter Ersatz für den Cluster-Admin-Benutzer.



Wenn Sie den Parameter verwenden `limitAggregateUsage`, sind Administratorberechtigungen für den Cluster erforderlich. Wenn Amazon FSX for NetApp ONTAP mit Trident verwendet wird, funktioniert der `limitAggregateUsage` Parameter nicht mit den `vsadmin` Benutzerkonten und `fsxadmin`. Der Konfigurationsvorgang schlägt fehl, wenn Sie diesen Parameter angeben.

Es ist zwar möglich, eine restriktivere Rolle in ONTAP zu erstellen, die ein Trident-Treiber verwenden kann, wir empfehlen sie jedoch nicht. Bei den meisten neuen Versionen von Trident sind zusätzliche APIs erforderlich, die berücksichtigt werden müssten, was Upgrades schwierig und fehleranfällig macht.

Bereiten Sie sich auf die Konfiguration eines Backend mit ONTAP-NAS-Treibern vor

Verstehen Sie die Anforderungen, Authentifizierungsoptionen und Exportrichtlinien für die Konfiguration eines ONTAP-Backends mit ONTAP-NAS-Treibern.

Anforderungen

- Für alle ONTAP-Back-Ends benötigt Trident der SVM mindestens ein Aggregat zugewiesen.
- Sie können mehrere Treiber ausführen und Speicherklassen erstellen, die auf den einen oder den anderen zeigen. Sie können beispielsweise eine Gold-Klasse konfigurieren, die den Treiber verwendet `ontap-nas`, und eine Bronze-Klasse, die den Treiber verwendet `ontap-nas-economy`.
- Alle Kubernetes-Worker-Nodes müssen über die entsprechenden NFS-Tools verfügen. ["Hier"](#) Weitere Informationen finden Sie unter.
- Trident unterstützt nur SMB Volumes, die in Pods gemountet sind, die nur auf Windows Nodes ausgeführt werden. Weitere Informationen finden Sie unter [Vorbereitung zur Bereitstellung von SMB Volumes](#).

Authentifizieren Sie das ONTAP-Backend

Trident bietet zwei Arten der Authentifizierung eines ONTAP-Backends.

- Anmeldeinformationsbasiert: Dieser Modus erfordert ausreichende Berechtigungen für das ONTAP-Backend. Es wird empfohlen, ein Konto zu verwenden, das einer vordefinierten Sicherheits-Login-Rolle zugeordnet ist, z. B. `admin` oder `vsadmin`, um maximale Kompatibilität mit ONTAP-Versionen sicherzustellen.
- Zertifikatsbasiert: Für diesen Modus ist ein Zertifikat auf dem Backend installiert, damit Trident mit einem ONTAP-Cluster kommunizieren kann. Hier muss die Backend-Definition Base64-kodierte Werte des Client-Zertifikats, des Schlüssels und des vertrauenswürdigen CA-Zertifikats enthalten, sofern verwendet (empfohlen).

Sie können vorhandene Back-Ends aktualisieren, um zwischen auf Anmeldeinformationen basierenden und zertifikatbasierten Methoden zu verschieben. Es wird jedoch immer nur eine Authentifizierungsmethode unterstützt. Um zu einer anderen Authentifizierungsmethode zu wechseln, müssen Sie die vorhandene Methode von der Backend-Konfiguration entfernen.



Wenn Sie versuchen, **sowohl Anmeldeinformationen als auch Zertifikate** bereitzustellen, schlägt die Backend-Erstellung mit einem Fehler fehl, dass mehr als eine Authentifizierungsmethode in der Konfigurationsdatei angegeben wurde.

Aktivieren Sie die Anmeldeinformationsbasierte Authentifizierung

Für die Kommunikation mit dem ONTAP-Back-End ist die Zugangsdaten an einen Administrator mit SVM-Umfang/Cluster-Umfang erforderlich Trident. Es wird empfohlen, standardmäßige, vordefinierte Rollen wie `vsadmin`` zu verwenden ``admin`. So wird die Kompatibilität mit zukünftigen ONTAP Versionen sichergestellt, die möglicherweise die FunktionAPIs für zukünftige Trident Versionen offenlegen. Eine benutzerdefinierte Sicherheits-Login-Rolle kann erstellt und mit Trident verwendet werden, wird aber nicht empfohlen.

Eine Beispiel-Back-End-Definition sieht folgendermaßen aus:

YAML

```
---
version: 1
backendName: ExampleBackend
storageDriverName: ontap-nas
managementLIF: 10.0.0.1
dataLIF: 10.0.0.2
svm: svm_nfs
username: vsadmin
password: password
```

JSON

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

Beachten Sie, dass die Backend-Definition der einzige Ort ist, an dem die Anmeldeinformationen im reinen Text gespeichert werden. Nach der Erstellung des Backend werden Benutzernamen/Passwörter mit Base64 codiert und als Kubernetes Secrets gespeichert. Die Erstellung/Aktualisierung eines Backend ist der einzige Schritt, der Kenntnisse der Anmeldeinformationen erfordert. Daher ist dieser Vorgang nur für Administratoren und wird vom Kubernetes-/Storage-Administrator ausgeführt.

Aktivieren Sie die zertifikatbasierte Authentifizierung

Neue und vorhandene Back-Ends können ein Zertifikat verwenden und mit dem ONTAP-Back-End kommunizieren. In der Backend-Definition sind drei Parameter erforderlich.

- **ClientCertificate:** Base64-codierter Wert des Clientzertifikats.
- **ClientPrivateKey:** Base64-kodierte Wert des zugeordneten privaten Schlüssels.
- **Trusted CACertificate:** Base64-codierter Wert des vertrauenswürdigen CA-Zertifikats. Bei Verwendung einer vertrauenswürdigen CA muss dieser Parameter angegeben werden. Dies kann ignoriert werden, wenn keine vertrauenswürdige CA verwendet wird.

Ein typischer Workflow umfasst die folgenden Schritte.

Schritte

1. Erzeugen eines Clientzertifikats und eines Schlüssels. Legen Sie beim Generieren den allgemeinen

Namen (CN) für den ONTAP-Benutzer fest, der sich authentifizieren soll als.

```
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"
```

2. Fügen Sie dem ONTAP-Cluster ein vertrauenswürdigen CA-Zertifikat hinzu. Dies kann möglicherweise bereits vom Storage-Administrator übernommen werden. Ignorieren, wenn keine vertrauenswürdige CA verwendet wird.

```
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. Installieren Sie das Client-Zertifikat und den Schlüssel (von Schritt 1) auf dem ONTAP-Cluster.

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

4. Vergewissern Sie sich, dass die ONTAP-Sicherheits-Anmeldungsrolle die Authentifizierungsmethode unterstützt cert.

```
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>
```

5. Testen Sie die Authentifizierung mithilfe des generierten Zertifikats. <ONTAP Management LIF> und <vServer Name> durch Management-LIF-IP und SVM-Namen ersetzen. Sie müssen sicherstellen, dass für die LIF-Service-Richtlinie auf festgelegt ist 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>'
```

6. Encodieren von Zertifikat, Schlüssel und vertrauenswürdigen CA-Zertifikat mit Base64.

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

7. Erstellen Sie das Backend mit den Werten, die aus dem vorherigen Schritt ermittelt wurden.

```
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...Vaaalllluuuuueeee",
  "clientPrivateKey": "LS0tFaKE...0VaLuES0tLS0K",
  "storagePrefix": "myPrefix_"
}

#Update backend with tridentctl
tridentctl update backend NasBackend -f cert-backend-updated.json -n
trident
```

NAME	STORAGE DRIVER	UUID
NasBackend	ontap-nas	98e19b74-aec7-4a3d-8dcf-128e5033b214

Aktualisieren Sie Authentifizierungsmethoden, oder drehen Sie die Anmeldedaten

Sie können ein vorhandenes Backend aktualisieren, um eine andere Authentifizierungsmethode zu verwenden oder ihre Anmeldedaten zu drehen. Das funktioniert auf beide Arten: Back-Ends, die einen Benutzernamen/ein Passwort verwenden, können aktualisiert werden, um Zertifikate zu verwenden; Back-Ends, die Zertifikate verwenden, können auf Benutzername/Passwort-basiert aktualisiert werden. Dazu müssen Sie die vorhandene Authentifizierungsmethode entfernen und die neue Authentifizierungsmethode hinzufügen. Verwenden Sie dann die aktualisierte Datei Backend.json, die die erforderlichen Parameter enthält `tridentctl update backend`.

```
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": "password",
  "storagePrefix": "myPrefix_"
}

#Update backend with tridentctl
tridentctl update backend NasBackend -f cert-backend-updated.json -n
trident
```

NAME	STORAGE DRIVER	UUID
NasBackend	ontap-nas	98e19b74-aec7-4a3d-8dcf-128e5033b214



Bei der Änderung von Passwörtern muss der Speicheradministrator das Kennwort für den Benutzer auf ONTAP aktualisieren. Auf diese Weise folgt ein Backend-Update. Beim Drehen von Zertifikaten können dem Benutzer mehrere Zertifikate hinzugefügt werden. Das Backend wird dann aktualisiert und verwendet das neue Zertifikat. Danach kann das alte Zertifikat aus dem ONTAP Cluster gelöscht werden.

Durch die Aktualisierung eines Backend wird der Zugriff auf Volumes, die bereits erstellt wurden, nicht unterbrochen, und auch die danach erstellten Volume-Verbindungen werden beeinträchtigt. Ein erfolgreiches Backend-Update zeigt an, dass Trident mit dem ONTAP Back-End kommunizieren und zukünftige Volume-Operationen verarbeiten kann.

Benutzerdefinierte ONTAP-Rolle für Trident erstellen

Sie können eine ONTAP-Cluster-Rolle mit minimaler Privileges erstellen, sodass Sie nicht die ONTAP-Administratorrolle verwenden müssen, um Vorgänge in Trident auszuführen. Wenn Sie den Benutzernamen in eine Trident-Back-End-Konfiguration aufnehmen, verwendet Trident die ONTAP-Cluster-Rolle, die Sie für die Durchführung der Vorgänge erstellt haben.

Weitere Informationen zum Erstellen benutzerdefinierter Trident-Rollen finden Sie unter ["Trident Custom-Role Generator"](#).

Verwenden der ONTAP CLI

1. Erstellen Sie eine neue Rolle mit dem folgenden Befehl:

```
security login role create <role_name\> -cmddirname "command" -access all  
-vserver <svm_name\>
```

2. Erstellen Sie einen Benutzernamen für den Trident-Benutzer:

```
security login create -username <user_name\> -application ontapi  
-authmethod <password\> -role <name_of_role_in_step_1\> -vserver  
<svm_name\> -comment "user_description"
```

3. Ordnen Sie die Rolle dem Benutzer zu:

```
security login modify username <user_name\> -vserver <svm_name\> -role  
<role_name\> -application ontapi -application console -authmethod  
<password\>
```

Verwenden Von System Manager

Führen Sie die folgenden Schritte im ONTAP System Manager durch:

1. Erstellen Sie eine benutzerdefinierte Rolle:

- a. Um eine benutzerdefinierte Rolle auf Cluster-Ebene zu erstellen, wählen Sie **Cluster > Einstellungen** aus.

(Oder) um eine benutzerdefinierte Rolle auf SVM-Ebene zu erstellen, wählen Sie **Storage > Storage VMs > > required svm Einstellungen > Benutzer und Rollen** aus.

- b. Wählen Sie das Pfeilsymbol (→) neben **Users and Roles**.
- c. Wählen Sie unter **Rollen +Hinzufügen** aus.
- d. Definieren Sie die Regeln für die Rolle und klicken Sie auf **Speichern**.

2. Rolle dem Trident-Benutzer zuordnen: + Führen Sie auf der Seite **Benutzer und Rollen** folgende Schritte aus:

- a. Wählen Sie unter **Benutzer** das Symbol Hinzufügen +.
- b. Wählen Sie den gewünschten Benutzernamen aus, und wählen Sie im Dropdown-Menü für **Rolle** eine Rolle aus.
- c. Klicken Sie Auf **Speichern**.

Weitere Informationen finden Sie auf den folgenden Seiten:

- ["Benutzerdefinierte Rollen für die Administration von ONTAP"](#) Oder ["Definieren benutzerdefinierter Rollen"](#)
- ["Arbeiten Sie mit Rollen und Benutzern"](#)

Management der NFS-Exportrichtlinien

Trident verwendet NFS-Exportrichtlinien, um den Zugriff auf die von ihm bereitstehenden Volumes zu kontrollieren.

Trident bietet zwei Optionen für die Arbeit mit Exportrichtlinien:

- Trident kann die Exportrichtlinie selbst dynamisch managen. In diesem Betriebsmodus gibt der Storage-Administrator eine Liste von CIDR-Blöcken an, die zulässige IP-Adressen darstellen. Trident fügt der Exportrichtlinie automatisch zum Veröffentlichungszeitpunkt anwendbare Node-IPs hinzu, die in diesen Bereichen fallen. Wenn keine CIDRs angegeben werden, werden alternativ alle global scoped Unicast-IPs, die auf dem Knoten gefunden werden, auf dem das Volume veröffentlicht wird, zur Exportrichtlinie hinzugefügt.
- Storage-Administratoren können eine Exportrichtlinie erstellen und Regeln manuell hinzufügen. Trident verwendet die standardmäßige Exportrichtlinie, es sei denn, in der Konfiguration ist ein anderer Name für die Exportrichtlinie angegeben.

Dynamisches Managen von Exportrichtlinien

Trident bietet die Möglichkeit, Richtlinien für den Export für ONTAP Back-Ends dynamisch zu managen. So kann der Storage-Administrator einen zulässigen Adressraum für Worker-Node-IPs festlegen, anstatt explizite Regeln manuell zu definieren. Dies vereinfacht das Management von Exportrichtlinien erheblich. Änderungen der Exportrichtlinie erfordern keine manuellen Eingriffe des Storage-Clusters mehr. Dies hilft darüber hinaus, den Zugriff auf das Storage-Cluster nur auf Arbeitsknoten zu beschränken, die Volumes mounten und IPs im angegebenen Bereich haben. Dies unterstützt ein granulares und automatisiertes Management.



Verwenden Sie keine Network Address Translation (NAT), wenn Sie dynamische Exportrichtlinien verwenden. Bei NAT erkennt der Speicher-Controller die Frontend-NAT-Adresse und nicht die tatsächliche IP-Host-Adresse, so dass der Zugriff verweigert wird, wenn in den Exportregeln keine Übereinstimmung gefunden wird.



Im Trident 24.10 `ontap-nas` funktioniert der Speichertreiber weiterhin wie in den früheren Versionen; für den ONTAP-nas-Treiber wurde keine Änderung vorgenommen. Nur der `ontap-nas-economy` Storage-Treiber verfügt in Trident 24.10 über eine Volume-basierte granulare Zugriffssteuerung.

Beispiel

Es müssen zwei Konfigurationsoptionen verwendet werden. Hier ist eine Beispiel-Backend-Definition:

```
---
version: 1
storageDriverName: ontap-nas-economy
backendName: ontap_nas_auto_export
managementLIF: 192.168.0.135
svm: svm1
username: vsadmin
password: password
autoExportCIDRs:
- 192.168.0.0/24
autoExportPolicy: true
```



Wenn Sie diese Funktion verwenden, müssen Sie sicherstellen, dass für die Root-Verbindung in Ihrer SVM eine zuvor erstellte Exportrichtlinie mit einer Exportregel vorhanden ist, die den CIDR-Block des Nodes zulässt (z. B. die standardmäßige Exportrichtlinie). Folgen Sie stets den von NetApp empfohlenen Best Practices, um eine SVM für Trident zu zuweisen.

Hier ist eine Erklärung, wie diese Funktion funktioniert, anhand des obigen Beispiels:

- `autoExportPolicy` Ist auf eingestellt `true`. Das zeigt an, dass Trident für jedes mit diesem Backend für die SVM bereitgestellte Volume eine Exportrichtlinie erstellt `svm1` und das Hinzufügen und Löschen von Regeln mithilfe von Adressblöcken handhabt `autoexportCIDRs`. Bis ein Volume mit einem Node verbunden ist, verwendet das Volume eine leere Exportrichtlinie ohne Regeln, um unerwünschten Zugriff auf dieses Volume zu verhindern. Wenn ein Volume auf einem Node veröffentlicht wird, erstellt Trident eine Exportrichtlinie mit demselben Namen wie der zugrunde liegende `qtree`, der die Node-IP innerhalb des angegebenen CIDR-Blocks enthält. Diese IPs werden auch zu der von der übergeordneten FlexVol verwendeten Exportrichtlinie hinzugefügt.
 - Beispiel:
 - Back-End UUID 403b5326-8482-40db-96d0-d83fb3f4daec
 - `autoExportPolicy` Stellen Sie auf ein `true`
 - Speicherpräfix `trident`
 - PVC UUID a79bcf5f-7b6d-4a40-9876-e2551f159c1c
 - Qtree namens `Trident_pvc_a79bcf5f_7b6d_4a40_9876_e2551f159c1c` erstellt eine Exportrichtlinie für die FlexVol namens `trident-403b5326-8482-40db96d0-d83fb3f4daec`, eine Exportrichtlinie für den genannten `qtree` `trident_pvc_a79bcf5f_7b6d_4a40_9876_e2551f159c1c` und eine leere Exportrichtlinie mit dem Namen `trident_empty` auf der SVM. Die Regeln für die FlexVol-Exportrichtlinie stellen eine Überlagerung sämtlicher Regeln dar, die in den `qtree` Exportrichtlinien enthalten sind. Die leere Exportrichtlinie wird von allen Volumes wiederverwendet, die nicht angehängt sind.
- `autoExportCIDRs` Enthält eine Liste von Adressblöcken. Dieses Feld ist optional und standardmäßig `[„0.0.0.0/0“, „:/0“]`. Wenn nicht definiert, fügt Trident alle global scoped Unicast-Adressen, die auf den Worker-Knoten mit Publikationen gefunden wurden, hinzu.

