Sicherheit

Trident

NetApp
July 01, 2026

Inhalt

Sicherheit

Sicherheit

Verwenden Sie die hier aufgeführten Empfehlungen, um sicherzustellen, dass Ihre Trident-Installation sicher ist.

Trident in einem eigenen Namensraum ausführen

Es ist wichtig, Anwendungen, Anwendungsadministratoren, Benutzern und Verwaltungsanwendungen den Zugriff auf Trident-Objektdefinitionen oder die Pods zu verwehren, um eine zuverlässige Speicherung zu gewährleisten und potenziell schädliche Aktivitäten zu verhindern.

Um andere Anwendungen und Benutzer von Trident zu trennen, installieren Sie Trident immer in einem eigenen Kubernetes-Namespace (`trident`). Wenn Trident in einem eigenen Namespace installiert wird, wird sichergestellt, dass nur das Kubernetes-Administrationspersonal Zugriff auf den Trident-Pod und die in den Namespaced-CRD-Objekten gespeicherten Artefakte (wie Backend- und CHAP-Secrets, falls zutreffend) hat. Sie sollten sicherstellen, dass nur Administratoren Zugriff auf den Trident-Namespace und damit auf die `tridentctl` Anwendung haben.

CHAP Authentifizierung mit ONTAP SAN-Backends verwenden

Trident unterstützt CHAP-basierte Authentifizierung für ONTAP SAN-Workloads (unter Verwendung der `ontap-san` und `ontap-san-economy` Treiber). NetApp empfiehlt die Verwendung von bidirektionalem CHAP mit Trident für die Authentifizierung zwischen einem Host und dem Storage-Backend.

Für ONTAP-Backends, die die SAN-Speichertreiber verwenden, kann Trident bidirektionales CHAP einrichten und CHAP-Benutzernamen und -Geheimnisse über `tridentctl` verwalten. Siehe "Bereiten Sie die Konfiguration des Backends mit ONTAP SAN-Treibern vor", um zu verstehen, wie Trident CHAP auf ONTAP-Backends konfiguriert.

Verwenden Sie die CHAP-Authentifizierung mit NetApp HCI und SolidFire Backends

NetApp empfiehlt die Bereitstellung von bidirektionalem CHAP, um die Authentifizierung zwischen einem Host und den NetApp HCI- und SolidFire-Backends sicherzustellen. Trident verwendet ein Secret-Objekt, das zwei CHAP-Passwörter pro Mandant enthält. Wenn Trident installiert ist, verwaltet es die CHAP-Secrets und speichert sie in einem `tridentvolume` CR-Objekt für das jeweilige PV. Wenn Sie ein PV erstellen, verwendet Trident die CHAP-Secrets, um eine iSCSI-Sitzung zu initiieren und über CHAP mit dem NetApp HCI- und SolidFire-System zu kommunizieren.

Die Volumes, die von Trident erstellt werden, sind keiner Volume-Zugriffsgruppe zugeordnet.

Verwenden Sie Trident mit NVE und NAE

NetApp ONTAP bietet Verschlüsselung ruhender Daten, um sensible Daten im Falle von Diebstahl, Rückgabe oder anderweitiger Verwendung einer Festplatte zu schützen. Weitere Informationen finden Sie unter "Konfigurieren Sie die Übersicht zur NetApp Volume Encryption".

- Wenn NAE im Backend aktiviert ist, wird jedes in Trident bereitgestellte Volume NAE-fähig sein.

    - Sie können das NVE-Verschlüsselungsflag auf `""` setzen, um NAE-fähige Volumes zu erstellen.
- Wenn NAE auf dem Backend nicht aktiviert ist, wird jedes in Trident bereitgestellte Volume NVE-fähig sein, es sei denn, das NVE-Verschlüsselungsflag ist auf `false` (den Standardwert) in der Backend-Konfiguration gesetzt.

