



# **Google Cloud NetApp Volumes**

**Trident**

NetApp  
January 15, 2026

This PDF was generated from <https://docs.netapp.com/de-de/trident-2506/trident-use/gcnv.html> on January 15, 2026. Always check [docs.netapp.com](https://docs.netapp.com) for the latest.

# Inhalt

Google Cloud NetApp Volumes . . . . .	1
Konfigurieren eines Google Cloud NetApp Volumes -Backends . . . . .	1
Details zum Google Cloud NetApp Volumes -Treiber . . . . .	1
Cloud-Identität für GKE . . . . .	1
Bereiten Sie die Konfiguration eines Google Cloud NetApp Volumes Backends vor. . . . .	3
Voraussetzungen für NFS-Volumes . . . . .	3
Google Cloud NetApp Volumes Backend-Konfigurationsoptionen und Beispiele . . . . .	3
Backend-Konfigurationsoptionen . . . . .	3
Volumenbereitstellungsoptionen . . . . .	5
Beispielkonfigurationen . . . . .	5
Wie geht es weiter? . . . . .	13
Speicherklassendefinitionen . . . . .	14

# Google Cloud NetApp Volumes

## Konfigurieren eines Google Cloud NetApp Volumes -Backends

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

### Details zum Google Cloud NetApp Volumes -Treiber

Trident bietet die `google-cloud-netapp-volumes` Der Treiber soll mit dem Cluster kommunizieren. Unterstützte Zugriffsmodi sind: `ReadWriteOnce` (RWO), `ReadOnlyMany` (ROX), `ReadWriteMany` (RWX), `ReadWriteOncePod` (RWOP).

Treiber	Protokoll	Lautstärke modus	Unterstützte Zugriffsmodi	Unterstützte Dateisysteme
<code>google-cloud-netapp-volumes</code>	NFS SMB	Dateisystem	RWO, ROX, RWX, RWOP	nfs, smb

### Cloud-Identität für GKE

Cloud Identity ermöglicht es Kubernetes-Pods, auf Google Cloud-Ressourcen zuzugreifen, indem sie sich als Workload-Identität authentifizieren, anstatt explizite Google Cloud-Anmeldeinformationen anzugeben.

Um die Cloud-Identität in Google Cloud nutzen zu können, benötigen Sie Folgendes:

- Ein mit GKE bereitgestellter Kubernetes-Cluster.
- Die Workload-Identität wurde im GKE-Cluster konfiguriert und der GKE MetaData Server auf den Knotenpools.
- Ein GCP-Dienstkonto mit der Rolle „Google Cloud NetApp Volumes Admin“ (`roles/netapp.admin`) oder einer benutzerdefinierten Rolle.
- Trident wurde installiert, einschließlich des Cloud-Providers, der "GCP" angibt, und der Cloud-Identität, die das neue GCP-Dienstkonto angibt. Nachfolgend ein Beispiel.

## Trident -Betreiber

Um Trident mithilfe des Trident -Operators zu installieren, bearbeiten Sie `tridentoperator_cr.yaml` einstellen `cloudProvider` Zu "GCP" und setzen `cloudIdentity` Zu `iam.gke.io/gcp-service-account: cloudvolumes-admin-sa@mygcpproject.iam.gserviceaccount.com`.

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 die Flags **cloud-provider (CP)** und **cloud-identity (CI)** mithilfe 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 konfiguriert es. `cloudProvider` für GCP unter Verwendung der Umgebungsvariablen `$CP` und stellt die `cloudIdentity` unter Verwendung der Umgebungsvariablen `$ANNOTATION` :

```
helm install trident trident-operator-100.6.0.tgz --set
cloudProvider=$CP --set cloudIdentity="$ANNOTATION"
```

## <code>tridentctl</code>

Legen Sie die Werte für die Flags **Cloud-Anbieter** und **Cloud-Identität** mithilfe 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 konfiguriert die `cloud-provider` Flagge an `$CP` , Und `cloud-identity` Zu `$ANNOTATION` :

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

## Bereiten Sie die Konfiguration eines Google Cloud NetApp Volumes Backends vor.

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

### Voraussetzungen für NFS-Volumes

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

Stellen Sie sicher, dass Sie Folgendes haben, bevor Sie das Google Cloud NetApp Volumes -Backend konfigurieren:

- Ein Google Cloud-Konto, das mit dem Google Cloud NetApp Volumes -Dienst konfiguriert ist. Siehe "[Google Cloud NetApp Volumes](#)".
- Projektnummer Ihres Google Cloud-Kontos. Siehe "[Projekte identifizieren](#)".
- Ein Google Cloud-Dienstkonto mit NetApp Volumes-Administratorrechten(`roles/netapp.admin`) Rolle. Siehe "[Rollen und Berechtigungen für Identitäts- und Zugriffsmanagement](#)".
- API-Schlüsseldatei für Ihr GCNV-Konto. Siehe "[Erstellen Sie einen Dienstkontoschlüssel](#)".
- Ein Speicherbecken. Siehe "[Übersicht der Speicherpools](#)".