In diesem Beispiel wird der `192.168.0.0/24` Adressraum angegeben. Das gibt an, dass Kubernetes-Node-IPs, die mit Publikationen innerhalb dieses Adressbereichs liegen, zur von Trident erstellten Exportrichtlinie hinzugefügt werden. Wenn Trident einen Knoten registriert, auf dem es ausgeführt wird, ruft es die IP-Adressen des Knotens ab und prüft diese anhand der in bereitgestellten Adressblöcke `autoExportCIDRs`. Nach dem Filtern der IPs erstellt Trident zum Zeitpunkt der Veröffentlichung die Exportrichtlinien für die Client-IPs für den Knoten, auf dem er veröffentlicht wird.

Sie können und `autoExportCIDRs` für Back-Ends aktualisieren `autoExportPolicy`, nachdem Sie sie erstellt haben. Sie können neue CIDRs für ein Backend anhängen, das automatisch verwaltet wird oder vorhandene CIDRs löschen. Beim Löschen von CIDRs Vorsicht walten lassen, um sicherzustellen, dass vorhandene Verbindungen nicht unterbrochen werden. Sie können auch für ein Backend deaktivieren `autoExportPolicy` und auf eine manuell erstellte Exportrichtlinie zurückgreifen. Dazu muss der Parameter in Ihrer Backend-Konfiguration festgelegt `exportPolicy` werden.

Nachdem Trident ein Backend erstellt oder aktualisiert hat, können Sie das Backend mit oder der entsprechenden `tridentbackend` CRD überprüfen `tridentctl`:

```
./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
```

Wenn ein Node entfernt wird, überprüft Trident alle Exportrichtlinien, um die dem Node entsprechenden Zugriffsregeln zu entfernen. Indem Trident diese Node-IP aus den Exportrichtlinien der Managed Back-Ends entfernt, verhindert es abnormale Mounts, sofern diese IP nicht von einem neuen Node im Cluster wiederverwendet wird.

Bei zuvor vorhandenen Back-Ends wird durch die Aktualisierung des Backend mit `tridentctl update backend` sichergestellt, dass Trident die Exportrichtlinien automatisch verwaltet. Dadurch werden zwei neue Export-Richtlinien erstellt, die nach der UUID und dem qtree-Namen des Backends benannt sind, wenn sie benötigt werden. Volumes, die auf dem Backend vorhanden sind, verwenden die neu erstellten Exportrichtlinien, nachdem sie abgehängt und wieder gemountet wurden.



Wenn Sie ein Backend mit automatisch gemanagten Exportrichtlinien löschen, wird die dynamisch erstellte Exportrichtlinie gelöscht. Wenn das Backend neu erstellt wird, wird es als neues Backend behandelt und erzeugt eine neue Exportrichtlinie.

Wenn die IP-Adresse eines aktiven Node aktualisiert wird, müssen Sie den Trident Pod auf dem Node neu starten. Trident aktualisiert dann die Exportrichtlinie für Back-Ends, die es verwaltet, um diese IP-Änderung widerzuspiegeln.

Vorbereitung zur Bereitstellung von SMB Volumes

Mit etwas zusätzlicher Vorbereitung können Sie SMB-Volumes mit Treibern bereitstellen `ontap-nas`.



Sie müssen auf der SVM sowohl NFS- als auch SMB/CIFS-Protokolle konfigurieren, um ein SMB-Volume für ONTAP vor Ort zu erstellen `ontap-nas-economy`. Ist eines dieser Protokolle nicht konfiguriert, schlägt die Erstellung von SMB Volumes fehl.



autoExportPolicy Wird für SMB-Volumes nicht unterstützt.

Bevor Sie beginnen

Bevor Sie SMB-Volumes bereitstellen können, müssen Sie über Folgendes verfügen:

- Kubernetes-Cluster mit einem Linux-Controller-Knoten und mindestens einem Windows-Worker-Node, auf dem Windows Server 2022 ausgeführt wird. Trident unterstützt nur SMB Volumes, die in Pods gemountet sind, die nur auf Windows Nodes ausgeführt werden.
- Mindestens ein Trident-Schlüssel, der Ihre Active Directory-Anmeldeinformationen enthält. So generieren Sie ein Geheimnis smbcreds:

```
kubectl create secret generic smbcreds --from-literal username=user  
--from-literal password='password'
```

- Ein CSI-Proxy, der als Windows-Dienst konfiguriert ist. Informationen zum Konfigurieren `csi-proxy` von finden Sie unter "[GitHub: CSI-Proxy](#)" oder "[GitHub: CSI Proxy für Windows](#)" für Kubernetes-Nodes, die unter Windows ausgeführt werden.

Schritte

1. Bei On-Premises-ONTAP können Sie optional eine SMB-Freigabe oder Trident eine für Sie erstellen.



SMB-Freigaben sind für Amazon FSX for ONTAP erforderlich.

Sie können die SMB-Administratorfreigaben auf zwei Arten erstellen, entweder mit dem "[Microsoft Management Console](#)" Snap-in für freigegebene Ordner oder mit der ONTAP-CLI. So erstellen Sie SMB-Freigaben mithilfe der ONTAP-CLI:

- a. Erstellen Sie bei Bedarf die Verzeichnispfadstruktur für die Freigabe.

Der `vserver cifs share create` Befehl überprüft den in der Option `-path` angegebenen Pfad während der Erstellung von Freigaben. Wenn der angegebene Pfad nicht vorhanden ist, schlägt der Befehl fehl.

- b. Erstellen einer mit der angegebenen SVM verknüpften SMB-Freigabe:

```
vserver cifs share create -vserver vserver_name -share-name  
share_name -path path [-share-properties share_properties,...]  
[other_attributes] [-comment text]
```

- c. Vergewissern Sie sich, dass die Freigabe erstellt wurde:

```
vserver cifs share show -share-name share_name
```



Weitere Informationen finden Sie unter "[Erstellen Sie eine SMB-Freigabe](#)".

2. Beim Erstellen des Backend müssen Sie Folgendes konfigurieren, um SMB-Volumes festzulegen. Für alle

FSX für ONTAP Backend-Konfigurationsoptionen, siehe ["FSX für ONTAP Konfigurationsoptionen und Beispiele"](#).

Parameter	Beschreibung	Beispiel
smbShare	Sie können eine der folgenden Optionen angeben: Den Namen einer SMB-Freigabe, die mit der Microsoft Verwaltungskonsole oder der ONTAP-CLI erstellt wurde, einen Namen, über den Trident die SMB-Freigabe erstellen kann, oder Sie können den Parameter leer lassen, um den Zugriff auf gemeinsame Freigaben auf Volumes zu verhindern. Dieser Parameter ist für On-Premises-ONTAP optional. Dieser Parameter ist für Amazon FSX for ONTAP-Back-Ends erforderlich und darf nicht leer sein.	smb-share
nasType	Muss auf. gesetzt werden smb Wenn Null, wird standardmäßig auf nfs.	smb
securityStyle	Sicherheitstyp für neue Volumes. Muss für SMB Volumes auf oder mixed gesetzt werden ntfs.	ntfs Oder mixed für SMB Volumes
unixPermissions	Modus für neue Volumes. Muss für SMB Volumes leer gelassen werden.	“ ”

ONTAP-NAS-Konfigurationsoptionen und Beispiele

Lernen Sie, wie Sie ONTAP NAS-Treiber mit Ihrer Trident-Installation erstellen und verwenden. Dieser Abschnitt enthält Beispiele und Details zur Back-End-Konfiguration für die Zuordnung von Back-Ends zu StorageClasses.

Back-End-Konfigurationsoptionen

Die Back-End-Konfigurationsoptionen finden Sie in der folgenden Tabelle:

Parameter	Beschreibung	Standard
version		Immer 1
storageDriverName	Name des Speichertreibers	„ontap-nas“, „ontap-nas-Economy“, „ontap-nas-flexgroup“, „ontap-san“, „ontap-san-Economy“
backendName	Benutzerdefinierter Name oder das Storage-Backend	Treibername + „_“ + DatenLIF

Parameter	Beschreibung	Standard
managementLIF	IP-Adresse eines Clusters oder einer SVM-Management-LIF Ein vollständig qualifizierter Domain-Name (FQDN) kann angegeben werden. Kann so eingestellt werden, dass IPv6-Adressen verwendet werden, wenn Trident mit dem IPv6-Flag installiert wurde. IPv6-Adressen müssen in eckigen Klammern definiert werden, z. B. [28e8:d9fb:a825:b7bf:69a8:d02f:9e7b:3555]. Informationen über die nahtlose MetroCluster-Umschaltung finden Sie im [mcc-best] .	„10.0.0.1“, „[2001:1234:abcd::fefe]“
dataLIF	IP-Adresse des LIF-Protokolls. Wir empfehlen die Angabe dataLIF. Falls nicht bereitgestellt, ruft Trident die Daten-LIFs von der SVM ab. Sie können einen vollständig qualifizierten Domänennamen (FQDN) angeben, der für die NFS-Mount-Vorgänge verwendet werden soll. Damit können Sie ein Round-Robin-DNS zum Load-Balancing über mehrere Daten-LIFs erstellen. Kann nach der Anfangseinstellung geändert werden. Siehe . Kann so eingestellt werden, dass IPv6-Adressen verwendet werden, wenn Trident mit dem IPv6-Flag installiert wurde. IPv6-Adressen müssen in eckigen Klammern definiert werden, z. B. [28e8:d9fb:a825:b7bf:69a8:d02f:9e7b:3555]. Für MetroCluster weglassen. Siehe [mcc-best] .	Angegebene Adresse oder abgeleitet von SVM, falls nicht angegeben (nicht empfohlen)
svm	Zu verwendende virtuelle Speichermaschine omit für MetroCluster. Siehe [mcc-best] .	Abgeleitet, wenn eine SVM managementLIF angegeben wird
autoExportPolicy	Aktivieren Sie die automatische Erstellung von Exportrichtlinien und aktualisieren Sie [Boolean]. Mithilfe der autoExportPolicy Optionen und autoExportCIDRs kann Trident Exportrichtlinien automatisch managen.	Falsch
autoExportCIDRs	Liste der CIDRs, nach denen die Node-IPs von Kubernetes gegen gefiltert werden sollen, wenn autoExportPolicy aktiviert ist. Mithilfe der autoExportPolicy Optionen und autoExportCIDRs kann Trident Exportrichtlinien automatisch managen.	[„0.0.0.0/0“, „:/0“]
labels	Satz willkürlicher JSON-formatierter Etiketten für Volumes	“ ”
clientCertificate	Base64-codierter Wert des Clientzertifikats. Wird für zertifikatbasierte Authentifizierung verwendet	“ ”
clientPrivateKey	Base64-kodierte Wert des privaten Client-Schlüssels. Wird für zertifikatbasierte Authentifizierung verwendet	“ ”
trustedCACertificate	Base64-kodierte Wert des vertrauenswürdigen CA-Zertifikats. Optional Wird für zertifikatbasierte Authentifizierung verwendet	“ ”

Parameter	Beschreibung	Standard
username	Benutzername für die Verbindung mit dem Cluster/SVM. Wird für Anmeldeinformationsbasierte verwendet	
password	Passwort für die Verbindung mit dem Cluster/SVM Wird für Anmeldeinformationsbasierte verwendet	
storagePrefix	Das Präfix wird beim Bereitstellen neuer Volumes in der SVM verwendet. Kann nicht aktualisiert werden, nachdem Sie sie festgelegt haben  Bei Verwendung von ONTAP-nas-Economy und einem storagePrefix, das aus 24 oder mehr Zeichen besteht, ist das Storage-Präfix für die qtrees nicht eingebettet, obwohl es sich im Volume-Namen befindet.	trident
aggregate	Aggregat für die Bereitstellung (optional, wenn eingestellt, muss der SVM zugewiesen werden) Für den <code>ontap-nas-flexgroup</code> Treiber wird diese Option ignoriert. Falls nicht, können alle verfügbaren Aggregate verwendet werden, um ein FlexGroup Volume bereitzustellen.  Wenn das Aggregat in einer SVM aktualisiert wird, wird es automatisch in Trident aktualisiert, indem es die SVM abfragt, ohne den Trident Controller neu starten zu müssen. Wenn Sie ein bestimmtes Aggregat in Trident für die Bereitstellung von Volumes konfiguriert haben, wird das Back-End Trident bei der Abfrage des SVM-Aggregats in den Status „Fehlgeschlagen“ verschoben. Sie müssen entweder das Aggregat zu einem auf der SVM vorhandenen Aggregat ändern oder es komplett entfernen, um das Back-End wieder online zu schalten.	“
limitAggregateUsage	Bereitstellung fehlgeschlagen, wenn die Nutzung über diesem Prozentsatz liegt. Gilt nicht für Amazon FSX für ONTAP	„ (nicht standardmäßig durchgesetzt)

Parameter	Beschreibung	Standard
FlexgroupAggregateList	Liste der Aggregate für die Bereitstellung (optional, muss dieser SVM zugewiesen werden, falls festgelegt) Zur Bereitstellung eines FlexGroup Volumes werden alle der SVM zugewiesenen Aggregate verwendet. Unterstützt für den ONTAP-nas-FlexGroup-Speichertreiber.  Bei einer Aktualisierung der Aggregatliste in der SVM wird die Liste automatisch in Trident aktualisiert, indem die SVM abgefragt wird, ohne den Trident Controller neu starten zu müssen. Wenn Sie in Trident eine bestimmte Aggregatliste für die Bereitstellung von Volumes konfiguriert haben und die Aggregatliste umbenannt oder von SVM entfernt wird, wird das Backend in Trident in den Fehlerzustand verschoben, während es das SVM Aggregat abfragt. Sie müssen entweder die Aggregatliste zu einer auf der SVM vorhandenen ändern oder sie komplett entfernen, um das Backend wieder online zu machen.	“
limitVolumeSize	Bereitstellung fehlgeschlagen, wenn die angeforderte Volume-Größe über diesem Wert liegt. Beschränkt darüber hinaus die maximale Größe der Volumes, die es für qtrees managt, und qtreesPerFlexvol ermöglicht die Anpassung der maximalen Anzahl an qtrees pro FlexVol.	„ (nicht standardmäßig durchgesetzt)
debugTraceFlags	Fehler-Flags bei der Fehlerbehebung beheben. Beispiel, {„API“:false, „method“:true} nicht verwenden debugTraceFlags, es sei denn, Sie beheben die Fehlerbehebung und benötigen einen detaillierten Log Dump.	Null
nasType	Konfiguration der Erstellung von NFS- oder SMB-Volumes Optionen sind nfs, smb oder Null. Einstellung auf null setzt standardmäßig auf NFS-Volumes.	nfs

Parameter	Beschreibung	Standard
nfsMountOptions	Kommagetrennte Liste von NFS-Mount-Optionen. Die Mount-Optionen für persistente Kubernetes-Volumes werden normalerweise in Storage-Klassen angegeben. Wenn jedoch keine Mount-Optionen in einer Storage-Klasse angegeben sind, verwendet Trident die Mount-Optionen, die in der Konfigurationsdatei des Storage-Backends angegeben sind. Wenn in der Storage-Klasse oder in der Konfigurationsdatei keine Mount-Optionen angegeben sind, legt Trident keine Mount-Optionen auf einem zugeordneten persistenten Volume fest.	“
qtreesPerFlexvol	Maximale Ques pro FlexVol, muss im Bereich [50, 300] liegen	„200“
smbShare	Sie können eine der folgenden Optionen angeben: Den Namen einer SMB-Freigabe, die mit der Microsoft Verwaltungskonsole oder der ONTAP-CLI erstellt wurde, einen Namen, über den Trident die SMB-Freigabe erstellen kann, oder Sie können den Parameter leer lassen, um den Zugriff auf gemeinsame Freigaben auf Volumes zu verhindern. Dieser Parameter ist für On-Premises-ONTAP optional. Dieser Parameter ist für Amazon FSX for ONTAP-Back-Ends erforderlich und darf nicht leer sein.	smb-share
useREST	Boolescher Parameter zur Verwendung von ONTAP REST-APIs. useREST Bei Einstellung auf true verwendet Trident ONTAP REST APIs zur Kommunikation mit dem Backend; bei Einstellung auf false verwendet Trident ONTAP ZAPI Aufrufe zur Kommunikation mit dem Backend. Diese Funktion erfordert ONTAP 9.11.1 und höher. Darüber hinaus muss die verwendete ONTAP-Anmelderolle Zugriff auf die Anwendung haben <code>ontap</code> . Dies wird durch die vordefinierten <code>vsadmin</code> Rollen und <code>cluster-admin</code> erreicht. Ab Trident 24.06-Version und ONTAP 9.15.1 oder höher useREST ist standardmäßig auf eingestellt true; ändern Sie useREST zu false ONTAP-ZAPI-Aufrufe verwenden.	true Für ONTAP 9.15.1 oder höher, andernfalls false.
limitVolumePoolSize	Maximale anforderbare FlexVol-Größe bei Verwendung von Qtrees im ONTAP-nas-Economy Backend.	„ (nicht standardmäßig durchgesetzt)
denyNewVolumePools	Schränkt das <code>ontap-nas-economy</code> Erstellen neuer FlexVol Volumes für Back-Ends ein, um ihre qtrees zu enthalten Zur Bereitstellung neuer PVS werden nur vorbestehende FlexVols verwendet.	

Back-End-Konfigurationsoptionen für die Bereitstellung von Volumes

Mit diesen Optionen können Sie die Standardbereitstellung im Abschnitt der Konfiguration steuern `defaults`. Ein Beispiel finden Sie unten in den Konfigurationsbeispielen.