> Volumes, die in Trident auf einem NAE-fähigen Backend erstellt wurden, müssen NVE- oder NAE-verschlüsselt sein.
>
> - Sie können das NVE-Verschlüsselungsflag auf `true` in der Trident-Backend-Konfiguration setzen, um die NAE-Verschlüsselung zu überschreiben und einen spezifischen Verschlüsselungsschlüssel pro Volume zu verwenden.
> - Das Setzen des NVE-Verschlüsselungsflags auf `false` einem NAE-fähigen Backend erstellt ein NAE-fähiges Volume. Sie können die NAE-Verschlüsselung nicht deaktivieren, indem Sie das NVE-Verschlüsselungsflag auf `false` setzen.

- Sie können ein NVE-Volume in Trident manuell erstellen, indem Sie das NVE-Verschlüsselungsflag explizit auf `true` setzen.

Weitere Informationen zu den Backend-Konfigurationsoptionen finden Sie unter:

- "ONTAP SAN-Konfigurationsoptionen"
- "ONTAP NAS-Konfigurationsoptionen"

Linux Unified Key Setup (LUKS)

Sie können Linux Unified Key Setup (LUKS) aktivieren, um ONTAP SAN- und ONTAP SAN ECONOMY-Volumes auf Trident zu verschlüsseln. Trident unterstützt die Rotation von Passphrasen und die Volume-Erweiterung für LUKS-verschlüsselte Volumes.

In Trident verwenden LUKS-verschlüsselte Volumes die aes-xts-plain64-Verschlüsselung und den Modus, wie empfohlen von "NIST".

> LUKS-Verschlüsselung wird für ASA r2-Systeme nicht unterstützt. Weitere Informationen zu ASA r2-Systemen finden Sie unter "Erfahren Sie mehr über ASA r2-Speichersysteme".

Bevor Sie beginnen

- Auf den Worker-Knoten muss cryptsetup Version 2.1 oder höher (aber niedriger als 3.0) installiert sein. Weitere Informationen finden Sie unter "Gitlab: cryptsetup".
- Aus Leistungsgründen empfiehlt NetApp, dass Worker-Knoten Advanced Encryption Standard New Instructions (AES-NI) unterstützen. Um die AES-NI-Unterstützung zu überprüfen, führen Sie den folgenden Befehl aus:

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

  Wird keine Antwort zurückgegeben, unterstützt Ihr Prozessor kein AES-NI. Weitere Informationen zu AES-NI finden Sie unter: "Intel: Advanced Encryption Standard Instructions (AES-NI)".

LUKS-Verschlüsselung aktivieren

Sie können die volumenbezogene, hostseitige Verschlüsselung mithilfe von Linux Unified Key Setup (LUKS) für ONTAP SAN und ONTAP SAN ECONOMY Volumes aktivieren.

Schritte

1. Definieren Sie die LUKS-Verschlüsselungsattribute in der Backend-Konfiguration. Weitere Informationen zu den Backend-Konfigurationsoptionen für ONTAP SAN finden Sie unter "ONTAP SAN-Konfigurationsoptionen".

```
{
  "storage": [
    {
      "labels": {
        "luks": "true"
      },
      "zone": "us_east_1a",
      "defaults": {
        "luksEncryption": "true"
      }
    },
    {
      "labels": {
        "luks": "false"
      },
      "zone": "us_east_1a",
      "defaults": {
        "luksEncryption": "false"
      }
    }
  ]
}
```

2. Verwenden Sie `parameters.selector` zur Definition der Speicherpools mit LUKS-Verschlüsselung. Beispiel:

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: luks
provisioner: csi.trident.netapp.io
parameters:
  selector: "luks=true"
  csi.storage.k8s.io/node-stage-secret-name: luks-${pvc.name}
  csi.storage.k8s.io/node-stage-secret-namespace: ${pvc.namespace}
```

3. Erstellen Sie ein Geheimnis, das die LUKS-Passphrase enthält. Beispiel:

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

Einschränkungen

LUKS-verschlüsselte Datenträger können die Deduplizierung und Komprimierung von ONTAP nicht nutzen.

Backend-Konfiguration für den Import von LUKS-Volumes

Um ein LUKS-Volume zu importieren, müssen Sie `luksEncryption` auf (`true` im Backend setzen. Die `luksEncryption` Option teilt Trident mit, ob das Volume LUKS-kompatibel (`true` ist oder nicht LUKS-kompatibel (`false`, wie im folgenden Beispiel gezeigt.