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

## Google Cloud NetApp Volumes Backend-Konfigurationsoptionen und Beispiele

Erfahren Sie mehr über die Backend-Konfigurationsoptionen für Google Cloud NetApp Volumes und sehen Sie sich Konfigurationsbeispiele an.

### Backend-Konfigurationsoptionen

Jedes Backend stellt Volumes in einer einzelnen Google Cloud-Region bereit. Um Volumes in anderen Regionen zu erstellen, können Sie zusätzliche Backends definieren.

Parameter	Beschreibung	Standard
version		Immer 1

Parameter	Beschreibung	Standard
storageDriverName	Name des Speichertreibers	Der Wert von storageDriverName muss als "google-cloud-netapp-volumes" angegeben werden.
backendName	(Optional) Benutzerdefinierter Name des Speicher-Backends	Fahrername + "_" + Teil des API-Schlüssels
storagePools	Optionaler Parameter zur Angabe von Speicherpools für die Volume-Erstellung.	
projectNumber	Google Cloud-Konto-Projektnummer. Den Wert finden Sie auf der Startseite des Google Cloud-Portals.	
location	Der Google Cloud-Speicherort, an dem Trident GCNV-Volumes erstellt. Beim Erstellen regionsübergreifender Kubernetes-Cluster werden Volumes, die in einem location kann in Workloads verwendet werden, die auf Knoten in mehreren Google Cloud-Regionen geplant sind. Für den Verkehr über Regionen hinweg fallen zusätzliche Kosten an.	
apiKey	API-Schlüssel für das Google Cloud-Dienstkonto mit dem netapp.admin Rolle. Sie enthält den JSON-formatierten Inhalt der privaten Schlüsseldatei eines Google Cloud-Dienstkontos (wörtlich in die Backend-Konfigurationsdatei kopiert). Der apiKey müssen Schlüssel-Wert-Paare für die folgenden Schlüssel enthalten: type , project_id , client_email , client_id , auth_uri , token_uri , auth_provider_x509_cert_url , Und client_x509_cert_url .	
nfsMountOptions	Feingranulare Steuerung der NFS-Mount-Optionen.	"nfsvers=3"
limitVolumeSize	Die Bereitstellung schlägt fehl, wenn die angeforderte Volume-Größe diesen Wert überschreitet.	"" (wird nicht standardmäßig erzwungen)
serviceLevel	Der Servicegrad eines Speicherpools und seiner Volumes. Die Werte sind flex , standard , premium , oder extreme .	
labels	Satz beliebiger JSON-formatierter Bezeichnungen, die auf Datenträger angewendet werden sollen	""
network	Das Google Cloud-Netzwerk wird für GCNV-Volumes verwendet.	
debugTraceFlags	Debug-Flags zur Verwendung bei der Fehlersuche. Beispiel, { "api":false, "method":true} . Verwenden Sie diese Funktion nur, wenn Sie eine Fehlerbehebung durchführen und einen detaillierten Protokollauszug benötigen.	null

Parameter	Beschreibung	Standard
nasType	Konfiguration der Erstellung von NFS- oder SMB-Volumes. Optionen sind nfs , smb oder null. Bei der Einstellung „null“ werden standardmäßig NFS-Volumes verwendet.	nfs
supportedTopologies	Stellt eine Liste der Regionen und Zonen dar, die von diesem Backend unterstützt werden. Weitere Informationen finden Sie unter " <a href="#">CSI-Topologie verwenden</a> ". Zum Beispiel: supportedTopologies: - topology.kubernetes.io/region: asia-east1 topology.kubernetes.io/zone: asia-east1-a	

## Volumenbereitstellungsoptionen

Sie können die Standard-Volume-Bereitstellung in der defaults Abschnitt der Konfigurationsdatei.

