Anwendungen wiederherstellen

Trident

NetApp
July 01, 2026

This PDF was generated from https://docs.netapp.com/de-de/trident-2602/trident-protect/trident-protect-restore-apps.html on July 01, 2026. Always check docs.netapp.com for the latest.

Inhalt

Anwendungen wiederherstellen

Anwendungen mit Trident Protect wiederherstellen

Mit Trident Protect können Sie Ihre Anwendung aus einem Snapshot oder einer Sicherung wiederherstellen. Die Wiederherstellung aus einem vorhandenen Snapshot ist schneller, wenn die Anwendung im selben Cluster wiederhergestellt wird.

- Wenn Sie eine Anwendung wiederherstellen, werden alle für die Anwendung konfigurierten Ausführungs-Hooks mit der Anwendung wiederhergestellt. Wenn ein Ausführungs-Hook nach der Wiederherstellung vorhanden ist, wird er automatisch als Teil des Wiederherstellungsvorgangs ausgeführt.
- Die Wiederherstellung aus einem Backup in einen anderen Namespace oder in den ursprünglichen Namespace wird für qtree Volumes unterstützt. Die Wiederherstellung aus einem Snapshot in einen anderen Namespace oder in den ursprünglichen Namespace wird für qtree Volumes jedoch nicht unterstützt.
- Sie können die Wiederherstellungsvorgänge mithilfe erweiterter Einstellungen anpassen. Weitere Informationen finden Sie unter "Verwenden Sie die erweiterten Trident Protect-Wiederherstellungseinstellungen".

Wiederherstellung aus einem Backup in einen anderen Namensraum

Wenn Sie eine Sicherung mithilfe einer BackupRestore CR in einem anderen Namespace wiederherstellen, stellt Trident Protect die Anwendung in einem neuen Namespace wieder her und erstellt eine Anwendungs-CR für die wiederhergestellte Anwendung. Um die wiederhergestellte Anwendung zu schützen, erstellen Sie bedarfsgesteuerte Backups oder Snapshots oder legen Sie einen Schutzzeitplan fest.

- Die Wiederherstellung einer Sicherung in einem anderen Namensraum mit vorhandenen Ressourcen ändert keine Ressourcen, die denselben Namen wie die in der Sicherung haben. Um alle Ressourcen in der Sicherung wiederherzustellen, löschen und erstellen Sie entweder den Zielnamensraum neu oder stellen Sie die Sicherung in einem neuen Namensraum wieder her.
- Wenn Sie eine CR zur Wiederherstellung in einem neuen Namespace verwenden, müssen Sie den Ziel-Namespace manuell erstellen, bevor Sie die CR anwenden. Trident Protect erstellt Namespaces automatisch nur bei Verwendung der CLI.

Bevor Sie beginnen

Stellen Sie sicher, dass die Gültigkeitsdauer des AWS-Sitzungstokens für alle länger dauernden s3-Wiederherstellungsvorgänge ausreichend ist. Wenn das Token während des Wiederherstellungsvorgangs abläuft, kann der Vorgang fehlschlagen.

- Weitere Informationen zum Prüfen des Ablaufs des aktuellen Sitzungstokens finden Sie in der "AWS API-Dokumentation".
- Weitere Informationen zu Anmeldeinformationen für AWS-Ressourcen finden Sie in der "AWS IAM-Dokumentation".

Wenn Sie Backups mit Kopia als Data Mover wiederherstellen, können Sie optional Anmerkungen in der CR oder über die CLI angeben, um das Verhalten des von Kopia verwendeten temporären Speichers zu steuern. Weitere Informationen über die Optionen, die Sie konfigurieren können, finden Sie in der "Kopia-Dokumentation". Verwenden Sie den `tridentctl-protect create --help`-Befehl, um weitere Informationen zum Angeben von Anmerkungen mit der Trident Protect CLI zu erhalten.