```
version: 1
storageDriverName: ontap-san
managementLIF: 10.0.0.1
dataLIF: 10.0.0.2
svm: trident_svm
username: admin
password: password
defaults:
  luksEncryption: 'true'
  spaceAllocation: 'false'
  snapshotPolicy: default
  snapshotReserve: '10'
```

PVC-Konfiguration für den Import von LUKS-Volumes

Um LUKS-Volumes dynamisch zu importieren, setzen Sie die Annotation `trident.netapp.io/luksEncryption` auf `true` und fügen Sie eine LUKS-fähige Speicherklasse in die PVC ein, wie in diesem Beispiel gezeigt.

```
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: luks-pvc
  namespace: trident
  annotations:
    trident.netapp.io/luksEncryption: "true"
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 1Gi
  storageClassName: luks-sc
```

Eine LUKS-Passphrase rotieren

Sie können die LUKS-Passphrase rotieren und die Rotation bestätigen.

> Vergessen Sie eine Passphrase erst, nachdem Sie überprüft haben, dass sie von keinem Volume, Snapshot oder Secret mehr referenziert wird. Geht eine referenzierte Passphrase verloren, können Sie das Volume möglicherweise nicht einbinden und die Daten bleiben verschlüsselt und unzugänglich.

Über diese Aufgabe

Die Rotation der LUKS-Passphrase erfolgt, wenn ein Pod, der das Volume einbindet, erstellt wird, nachdem eine neue LUKS-Passphrase festgelegt wurde. Wenn ein neuer Pod erstellt wird, vergleicht Trident die LUKS-Passphrase auf dem Volume mit der aktiven Passphrase im Secret.

- Stimmt die Passphrase des Volumes nicht mit der aktiven Passphrase im Secret überein, findet eine Rotation statt.
- Wenn die Passphrase des Volumes mit der aktiven Passphrase im Geheimnis übereinstimmt, wird der `previous-luks-passphrase` Parameter ignoriert.

Schritte

1. Fügen Sie die `node-publish-secret-name` und `node-publish-secret-namespace` StorageClass-Parameter hinzu. Beispiel:

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: csi-san
provisioner: csi.trident.netapp.io
parameters:
  trident.netapp.io/backendType: "ontap-san"
  csi.storage.k8s.io/node-stage-secret-name: luks
  csi.storage.k8s.io/node-stage-secret-namespace: ${pvc.namespace}
  csi.storage.k8s.io/node-publish-secret-name: luks
  csi.storage.k8s.io/node-publish-secret-namespace: ${pvc.namespace}
```

2. Identifizieren Sie vorhandene Passphrasen auf dem Volume oder Snapshot.

Volumen

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

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

Schnappschuss

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

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

3. Aktualisieren Sie das LUKS-Geheimnis für das Volume, um die neue und die vorherige Passphrase anzugeben. Stellen Sie sicher, dass `previous-luke-passphrase-name` und `previous-luks-passphrase` mit der vorherigen Passphrase übereinstimmen.

```
apiVersion: v1
kind: Secret
metadata:
  name: luks-pvc1
stringData:
  luks-passphrase-name: B
  luks-passphrase: secretB
  previous-luks-passphrase-name: A
  previous-luks-passphrase: secretA
```

4. Erstellen Sie einen neuen Pod, der das Volume einbindet. Dies ist erforderlich, um die Rotation zu initiieren.

5. Überprüfen Sie, ob die Passphrase geändert wurde.

Volumen

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

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

Schnappschuss

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

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

Ergebnisse

Die Passphrase wurde geändert, wenn auf dem Volume und im Snapshot nur die neue Passphrase zurückgegeben wird.

> Wenn zwei Passphrasen zurückgegeben werden, zum Beispiel `luksPassphraseNames: ["B", "A"]`, ist die Rotation unvollständig. Sie können einen neuen Pod auslösen, um zu versuchen, die Rotation abzuschließen.

Volumenerweiterung aktivieren

Sie können die Volumenerweiterung auf einem LUKS-verschlüsselten Volume aktivieren.