Parameter	Beschreibung	Standard
exportRule	Die Ausfuhrbestimmungen für neue Bände. Es muss sich um eine durch Kommas getrennte Liste beliebiger Kombinationen von IPv4-Adressen handeln.	"0.0.0.0/0"
snapshotDir	Zugang zu .snapshot Verzeichnis	"true" für NFSv4, "false" für NFSv3
snapshotReserve	Prozentsatz des für Snapshots reservierten Speichervolumens	"" (Standardwert 0 akzeptieren)
unixPermissions	Die Unix-Berechtigungen für neue Datenträger (4 Oktalstellen).	""

## Beispielkonfigurationen

Die folgenden Beispiele zeigen Basiskonfigurationen, bei denen die meisten Parameter auf Standardwerte eingestellt bleiben. Dies ist die einfachste Möglichkeit, ein Backend zu definieren.

## Minimale Konfiguration

Dies ist die absolute Minimalkonfiguration im Backend. Mit dieser Konfiguration erkennt Trident alle Ihre Speicherpools, die an Google Cloud NetApp Volumes delegiert sind, am konfigurierten Speicherort und platziert neue Volumes zufällig in einem dieser Pools. Weil `nasType` wird ausgelassen, die `nfs` Es gelten die Standardeinstellungen, und das Backend stellt NFS-Volumes bereit.

Diese Konfiguration ist ideal, wenn Sie gerade erst mit Google Cloud NetApp Volumes beginnen und verschiedene Funktionen ausprobieren möchten. In der Praxis werden Sie jedoch höchstwahrscheinlich zusätzliche Einschränkungen für die von Ihnen bereitgestellten Volumes benötigen.

```

---
apiVersion: v1
kind: Secret
metadata:
  name: backend-tbc-gcnv-secret
type: Opaque
stringData:
  private_key_id: f2cb6ed6d7cc10c453f7d3406fc700c5df0ab9ec
  private_key: |
    -----BEGIN PRIVATE KEY-----\n
    znHczZsrrtHisIsAbOguSaPIKeyAZNchRAGzlzzE4jK3b1/qp8B4Kws8zX5ojY9m\n
    znHczZsrrtHisIsAbOguSaPIKeyAZNchRAGzlzzE4jK3b1/qp8B4Kws8zX5ojY9m\n
    znHczZsrrtHisIsAbOguSaPIKeyAZNchRAGzlzzE4jK3b1/qp8B4Kws8zX5ojY9m\n
    XsYg6gyxy4zq70lwWgLwGa==\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

```

## Konfiguration für SMB-Volumes

```
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-tbc-gcnv1
  namespace: trident
spec:
  version: 1
  storageDriverName: google-cloud-netapp-volumes
  projectNumber: "123456789"
  location: asia-east1
  serviceLevel: flex
  nasType: smb
  apiKey:
    type: service_account
    project_id: cloud-native-data
    client_email: trident-sample@cloud-native-
data.iam.gserviceaccount.com
    client_id: "123456789737813416734"
    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/trident-
sample%40cloud-native-data.iam.gserviceaccount.com
  credentials:
    name: backend-tbc-gcnv-secret
```

## Konfiguration mit StoragePools-Filter

```

-----
apiVersion: v1
kind: Secret
metadata:
  name: backend-tbc-gcnv-secret
type: Opaque
stringData:
  private_key_id: f2cb6ed6d7cc10c453f7d3406fc700c5df0ab9ec
  private_key: |
    -----BEGIN PRIVATE KEY-----
    znHczZsrrtHisIsAbOguSaPIKeyAZNchRAGzlzzE4jK3b1/qp8B4Kws8zX5ojY9m
    znHczZsrrtHisIsAbOguSaPIKeyAZNchRAGzlzzE4jK3b1/qp8B4Kws8zX5ojY9m
    znHczZsrrtHisIsAbOguSaPIKeyAZNchRAGzlzzE4jK3b1/qp8B4Kws8zX5ojY9m
    XsYg6gyxy4zq70lwWgLwGa==
    -----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
  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 eines virtuellen Pools

Diese Backend-Konfiguration definiert mehrere virtuelle Pools in einer einzigen Datei. Virtuelle Pools werden definiert in der storage Abschnitt. Sie sind nützlich, wenn Sie mehrere Speicherpools haben, die unterschiedliche Servicelevel unterstützen, und Sie Speicherklassen in Kubernetes erstellen möchten, die diese repräsentieren. Virtuelle Poolbezeichnungen dienen zur Unterscheidung der Pools. Zum Beispiel im folgenden Beispiel performance Etikett und serviceLevel Der Typ wird verwendet, um virtuelle Pools zu unterscheiden.

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

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