Verwenden Sie einen CR

Schritte

1. Erstellen Sie die benutzerdefinierte Ressourcendatei (CR) und benennen Sie sie `trident-protect-backup-restore-cr.yaml`.
2. Konfigurieren Sie in der von Ihnen erstellten Datei die folgenden Attribute:
   - **metadata.name**: (*Erforderlich*) Der Name dieser benutzerdefinierten Ressource; wählen Sie einen eindeutigen und sinnvollen Namen für Ihre Umgebung.
   - **spec.appArchivePath**: Der Pfad innerhalb von AppVault, in dem die Sicherungsinhalte gespeichert sind. Sie können den folgenden Befehl verwenden, um diesen Pfad zu finden:

     ```
     kubectl get backups <BACKUP_NAME> -n my-app-namespace -o
     jsonpath='{.status.appArchivePath}'
     ```

   - **spec.appVaultRef**: (*Erforderlich*) Der Name des AppVault, in dem die Sicherungsinhalte gespeichert sind.
   - **spec.destinationApplicationName**: (*Optional*) Der Name für die wiederhergestellte Anwendung. Falls angegeben, verwendet die wiederhergestellte Anwendung diesen Namen. Falls nicht angegeben, verwendet die wiederhergestellte Anwendung den Namen der Quellanwendung.
   - **spec.namespaceMapping**: Die Zuordnung des Quell-Namespace des Wiederherstellungsvorgangs zum Ziel-Namespace. Ersetzen Sie `my-source-namespace` und `my-destination-namespace` durch Informationen aus Ihrer Umgebung.

     ```
     ---
     apiVersion: protect.trident.netapp.io/v1
     kind: BackupRestore
     metadata:
       name: my-cr-name
       namespace: my-destination-namespace
     spec:
       appArchivePath: my-backup-path
       appVaultRef: appvault-name
       destinationApplicationName: my-new-app-name
       namespaceMapping: [{"source": "my-source-namespace",
     "destination": "my-destination-namespace"}]
     ```

3. (*Optional*) Falls Sie nur bestimmte Ressourcen der Anwendung für die Wiederherstellung auswählen möchten, fügen Sie Filter hinzu, die Ressourcen mit bestimmten Bezeichnungen ein- oder ausschließen:

   Trident Protect wählt bestimmte Ressourcen automatisch aus, weil sie mit den von Ihnen ausgewählten Ressourcen in Beziehung stehen. Wenn Sie beispielsweise eine Ressource für einen persistenten Volume-Claim auswählen und diese einen zugehörigen Pod hat, wird Trident Protect auch den zugehörigen Pod wiederherstellen.

- **resourceFilter.resourceSelectionCriteria**: (Für die Filterung erforderlich) Verwenden Sie `Include` oder `Exclude`, um eine in resourceMatchers definierte Ressource ein- oder auszuschließen. Fügen Sie die folgenden resourceMatchers-Parameter hinzu, um die Ressourcen zu definieren, die ein- oder auszuschließen sind:
  - **resourceFilter.resourceMatchers**: Ein Array von resourceMatcher-Objekten. Wenn Sie mehrere Elemente in diesem Array definieren, werden diese mit einer ODER-Verknüpfung verglichen und die Felder innerhalb jedes Elements (group, kind, version) werden mit einer UND-Verknüpfung verglichen.
    - **resourceMatchers[].group**: (*Optional*) Gruppe der zu filternden Ressource.
    - **resourceMatchers[].kind**: (*Optional*) Art der zu filternden Ressource.
    - **resourceMatchers[].version**: (*Optional*) Version der zu filternden Ressource.
    - **resourceMatchers[].names**: (*Optional*) Namen im Kubernetes metadata.name-Feld der Ressource, die gefiltert werden soll.
    - **resourceMatchers[].namespaces**: (*Optional*) Namespaces im Kubernetes metadata.name-Feld der Ressource, die gefiltert werden soll.
    - **resourceMatchers[].labelSelectors**: (*Optional*) Label-Selektorzeichenfolge im Kubernetes-Metadatenfeld name der Ressource, wie definiert in der "Kubernetes-Dokumentation". Zum Beispiel: `"trident.netapp.io/os=linux"`.

      Beispiel:

```
spec:
  resourceFilter:
    resourceSelectionCriteria: "Include"
    resourceMatchers:
      - group: my-resource-group-1
        kind: my-resource-kind-1
        version: my-resource-version-1
        names: ["my-resource-names"]
        namespaces: ["my-resource-namespaces"]
        labelSelectors: ["trident.netapp.io/os=linux"]
      - group: my-resource-group-2
        kind: my-resource-kind-2
        version: my-resource-version-2
        names: ["my-resource-names"]
        namespaces: ["my-resource-namespaces"]
        labelSelectors: ["trident.netapp.io/os=linux"]
```