Schritte

1. Aktivieren Sie das `CSINodeExpandSecret` Feature-Gate (Beta 1.25+). Siehe "Kubernetes 1.25: Verwendung von Secrets für die knotengesteuerte Erweiterung von CSI-Volumes" für Details.
2. Fügen Sie die `node-expand-secret-name` und `node-expand-secret-namespace` StorageClass-Parameter hinzu. Beispiel:

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: luks
provisioner: csi.trident.netapp.io
parameters:
  selector: "luks=true"
  csi.storage.k8s.io/node-stage-secret-name: luks-${pvc.name}
  csi.storage.k8s.io/node-stage-secret-namespace: ${pvc.namespace}
  csi.storage.k8s.io/node-expand-secret-name: luks-${pvc.name}
  csi.storage.k8s.io/node-expand-secret-namespace: ${pvc.namespace}
allowVolumeExpansion: true
```

Ergebnisse

Wenn Sie die Online-Speichererweiterung initiieren, übergibt das kubelet die entsprechenden Anmeldeinformationen an den Treiber.

Kerberos-In-Flight-Verschlüsselung

Durch die Verwendung der Kerberos-In-Flight-Verschlüsselung können Sie die Sicherheit des Datenzugriffs verbessern, indem Sie die Verschlüsselung für den Datenverkehr zwischen Ihrem verwalteten Cluster und dem Storage-Backend aktivieren.

Trident unterstützt Kerberos-Verschlüsselung für ONTAP als Storage-Backend:

- **On-premise ONTAP** - Trident unterstützt die Kerberos-Verschlüsselung über NFSv3- und NFSv4-Verbindungen von Red Hat OpenShift und Upstream-Kubernetes-Clustern zu On-premise ONTAP Volumes.

Sie können Volumes erstellen, löschen, ihre Größe ändern, Snapshots erstellen, klonen, schreibgeschützte Klone erstellen und importieren, die NFS-Verschlüsselung verwenden.

Konfigurieren Sie die Kerberos-Verschlüsselung während der Übertragung mit lokalen ONTAP-Volumes

Sie können die Kerberos-Verschlüsselung für den Speicherdatenverkehr zwischen Ihrem verwalteten Cluster und einem lokalen ONTAP Storage-Backend aktivieren.

> Die Kerberos-Verschlüsselung für NFS-Datenverkehr mit On-Premise ONTAP-Speicher-Backends wird nur mit dem `ontap-nas` storage driver unterstützt.

Bevor Sie beginnen

- Stellen Sie sicher, dass Sie Zugriff auf das `tridentctl` Dienstprogramm haben.
- Stellen Sie sicher, dass Sie über Administratorzugriff auf das ONTAP-Speicher-Backend verfügen.
- Stellen Sie sicher, dass Sie den Namen des oder der Volumes kennen, die Sie vom ONTAP Storage-Backend freigeben werden.

- Stellen Sie sicher, dass Sie die ONTAP Storage-VM für die Unterstützung der Kerberos-Verschlüsselung für NFS-Volumes vorbereitet haben. Siehe "Kerberos auf einem dataLIF aktivieren" für Anweisungen.
- Stellen Sie sicher, dass alle NFSv4-Volumes, die Sie mit Kerberos-Verschlüsselung verwenden, korrekt konfiguriert sind. Siehe den Abschnitt NetApp NFSv4-Domänenkonfiguration (Seite 13) des "NetApp NFSv4-Verbesserungen und Best Practices-Leitfaden".

ONTAP Exportrichtlinien hinzufügen oder ändern

Sie müssen bestehenden ONTAP-Exportrichtlinien Regeln hinzufügen oder neue Exportrichtlinien erstellen, die die Kerberos-Verschlüsselung für das ONTAP Storage-VM-Root-Volume sowie für alle ONTAP-Volumes, die mit dem Upstream-Kubernetes-Cluster gemeinsam genutzt werden, unterstützen. Die Exportrichtlinienregeln, die Sie hinzufügen, oder neuen Exportrichtlinien, die Sie erstellen, müssen die folgenden Zugriffsprotokolle und Zugriffsberechtigungen unterstützen:

Zugriffsprotokolle

Konfigurieren Sie die Exportrichtlinie mit den Zugriffsprotokollen NFS, NFSv3 und NFSv4.

Zugangsdaten

Je nach Ihren Anforderungen an das Volume können Sie eine von drei verschiedenen Versionen der Kerberos-Verschlüsselung konfigurieren:

- **Kerberos 5** - (Authentifizierung und Verschlüsselung)
- **Kerberos 5i** - (Authentifizierung und Verschlüsselung mit Identitätsschutz)
- **Kerberos 5p** - (Authentifizierung und Verschlüsselung mit Identitäts- und Privatsphärenschutz)

Konfigurieren Sie die ONTAP-Exportrichtlinienregel mit den entsprechenden Zugriffsberechtigungen. Wenn Cluster beispielsweise die NFS-Volumes mit einer Mischung aus Kerberos 5i und Kerberos 5p Verschlüsselung einbinden, verwenden Sie die folgenden Zugriffseinstellungen:

Typ	Nur-Lese-Zugriff	Lese-/Schreibzugriff	Superuser-Zugriff
UNIX	Aktiviert	Aktiviert	Aktiviert
Kerberos 5i	Aktiviert	Aktiviert	Aktiviert
Kerberos 5p	Aktiviert	Aktiviert	Aktiviert

Siehe die folgende Dokumentation, um zu erfahren, wie Sie ONTAP Exportrichtlinien und Exportrichtlinienregeln erstellen:

- "Erstellen Sie eine Exportrichtlinie"
- "Fügen Sie einer Exportrichtlinie eine Regel hinzu"

Erstellen Sie ein Storage-Backend

Sie können eine Trident-Speicher-Backend-Konfiguration erstellen, die die Kerberos-Verschlüsselungsfunktion umfasst.

Über diese Aufgabe

Wenn Sie eine Storage-Backend-Konfigurationsdatei erstellen, die die Kerberos-Verschlüsselung konfiguriert, können Sie mit dem `spec.nfsMountOptions`-Parameter eine von drei verschiedenen Versionen der Kerberos-Verschlüsselung angeben:

- `spec.nfsMountOptions: sec=krb5` (Authentifizierung und Verschlüsselung)
- `spec.nfsMountOptions: sec=krb5i` (Authentifizierung und Verschlüsselung mit Identitätsschutz)
- `spec.nfsMountOptions: sec=krb5p` (Authentifizierung und Verschlüsselung mit Identitäts- und Datenschutz)

Geben Sie nur eine Kerberos-Verschlüsselungsstufe an. Wenn Sie in der Parameterliste mehr als eine Kerberos-Verschlüsselungsstufe angeben, wird nur die erste Option verwendet.

Schritte

1. Erstellen Sie auf dem verwalteten Cluster eine Speicher-Backend-Konfigurationsdatei anhand des folgenden Beispiels. Ersetzen Sie Werte in Klammern <> mit Informationen aus Ihrer Umgebung:

```
apiVersion: v1
kind: Secret
metadata:
  name: backend-ontap-nas-secret
type: Opaque
stringData:
  clientID: <CLIENT_ID>
  clientSecret: <CLIENT_SECRET>
---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-ontap-nas
spec:
  version: 1
  storageDriverName: "ontap-nas"
  managementLIF: <STORAGE_VM_MGMT_LIF_IP_ADDRESS>
  dataLIF: <PROTOCOL_LIF_FQDN_OR_IP_ADDRESS>
  svm: <STORAGE_VM_NAME>
  username: <STORAGE_VM_USERNAME_CREDENTIAL>
  password: <STORAGE_VM_PASSWORD_CREDENTIAL>
  nasType: nfs
  nfsMountOptions: ["sec=krb5i"] #can be krb5, krb5i, or krb5p
  qtreesPerFlexvol:
  credentials:
    name: backend-ontap-nas-secret
```

2. Verwenden Sie die Konfigurationsdatei, die Sie im vorherigen Schritt erstellt haben, um das Backend zu erstellen:

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

Schlägt die Backend-Erstellung fehl, liegt ein Fehler in der Backend-Konfiguration vor. Sie können die

Protokolle einsehen, 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 den create-Befehl erneut ausführen.

Erstellen Sie eine Speicherklasse

Sie können eine Speicherklasse erstellen, um Volumes mit Kerberos-Verschlüsselung bereitzustellen.

Über diese Aufgabe

Wenn Sie ein Speicherklassenobjekt erstellen, können Sie mit dem `mountOptions`-Parameter eine von drei verschiedenen Versionen der Kerberos-Verschlüsselung angeben:

- `mountOptions: sec=krb5` (Authentifizierung und Verschlüsselung)
- `mountOptions: sec=krb5i` (Authentifizierung und Verschlüsselung mit Identitätsschutz)
- `mountOptions: sec=krb5p` (Authentifizierung und Verschlüsselung mit Identitäts- und Datenschutz)

Geben Sie nur eine Kerberos-Verschlüsselungsstufe an. Wenn Sie in der Parameterliste mehr als eine Kerberos-Verschlüsselungsstufe angeben, wird nur die erste Option verwendet. Wenn die von Ihnen in der Speicher-Backend-Konfiguration angegebene Verschlüsselungsstufe von der Stufe abweicht, die Sie im Speicherklassenobjekt angeben, hat das Speicherklassenobjekt Vorrang.

Schritte

1. Erstellen Sie ein StorageClass Kubernetes-Objekt anhand des folgenden Beispiels:

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: ontap-nas-sc
provisioner: csi.trident.netapp.io
mountOptions:
  - sec=krb5i #can be krb5, krb5i, or krb5p
parameters:
  backendType: ontap-nas
  storagePools: ontapnas_pool
  trident.netapp.io/nasType: nfs
allowVolumeExpansion: true
```