Parameter	Beschreibung	Standard
<code>spaceAllocation</code>	Platzzuweisung für Qtrees	„Wahr“
<code>spaceReserve</code>	Modus für Speicherplatzreservierung; „none“ (Thin) oder „Volume“ (Thick)	„Keine“
<code>snapshotPolicy</code>	Die Snapshot-Richtlinie zu verwenden	„Keine“
<code>qosPolicy</code>	QoS-Richtliniengruppe zur Zuweisung für erstellte Volumes Wählen Sie eine der <code>qosPolicy</code> oder <code>adaptiveQosPolicy</code> pro Storage Pool/Backend	“
<code>adaptiveQosPolicy</code>	Adaptive QoS-Richtliniengruppe mit Zuordnung für erstellte Volumes Wählen Sie eine der <code>qosPolicy</code> oder <code>adaptiveQosPolicy</code> pro Storage Pool/Backend. Nicht unterstützt durch <code>ontap-nas</code> -Ökonomie	“
<code>snapshotReserve</code>	Prozentsatz des für Snapshots reservierten Volumes	„0“, wenn <code>snapshotPolicy</code> „keine“ ist, andernfalls „“
<code>splitOnClone</code>	Teilen Sie einen Klon bei der Erstellung von seinem übergeordneten Objekt auf	„Falsch“
<code>encryption</code>	Aktivieren Sie NetApp Volume Encryption (NVE) auf dem neuen Volume, Standardeinstellung ist <code>false</code> . NVE muss im Cluster lizenziert und aktiviert sein, damit diese Option verwendet werden kann. Wenn auf dem Backend NAE aktiviert ist, wird jedes in Trident bereitgestellte Volume NAE aktiviert. Weitere Informationen finden Sie unter " Funktionsweise von Trident mit NVE und NAE ".	„Falsch“
<code>tieringPolicy</code>	Tiering-Richtlinie, die zu „keinen“ verwendet wird	„Nur snapshot“ für eine SVM-DR-Konfiguration vor ONTAP 9.5
<code>unixPermissions</code>	Modus für neue Volumes	„777“ für NFS Volumes; leer (nicht zutreffend) für SMB Volumes
<code>snapshotDir</code>	Steuert den Zugriff auf das <code>.snapshot</code> Verzeichnis	„Wahr“ für NFSv4 „falsch“ für NFSv3
<code>exportPolicy</code>	Zu verwendende Exportrichtlinie	„Standard“
<code>securityStyle</code>	Sicherheitstyp für neue Volumes. NFS-Unterstützung <code>mixed</code> und <code>unix</code> -Sicherheitsstile. SMB-Unterstützung <code>mixed</code> und <code>ntfs</code> Sicherheitsstile.	NFS-Standard ist <code>unix</code> . SMB-Standard ist <code>ntfs</code> .
<code>nameTemplate</code>	Vorlage zum Erstellen benutzerdefinierter Volume-Namen.	“



Für die Verwendung von QoS-Richtliniengruppen mit Trident ist ONTAP 9.8 oder höher erforderlich. Sie sollten eine nicht gemeinsam genutzte QoS-Richtliniengruppe verwenden und sicherstellen, dass die Richtliniengruppe auf jede Komponente einzeln angewendet wird. Eine Shared-QoS-Richtliniengruppe erzwingt die Obergrenze für den Gesamtdurchsatz aller Workloads.

Beispiele für die Volume-Bereitstellung

Hier ein Beispiel mit definierten Standardwerten:

```
---
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'
```

Für `ontap-nas` und `ontap-nas-flexgroups` verwendet Trident jetzt eine neue Berechnung, um sicherzustellen, dass die FlexVol korrekt mit der Snapshot Reserve Prozentsatz und PVC-Größe ist. Wenn der Benutzer eine PVC anfordert, erstellt Trident mithilfe der neuen Berechnung die ursprüngliche FlexVol mit mehr Speicherplatz. Diese Berechnung stellt sicher, dass der Benutzer den beschreibbaren Speicherplatz erhält, für den er in der PVC benötigt wird, und nicht weniger Speicherplatz als der angeforderte. Vor Version 2.07, wenn der Benutzer eine PVC anfordert (z. B. 5 gib), bei der SnapshotReserve auf 50 Prozent, erhalten sie nur 2,5 gib schreibbaren Speicherplatz. Der Grund dafür ist, dass der Benutzer das gesamte Volume angefordert hat und einen prozentualen Anteil davon darstellt. `snapshotReserve` Bei Trident 21.07 fordert der Benutzer den beschreibbaren Speicherplatz an, und Trident definiert die `snapshotReserve` Zahl als Prozentsatz des gesamten Volumes. Dies gilt nicht für `ontap-nas-economy`. Im folgenden Beispiel sehen Sie, wie das funktioniert:

Die Berechnung ist wie folgt:

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

Für die `snapshotReserve = 50 %`, und die PVC-Anfrage = 5 gib, beträgt die Gesamtgröße des Volumes $2/5 = 10$ gib, und die verfügbare Größe beträgt 5 gib. Dies entspricht dem, was der Benutzer in der PVC-Anfrage angefordert hat. Der `volume show` Befehl sollte die Ergebnisse ähnlich wie in diesem Beispiel anzeigen:

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.							

Vorhandene Back-Ends von vorherigen Installationen stellen Volumes wie oben beschrieben beim Upgrade von Trident bereit. Bei Volumes, die Sie vor dem Upgrade erstellt haben, sollten Sie die Größe ihrer Volumes entsprechend der zu beobachtenden Änderung anpassen. Ein Beispiel: Eine PVC mit 2 gib und einer früheren Version `snapshotReserve=50` führte zu einem Volume, das 1 gib schreibbaren Speicherplatz bereitstellt. Wenn Sie die Größe des Volumes auf 3 gib ändern, z. B. stellt die Applikation auf einem 6 gib an beschreibbarem Speicherplatz bereit.

Minimale Konfigurationsbeispiele

Die folgenden Beispiele zeigen grundlegende Konfigurationen, bei denen die meisten Parameter standardmäßig belassen werden. Dies ist der einfachste Weg, ein Backend zu definieren.



Wenn Sie Amazon FSX auf NetApp ONTAP mit Trident verwenden, empfiehlt es sich, DNS-Namen für LIFs anstelle von IP-Adressen anzugeben.

Beispiel für die NAS-Ökonomie von ONTAP

```
---
version: 1
storageDriverName: ontap-nas-economy
managementLIF: 10.0.0.1
dataLIF: 10.0.0.2
svm: svm_nfs
username: vsadmin
password: password
```

Beispiel für ONTAP NAS FlexGroup

```
---
version: 1
storageDriverName: ontap-nas-flexgroup
managementLIF: 10.0.0.1
dataLIF: 10.0.0.2
svm: svm_nfs
username: vsadmin
password: password
```

Beispiel: MetroCluster

Sie können das Backend konfigurieren, um zu vermeiden, dass die Backend-Definition nach Umschaltung und Switchback während manuell aktualisiert ["SVM-Replizierung und Recovery"](#) werden muss.

Geben Sie für ein nahtloses Switchover und Switchback die SVM mit an managementLIF und lassen Sie die Parameter und svm weg dataLIF. Beispiel:

```
---
version: 1
storageDriverName: ontap-nas
managementLIF: 192.168.1.66
username: vsadmin
password: password
```

Beispiel: SMB Volumes

```
---
version: 1
backendName: ExampleBackend
storageDriverName: ontap-nas
managementLIF: 10.0.0.1
nasType: smb
securityStyle: ntfs
unixPermissions: ""
dataLIF: 10.0.0.2
svm: svm_nfs
username: vsadmin
password: password
```

Beispiel für die zertifikatbasierte Authentifizierung

Dies ist ein minimales Beispiel für die Backend-Konfiguration. `clientCertificate`, `clientPrivateKey` und `trustedCACertificate` (optional, wenn vertrauenswürdige CA verwendet wird) werden eingetragen `backend.json` und nehmen die base64-kodierten Werte des Clientzertifikats, des privaten Schlüssels und des vertrauenswürdigen CA-Zertifikats an.

```
---
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_
```

Beispiel für eine Richtlinie für den automatischen Export

Dieses Beispiel zeigt, wie Sie Trident anweisen können, dynamische Exportrichtlinien zu verwenden, um die Exportrichtlinie automatisch zu erstellen und zu verwalten. Dies funktioniert für die `ontap-nas-flexgroup`-Treiber gleich `ontap-nas-economy`.

```
---
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: password
nfsMountOptions: nfsvers=4
```

Beispiel für IPv6-Adressen

Dieses Beispiel zeigt managementLIF die Verwendung einer IPv6-Adresse.

```
---
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: password
```

Amazon FSX für ONTAP mit SMB-Volumes – Beispiel

Der smbShare Parameter ist für FSX for ONTAP mit SMB-Volumes erforderlich.

```
---
version: 1
backendName: SMBBackend
storageDriverName: ontap-nas
managementLIF: example.mgmt.fqdn.aws.com
nasType: smb
dataLIF: 10.0.0.15
svm: nfs_svm
smbShare: smb-share
clientCertificate: ZXR0ZXJwYXB...ICMgJ3BhcGVyc2
clientPrivateKey: vciwKIyAgZG...0cnksIGRlc2NyaX
trustedCACertificate: zcyBbaG...b3Igb3duIGNsYXNz
storagePrefix: myPrefix_
```

Back-End-Konfigurationsbeispiel mit nameTemplate

```
---
version: 1
storageDriverName: ontap-nas
backendName: ontap-nas-backend
managementLIF: <ip address>
svm: svm0
username: <admin>
password: <password>
defaults: {
  "nameTemplate":
    "{{.volume.Name}}_{{.labels.cluster}}_{{.volume.Namespace}}_{{.volume.R
    equestName}}"
},
"labels": {"cluster": "ClusterA", "PVC":
  "{{.volume.Namespace}}_{{.volume.RequestName}}"}
}
```

Beispiele für Back-Ends mit virtuellen Pools

In den unten gezeigten Beispieldateien für die Backend-Definition werden spezifische Standardwerte für alle Speicherpools festgelegt, z. B. `spaceReserve` bei „none“, `spaceAllocation` „false“ und „false encryption“. Die virtuellen Pools werden im Abschnitt Speicher definiert.

Trident legt die Bereitstellungsetiketten im Feld „Kommentare“ fest. Kommentare werden auf FlexVol für oder FlexGroup für `ontap-nas-flexgroup` gesetzt `ontap-nas`. Trident kopiert bei der Bereitstellung alle Labels, die sich in einem virtuellen Pool befinden, auf das Storage-Volume. Storage-Administratoren können Labels je virtuellen Pool definieren und Volumes nach Label gruppieren.

In diesen Beispielen legen einige Speicherpools eigene Werte , `spaceAllocation` und fest `spaceReserve`, und `encryption` einige Pools überschreiben die Standardwerte.

Beispiel: ONTAP NAS

```
---
version: 1
storageDriverName: ontap-nas
managementLIF: 10.0.0.1
svm: svm_nfs
username: admin
password: <password>
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:
  department: legal
  creditpoints: '5000'
  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'
```

Beispiel für ONTAP NAS FlexGroup

```
---
version: 1
storageDriverName: ontap-nas-flexgroup
managementLIF: 10.0.0.1
svm: svm_nfs
username: vsadmin
password: <password>
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'
```

Beispiel für die NAS-Ökonomie von ONTAP

```
---
version: 1
storageDriverName: ontap-nas-economy
managementLIF: 10.0.0.1
svm: svm_nfs
username: vsadmin
password: <password>
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:
  protection: bronze
  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'
```

Back-Ends StorageClasses zuordnen

Die folgenden StorageClass-Definitionen finden Sie unter [Beispiele für Back-Ends mit virtuellen Pools](#). Mit dem `parameters.selector` Feld ruft jede StorageClass ab, welche virtuellen Pools zum Hosten eines Volumes verwendet werden können. Auf dem Volume werden die Aspekte im ausgewählten virtuellen Pool definiert.

- Die `protection-gold` StorageClass wird dem ersten und zweiten virtuellen Pool im Backend zugeordnet `ontap-nas-flexgroup`. Dies sind die einzigen Pools, die Gold-Level-Schutz bieten.

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: protection-gold
provisioner: csi.trident.netapp.io
parameters:
  selector: "protection=gold"
  fsType: "ext4"
```

- Die `protection-not-gold` StorageClass wird dem dritten und vierten virtuellen Pool im Backend zugeordnet `ontap-nas-flexgroup`. Dies sind die einzigen Pools, die Schutz Level nicht Gold bieten.

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: protection-not-gold
provisioner: csi.trident.netapp.io
parameters:
  selector: "protection!=gold"
  fsType: "ext4"
```

- Die `app-mysqldb` StorageClass wird dem vierten virtuellen Pool im Backend zugeordnet `ontap-nas`. Dies ist der einzige Pool, der Storage-Pool-Konfiguration für `mysqldb`-Typ-App bietet.

```

apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: app-mysqldb
provisioner: csi.trident.netapp.io
parameters:
  selector: "app=mysqldb"
  fsType: "ext4"

```

- Die `protection-silver-creditpoints-20k` StorageClass wird dem dritten virtuellen Pool im Backend zugeordnet `ontap-nas-flexgroup`. Dies ist der einzige Pool mit Silber-Level-Schutz und 20000 Kreditpunkte.

```

apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: protection-silver-creditpoints-20k
provisioner: csi.trident.netapp.io
parameters:
  selector: "protection=silver; creditpoints=20000"
  fsType: "ext4"

```

- Die `creditpoints-5k` StorageClass wird dem dritten virtuellen Pool im Backend und dem zweiten virtuellen Pool im Backend `ontap-nas-economy` zugeordnet `ontap-nas`. Dies sind die einzigen Poolangebote mit 5000 Kreditpunkten.

```

apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: creditpoints-5k
provisioner: csi.trident.netapp.io
parameters:
  selector: "creditpoints=5000"
  fsType: "ext4"

```

Trident entscheidet, welcher virtuelle Pool ausgewählt wird, und stellt sicher, dass die Speicheranforderungen erfüllt werden.

Nach der Erstkonfiguration aktualisieren `dataLIF`

Sie können die Daten-LIF nach der Erstkonfiguration ändern, indem Sie den folgenden Befehl ausführen, um die neue Backend-JSON-Datei mit aktualisierten Daten-LIF bereitzustellen.

```
tridentctl update backend <backend-name> -f <path-to-backend-json-file-with-updated-dataLIF>
```



Wenn PVCs an einen oder mehrere Pods angeschlossen sind, müssen Sie alle entsprechenden Pods herunterfahren und sie dann wieder zurückbringen, damit die neue logische Daten wirksam werden.

Amazon FSX für NetApp ONTAP

Verwenden Sie Trident mit Amazon FSX für NetApp ONTAP

"[Amazon FSX für NetApp ONTAP](#)" Ist ein vollständig gemanagter AWS Service, mit dem Kunden Filesysteme mit NetApp ONTAP Storage-Betriebssystem starten und ausführen können. Mit FSX für ONTAP können Sie bekannte NetApp Funktionen sowie die Performance und Administration nutzen und gleichzeitig die Einfachheit, Agilität, Sicherheit und Skalierbarkeit beim Speichern von Daten in AWS nutzen. FSX für ONTAP unterstützt ONTAP Dateisystemfunktionen und Administrations-APIs.

Die Integration des Filesystems Amazon FSX for NetApp ONTAP mit Trident stellt sicher, dass Kubernetes-Cluster, die in Amazon Elastic Kubernetes Service (EKS) ausgeführt werden, persistente Block- und dateibasierte Volumes mit ONTAP bereitstellen können.

Ein Dateisystem ist die primäre Ressource in Amazon FSX, analog zu einem ONTAP-Cluster vor Ort. Innerhalb jeder SVM können Sie ein oder mehrere Volumes erstellen, bei denen es sich um Daten-Container handelt, die die Dateien und Ordner im Filesystem speichern. Amazon FSX für NetApp ONTAP wird Data ONTAP als gemanagtes Dateisystem in der Cloud zur Verfügung stellen. Der neue Dateisystemtyp heißt **NetApp ONTAP**.

Durch den Einsatz von Trident mit Amazon FSX for NetApp ONTAP können Sie sicherstellen, dass Kubernetes-Cluster, die im Amazon Elastic Kubernetes Service (EKS) ausgeführt werden, persistente Block- und dateibasierte Volumes bereitstellen können, die von ONTAP unterstützt werden.

Anforderungen

"[Trident-Anforderungen erfüllt](#)" Um FSX for ONTAP mit Trident zu integrieren, benötigen Sie zusätzlich:

- Ein vorhandener Amazon EKS Cluster oder selbstverwalteter Kubernetes-Cluster mit `kubectl` installierter Installation.
- Ein vorhandenes Amazon FSX for NetApp ONTAP-Filesystem und eine Storage Virtual Machine (SVM), die über die Worker-Nodes Ihres Clusters erreichbar ist.
- Worker-Knoten, die für vorbereitet sind "[NFS oder iSCSI](#)".



Stellen Sie sicher, dass Sie die erforderlichen Schritte zur Knotenvorbereitung für Amazon Linux und Ubuntu (Amis) je nach EKS AMI-Typ befolgen "[Amazon Machine Images](#)".

Überlegungen

- SMB Volumes:
 - SMB-Volumes werden nur über den Treiber unterstützt `ontap-nas`.
 - SMB-Volumes werden vom Trident EKS Add-on nicht unterstützt.
 - Trident unterstützt nur SMB Volumes, die in Pods gemountet sind, die nur auf Windows Nodes ausgeführt werden. Weitere Informationen finden Sie unter ["Vorbereitung zur Bereitstellung von SMB Volumes"](#).
- Vor Trident 24.02 konnten auf Amazon FSX-Dateisystemen erstellte Volumes, bei denen automatische Backups aktiviert sind, von Trident nicht gelöscht werden. Um dieses Problem in Trident 24.02 oder höher zu vermeiden, geben Sie `apiKey` in der Backend-Konfigurationsdatei für AWS FSX für ONTAP, `APIRegion` und `AWS secretKey` an `fsxFilesystemID`.



Wenn Sie eine IAM-Rolle als Trident angeben, können Sie die Felder `apiKey` und `secretKey` explizit als Trident auslassen `APIRegion`. Weitere Informationen finden Sie unter ["FSX für ONTAP Konfigurationsoptionen und Beispiele"](#).

Authentifizierung

Trident bietet zwei Authentifizierungsmodi.

- Anmeldeinformationsbasiert (empfohlen): Speichert Anmeldeinformationen sicher in AWS Secrets Manager. Sie können den Benutzer für Ihr Dateisystem oder den für Ihre SVM konfigurierten Benutzer verwenden `fsxadmin` `vsadmin`.