```
---  
apiVersion: v1  
kind: Secret  
metadata:  
  name: backend-tbc-gcnv-secret  
type: Opaque  
stringData:  
  private_key_id: f2cb6ed6d7cc10c453f7d3406fc700c5df0ab9ec  
  private_key: |  
    -----BEGIN PRIVATE KEY-----  
    znHczZsrrtHisIsAbOguSaPIKeyAZNchRAGzlzzE4jK3b1/qp8B4Kws8zX5ojY9m  
    znHczZsrrtHisIsAbOguSaPIKeyAZNchRAGzlzzE4jK3b1/qp8B4Kws8zX5ojY9m  
    znHczZsrrtHisIsAbOguSaPIKeyAZNchRAGzlzzE4jK3b1/qp8B4Kws8zX5ojY9m  
    znHczZsrrtHisIsAbOguSaPIKeyAZNchRAGzlzzE4jK3b1/qp8B4Kws8zX5ojY9m  
    XsYg6gyxy4zq70lwWgLwGa==  
    -----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

```

## Unterstützte Topologiekonfiguration

Trident ermöglicht die Bereitstellung von Volumes für Workloads basierend auf Regionen und Verfügbarkeitszonen. Der `supportedTopologies` Block in dieser Backend-Konfiguration dient dazu, eine Liste von Regionen und Zonen pro Backend bereitzustellen. Die hier angegebenen Regions- und Zonenwerte müssen mit den Regions- und Zonenwerten der Labels auf jedem Kubernetes-Clusterknoten übereinstimmen. Diese Regionen und Zonen stellen die Liste der zulässigen Werte dar, die in einer Speicherklasse angegeben werden können. Für Speicherklassen, die eine Teilmenge der im Backend bereitgestellten Regionen und Zonen enthalten, erstellt Trident Volumes in der genannten Region und Zone. Weitere Informationen finden Sie unter "[CSI-Topologie verwenden](#)".

```
---  
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
```

## Wie geht es weiter?

Nachdem Sie die Backend-Konfigurationsdatei erstellt haben, führen Sie folgenden Befehl aus:

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

Um zu überprüfen, ob das Backend erfolgreich erstellt wurde, führen Sie folgenden Befehl aus:

```
kubectl get tridentbackendconfig  
  
NAME          BACKEND NAME      BACKEND UUID  
PHASE  STATUS  
backend-tbc-gcnv  backend-tbc-gcnv  b2fd1ff9-b234-477e-88fd-713913294f65  
Bound   Success
```

Wenn die Backend-Erstellung fehlschlägt, stimmt etwas mit der Backend-Konfiguration nicht. Sie können das Backend mithilfe des folgenden beschreiben: `kubectl get tridentbackendconfig <backend-name>` Um die Ursache zu ermitteln, führen Sie den folgenden Befehl aus oder überprüfen Sie die Protokolle:

```
tridentctl logs
```

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

## Speicherklassendefinitionen

Im Folgenden finden Sie eine grundlegende StorageClass Definition, die sich auf das oben genannte Backend 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 unter Verwendung der `parameter.selector` Feld:

Verwenden `parameter.selector` Sie können für jedes einzelne festlegen StorageClass Die "virtueller Pool" Das wird zum Hosten eines Volumes verwendet. Das Volumen wird die im gewählten Pool definierten Aspekte aufweisen.

```

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 Einzelheiten zu Speicherklassen finden Sie unter "[Erstellen einer Speicherklasse](#)".

### **Beispieldefinitionen für SMB-Volumes**

Verwenden `nasType`, `node-stage-secret-name`, Und `node-stage-secret-namespace` Sie können ein SMB-Volume angeben und die erforderlichen Active Directory-Anmeldeinformationen bereitstellen. Für das Node-Stage-Secret kann jeder Active Directory-Benutzer/jedes Passwort mit beliebigen/keinen Berechtigungen verwendet werden.

## Grundkonfiguration im Standard-Namespace

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

## Verwendung unterschiedlicher Geheimnisse pro Namensraum

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

## Verwendung unterschiedlicher Geheimnisse pro Band

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

## PVC-Definitionsbeispiel

```
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-Verbindung hergestellt ist, führen Sie folgenden Befehl aus:

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

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

## **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 BESCHRÄNKKT 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 BESCHRÄNKKT 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 IRGENDERWEINER 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.