2. Erstellen Sie die Speicherklasse:

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

3. Stellen Sie sicher, dass die Speicherklasse erstellt wurde:

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

Sie sollten eine Ausgabe ähnlich der folgenden sehen:

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

Volumes bereitstellen

Nachdem Sie ein Speicher-Backend und eine Speicherklasse erstellt haben, können Sie nun ein Volume bereitstellen. Anweisungen finden Sie unter "Ein Volume bereitstellen".

Konfigurieren der Kerberos-Verschlüsselung während der Übertragung mit Azure NetApp Files-Volumes

Sie können die Kerberos-Verschlüsselung für den Speicherdatenverkehr zwischen Ihrem verwalteten Cluster und einem einzelnen Azure NetApp Files-Speicher-Backend oder einem virtuellen Pool von Azure NetApp Files-Speicher-Backends aktivieren.

Bevor Sie beginnen

- Stellen Sie sicher, dass Sie Trident auf dem verwalteten Red Hat OpenShift Cluster aktiviert haben.
- Stellen Sie sicher, dass Sie Zugriff auf das `tridentctl` Dienstprogramm haben.
- Stellen Sie sicher, dass Sie das Azure NetApp Files Storage-Backend für die Kerberos-Verschlüsselung vorbereitet haben, indem Sie die Anforderungen beachten und den Anweisungen in "Azure NetApp Files-Dokumentation"folgen.
- Stellen Sie sicher, dass alle NFSv4-Volumes, die Sie mit Kerberos-Verschlüsselung verwenden, korrekt konfiguriert sind. Siehe den Abschnitt NetApp NFSv4-Domänenkonfiguration (Seite 13) des "NetApp NFSv4-Verbesserungen und Best Practices-Leitfaden".

Erstellen Sie ein Storage-Backend

Sie können eine Azure NetApp Files-Speicher-Backend-Konfiguration erstellen, die die Kerberos-Verschlüsselungsfunktion beinhaltet.

Über diese Aufgabe

Wenn Sie eine Konfigurationsdatei für das Storage-Backend erstellen, die die Kerberos-Verschlüsselung konfiguriert, können Sie festlegen, dass sie auf einer von zwei möglichen Ebenen angewendet werden soll:

- Die **Speicher-Backend-Ebene** unter Verwendung des `spec.kerberos` Felds
- Der **virtuelle Poolpegel** unter Verwendung des `spec.storage.kerberos` Feldes

Wenn Sie die Konfiguration auf Ebene des virtuellen Pools definieren, wird der Pool anhand der Bezeichnung in der Speicherklasse ausgewählt.