Trident wird voraussichtlich als SVM-Benutzer oder als Benutzer mit einem anderen Namen, der dieselbe Rolle hat, ausgeführt `vsadmin`. Amazon FSX for NetApp ONTAP hat einen `fsxadmin` Benutzer, der den ONTAP-Cluster-Benutzer nur eingeschränkt ersetzt `admin`. Wir empfehlen die Verwendung `vsadmin` mit Trident.

- Zertifikat-basiert: Trident kommuniziert über ein auf Ihrer SVM installiertes Zertifikat mit der SVM auf Ihrem FSX Filesystem.

Weitere Informationen zur Aktivierung der Authentifizierung finden Sie in der Authentifizierung für Ihren Treibertyp:

- ["ONTAP NAS-Authentifizierung"](#)
- ["ONTAP SAN-Authentifizierung"](#)

Getestete Amazon Machine Images (Amis)

Der EKS Cluster unterstützt zwar verschiedene Betriebssysteme, AWS hat jedoch bestimmte Amazon Machine Images (Amis) für Container und EKS optimiert. Die folgenden Amis wurden mit Trident 24.10 getestet.

AMI	NAS	NAS-Economy	San	SAN-Economy
AL2023_x86_64_STANDARD	Ja.	Ja.	Ja.	Ja.
AL2_x86_64	Ja.	Ja.	Ja**	Ja**

BOTTLEROCKET_x86_64	Ja*	Ja.	K. A.	K. A.
AL2023_ARM_64_STANDARD	Ja.	Ja.	Ja.	Ja.
AL2_ARM_64	Ja.	Ja.	Ja**	Ja**
BOTTLEROCKET_ARM_64	Ja*	Ja.	K. A.	K. A.

- *Muss „nolock“ in Mount-Optionen verwenden.
- ** Das PV kann nicht gelöscht werden, ohne den Knoten neu zu starten



Wenn Ihr gewünschtes AMI hier nicht aufgeführt ist, bedeutet dies nicht, dass es nicht unterstützt wird, sondern dass es einfach nicht getestet wurde. Diese Liste dient als Leitfaden für Amis, die bekannt sind zu arbeiten.

Tests durchgeführt mit:

- EKS-Version: 1.30
- Installationsmethode: Helm und als AWS Add-on
- Für NAS wurden sowohl NFSv3 als auch NFSv4.1 getestet.
- Für SAN wurde nur iSCSI getestet, nicht NVMe-of.

Durchgeführte Tests:

- Erstellen: Storage-Klasse, pvc, POD
- Löschen: Pod, pvc (normal, qtree/lun – Economy, NAS mit AWS Backup)

Weitere Informationen

- ["Dokumentation zu Amazon FSX für NetApp ONTAP"](#)
- ["Blogbeitrag zu Amazon FSX für NetApp ONTAP"](#)

IAM-Rolle und AWS Secret erstellen

Sie können Kubernetes-Pods für den Zugriff auf AWS-Ressourcen konfigurieren, indem Sie sich als AWS IAM-Rolle authentifizieren anstatt dafür explizite AWS-Anmeldedaten bereitstellen zu müssen.



Um sich mit einer AWS IAM-Rolle zu authentifizieren, müssen Sie über ein Kubernetes-Cluster mit EKS verfügen.

Erstellen Sie den AWS Secret Manager-Schlüssel

Dieses Beispiel erstellt einen AWS Secret Manager-Schlüssel, um Trident-CSI-Anmeldedaten zu speichern:

```
aws secretsmanager create-secret --name trident-secret --description
"Trident CSI credentials"\
  --secret-string
"{\"username\": \"vsadmin\", \"password\": \"<svmpassword>\"}"
```

IAM-Richtlinie erstellen

In den folgenden Beispielen wird eine IAM-Richtlinie über die AWS-CLI erstellt:

```
aws iam create-policy --policy-name AmazonFSxNCSIDriverPolicy --policy
-document file://policy.json
  --description "This policy grants access to Trident CSI to FSxN and
Secret manager"
```

Richtlinien-JSON-Datei:

```
policy.json:
{
  "Statement": [
    {
      "Action": [
        "fsx:DescribeFileSystems",
        "fsx:DescribeVolumes",
        "fsx:CreateVolume",
        "fsx:RestoreVolumeFromSnapshot",
        "fsx:DescribeStorageVirtualMachines",
        "fsx:UntagResource",
        "fsx:UpdateVolume",
        "fsx:TagResource",
        "fsx>DeleteVolume"
      ],
      "Effect": "Allow",
      "Resource": "*"
    },
    {
      "Action": "secretsmanager:GetSecretValue",
      "Effect": "Allow",
      "Resource": "arn:aws:secretsmanager:<aws-region>:<aws-account-
id>:secret:<aws-secret-manager-name>*"
    }
  ],
  "Version": "2012-10-17"
}
```

Erstellen Sie eine IAM-Rolle für das Dienstkonto

AWS CLI

```
aws iam create-role --role-name trident-controller \  
--assume-role-policy-document file://trust-relationship.json
```

Trust-Relationship.json-Datei:

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    { "Effect": "Allow",  
      "Principal": {  
        "Federated": "arn:aws:iam::<account_id>:oidc-  
provider/<oidc_provider>"  
      },  
      "Action": "sts:AssumeRoleWithWebIdentity",  
      "Condition": {  
        "StringEquals": {  
          "<oidc_provider>:aud": "sts.amazonaws.com",  
          "<oidc_provider>:sub":  
"system:serviceaccount:trident:trident-controller"  
        }  
      }  
    }  
  ]  
}
```

Aktualisieren Sie die folgenden Werte in der trust-relationship.json Datei:

- **<account_id>** - Ihre AWS-Konto-ID
- **<oidc_provider>** - das OIDC Ihres EKS-Clusters. Sie können den oidc_Provider erhalten, indem Sie Folgendes ausführen:

```
aws eks describe-cluster --name my-cluster --query  
"cluster.identity.oidc.issuer"\  
--output text | sed -e "s/^https:\\\\\\"
```

Die IAM-Rolle mit der IAM-Richtlinie verknüpfen:

Nachdem die Rolle erstellt wurde, hängen Sie die Richtlinie (die im obigen Schritt erstellt wurde) mit diesem Befehl an die Rolle an:

```
aws iam attach-role-policy --role-name my-role --policy-arn <IAM policy ARN>
```

Verify OIDC Provider is associated:

Vergewissern Sie sich, dass der OIDC-Anbieter dem Cluster zugeordnet ist. Sie können sie mit diesem Befehl überprüfen:

```
aws iam list-open-id-connect-providers | grep $oidc_id | cut -d "/" -f4
```

Verwenden Sie den folgenden Befehl, um IAM OIDC dem Cluster zuzuordnen:

```
eksctl utils associate-iam-oidc-provider --cluster $cluster_name  
--approve
```

Eksctl

Im folgenden Beispiel wird eine IAM-Rolle für das Dienstkonto in EKS erstellt:

```
eksctl create iamserviceaccount --name trident-controller --namespace  
trident \  
    --cluster <my-cluster> --role-name <AmazonEKS_FSxN_CSI_DriverRole>  
--role-only \  
    --attach-policy-arn <IAM-Policy ARN> --approve
```

Installation Von Trident

Trident optimiert das Amazon FSX für NetApp ONTAP Storage-Management in Kubernetes, damit sich Ihre Entwickler und Administratoren voll und ganz auf den Applikationseinsatz konzentrieren können.

Sie können Trident mit einer der folgenden Methoden installieren:

- Helm
- EKS-Add-on

Wenn Sie die Snapshot-Funktionalität nutzen möchten, installieren Sie das Add-On für den CSI-Snapshot-Controller. Weitere Informationen finden Sie unter ["Snapshot-Funktionalität für CSI-Volumes aktivieren"](#).

Trident über Helm installieren

1. Laden Sie das Trident-Installationspaket herunter

Das Trident-Installationspaket enthält alles, was Sie für die Bereitstellung des Trident-Bedieners und die Installation von Trident benötigen. Laden Sie die neueste Version des Trident-Installers herunter und

extrahieren Sie sie aus dem Abschnitt „Assets“ auf GitHub.

```
wget https://github.com/NetApp/trident/releases/download/v24.10.0/trident-
installer-24.10.0.tar.gz
tar -xf trident-installer-24.10.0.tar.gz
cd trident-installer/helm
```

2. Legen Sie die Werte für **Cloud Provider** und **Cloud Identity** unter Verwendung der folgenden Umgebungsvariablen fest:

Das folgende Beispiel installiert Trident und setzt das `cloud-provider` Flag auf `$CP`, und `cloud-identity` auf `$CI`:

```
helm install trident trident-operator-100.2410.0.tgz --set
cloudProvider="AWS" \

--set cloudIdentity="'eks.amazonaws.com/role-arn:
arn:aws:iam::<accountID>:role/<AmazonEKS_FSxN_CSI_DriverRole>'" \
--namespace trident --create-namespace
```

Mit dem Befehl können `helm list` Sie Installationsdetails wie Name, Namespace, Diagramm, Status, App-Version und Revisionsnummer überprüfen.

```
helm list -n trident
```

NAME	NAMESPACE	REVISION	UPDATED
STATUS	CHART		APP VERSION
trident-operator	trident	1	2024-10-14 14:31:22.463122
+0300 IDT	deployed	trident-operator-100.2410.0	24.10.0

Installieren Sie Trident über das EKS-Add-on

Das Trident EKS Add-on enthält die neuesten Sicherheitspatches und Bug Fixes. Es wurde von AWS für die Zusammenarbeit mit Amazon EKS validiert. Mit dem EKS-Add-on können Sie sicherstellen, dass Ihre Amazon EKS-Cluster sicher und stabil sind und den Arbeitsaufwand für die Installation, Konfiguration und Aktualisierung von Add-Ons verringern.

Voraussetzungen

Stellen Sie sicher, dass folgende Voraussetzungen erfüllt sind, bevor Sie das Trident Add-on für AWS EKS konfigurieren:

- Ein Amazon EKS Cluster-Konto mit Add-on-Abonnement
- AWS Berechtigungen für den AWS Marketplace:
"aws-marketplace:ViewSubscriptions",
"aws-marketplace:Subscribe",

"aws-marketplace:Unsubscribe

- AMI-Typ: Amazon Linux 2 (AL2_x86_64) oder Amazon Linux 2 ARM (AL2_ARM_64)
- Knotentyp: AMD oder ARM
- Ein bestehendes Amazon FSX für NetApp ONTAP-Filesystem

Aktivieren Sie das Trident Add-on für AWS

Eksctl

Mit den folgenden Beispielbefehlen wird das Trident EKS Add-On installiert:

```
eksctl create addon --name netapp_trident-operator --cluster  
<cluster_name> \  
    --service-account-role-arn  
arn:aws:iam::<account_id>:role/<role_name> --force
```

Management-Konsole

1. Öffnen Sie die Amazon EKS Konsole unter <https://console.aws.amazon.com/eks/home#/clusters>.
2. Klicken Sie im linken Navigationsbereich auf **Cluster**.
3. Klicken Sie auf den Namen des Clusters, für den Sie das NetApp Trident-CSI-Add-On konfigurieren möchten.
4. Klicken Sie auf **Add-ons** und dann auf **Weitere Add-Ons** erhalten.
5. Gehen Sie auf der Seite **Add-ons auswählen** wie folgt vor:
 - a. Aktivieren Sie im Abschnitt EKS-Addons des AWS Marketplace das Kontrollkästchen **Trident by NetApp**.
 - b. Klicken Sie Auf **Weiter**.
6. Gehen Sie auf der Seite **Ausgewählte Add-Ons konfigurieren**-Einstellungen wie folgt vor:
 - a. Wählen Sie die **Version** aus, die Sie verwenden möchten.
 - b. Für **IAM-Rolle auswählen** lassen Sie bei **nicht gesetzt**.
 - c. Erweitern Sie die **Optionale Konfigurationseinstellungen**, folgen Sie dem **Add-On Konfigurationsschema** und setzen Sie den Parameter configurationValues im Abschnitt **Konfigurationswerte** auf die Rolle-arn, die Sie im vorherigen Schritt erstellt haben (Wert sollte im folgenden Format sein: eks.amazonaws.com/role-arn: arn:aws:iam::464262061435:role/AmazonEKS_FSXN_CSI_DriverRole). Wenn Sie für die Konfliktlösungsmethode Überschreiben auswählen, können eine oder mehrere Einstellungen für das vorhandene Add-On mit den Amazon EKS-Zusatzeinstellungen überschrieben werden. Wenn Sie diese Option nicht aktivieren und es einen Konflikt mit Ihren bestehenden Einstellungen gibt, schlägt der Vorgang fehl. Sie können die resultierende Fehlermeldung verwenden, um den Konflikt zu beheben. Bevor Sie diese Option auswählen, stellen Sie sicher, dass das Amazon EKS-Add-On keine Einstellungen verwaltet, die Sie selbst verwalten müssen.
7. Wählen Sie **Weiter**.
8. Wählen Sie auf der Seite **Überprüfen und Hinzufügen Erstellen**.

Nachdem die Installation des Add-ons abgeschlossen ist, wird das installierte Add-on angezeigt.

AWS CLI

1. Erstellen Sie die add-on.json Datei:

```
add-on.json
{
    "clusterName": "<eks-cluster>",
    "addonName": "netapp_trident-operator",
    "addonVersion": "v24.10.0-eksbuild.1",
    "serviceAccountRoleArn": "<arn:aws:iam::123456:role/astratrident-
role>",
    "configurationValues": "{\"cloudIdentity\":
    \"'eks.amazonaws.com/role-arn:
    <arn:aws:iam::123456:role/astratrident-role>'\",
    \"cloudProvider\": \"AWS\"}"
}
```

2. Trident EKS-Add-On installieren“

```
aws eks create-addon --cli-input-json file://add-on.json
```

Aktualisieren Sie das Trident EKS-Add-On

Eksctl

- Überprüfen Sie die aktuelle Version des FSxN Trident CSI-Add-ons. Ersetzen Sie `my-cluster` den Cluster-Namen.

```
eksctl get addon --name netapp_trident-operator --cluster my-cluster
```

Beispielausgabe:

NAME	VERSION	STATUS	ISSUES
IAMROLE	UPDATE AVAILABLE	CONFIGURATION VALUES	
netapp_trident-operator	v24.10.0-eksbuild.1	ACTIVE	0
{ "cloudIdentity": "'eks.amazonaws.com/role-arn: arn:aws:iam::139763910815:role/AmazonEKS_FSXN_CSI_DriverRole'" }			

- Aktualisieren Sie das Add-on auf die Version, DIE unter UPDATE zurückgegeben wurde, DIE in der Ausgabe des vorherigen Schritts VERFÜGBAR ist.

```
eksctl update addon --name netapp_trident-operator --version v24.10.0-eksbuild.1 --cluster my-cluster --force
```

Wenn Sie die Option entfernen `--force` und eine der Amazon EKS-Zusatz Einstellungen mit Ihren vorhandenen Einstellungen in Konflikt steht, schlägt die Aktualisierung des Amazon EKS-Zusatzes fehl. Sie erhalten eine Fehlermeldung, um den Konflikt zu beheben. Bevor Sie diese Option angeben, stellen Sie sicher, dass das Amazon EKS-Add-On keine Einstellungen verwaltet, die Sie verwalten müssen, da diese Einstellungen mit dieser Option überschrieben werden. Weitere Informationen zu anderen Optionen für diese Einstellung finden Sie unter "[Add-Ons](#)". Weitere Informationen zum Field Management von Amazon EKS Kubernetes finden Sie unter "[Außendienstmanagement von Kubernetes](#)".

Management-Konsole

1. Öffnen Sie die Amazon EKS Konsole <https://console.aws.amazon.com/eks/home#/clusters>.
2. Klicken Sie im linken Navigationsbereich auf **Cluster**.
3. Klicken Sie auf den Namen des Clusters, für den Sie das NetApp Trident-CSI-Add-On aktualisieren möchten.
4. Klicken Sie auf die Registerkarte **Add-ons**.
5. Klicken Sie auf **Trident by NetApp** und dann auf **Bearbeiten**.
6. Gehen Sie auf der Seite **Configure Trident by NetApp** wie folgt vor:
 - a. Wählen Sie die **Version** aus, die Sie verwenden möchten.
 - b. Erweitern Sie die **Optionale Konfigurationseinstellungen** und ändern Sie sie nach Bedarf.
 - c. Klicken Sie auf **Änderungen speichern**.

AWS CLI

Im folgenden Beispiel wird das EKS-Add-on aktualisiert:

```
aws eks update-addon --cluster-name my-cluster netapp_trident-operator
vpc-cni --addon-version v24.6.1-eksbuild.1 \
    --service-account-role-arn arn:aws:iam::111122223333:role/role-name
--configuration-values '{}' --resolve-conflicts --preserve
```

Deinstallieren Sie das Trident EKS-Add-On bzw. entfernen Sie es

Sie haben zwei Optionen zum Entfernen eines Amazon EKS-Add-ons:

- **Add-on-Software auf Ihrem Cluster beibehalten** – Diese Option entfernt die Amazon EKS-Verwaltung aller Einstellungen. Amazon EKS kann Sie auch nicht mehr über Updates informieren und das Amazon EKS-Add-On automatisch aktualisieren, nachdem Sie ein Update gestartet haben. Die Add-on-Software auf dem Cluster bleibt jedoch erhalten. Mit dieser Option wird das Add-On zu einer selbstverwalteten Installation anstatt zu einem Amazon EKS-Add-on. Bei dieser Option haben Add-on keine Ausfallzeiten. Behalten Sie die Option im Befehl bei `--preserve`, um das Add-on beizubehalten.
- **Entfernen Sie Add-on-Software komplett aus Ihrem Cluster** – Wir empfehlen, das Amazon EKS-Add-on nur dann aus Ihrem Cluster zu entfernen, wenn es keine Ressourcen auf Ihrem Cluster gibt, die davon abhängen. Entfernen Sie die `--preserve` Option aus dem `delete` Befehl, um das Add-On zu entfernen.



Wenn dem Add-On ein IAM-Konto zugeordnet ist, wird das IAM-Konto nicht entfernt.

Eksctl

Mit dem folgenden Befehl wird das Trident EKS-Add-On deinstalliert:

```
eksctl delete addon --cluster K8s-arm --name netapp_trident-operator
```

Management-Konsole

1. Öffnen Sie die Amazon EKS Konsole unter <https://console.aws.amazon.com/eks/home#/clusters>.
2. Klicken Sie im linken Navigationsbereich auf **Cluster**.
3. Klicken Sie auf den Namen des Clusters, für den Sie das NetApp Trident-CSI-Add-On entfernen möchten.
4. Klicken Sie auf die Registerkarte **Add-ons** und dann auf **Trident by NetApp**.*
5. Klicken Sie Auf **Entfernen**.
6. Gehen Sie im Dialogfeld **Remove netapp_Trident-Operator confirmation** wie folgt vor:
 - a. Wenn Amazon EKS die Verwaltung der Einstellungen für das Add-On einstellen soll, wählen Sie **auf Cluster beibehalten** aus. Führen Sie diese Option aus, wenn Sie die Add-on-Software auf dem Cluster beibehalten möchten, damit Sie alle Einstellungen des Add-ons selbst verwalten können.
 - b. Geben Sie **netapp_Trident-Operator** ein.
 - c. Klicken Sie Auf **Entfernen**.

AWS CLI

Ersetzen `my-cluster` Sie den Namen des Clusters, und führen Sie dann den folgenden Befehl aus.

```
aws eks delete-addon --cluster-name my-cluster --addon-name netapp_trident-operator --preserve
```

Konfigurieren Sie das Speicher-Back-End

Integration von ONTAP-SAN- und NAS-Treibern

Um ein Storage-Backend zu erstellen, müssen Sie eine Konfigurationsdatei im JSON- oder YAML-Format erstellen. Die Datei muss den gewünschten Speichertyp (NAS oder SAN), das Dateisystem und die SVM angeben, von der sie abgerufen werden soll, und wie die Authentifizierung mit ihr durchgeführt werden soll. Im folgenden Beispiel wird gezeigt, wie NAS-basierter Storage definiert wird und wie ein AWS-Schlüssel zum Speichern der Zugangsdaten für die zu verwendende SVM verwendet wird:

YAML

```
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-tbc-ontap-nas
  namespace: trident
spec:
  version: 1
  storageDriverName: ontap-nas
  backendName: tbc-ontap-nas
  svm: svm-name
  aws:
    fsxFilesystemID: fs-xxxxxxxxxx
  credentials:
    name: "arn:aws:secretsmanager:us-west-2:xxxxxxx:secret:secret-
name"
    type: awsarn
```

JSON

```
{
  "apiVersion": "trident.netapp.io/v1",
  "kind": "TridentBackendConfig",
  "metadata": {
    "name": "backend-tbc-ontap-nas"
    "namespace": "trident"
  },
  "spec": {
    "version": 1,
    "storageDriverName": "ontap-nas",
    "backendName": "tbc-ontap-nas",
    "svm": "svm-name",
    "aws": {
      "fsxFilesystemID": "fs-xxxxxxxxxx"
    },
    "managementLIF": null,
    "credentials": {
      "name": "arn:aws:secretsmanager:us-west-2:xxxxxxx:secret:secret-
name",
      "type": "awsarn"
    }
  }
}
```

Führen Sie die folgenden Befehle aus, um die Trident-Backend-Konfiguration (TBC) zu erstellen und zu validieren:

- Erstellen Sie die Trident-Backend-Konfiguration (TBC) aus der yaml-Datei, und führen Sie den folgenden Befehl aus:

```
kubectl create -f backendconfig.yaml -n trident
```

```
tridentbackendconfig.trident.netapp.io/backend-tbc-ontap-nas created
```

- Überprüfen Sie, ob die Trident-Backend-Konfiguration (TBC) erfolgreich erstellt wurde:

```
Kubectl get tbc -n trident
```

NAME	BACKEND NAME	BACKEND UUID
PHASE	STATUS	
backend-tbc-ontap-nas	tbc-ontap-nas	933e0071-66ce-4324-
b9ff-f96d916ac5e9	Bound	Success

FSX für ONTAP-Treiber Details

Sie können Trident mithilfe der folgenden Treiber in Amazon FSX for NetApp ONTAP integrieren:

- **ontap-san:** Jedes bereitgestellte PV ist eine LUN innerhalb seines eigenen Amazon FSX für NetApp ONTAP-Volumens. Empfohlen für Blocklagerung.
- **ontap-nas:** Jedes bereitgestellte PV ist ein vollständiges Amazon FSX für NetApp ONTAP Volumen. Für NFS und SMB empfohlen.
- **ontap-san-economy:** Jedes bereitgestellte PV ist eine LUN mit einer konfigurierbaren Anzahl von LUNs pro Amazon FSX für NetApp ONTAP Volumen.
- **ontap-nas-economy:** Jedes bereitgestellte PV ist ein qtree, mit einer konfigurierbaren Anzahl von qtrees pro Amazon FSX für NetApp ONTAP Volumen.
- **ontap-nas-flexgroup:** Jedes bereitgestellte PV ist ein vollständiges Amazon FSX für NetApp ONTAP FlexGroup Volumen.

Informationen zum Treiber finden Sie unter ["NAS-Treiber"](#) und ["SAN-Treiber"](#).

Nachdem die Konfigurationsdatei erstellt wurde, führen Sie diesen Befehl aus, um sie in Ihrem EKS zu erstellen:

```
kubectl create -f configuration_file
```

Führen Sie den folgenden Befehl aus, um den Status zu überprüfen:

```
kubectl get tbc -n trident
```

NAME	BACKEND NAME	BACKEND UUID
PHASE STATUS		
backend-fsx-ontap-nas	backend-fsx-ontap-nas	7a551921-997c-4c37-a1d1-f2f4c87fa629
Bound	Success	

Erweiterte Back-End-Konfiguration und Beispiele

Die Back-End-Konfigurationsoptionen finden Sie in der folgenden Tabelle:

Parameter	Beschreibung	Beispiel
version		Immer 1
storageDriverName	Name des Speichertreibers	ontap-nas, ontap-nas-economy, ontap-nas-flexgroup, ontap-san, ontap-san-economy
backendName	Benutzerdefinierter Name oder das Storage-Backend	Treibername + „_“ + DatenLIF
managementLIF	IP-Adresse eines Clusters oder einer SVM-Management-LIF Ein vollständig qualifizierter Domain-Name (FQDN) kann angegeben werden. Kann so eingestellt werden, dass IPv6-Adressen verwendet werden, wenn Trident mit dem IPv6-Flag installiert wurde. IPv6-Adressen müssen in eckigen Klammern definiert werden, z. B. [28e8:d9fb:a825:b7bf:69a8:d02f:9e7b:3555]. Wenn Sie den im aws Feld angeben fsxFilesystemID, müssen Sie den nicht angeben managementLIF, da Trident die SVM-Informationen von AWS abrufen managementLIF. Daher müssen Sie die Anmeldedaten für einen Benutzer unter der SVM (z. B. vsadmin) angeben, und der Benutzer muss über die Rolle verfügen vsadmin .	„10.0.0.1“, „[2001:1234:abcd::fefe]“

Parameter	Beschreibung	Beispiel
dataLIF	IP-Adresse des LIF-Protokolls. ONTAP NAS drivers: Wir empfehlen die Angabe von dataLIF. Falls nicht bereitgestellt, ruft Trident die Daten-LIFs von der SVM ab. Sie können einen vollständig qualifizierten Domännennamen (FQDN) angeben, der für die NFS-Mount-Vorgänge verwendet werden soll. Damit können Sie ein Round-Robin-DNS zum Load-Balancing über mehrere Daten-LIFs erstellen. Kann nach der Anfangseinstellung geändert werden. Siehe . ONTAP-SAN-Treiber: Geben Sie nicht für iSCSI an. Trident verwendet die selektive LUN-Zuordnung von ONTAP, um die iSCSI LIFs zu ermitteln, die für die Einrichtung einer Multi-Path-Sitzung erforderlich sind. Eine Warnung wird erzeugt, wenn dataLIF explizit definiert ist. Kann so eingestellt werden, dass IPv6-Adressen verwendet werden, wenn Trident mit dem IPv6-Flag installiert wurde. IPv6-Adressen müssen in eckigen Klammern definiert werden, z. B. [28e8:d9fb:a825:b7bf:69a8:d02f:9e7b:3555].	
autoExportPolicy	Aktivieren Sie die automatische Erstellung von Exportrichtlinien und aktualisieren Sie [Boolean]. Mithilfe der autoExportPolicy Optionen und autoExportCIDRs kann Trident Exportrichtlinien automatisch managen.	false
autoExportCIDRs	Liste der CIDRs, nach denen die Node-IPs von Kubernetes gegen gefiltert werden sollen, wenn autoExportPolicy aktiviert ist. Mithilfe der autoExportPolicy Optionen und autoExportCIDRs kann Trident Exportrichtlinien automatisch managen.	„[„0.0.0.0/0“, „:/0“]“
labels	Satz willkürlicher JSON-formatierter Etiketten für Volumes	“

Parameter	Beschreibung	Beispiel
clientCertificate	Base64-codierter Wert des Clientzertifikats. Wird für zertifikatbasierte Authentifizierung verwendet	“ ”
clientPrivateKey	Base64-kodierte Wert des privaten Client-Schlüssels. Wird für zertifikatbasierte Authentifizierung verwendet	“ ”
trustedCACertificate	Base64-kodierte Wert des vertrauenswürdigen CA-Zertifikats. Optional Wird für die zertifikatbasierte Authentifizierung verwendet.	“ ”
username	Benutzername zum Herstellen einer Verbindung zum Cluster oder zur SVM. Wird für die Anmeldeinformationsbasierte Authentifizierung verwendet. Beispiel: Vsadmin.	
password	Passwort für die Verbindung mit dem Cluster oder der SVM Wird für die Anmeldeinformationsbasierte Authentifizierung verwendet.	
svm	Zu verwendende Storage Virtual Machine	Abgeleitet, wenn eine SVM Management LIF angegeben ist.
storagePrefix	Das Präfix wird beim Bereitstellen neuer Volumes in der SVM verwendet. Kann nach der Erstellung nicht geändert werden. Um diesen Parameter zu aktualisieren, müssen Sie ein neues Backend erstellen.	trident
limitAggregateUsage	Nicht für Amazon FSX für NetApp ONTAP angeben. Die angegebenen <code>fsxadmin</code> und <code>vsadmin</code> enthalten nicht die erforderlichen Berechtigungen, um die aggregierte Nutzung abzurufen und sie mit Trident zu begrenzen.	Verwenden Sie ihn nicht.

Parameter	Beschreibung	Beispiel
limitVolumeSize	Bereitstellung fehlgeschlagen, wenn die angeforderte Volume-Größe über diesem Wert liegt. Beschränkt darüber hinaus die maximale Größe der Volumes, die es über qtrees und LUNs verwaltet, und qtreesPerFlexvol ermöglicht die Anpassung der maximalen Anzahl von qtrees pro FlexVol.	„ (nicht standardmäßig durchgesetzt)
lunsPerFlexvol	Die maximale Anzahl an LUNs pro FlexVol muss im Bereich [50, 200] liegen. Nur SAN	„100“
debugTraceFlags	Fehler-Flags bei der Fehlerbehebung beheben. Beispiel, {„API“:false, „method“:true} nicht verwenden debugTraceFlags, es sei denn, Sie beheben die Fehlerbehebung und erfordern einen detaillierten Log Dump.	Null
nfsMountOptions	Kommagetrennte Liste von NFS-Mount-Optionen. Die Mount-Optionen für persistente Kubernetes-Volumes werden normalerweise in Storage-Klassen angegeben. Wenn jedoch keine Mount-Optionen in einer Storage-Klasse angegeben sind, verwendet Trident die Mount-Optionen, die in der Konfigurationsdatei des Storage-Backends angegeben sind. Wenn in der Storage-Klasse oder in der Konfigurationsdatei keine Mount-Optionen angegeben sind, legt Trident keine Mount-Optionen auf einem zugeordneten persistenten Volume fest.	“
nasType	Konfiguration der Erstellung von NFS- oder SMB-Volumes Optionen sind nfs, , smb oder Null. Muss für SMB-Volumes auf gesetzt smb werden. Einstellung auf null setzt standardmäßig auf NFS-Volumes.	nfs
qtreesPerFlexvol	Maximale Ques pro FlexVol, muss im Bereich [50, 300] liegen	"200"

Parameter	Beschreibung	Beispiel
smbShare	Sie können eine der folgenden Optionen angeben: Den Namen einer SMB-Freigabe, die mit der Microsoft Verwaltungskonsole oder der ONTAP-CLI erstellt wurde, oder einen Namen, mit dem Trident die SMB-Freigabe erstellen kann. Dieser Parameter ist für Amazon FSX for ONTAP Back-Ends erforderlich.	smb-share
useREST	Boolescher Parameter zur Verwendung von ONTAP REST-APIs. Tech Preview useREST wird als Tech Preview bereitgestellt, die für Testumgebungen und nicht für Produktions-Workloads empfohlen wird. Wenn auf festgelegt <code>true</code> , verwendet Trident ONTAP REST APIs, um mit dem Backend zu kommunizieren. Diese Funktion erfordert ONTAP 9.11.1 und höher. Darüber hinaus muss die verwendete ONTAP-Anmelderolle Zugriff auf die Anwendung haben <code>ontap</code> . Dies wird durch die vordefinierten <code>vsadmin</code> Rollen und <code>cluster-admin</code> erreicht.	false
aws	Sie können Folgendes in der Konfigurationsdatei für AWS FSX für ONTAP angeben: - fsxFilesystemID: Geben Sie die ID des AWS FSX Dateisystems an. - apiRegion: Name der AWS API-Region. - apikey: AWS API-Schlüssel. - secretKey: AWS Geheimschlüssel.	"" "" ""
credentials	Geben Sie die FSX SVM-Anmeldeinformationen an, die in AWS Secret Manager zu speichern sind. - name: Amazon Resource Name (ARN) des Geheimnisses, das die Zugangsdaten von SVM enthält. - type: Gesetzt auf <code>awsarn</code> . Weitere Informationen finden Sie unter " Erstellen Sie einen AWS Secrets Manager-Schlüssel ".	

Back-End-Konfigurationsoptionen für die Bereitstellung von Volumes

Mit diesen Optionen können Sie die Standardbereitstellung im Abschnitt der Konfiguration steuern `defaults`. Ein Beispiel finden Sie unten in den Konfigurationsbeispielen.

Parameter	Beschreibung	Standard
<code>spaceAllocation</code>	Speicherplatzzuweisung für LUNs	<code>true</code>
<code>spaceReserve</code>	Space Reservation Mode; „none“ (Thin) oder „Volume“ (Thick)	<code>none</code>
<code>snapshotPolicy</code>	Die Snapshot-Richtlinie zu verwenden	<code>none</code>
<code>qosPolicy</code>	QoS-Richtliniengruppe zur Zuweisung für erstellte Volumes Wählen Sie eine der <code>qosPolicy</code> oder <code>adaptiveQosPolicy</code> pro Storage-Pool oder Backend. Für die Verwendung von QoS-Richtliniengruppen mit Trident ist ONTAP 9.8 oder höher erforderlich. Sie sollten eine nicht gemeinsam genutzte QoS-Richtliniengruppe verwenden und sicherstellen, dass die Richtliniengruppe auf jede Komponente einzeln angewendet wird. Eine Shared-QoS-Richtliniengruppe erzwingt die Obergrenze für den Gesamtdurchsatz aller Workloads.	“
<code>adaptiveQosPolicy</code>	Adaptive QoS-Richtliniengruppe mit Zuordnung für erstellte Volumes Wählen Sie eine der <code>qosPolicy</code> oder <code>adaptiveQosPolicy</code> pro Storage-Pool oder Backend. Nicht unterstützt durch <code>ontap-nas</code> -Ökonomie	“
<code>snapshotReserve</code>	Prozentsatz des für Snapshots reservierten Volumens „0“	Wenn <code>snapshotPolicy</code> ist <code>none</code> , else „
<code>splitOnClone</code>	Teilen Sie einen Klon bei der Erstellung von seinem übergeordneten Objekt auf	<code>false</code>

Parameter	Beschreibung	Standard
encryption	Aktivieren Sie NetApp Volume Encryption (NVE) auf dem neuen Volume, Standardeinstellung ist <code>false</code> . NVE muss im Cluster lizenziert und aktiviert sein, damit diese Option verwendet werden kann. Wenn auf dem Backend NAE aktiviert ist, wird jedes in Trident bereitgestellte Volume NAE aktiviert. Weitere Informationen finden Sie unter "Funktionsweise von Trident mit NVE und NAE" .	<code>false</code>
luksEncryption	Aktivieren Sie die LUKS-Verschlüsselung. Siehe "Linux Unified Key Setup (LUKS) verwenden" . Nur SAN	<code>"</code>
tieringPolicy	Tiering-Richtlinie für die Nutzung <code>none</code>	<code>snapshot-only</code> Für Konfiguration vor ONTAP 9.5 SVM-DR
unixPermissions	Modus für neue Volumes. Leere leer für SMB Volumen.	<code>"</code>
securityStyle	Sicherheitstyp für neue Volumes. NFS-Unterstützung <code>mixed</code> und <code>unix</code> -Sicherheitsstile. SMB-Unterstützung <code>mixed</code> und <code>ntfs</code> Sicherheitsstile.	NFS-Standard ist <code>unix</code> . SMB-Standard ist <code>ntfs</code> .

Vorbereitung zur Bereitstellung von SMB Volumes

Sie können SMB-Volumes mit dem Treiber bereitstellen `ontap-nas`. Führen Sie die folgenden Schritte aus, bevor Sie [Integration von ONTAP-SAN- und NAS-Treibern](#) die Schritte ausführen.

Bevor Sie beginnen

Bevor Sie SMB-Volumes mit dem Treiber bereitstellen können `ontap-nas`, müssen Sie Folgendes haben:

- Kubernetes-Cluster mit einem Linux-Controller-Knoten und mindestens einem Windows-Worker-Node, auf dem Windows Server 2019 ausgeführt wird. Trident unterstützt nur SMB Volumes, die in Pods gemountet sind, die nur auf Windows Nodes ausgeführt werden.
- Mindestens ein Trident-Schlüssel, der Ihre Active Directory-Anmeldeinformationen enthält. So generieren Sie ein Geheimnis `smbcreds`:

```
kubectl create secret generic smbcreds --from-literal username=user
--from-literal password='password'
```

- Ein CSI-Proxy, der als Windows-Dienst konfiguriert ist. Informationen zum Konfigurieren `csi-proxy` von finden Sie unter ["GitHub: CSI-Proxy"](#) oder ["GitHub: CSI Proxy für Windows"](#) für Kubernetes-Nodes, die unter Windows ausgeführt werden.

Schritte

1. Erstellen von SMB-Freigaben Sie können die SMB-Administratorfreigaben auf zwei Arten erstellen, entweder mit dem "[Microsoft Management Console](#)" Snap-in für freigegebene Ordner oder mit der ONTAP-CLI. So erstellen Sie SMB-Freigaben mithilfe der ONTAP-CLI:

- a. Erstellen Sie bei Bedarf die Verzeichnispfadstruktur für die Freigabe.

Der `vserver cifs share create` Befehl überprüft den in der Option `-path` angegebenen Pfad während der Erstellung von Freigaben. Wenn der angegebene Pfad nicht vorhanden ist, schlägt der Befehl fehl.

- b. Erstellen einer mit der angegebenen SVM verknüpften SMB-Freigabe:

```
vserver cifs share create -vserver vserver_name -share-name
share_name -path path [-share-properties share_properties,...]
[other_attributes] [-comment text]
```

- c. Vergewissern Sie sich, dass die Freigabe erstellt wurde:

```
vserver cifs share show -share-name share_name
```



Weitere Informationen finden Sie unter "[Erstellen Sie eine SMB-Freigabe](#)".

2. Beim Erstellen des Backend müssen Sie Folgendes konfigurieren, um SMB-Volumes festzulegen. Für alle FSX für ONTAP Backend-Konfigurationsoptionen, siehe "[FSX für ONTAP Konfigurationsoptionen und Beispiele](#)".

Parameter	Beschreibung	Beispiel
smbShare	Sie können eine der folgenden Optionen angeben: Den Namen einer SMB-Freigabe, die mit der Microsoft Verwaltungskonsole oder der ONTAP-CLI erstellt wurde, oder einen Namen, mit dem Trident die SMB-Freigabe erstellen kann. Dieser Parameter ist für Amazon FSX for ONTAP Back-Ends erforderlich.	smb-share
nasType	Muss auf. gesetzt werden <code>smb</code> Wenn Null, wird standardmäßig auf <code>nfs</code> .	smb
securityStyle	Sicherheitstyp für neue Volumes. Muss für SMB Volumes auf oder mixed gesetzt werden ntfs.	ntfs Oder mixed für SMB Volumes
unixPermissions	Modus für neue Volumes. Muss für SMB Volumes leer gelassen werden.	“

Konfigurieren Sie eine Storage-Klasse und PVC

Konfigurieren Sie ein Kubernetes StorageClass-Objekt und erstellen Sie die Storage-Klasse, um Trident anzuweisen, wie Volumes bereitgestellt werden. Erstellen Sie ein PersistentVolume (PV) und ein PersistentVolumeClaim (PVC), das die konfigurierte Kubernetes StorageClass verwendet, um Zugriff auf das PV anzufordern. Anschließend können Sie das PV an einem Pod montieren.

Erstellen Sie eine Speicherklasse

Konfigurieren Sie ein Kubernetes StorageClass-Objekt

Trident wird von als bereitstellung identifiziert, die für diese Klasse verwendet wird. Trident wird darin ["Kubernetes StorageClass-Objekt"](#) angewiesen, ein Volume bereitzustellen. Beispiel:

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: ontap-gold
provisioner: csi.trident.netapp.io
parameters:
  backendType: "ontap-nas"
  media: "ssd"
  provisioningType: "thin"
  snapshots: "true"
```

Einzelheiten zur Interaktion von Storage-Klassen mit den PersistentVolumeClaim Parametern und zur Steuerung, wie Trident Volumes provisioniert, finden Sie unter ["Kubernetes und Trident Objekte"](#).

Erstellen Sie eine Speicherklasse

Schritte

1. Dies ist ein Kubernetes-Objekt. Verwenden Sie es also `kubectl`, um es in Kubernetes zu erstellen.

```
kubectl create -f storage-class-ontapnas.yaml
```

2. Sie sollten nun eine **Basic-csi** Storage-Klasse sowohl in Kubernetes als auch in Trident sehen, und Trident hätte die Pools auf dem Backend entdeckt haben sollen.

```
kubectl get sc basic-csi
NAME          PROVISIONER          AGE
basic-csi     csi.trident.netapp.io 15h
```

Erstellen Sie das PV und die PVC

Ein "*PersistentVolume*" (PV) ist eine physische Speicherressource, die vom Clusteradministrator auf einem Kubernetes-Cluster bereitgestellt wird. Die "*PersistentVolumeClaim*" (PVC) ist eine Anforderung für den Zugriff auf das PersistentVolume auf dem Cluster.

Die PVC kann so konfiguriert werden, dass eine Speicherung einer bestimmten Größe oder eines bestimmten Zugriffsmodus angefordert wird. Mithilfe der zugehörigen StorageClass kann der Clusteradministrator mehr als die Größe des PersistentVolume und den Zugriffsmodus steuern, z. B. die Performance oder das Service-Level.

Nachdem Sie das PV und die PVC erstellt haben, können Sie das Volume in einem Pod einbinden.

Beispielmanifeste

PersistentVolume-Beispielmanifest

Dieses Beispielmanifest zeigt ein Basis-PV von 10Gi, das mit StorageClass verknüpft ist `basic-csi`.

```
apiVersion: v1
kind: PersistentVolume
metadata:
  name: pv-storage
  labels:
    type: local
spec:
  storageClassName: basic-csi
  capacity:
    storage: 10Gi
  accessModes:
    - ReadWriteMany
  hostPath:
    path: "/my/host/path"
```

PersistentVolumeClaim-Beispielmanifeste

Diese Beispiele zeigen grundlegende PVC-Konfigurationsoptionen.

PVC mit RWX-Zugang

Dieses Beispiel zeigt ein einfaches PVC mit RWX-Zugriff, das mit einer StorageClass namens verknüpft ist `basic-csi`.

```
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: pvc-storage
spec:
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 1Gi
  storageClassName: basic-csi
```

PVC mit NVMe/TCP

Dieses Beispiel zeigt eine grundlegende PVC für NVMe/TCP mit RWO-Zugriff, die einer StorageClass namens zugeordnet ist `protection-gold`.

```
---
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: pvc-san-nvme
spec:
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 300Mi
  storageClassName: protection-gold
```

Erstellen Sie das PV und die PVC

Schritte

1. Erstellen Sie das PV.

```
kubectl create -f pv.yaml
```

2. Überprüfen Sie den PV-Status.

```
kubectl get pv
NAME          CAPACITY  ACCESS MODES  RECLAIM POLICY  STATUS  CLAIM
STORAGECLASS  REASON    AGE
pv-storage    4Gi       RWO           Retain          Available
7s
```

3. Erstellen Sie das PVC.

```
kubectl create -f pvc.yaml
```

4. Überprüfen Sie den PVC-Status.

```
kubectl get pvc
NAME          STATUS  VOLUME      CAPACITY  ACCESS MODES  STORAGECLASS  AGE
pvc-storage   Bound   pv-name     2Gi       RWO           storageclass  5m
```

Einzelheiten zur Interaktion von Storage-Klassen mit den PersistentVolumeClaim Parametern und zur Steuerung, wie Trident Volumes provisioniert, finden Sie unter ["Kubernetes und Trident Objekte"](#).

Trident-Attribute

Diese Parameter bestimmen, welche in Trident gemanagten Storage Pools zur Bereitstellung von Volumes eines bestimmten Typs verwendet werden sollten.

Attribut	Typ	Werte	Angebot	Anfrage	Unterstützt von
Medien ¹	Zeichenfolge	hdd, Hybrid, ssd	Pool enthält Medien dieser Art. Beides bedeutet Hybrid	Medientyp angeben	ontap-nas, ontap-nas-Economy, ontap-nas-Flexgroup, ontap-san, solidfire-san
Bereitstellungstyp	Zeichenfolge	Dünn, dick	Pool unterstützt diese Bereitstellungsmethode	Bereitstellungsmethode angeben	Thick: All ONTAP; Thin: Alle ONTAP und solidfire-san

Attribut	Typ	Werte	Angebot	Anfrage	Unterstützt von
BackendType	Zeichenfolge	ontap-nas, ontap-nas-Economy, ontap-nas-Flexgroup, ontap-san, solidfire-san, gcp-cvs, Azure-netapp-Files, ontap-san-Wirtschaftlichkeit	Pool gehört zu dieser Art von Backend	Back-End angegeben	Alle Treiber
Snapshots	bool	Richtig, falsch	Pool unterstützt Volumes mit Snapshots	Volume mit aktivierten Snapshots	ontap-nas, ontap-san, solidfire-san, gcp-cvs
Klone	bool	Richtig, falsch	Pool unterstützt das Klonen von Volumes	Volume mit aktivierten Klonen	ontap-nas, ontap-san, solidfire-san, gcp-cvs
Verschlüsselung	bool	Richtig, falsch	Pool unterstützt verschlüsselte Volumes	Volume mit aktivierter Verschlüsselung	ontap-nas, ontap-nas-Economy, ontap-nas-Flexgroups, ontap-san
IOPS	Int	Positive Ganzzahl	Pool kann IOPS in diesem Bereich garantieren	Volume hat diese IOPS garantiert	solidfire-san

1: Nicht unterstützt von ONTAP Select-Systemen

Beispielanwendung bereitstellen

Beispielanwendung bereitstellen.

Schritte

1. Mounten Sie das Volume in einem Pod.

```
kubectl create -f pv-pod.yaml
```

Diese Beispiele zeigen grundlegende Konfigurationen zum Anbringen der PVC an einem POD:

Grundkonfiguration:

```

kind: Pod
apiVersion: v1
metadata:
  name: pv-pod
spec:
  volumes:
    - name: pv-storage
      persistentVolumeClaim:
        claimName: basic
  containers:
    - name: pv-container
      image: nginx
      ports:
        - containerPort: 80
          name: "http-server"
      volumeMounts:
        - mountPath: "/my/mount/path"
          name: pv-storage

```



Sie können den Fortschritt mit überwachen `kubectl get pod --watch`.

2. Vergewissern Sie sich, dass das Volume auf gemountet ist `/my/mount/path`.

```
kubectl exec -it pv-pod -- df -h /my/mount/path
```

Filesystem	Size
Used Avail Use% Mounted on	
192.168.188.78:/trident_pvc_ae45ed05_3ace_4e7c_9080_d2a83ae03d06	1.1G
320K 1.0G 1% /my/mount/path	

Sie können den Pod jetzt löschen. Die Pod Applikation wird nicht mehr existieren, aber das Volume bleibt erhalten.

```
kubectl delete pod pv-pod
```

Konfigurieren Sie das Trident EKS-Add-on auf einem EKS-Cluster

NetApp Trident optimiert das Amazon FSX für NetApp ONTAP Storage-Management in Kubernetes, damit sich Ihre Entwickler und Administratoren voll und ganz auf den Applikationseinsatz konzentrieren können. Das NetApp Trident EKS Add-on enthält die neuesten Sicherheitspatches und Bug Fixes. Es wurde von AWS für die Zusammenarbeit

mit Amazon EKS validiert. Mit dem EKS-Add-on können Sie sicherstellen, dass Ihre Amazon EKS-Cluster sicher und stabil sind und den Arbeitsaufwand für die Installation, Konfiguration und Aktualisierung von Add-Ons verringern.

Voraussetzungen

Stellen Sie sicher, dass folgende Voraussetzungen erfüllt sind, bevor Sie das Trident Add-on für AWS EKS konfigurieren:

- Ein Amazon EKS-Cluster-Konto mit Berechtigungen zum Arbeiten mit Add-ons. Siehe ["Amazon EKS-Add-ons"](#).
- AWS Berechtigungen für den AWS Marketplace:
"aws-marketplace:ViewSubscriptions",
"aws-marketplace:Subscribe",
"aws-marketplace:Unsubscribe"
- AMI-Typ: Amazon Linux 2 (AL2_x86_64) oder Amazon Linux 2 ARM (AL2_ARM_64)
- Knotentyp: AMD oder ARM
- Ein bestehendes Amazon FSX für NetApp ONTAP-Filesystem

Schritte

1. Erstellen Sie unbedingt eine IAM-Rolle und einen AWS Secret, damit EKS-Pods auf AWS Ressourcen zugreifen können. Anweisungen hierzu finden Sie unter ["IAM-Rolle und AWS Secret erstellen"](#).
2. Navigieren Sie auf Ihrem EKS Kubernetes-Cluster zur Registerkarte **Add-ons**.
3. Gehen Sie zu **AWS Marketplace Add-ons** und wählen Sie die Kategorie *Storage*.
4. Suchen Sie **NetApp Trident** und aktivieren Sie das Kontrollkästchen für das Trident-Add-on, und klicken Sie auf **Weiter**.
5. Wählen Sie die gewünschte Version des Add-ons aus.
6. Wählen Sie die Option IAM-Rolle aus, die vom Knoten übernommen werden soll.
7. Konfigurieren Sie die optionalen Konfigurationseinstellungen nach Bedarf, und wählen Sie **Weiter**.

Folgen Sie dem **Add-on-Konfigurationsschema** und setzen Sie den Parameter Konfigurationswerte im Abschnitt **Konfigurationswerte** auf die Rolle-arn, die Sie im vorherigen Schritt (Schritt 1) erstellt haben (Wert sollte im folgenden Format sein: `eks.amazonaws.com/role-arn:arn:aws:iam::464262061435:role/AmazonEKS_FSXN_CSI_DriverRole`). HINWEIS: Wenn Sie für die Konfliktlösungsmethode Überschreiben auswählen, können eine oder mehrere Einstellungen für das vorhandene Add-On mit den Amazon EKS-Add-on-Einstellungen überschrieben werden. Wenn Sie diese Option nicht aktivieren und es einen Konflikt mit Ihren bestehenden Einstellungen gibt, schlägt der Vorgang fehl. Sie können die resultierende Fehlermeldung verwenden, um den Konflikt zu beheben. Bevor Sie diese Option auswählen, stellen Sie sicher, dass das Amazon EKS-Add-On keine Einstellungen verwaltet, die Sie selbst verwalten müssen.

8. Wählen Sie **Erstellen**.

9. Überprüfen Sie, ob der Status des Add-ons *Active* lautet.

10. Führen Sie den folgenden Befehl aus, um zu überprüfen, ob Trident ordnungsgemäß auf dem Cluster installiert ist:

```
kubectl get pods -n trident
```

11. Setzen Sie die Einrichtung fort und konfigurieren Sie das Storage-Back-End. Weitere Informationen finden Sie unter ["Konfigurieren Sie das Speicher-Back-End"](#).

Installieren/deinstallieren Sie das Trident EKS-Add-On über CLI

Installieren Sie das NetApp Trident EKS-Add-On über CLI:

Mit dem folgenden Beispielfehl wird das Trident EKS Add-On installiert:

```
eksctl create addon --name aws-ebs-csi-driver --cluster <cluster_name>
--service-account-role-arn arn:aws:iam::<account_id>:role/<role_name>
--force
```

Deinstallieren Sie das NetApp Trident EKS-Add-On über CLI:

Mit dem folgenden Befehl wird das Trident EKS-Add-On deinstalliert:

```
eksctl delete addon --cluster K8s-arm --name netapp_trident-operator
```

Back-Ends mit kubectl erstellen

Ein Backend definiert die Beziehung zwischen Trident und einem Storage-System. Er erzählt Trident, wie man mit diesem Storage-System kommuniziert und wie Trident Volumes daraus bereitstellen sollte. Nach der Installation von Trident wird im nächsten Schritt ein Backend erstellt. Mit der `TridentBackendConfig` CRD-Definition (Custom Resource Definition) können Sie Trident Back-Ends direkt über die Kubernetes-Schnittstelle erstellen und managen. Sie können dies mit `kubectl` oder mit dem entsprechenden CLI-Tool für Ihre Kubernetes-Distribution tun.

TridentBackendConfig

`TridentBackendConfig` (`tbc`, `tbconfig`, `tbackendconfig`) ist ein Frontend, named CRD, das Ihnen ermöglicht, Trident-Backends mit `kubectl` zu verwalten. Kubernetes- und Storage-Administratoren können jetzt Back-Ends direkt über die Kubernetes-CLI erstellen und managen (`tridentctl`, ohne dass ein dediziertes Befehlszeilendienstprogramm erforderlich ist).

Bei der Erstellung eines `TridentBackendConfig` Objekts geschieht Folgendes:

- Basierend auf der von Ihnen bereitgestellten Konfiguration wird von Trident automatisch ein Backend erstellt. Dies wird intern als (tbe, tridentbackend) CR dargestellt TridentBackend.
- Das TridentBackendConfig ist eindeutig an ein gebunden TridentBackend, das von Trident erstellt wurde.

Jede TridentBackendConfig verwaltet ein One-to-One Mapping mit einem TridentBackend. ersteres ist die Schnittstelle, die dem Benutzer zur Gestaltung und Konfiguration von Backends zur Verfügung gestellt wird; Letzteres ist, wie Trident das eigentliche Backend-Objekt darstellt.



TridentBackend CRS werden automatisch von Trident erstellt. Sie sollten diese nicht ändern. Wenn Sie Änderungen an Back-Ends vornehmen möchten, ändern Sie das TridentBackendConfig Objekt.