Auf beiden Ebenen können Sie eine von drei verschiedenen Versionen der Kerberos-Verschlüsselung angeben:

- `kerberos: sec=krb5` (Authentifizierung und Verschlüsselung)
- `kerberos: sec=krb5i` (Authentifizierung und Verschlüsselung mit Identitätsschutz)
- `kerberos: sec=krb5p` (Authentifizierung und Verschlüsselung mit Identitäts- und Datenschutz)

Schritte

1. Erstellen Sie auf dem verwalteten Cluster eine Storage-Backend-Konfigurationsdatei anhand eines der folgenden Beispiele, je nachdem, wo Sie das Storage-Backend definieren müssen (Storage-Backend-Ebene oder virtuelle Pool-Ebene). Ersetzen Sie Werte in Klammern <> mit Informationen aus Ihrer Umgebung:

Beispiel auf Storage-Backend-Ebene

```
apiVersion: v1
kind: Secret
metadata:
  name: backend-tbc-secret
type: Opaque
stringData:
  clientID: <CLIENT_ID>
  clientSecret: <CLIENT_SECRET>

---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-tbc
spec:
  version: 1
  storageDriverName: azure-netapp-files
  subscriptionID: <SUBSCRIPTION_ID>
  tenantID: <TENANT_ID>
  location: <AZURE_REGION_LOCATION>
  serviceLevel: Standard
  networkFeatures: Standard
  capacityPools: <CAPACITY_POOL>
  resourceGroups: <RESOURCE_GROUP>
  netappAccounts: <NETAPP_ACCOUNT>
  virtualNetwork: <VIRTUAL_NETWORK>
  subnet: <SUBNET>
  nasType: nfs
  kerberos: sec=krb5i #can be krb5, krb5i, or krb5p
  credentials:
    name: backend-tbc-secret
```

Beispiel auf virtueller Pool-Ebene

```
---
apiVersion: v1
kind: Secret
metadata:
  name: backend-tbc-secret
type: Opaque
stringData:
  clientID: <CLIENT_ID>
  clientSecret: <CLIENT_SECRET>

---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-tbc
spec:
  version: 1
  storageDriverName: azure-netapp-files
  subscriptionID: <SUBSCRIPTION_ID>
  tenantID: <TENANT_ID>
  location: <AZURE_REGION_LOCATION>
  serviceLevel: Standard
  networkFeatures: Standard
  capacityPools: <CAPACITY_POOL>
  resourceGroups: <RESOURCE_GROUP>
  netappAccounts: <NETAPP_ACCOUNT>
  virtualNetwork: <VIRTUAL_NETWORK>
  subnet: <SUBNET>
  nasType: nfs
  storage:
    - labels:
        type: encryption
      kerberos: sec=krb5i #can be krb5, krb5i, or krb5p
  credentials:
    name: backend-tbc-secret
```

2. Verwenden Sie die Konfigurationsdatei, die Sie im vorherigen Schritt erstellt haben, um das Backend zu erstellen:

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

Schlägt die Backend-Erstellung fehl, liegt ein Fehler in der Backend-Konfiguration vor. Sie können die Protokolle einsehen, 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 den create-Befehl erneut ausführen.

Erstellen Sie eine Speicherklasse

Sie können eine Speicherklasse erstellen, um Volumes mit Kerberos-Verschlüsselung bereitzustellen.

Schritte

1. Erstellen Sie ein StorageClass Kubernetes-Objekt anhand des folgenden Beispiels:

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: sc-nfs
provisioner: csi.trident.netapp.io
parameters:
  backendType: azure-netapp-files
  trident.netapp.io/nasType: nfs
  selector: type=encryption
```

2. Erstellen Sie die Speicherklasse:

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

3. Stellen Sie sicher, dass die Speicherklasse erstellt wurde:

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

Sie sollten eine Ausgabe ähnlich der folgenden sehen:

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

Volumes bereitstellen

Nachdem Sie ein Speicher-Backend und eine Speicherklasse erstellt haben, können Sie nun ein Volume bereitstellen. Anweisungen finden Sie unter "Ein Volume bereitstellen".