Das folgende Beispiel zeigt das CR-Format TridentBackendConfig:

```
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-tbc-ontap-san
spec:
  version: 1
  backendName: ontap-san-backend
  storageDriverName: ontap-san
  managementLIF: 10.0.0.1
  dataLIF: 10.0.0.2
  svm: trident_svm
  credentials:
    name: backend-tbc-ontap-san-secret
```

Sie können sich auch die Beispiele im "[trident-Installationsprogramm](#)" Verzeichnis für Beispielkonfigurationen für die gewünschte Speicherplattform/den gewünschten Service ansehen.

Das spec übernimmt Backend-spezifische Konfigurationsparameter. In diesem Beispiel verwendet das Backend den ontap-san Speichertreiber und verwendet die hier tabellierten Konfigurationsparameter. Eine Liste der Konfigurationsoptionen für den gewünschten Speichertreiber finden Sie im "[Back-End-Konfigurationsinformationen für Ihren Speichertreiber](#)".

Der spec Abschnitt enthält auch credentials und deletionPolicy Felder, die neu im CR eingeführt werden TridentBackendConfig:

- **credentials:** Dieser Parameter ist ein Pflichtfeld und enthält die Anmeldeinformationen, die zur Authentifizierung mit dem Speichersystem/Service verwendet werden. Dies ist auf ein vom Benutzer erstelltes Kubernetes Secret festgelegt. Die Anmeldeinformationen können nicht im Klartext weitergegeben werden und führen zu einem Fehler.
- **deletionPolicy:** Dieses Feld definiert, was passieren soll, wenn das TridentBackendConfig gelöscht wird. Es kann einen von zwei möglichen Werten annehmen:
 - **delete:** Dies führt zum Löschen von TridentBackendConfig CR und dem zugehörigen Backend. Dies ist der Standardwert.

- `retain`: Wenn ein `TridentBackendConfig` CR gelöscht wird, ist die Backend-Definition weiterhin vorhanden und kann mit verwaltet werden `tridentctl`. Durch Festlegen der Löschrichtlinie auf `retain` können Benutzer ein Downgrade auf eine frühere Version (vor 21.04) durchführen und die erstellten Back-Ends beibehalten. Der Wert für dieses Feld kann aktualisiert werden, nachdem ein `TridentBackendConfig` erstellt wurde.



Der Name eines Backends wird mit `spec.backendName` gesetzt. Wenn nicht angegeben, wird der Name des Backends auf den Namen des Objekts (`metadata.name`) gesetzt `TridentBackendConfig`. Es wird empfohlen, Backend-Namen explizit mitzu setzen `spec.backendName`.



Back-Ends, die mit `tridentctl` erstellt wurden, haben kein zugeordnetes `TridentBackendConfig` Objekt. Sie können diese Back-Ends mit `kubectl` verwalten, indem Sie ein CR erstellen `TridentBackendConfig`. Es ist darauf zu achten, identische Konfigurationsparameter anzugeben (z. B. `spec.backendName`, `spec.storagePrefix`, `spec.storageDriverName` und so weiter). `Trident` bindet das neu erstellte mit dem bereits vorhandenen Backend automatisch `TridentBackendConfig`.

Schritte im Überblick

Um ein neues Backend mit zu erstellen `kubectl`, sollten Sie Folgendes tun:

1. Erstellen Sie ein **"Kubernetes Secret"**. das Geheimnis enthält die Anmeldeinformationen, die `Trident` benötigt, um mit dem Speicher-Cluster/Service zu kommunizieren.
2. Erstellen Sie ein `TridentBackendConfig` Objekt. Dies enthält Angaben zum Storage-Cluster/Service und verweist auf das im vorherigen Schritt erstellte Geheimnis.

Nachdem Sie ein Backend erstellt haben, können Sie dessen Status mithilfe von beobachten `kubectl get tbc <tbc-name> -n <trident-namespace>` und weitere Details erfassen.

Schritt: Ein Kubernetes Secret erstellen

Erstellen Sie einen geheimen Schlüssel, der die Anmeldedaten für den Zugriff für das Backend enthält. Dies ist nur bei jedem Storage Service/jeder Plattform möglich. Hier ein Beispiel:

```
kubectl -n trident create -f backend-tbc-ontap-san-secret.yaml
apiVersion: v1
kind: Secret
metadata:
  name: backend-tbc-ontap-san-secret
type: Opaque
stringData:
  username: cluster-admin
  password: password
```

In dieser Tabelle sind die Felder zusammengefasst, die für jede Speicherplattform im Secret enthalten sein müssen:

Beschreibung der geheimen Felder der Speicherplattform	Geheim	Feldbeschreibung
Azure NetApp Dateien	Client-ID	Die Client-ID aus einer App-Registrierung
Cloud Volumes Service für GCP	Private_Schlüssel_id	ID des privaten Schlüssels. Teil des API-Schlüssels für GCP-Servicekonto mit CVS-Administratorrolle
Cloud Volumes Service für GCP	Privater_Schlüssel	Privater Schlüssel. Teil des API-Schlüssels für GCP-Servicekonto mit CVS-Administratorrolle
Element (NetApp HCI/SolidFire)	Endpunkt	MVIP für den SolidFire-Cluster mit Mandanten-Anmeldedaten
ONTAP	Benutzername	Benutzername für die Verbindung mit dem Cluster/SVM. Wird für die Anmeldeinformationsbasierte Authentifizierung verwendet
ONTAP	Passwort	Passwort für die Verbindung mit dem Cluster/SVM Wird für die Anmeldeinformationsbasierte Authentifizierung verwendet
ONTAP	KundenPrivateKey	Base64-kodierte Wert des privaten Client-Schlüssels. Wird für die zertifikatbasierte Authentifizierung verwendet
ONTAP	ChapUsername	Eingehender Benutzername. Erforderlich, wenn usCHAP=true verwendet wird. Für <code>ontap-san</code> und <code>ontap-san-economy</code>
ONTAP	ChapInitiatorSecret	CHAP-Initiatorschlüssel. Erforderlich, wenn usCHAP=true verwendet wird. Für <code>ontap-san</code> und <code>ontap-san-economy</code>
ONTAP	ChapTargetBenutzername	Zielbenutzername. Erforderlich, wenn usCHAP=true verwendet wird. Für <code>ontap-san</code> und <code>ontap-san-economy</code>

Beschreibung der geheimen Felder der Speicherplattform	Geheim	Feldbeschreibung
ONTAP	ChapTargetInitiatorSecret	Schlüssel für CHAP-Zielinitiator. Erforderlich, wenn usCHAP=true verwendet wird. Für ontap-san und ontap-san-economy

Der in diesem Schritt erstellte Schlüssel wird im Feld des `TridentBackendConfig` Objekts referenziert `spec.credentials`, das im nächsten Schritt erstellt wird.

Schritt 2: Erstellen Sie den `TridentBackendConfig` CR

Sie können jetzt Ihren CR erstellen `TridentBackendConfig`. In diesem Beispiel wird mithilfe des unten dargestellten Objekts ein Backend erstellt, das den Treiber `TridentBackendConfig` verwendet `ontap-san`:

```
kubectl -n trident create -f backend-tbc-ontap-san.yaml
```

```
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-tbc-ontap-san
spec:
  version: 1
  backendName: ontap-san-backend
  storageDriverName: ontap-san
  managementLIF: 10.0.0.1
  dataLIF: 10.0.0.2
  svm: trident_svm
  credentials:
    name: backend-tbc-ontap-san-secret
```

Schritt 3: Überprüfen Sie den Status des `TridentBackendConfig` CR

Nachdem Sie den CR erstellt `TridentBackendConfig` haben, können Sie den Status überprüfen. Das folgende Beispiel zeigt:

```
kubectl -n trident get tbc backend-tbc-ontap-san
NAME                                BACKEND NAME                BACKEND UUID
PHASE    STATUS
backend-tbc-ontap-san    ontap-san-backend          8d24fce7-6f60-4d4a-8ef6-
bab2699e6ab8    Bound    Success
```

Ein Backend wurde erfolgreich erstellt und an den CR gebunden `TridentBackendConfig`.

Die Phase kann einen der folgenden Werte annehmen:

- **Bound:** Der `TridentBackendConfig` CR ist mit einem Backend verbunden, und das Backend enthält `configRef` gesetzt auf die UID des `TridentBackendConfig` CR.
- **Unbound:** Dargestellt mit `""`. Das `TridentBackendConfig` Objekt ist nicht an ein Backend gebunden. Alle neu erstellten `TridentBackendConfig` CRS befinden sich standardmäßig in dieser Phase. Wenn die Phase sich ändert, kann sie nicht wieder auf Unbound zurückgesetzt werden.
- **Deleting:** Die `TridentBackendConfig` CR's `deletionPolicy` wurden auf Löschen gesetzt. Wenn der `TridentBackendConfig` CR gelöscht wird, wechselt er in den Löschstatus.
 - Wenn auf dem Backend keine Persistent Volume Claims (PVCs) vorhanden sind, führt das Löschen des `TridentBackendConfig` dazu, dass Trident das Backend sowie den CR löscht `TridentBackendConfig`.
 - Wenn ein oder mehrere VES im Backend vorhanden sind, wechselt es in den Löschzustand. Anschließend geht der `TridentBackendConfig` CR auch in die Löschphase über. Das Backend und `TridentBackendConfig` werden erst gelöscht, nachdem alle VES gelöscht wurden.
- **Lost:** Das mit dem CR verknüpfte Backend `TridentBackendConfig` wurde versehentlich oder absichtlich gelöscht und der `TridentBackendConfig` CR hat noch einen Verweis auf das gelöschte Backend. Der `TridentBackendConfig` CR kann unabhängig vom Wert gelöscht werden `deletionPolicy`.
- **Unknown:** Trident kann den Status oder die Existenz des mit dem CR verknüpften Backends nicht bestimmen `TridentBackendConfig`. Beispiel: Wenn der API-Server nicht reagiert oder die `tridentbackends.trident.netapp.io` CRD fehlt. Dies kann Eingriffe erfordern.

In dieser Phase wird erfolgreich ein Backend erstellt! Es gibt mehrere Operationen, die zusätzlich bearbeitet werden können, wie ["Back-End-Updates und Löschungen am Back-End"](#)z. B. .

(Optional) Schritt 4: Weitere Informationen

Sie können den folgenden Befehl ausführen, um weitere Informationen über Ihr Backend zu erhalten:

```
kubectl -n trident get tbc backend-tbc-ontap-san -o wide
```

NAME	BACKEND NAME	BACKEND UUID	
PHASE	STATUS	STORAGE DRIVER	DELETION POLICY
backend-tbc-ontap-san	ontap-san-backend	8d24fce7-6f60-4d4a-8ef6-bab2699e6ab8	Bound Success ontap-san delete

Zusätzlich können Sie auch einen YAML/JSON Dump von erhalten `TridentBackendConfig`.

```
kubectl -n trident get tbc backend-tbc-ontap-san -o yaml
```

```

apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  creationTimestamp: "2021-04-21T20:45:11Z"
  finalizers:
  - trident.netapp.io
  generation: 1
  name: backend-tbc-ontap-san
  namespace: trident
  resourceVersion: "947143"
  uid: 35b9d777-109f-43d5-8077-c74a4559d09c
spec:
  backendName: ontap-san-backend
  credentials:
    name: backend-tbc-ontap-san-secret
  managementLIF: 10.0.0.1
  dataLIF: 10.0.0.2
  storageDriverName: ontap-san
  svm: trident_svm
  version: 1
status:
  backendInfo:
    backendName: ontap-san-backend
    backendUUID: 8d24fce7-6f60-4d4a-8ef6-bab2699e6ab8
  deletionPolicy: delete
  lastOperationStatus: Success
  message: Backend 'ontap-san-backend' created
  phase: Bound

```

`backendInfo` Enthält die `backendName` und die `backendUUID` des Backends, das als Antwort auf den CR erstellt wurde `TridentBackendConfig`. Das `lastOperationStatus` Feld stellt den Status des letzten Vorgangs des CR dar `TridentBackendConfig`, der vom Benutzer ausgelöst werden kann (z. B. Benutzer hat etwas in geändert `spec`) oder durch Trident ausgelöst (z. B. beim Neustart von Trident). Es kann entweder erfolgreich oder fehlgeschlagen sein. `phase` Stellt den Status der Beziehung zwischen dem CR und dem Backend dar `TridentBackendConfig`. Im obigen Beispiel `phase` hat der Wert gebunden, was bedeutet, dass der `TridentBackendConfig` CR mit dem Backend verknüpft ist.

Sie können den Befehl ausführen `kubectl -n trident describe tbc <tbc-cr-name>`, um Details der Ereignisprotokolle zu erhalten.



Sie können ein Backend, das ein zugeordnetes Objekt enthält, mit `tridentctl` nicht aktualisieren oder löschen `TridentBackendConfig`. Um die Schritte beim Wechsel zwischen und `TridentBackendConfig` zu verstehen `tridentctl`, ["Sehen Sie hier"](#).

Back-Ends managen

Führen Sie das Back-End-Management mit kubectl durch

Erfahren Sie, wie Sie Back-End-Management-Operationen mit durchführen `kubectl`.

Löschen Sie ein Back-End

Durch das Löschen eines `TridentBackendConfig`, weisen Sie Trident an, Back-Ends zu löschen/zu behalten (basierend auf `deletionPolicy`). Um ein Backend zu löschen, stellen Sie sicher, dass `deletionPolicy` es auf „Löschen“ gesetzt ist. Um nur die zu löschen `TridentBackendConfig`, stellen Sie sicher, dass `deletionPolicy` auf beibehalten gesetzt ist. Dadurch wird sichergestellt, dass das Backend noch vorhanden ist und mit verwaltet werden kann `tridentctl`.

Führen Sie den folgenden Befehl aus:

```
kubectl delete tbc <tbc-name> -n trident
```

Trident löscht die Kubernetes-Geheimnisse nicht, die von verwendet wurden `TridentBackendConfig`. Der Kubernetes-Benutzer ist für die Bereinigung von Geheimnissen verantwortlich. Beim Löschen von Geheimnissen ist Vorsicht zu nehmen. Sie sollten Geheimnisse nur löschen, wenn sie nicht von den Back-Ends verwendet werden.

Zeigen Sie die vorhandenen Back-Ends an

Führen Sie den folgenden Befehl aus:

```
kubectl get tbc -n trident
```

Sie können auch ausführen `tridentctl get backend -n trident` oder `tridentctl get backend -o yaml -n trident` eine Liste aller vorhandenen Back-Ends erhalten. Diese Liste enthält auch Backends, die mit erstellt wurden `tridentctl`.

Aktualisieren Sie ein Backend

Es gibt mehrere Gründe für die Aktualisierung eines Backend:

- Die Anmeldeinformationen für das Speichersystem wurden geändert. Zum Aktualisieren der Zugangsdaten muss der im Objekt verwendete Kubernetes Secret `TridentBackendConfig` aktualisiert werden. Trident aktualisiert das Backend automatisch mit den neuesten Anmeldeinformationen. Führen Sie den folgenden Befehl aus, um den Kubernetes Secret zu aktualisieren:

```
kubectl apply -f <updated-secret-file.yaml> -n trident
```

- Parameter (wie der Name der verwendeten ONTAP-SVM) müssen aktualisiert werden.
 - Mit dem folgenden Befehl können Sie Objekte direkt über Kubernetes aktualisieren `TridentBackendConfig`:

```
kubectl apply -f <updated-backend-file.yaml>
```

- Alternativ können Sie mit dem folgenden Befehl Änderungen am vorhandenen CR vornehmen `TridentBackendConfig`:

```
kubectl edit tbc <tbc-name> -n trident
```



- Wenn ein Backend-Update fehlschlägt, bleibt das Backend in seiner letzten bekannten Konfiguration erhalten. Sie können die Protokolle anzeigen, um die Ursache zu ermitteln, indem Sie `kubectl describe tbc <tbc-name> -n trident` ausführen oder `kubectl get tbc <tbc-name> -o yaml -n trident`.
- Nachdem Sie das Problem mit der Konfigurationsdatei erkannt und behoben haben, können Sie den Befehl `Update` erneut ausführen.

Back-End-Management mit `tridentctl`

Erfahren Sie, wie Sie Back-End-Management-Operationen mit `tridentctl` durchführen.

Erstellen Sie ein Backend

"[Back-End-Konfigurationsdatei](#)" Führen Sie nach dem Erstellen eines den folgenden Befehl aus:

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

Wenn die Back-End-Erstellung fehlschlägt, ist mit der Back-End-Konfiguration ein Fehler aufgetreten. Sie können die Protokolle zur Bestimmung der Ursache anzeigen, indem Sie den folgenden Befehl ausführen:

```
tridentctl logs -n trident
```

Nachdem Sie das Problem mit der Konfigurationsdatei identifiziert und behoben haben, können Sie den Befehl einfach erneut ausführen `create`.

Löschen Sie ein Back-End

Gehen Sie folgendermaßen vor, um ein Backend aus Trident zu löschen:

1. Abrufen des Back-End-Namens:

```
tridentctl get backend -n trident
```

2. Back-End löschen:

```
tridentctl delete backend <backend-name> -n trident
```



Wenn Trident Volumes und Snapshots von diesem Backend bereitgestellt hat, die noch vorhanden sind, verhindert das Löschen des Backends, dass neue Volumes bereitgestellt werden. Das Backend wird weiterhin in einem „Deleting“ Zustand vorhanden sein und Trident wird weiterhin diese Volumes und Snapshots verwalten, bis sie gelöscht werden.

Zeigen Sie die vorhandenen Back-Ends an

Gehen Sie zum Anzeigen der von Trident verwendeten Back-Ends wie folgt vor:

- Führen Sie den folgenden Befehl aus, um eine Zusammenfassung anzuzeigen:

```
tridentctl get backend -n trident
```

- Um alle Details anzuzeigen, führen Sie den folgenden Befehl aus:

```
tridentctl get backend -o json -n trident
```

Aktualisieren Sie ein Backend

Führen Sie nach dem Erstellen einer neuen Backend-Konfigurationsdatei den folgenden Befehl aus:

```
tridentctl update backend <backend-name> -f <backend-file> -n trident
```

Wenn das Backend-Update fehlschlägt, ist bei der Backend-Konfiguration ein Fehler aufgetreten oder Sie haben ein ungültiges Update versucht. Sie können die Protokolle zur Bestimmung der Ursache anzeigen, indem Sie den folgenden Befehl ausführen:

```
tridentctl logs -n trident
```

Nachdem Sie das Problem mit der Konfigurationsdatei identifiziert und behoben haben, können Sie den Befehl einfach erneut ausführen `update`.

Identifizieren Sie die Storage-Klassen, die ein Backend nutzen

Dies ist ein Beispiel für die Art von Fragen, die Sie mit der JSON beantworten können, die `tridentctl` für Backend-Objekte ausgegeben wird. Hierbei wird das Dienstprogramm verwendet `jq`, das Sie installieren müssen.

```
tridentctl get backend -o json | jq '[.items[] | {backend: .name, storageClasses: [.storage[].storageClasses]|unique}]'
```

Dies gilt auch für Backends, die durch die Verwendung von erstellt wurden `TridentBackendConfig`.

Wechseln Sie zwischen den Back-End-Managementoptionen

Erfahren Sie mehr über die verschiedenen Möglichkeiten für das Management von Back-Ends in Trident.

Optionen für das Management von Back-Ends

Mit der Einführung von `TridentBackendConfig` haben Administratoren nun zwei einzigartige Möglichkeiten, Back-Ends zu managen. Dies stellt die folgenden Fragen:

- Können Back-Ends erstellt `tridentctl` werden mit `TridentBackendConfig`?
- Können Back-Ends erstellt mit `TridentBackendConfig` verwaltet werden `tridentctl` ?

Managen von `tridentctl` Back-Ends mit `TridentBackendConfig`

In diesem Abschnitt werden die Schritte zum Management von Back-Ends behandelt, die durch das Erstellen von Objekten direkt über die Kubernetes-Schnittstelle erstellt `TridentBackendConfig` wurden `tridentctl`.

Dies gilt für die folgenden Szenarien:

- Bereits vorhandene Backends, die nicht über ein verfügen `TridentBackendConfig`, weil sie mit erstellt wurden `tridentctl`.
- Neue Backends, die mit erstellt wurden `tridentctl`, während andere `TridentBackendConfig` Objekte existieren.

In beiden Szenarien sind Back-Ends weiterhin vorhanden, wobei Trident Volumes terminiert und darauf ausgeführt werden. Administratoren können hier eine von zwei Möglichkeiten wählen:

- Verwenden Sie weiter `tridentctl`, um Back-Ends zu verwalten, die mit ihm erstellt wurden.
- Binden von Back-Ends, die mit erstellt `tridentctl` wurden, an ein neues `TridentBackendConfig` Objekt. Dies würde bedeuten, dass die Back-Ends mit und nicht `tridentctl` verwaltet werden `kubect1`.

Um ein vorvorhandenes Backend mit zu verwalten `kubect1`, müssen Sie ein erstellen `TridentBackendConfig`, das an das vorhandene Backend bindet. Hier eine Übersicht über die Funktionsweise:

1. Kubernetes Secret erstellen: Das Geheimnis enthält die Anmeldeinformationen, die Trident zur Kommunikation mit dem Storage-Cluster/Service benötigt.
2. Erstellen Sie ein `TridentBackendConfig` Objekt. Dies enthält Angaben zum Storage-Cluster/Service und verweist auf das im vorherigen Schritt erstellte Geheimnis. Es ist darauf zu achten, identische Konfigurationsparameter anzugeben (z. B. `spec.backendName`, , `spec.storagePrefix` `spec.storageDriverName` und so weiter). `spec.backendName` Muss auf den Namen des vorhandenen Backends gesetzt werden.

Schritt 0: Identifizieren Sie das Backend

Um ein zu erstellen `TridentBackendConfig`, das an ein vorhandenes Backend bindet, müssen Sie die Backend-Konfiguration abrufen. In diesem Beispiel nehmen wir an, dass ein Backend mithilfe der folgenden

JSON-Definition erstellt wurde:

```
tridentctl get backend ontap-nas-backend -n trident
```

```
+-----+-----+
+-----+-----+-----+-----+
|          NAME          | STORAGE DRIVER |          UUID          |
| STATE | VOLUMES |                  |
+-----+-----+-----+-----+
+-----+-----+-----+-----+
| ontap-nas-backend      | ontap-nas      | 52f2eb10-e4c6-4160-99fc-
96b3be5ab5d7 | online |          25 |
+-----+-----+-----+-----+
+-----+-----+-----+-----+
```

```
cat ontap-nas-backend.json
```

```
{
  "version": 1,
  "storageDriverName": "ontap-nas",
  "managementLIF": "10.10.10.1",
  "dataLIF": "10.10.10.2",
  "backendName": "ontap-nas-backend",
  "svm": "trident_svm",
  "username": "cluster-admin",
  "password": "admin-password",

  "defaults": {
    "spaceReserve": "none",
    "encryption": "false"
  },
  "labels": {"store": "nas_store"},
  "region": "us_east_1",
  "storage": [
    {
      "labels": {"app": "msoffice", "cost": "100"},
      "zone": "us_east_1a",
      "defaults": {
        "spaceReserve": "volume",
        "encryption": "true",
        "unixPermissions": "0755"
      }
    },
    {
      "labels": {"app": "mysqldb", "cost": "25"},
      "zone": "us_east_1d",
      "defaults": {
```

```

        "spaceReserve": "volume",
        "encryption": "false",
        "unixPermissions": "0775"
    }
}
]
}

```

Schritt: Ein Kubernetes Secret erstellen

Erstellen Sie einen geheimen Schlüssel, der die Anmeldeinformationen für das Backend enthält, wie in diesem Beispiel gezeigt:

```

cat tbc-ontap-nas-backend-secret.yaml

apiVersion: v1
kind: Secret
metadata:
  name: ontap-nas-backend-secret
type: Opaque
stringData:
  username: cluster-admin
  password: admin-password

kubectl create -f tbc-ontap-nas-backend-secret.yaml -n trident
secret/backend-tbc-ontap-san-secret created

```

Schritt 2: Erstellen eines `TridentBackendConfig` CR

Im nächsten Schritt wird ein CR erstellt `TridentBackendConfig`, der automatisch an das bereits vorhandene bindet `ontap-nas-backend` (wie in diesem Beispiel). Stellen Sie sicher, dass folgende Anforderungen erfüllt sind:

- Der gleiche Backend-Name ist in definiert `spec.backendName`.
- Die Konfigurationsparameter sind mit dem ursprünglichen Back-End identisch.
- Virtuelle Pools (falls vorhanden) müssen dieselbe Reihenfolge wie im ursprünglichen Backend beibehalten.
- Anmeldedaten werden bei einem Kubernetes Secret und nicht im Klartext bereitgestellt.

In diesem Fall sieht das `TridentBackendConfig` wie folgt aus:

```

cat backend-tbc-ontap-nas.yaml
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: tbc-ontap-nas-backend
spec:
  version: 1
  storageDriverName: ontap-nas
  managementLIF: 10.10.10.1
  dataLIF: 10.10.10.2
  backendName: ontap-nas-backend
  svm: trident_svm
  credentials:
    name: mysecret
  defaults:
    spaceReserve: none
    encryption: 'false'
  labels:
    store: nas_store
  region: us_east_1
  storage:
  - labels:
      app: msoffice
      cost: '100'
      zone: us_east_1a
      defaults:
        spaceReserve: volume
        encryption: 'true'
        unixPermissions: '0755'
  - labels:
      app: mysqldb
      cost: '25'
      zone: us_east_1d
      defaults:
        spaceReserve: volume
        encryption: 'false'
        unixPermissions: '0775'

kubectl create -f backend-tbc-ontap-nas.yaml -n trident
tridentbackendconfig.trident.netapp.io/tbc-ontap-nas-backend created

```

Schritt 3: Überprüfen Sie den Status des `TridentBackendConfig` CR

Nachdem der `TridentBackendConfig` erstellt wurde, muss seine Phase sein `Bound`. Sie sollte außerdem den gleichen Backend-Namen und die gleiche UUID wie das vorhandene Backend widerspiegeln.

```
kubectl get tbc tbc-ontap-nas-backend -n trident
```

NAME	BACKEND NAME	BACKEND UUID
tbc-ontap-nas-backend	ontap-nas-backend	52f2eb10-e4c6-4160-99fc-96b3be5ab5d7

```
tridentctl get backend -n trident
```

NAME	STATE	VOLUMES	STORAGE DRIVER	UUID
ontap-nas-backend	online	25	ontap-nas	52f2eb10-e4c6-4160-99fc-96b3be5ab5d7

Das Backend wird nun vollständig über das Objekt verwaltet tbc-ontap-nas-backend TridentBackendConfig.

Managen von TridentBackendConfig Back-Ends mit tridentctl

`tridentctl` Kann verwendet werden, um Back-Ends aufzulisten, die mit erstellt wurden `TridentBackendConfig`. Darüber hinaus können Administratoren auch wählen, um vollständig verwalten solche Back-Ends durch durch `tridentctl` Löschen `TridentBackendConfig` und sicherstellen, `spec.deletionPolicy` ist auf gesetzt `retain`.

Schritt 0: Identifizieren Sie das Backend

Nehmen wir zum Beispiel an, dass das folgende Backend mit erzeugt wurde TridentBackendConfig:

```
kubectl get tbc backend-tbc-ontap-san -n trident -o wide
NAME                                BACKEND NAME                BACKEND UUID
PHASE    STATUS    STORAGE DRIVER    DELETION POLICY
backend-tbc-ontap-san    ontap-san-backend    81abcb27-ea63-49bb-b606-
0a5315ac5f82    Bound    Success    ontap-san    delete

tridentctl get backend ontap-san-backend -n trident
+-----+-----+
+-----+-----+-----+-----+
|      NAME      | STORAGE DRIVER |                      UUID
| STATE  | VOLUMES |
+-----+-----+
+-----+-----+-----+-----+
| ontap-san-backend | ontap-san      | 81abcb27-ea63-49bb-b606-
0a5315ac5f82 | online |      33 |
+-----+-----+
+-----+-----+-----+-----+
```

Aus der Ausgabe wird ersichtlich, dass sie `TridentBackendConfig` erfolgreich erstellt wurde und an ein Backend gebunden ist [Observe the Backend's UUID].

Schritt 1: Bestätigen `deletionPolicy` ist auf eingestellt `retain`

Lassen Sie uns einen Blick auf den Wert von `deletionPolicy`. Dies muss auf eingestellt werden `retain`. Dadurch wird sichergestellt, dass beim Löschen eines `TridentBackendConfig` CR die Backend-Definition weiterhin vorhanden ist und mit verwaltet werden kann `tridentctl`.

```
kubectl get tbc backend-tbc-ontap-san -n trident -o wide
NAME                                BACKEND NAME                BACKEND UUID
PHASE    STATUS    STORAGE DRIVER    DELETION POLICY
backend-tbc-ontap-san    ontap-san-backend    81abcb27-ea63-49bb-b606-
0a5315ac5f82    Bound    Success    ontap-san    delete

# Patch value of deletionPolicy to retain
kubectl patch tbc backend-tbc-ontap-san --type=merge -p
'{"spec":{"deletionPolicy":"retain"}}' -n trident
tridentbackendconfig.trident.netapp.io/backend-tbc-ontap-san patched

#Confirm the value of deletionPolicy
kubectl get tbc backend-tbc-ontap-san -n trident -o wide
NAME                                BACKEND NAME                BACKEND UUID
PHASE    STATUS    STORAGE DRIVER    DELETION POLICY
backend-tbc-ontap-san    ontap-san-backend    81abcb27-ea63-49bb-b606-
0a5315ac5f82    Bound    Success    ontap-san    retain
```



Fahren Sie nicht mit dem nächsten Schritt fort, es sei denn, es `deletionPolicy` ist auf `retain`.

Schritt 2: Löschen Sie den `TridentBackendConfig` CR

Der letzte Schritt besteht darin, den CR zu löschen `TridentBackendConfig`. Nach der Bestätigung, dass der `deletionPolicy` auf `gesetzt` ist `retain`, können Sie mit dem Löschen fortfahren:

```
kubectl delete tbc backend-tbc-ontap-san -n trident
tridentbackendconfig.trident.netapp.io "backend-tbc-ontap-san" deleted

tridentctl get backend ontap-san-backend -n trident
+-----+-----+
+-----+-----+-----+-----+
|          NAME          | STORAGE DRIVER |                               UUID
| STATE  | VOLUMES |
+-----+-----+
+-----+-----+-----+-----+
| ontap-san-backend | ontap-san      | 81abcb27-ea63-49bb-b606-
0a5315ac5f82 | online |      33 |
+-----+-----+
+-----+-----+-----+-----+
```

Beim Löschen des `TridentBackendConfig` Objekts entfernt Trident es einfach, ohne das Backend selbst zu löschen.

Copyright-Informationen

Copyright © 2026 NetApp. Alle Rechte vorbehalten. Gedruckt in den USA. Dieses urheberrechtlich geschützte Dokument darf ohne die vorherige schriftliche Genehmigung des Urheberrechtsinhabers in keiner Form und durch keine Mittel – weder grafische noch elektronische oder mechanische, einschließlich Fotokopieren, Aufnehmen oder Speichern in einem elektronischen Abrufsystem – auch nicht in Teilen, vervielfältigt werden.

Software, die von urheberrechtlich geschütztem NetApp Material abgeleitet wird, unterliegt der folgenden Lizenz und dem folgenden Haftungsausschluss:

DIE VORLIEGENDE SOFTWARE WIRD IN DER VORLIEGENDEN FORM VON NETAPP ZUR VERFÜGUNG GESTELLT, D. H. OHNE JEGLICHE EXPLIZITE ODER IMPLIZITE GEWÄHRLEISTUNG, EINSCHLIESSLICH, JEDOCH NICHT BESCHRÄNKT AUF DIE STILLSCHWEIGENDE GEWÄHRLEISTUNG DER MARKTGÄNGIGKEIT UND EIGNUNG FÜR EINEN BESTIMMTEN ZWECK, DIE HIERMIT AUSGESCHLOSSEN WERDEN. NETAPP ÜBERNIMMT KEINERLEI HAFTUNG FÜR DIREKTE, INDIREKTE, ZUFÄLLIGE, BESONDERE, BEISPIELHAFTE SCHÄDEN ODER FOLGESCHÄDEN (EINSCHLIESSLICH, JEDOCH NICHT BESCHRÄNKT AUF DIE BESCHAFFUNG VON ERSATZWAREN ODER -DIENSTLEISTUNGEN, NUTZUNGS-, DATEN- ODER GEWINNVERLUSTE ODER UNTERBRECHUNG DES GESCHÄFTSBETRIEBS), UNABHÄNGIG DAVON, WIE SIE VERURSACHT WURDEN UND AUF WELCHER HAFTUNGSTHEORIE SIE BERUHEN, OB AUS VERTRAGLICH FESTGELEGTER HAFTUNG, VERSCHULDENSUNABHÄNGIGER HAFTUNG ODER DELIKTSHAFTUNG (EINSCHLIESSLICH FAHRLÄSSIGKEIT ODER AUF ANDEREM WEGE), DIE IN IRGEND EINER WEISE AUS DER NUTZUNG DIESER SOFTWARE RESULTIEREN, SELBST WENN AUF DIE MÖGLICHKEIT DERARTIGER SCHÄDEN HINGEWIESEN WURDE.

NetApp behält sich das Recht vor, die hierin beschriebenen Produkte jederzeit und ohne Vorankündigung zu ändern. NetApp übernimmt keine Verantwortung oder Haftung, die sich aus der Verwendung der hier beschriebenen Produkte ergibt, es sei denn, NetApp hat dem ausdrücklich in schriftlicher Form zugestimmt. Die Verwendung oder der Erwerb dieses Produkts stellt keine Lizenzierung im Rahmen eines Patentrechts, Markenrechts oder eines anderen Rechts an geistigem Eigentum von NetApp dar.

Das in diesem Dokument beschriebene Produkt kann durch ein oder mehrere US-amerikanische Patente, ausländische Patente oder anhängige Patentanmeldungen geschützt sein.

ERLÄUTERUNG ZU „RESTRICTED RIGHTS“: Nutzung, Vervielfältigung oder Offenlegung durch die US-Regierung unterliegt den Einschränkungen gemäß Unterabschnitt (b)(3) der Klausel „Rights in Technical Data – Noncommercial Items“ in DFARS 252.227-7013 (Februar 2014) und FAR 52.227-19 (Dezember 2007).

Die hierin enthaltenen Daten beziehen sich auf ein kommerzielles Produkt und/oder einen kommerziellen Service (wie in FAR 2.101 definiert) und sind Eigentum von NetApp, Inc. Alle technischen Daten und die Computersoftware von NetApp, die unter diesem Vertrag bereitgestellt werden, sind gewerblicher Natur und wurden ausschließlich unter Verwendung privater Mittel entwickelt. Die US-Regierung besitzt eine nicht ausschließliche, nicht übertragbare, nicht unterlizenzierbare, weltweite, limitierte unwiderrufliche Lizenz zur Nutzung der Daten nur in Verbindung mit und zur Unterstützung des Vertrags der US-Regierung, unter dem die Daten bereitgestellt wurden. Sofern in den vorliegenden Bedingungen nicht anders angegeben, dürfen die Daten ohne vorherige schriftliche Genehmigung von NetApp, Inc. nicht verwendet, offengelegt, vervielfältigt, geändert, aufgeführt oder angezeigt werden. Die Lizenzrechte der US-Regierung für das US-Verteidigungsministerium sind auf die in DFARS-Klausel 252.227-7015(b) (Februar 2014) genannten Rechte beschränkt.

Markeninformationen

NETAPP, das NETAPP Logo und die unter <http://www.netapp.com/TM> aufgeführten Marken sind Marken von NetApp, Inc. Andere Firmen und Produktnamen können Marken der jeweiligen Eigentümer sein.