4. Nachdem Sie die `trident-protect-backup-restore-cr.yaml` Datei mit den korrekten Werten gefüllt haben, wenden Sie die CR an:

```
kubectl apply -f trident-protect-backup-restore-cr.yaml
```

Verwenden Sie die Befehlszeile

Schritte

1. Stellen Sie die Sicherung in einem anderen Namensraum wieder her, indem Sie die Werte in Klammern durch Informationen aus Ihrer Umgebung ersetzen. Das `namespace-mapping` Argument verwendet durch Doppelpunkte getrennte Namensräume, um Quell-Namensräume den korrekten Ziel-Namensräumen im Format `source1:dest1,source2:dest2` zuzuordnen. Beispiel:

```
tridentctl-protect create backuprestore <my_restore_name> \
--backup <backup_namespace>/<backup_to_restore> \
--namespace-mapping <source_to_destination_namespace_mapping> \
--destination-app-name<custom_app_name>\
-n <application_namespace>
```

Stellen Sie aus einer Sicherung in den ursprünglichen Namensraum wieder her

Sie können eine Sicherung jederzeit im ursprünglichen Namensraum wiederherstellen. Wenn Sie eine Wiederherstellung vor Ort durchführen, verwaltet Trident Protect automatisch Schutzzeitpläne und laufende Vorgänge, um ungültige Wiederherstellungspunkte zu verhindern:

- Alle für die Anwendung aktivierten Schutzzeitpläne werden vor Beginn der Wiederherstellung deaktiviert. Dadurch wird verhindert, dass geplante Sicherungen oder Snapshots ausgeführt werden, während die Anwendungsressourcen wiederhergestellt werden.
- Nach erfolgreicher Wiederherstellung werden nur die vor der Wiederherstellung aktivierten Zeitpläne wieder aktiviert. Bereits deaktivierte Zeitpläne bleiben deaktiviert.
- Laufende Sicherungs- oder Snapshot-Vorgänge werden vor Beginn der Wiederherstellung abgebrochen. Wird ein Vorgang nicht innerhalb von 5 Minuten abgebrochen, wird die Wiederherstellung fortgesetzt und eine Warnung im Wiederherstellungs-CR-Status protokolliert.

Bevor Sie beginnen

Stellen Sie sicher, dass die Gültigkeitsdauer des AWS-Sitzungstokens für alle länger dauernden s3-Wiederherstellungsvorgänge ausreichend ist. Wenn das Token während des Wiederherstellungsvorgangs abläuft, kann der Vorgang fehlschlagen.

- Weitere Informationen zum Prüfen des Ablaufs des aktuellen Sitzungstokens finden Sie in der "AWS API-Dokumentation".
- Weitere Informationen zu Anmeldeinformationen für AWS-Ressourcen finden Sie in der "AWS IAM-Dokumentation".

> Wenn Sie Backups mit Kopia als Data Mover wiederherstellen, können Sie optional Anmerkungen in der CR oder über die CLI angeben, um das Verhalten des von Kopia verwendeten temporären Speichers zu steuern. Weitere Informationen über die Optionen, die Sie konfigurieren können, finden Sie in der "Kopia-Dokumentation". Verwenden Sie den `tridentctl-protect create --help`-Befehl, um weitere Informationen zum Angeben von Anmerkungen mit der Trident Protect CLI zu erhalten.

Verwenden Sie einen CR

Schritte

1. Erstellen Sie die benutzerdefinierte Ressourcendatei (CR) und benennen Sie sie `trident-protect-backup-ipr-cr.yaml`.
2. Konfigurieren Sie in der von Ihnen erstellten Datei die folgenden Attribute:

   - **metadata.name**: (*Erforderlich*) Der Name dieser benutzerdefinierten Ressource; wählen Sie einen eindeutigen und sinnvollen Namen für Ihre Umgebung.
   - **spec.appArchivePath**: Der Pfad innerhalb von AppVault, in dem die Sicherungsinhalte gespeichert sind. Sie können den folgenden Befehl verwenden, um diesen Pfad zu finden:

     ```
     kubectl get backups <BACKUP_NAME> -n my-app-namespace -o jsonpath='{.status.appArchivePath}'
     ```

   - **spec.appVaultRef**: (*Erforderlich*) Der Name des AppVault, in dem die Sicherungsinhalte gespeichert sind.

     Beispiel:

```
---
apiVersion: protect.trident.netapp.io/v1
kind: BackupInplaceRestore
metadata:
  name: my-cr-name
  namespace: my-app-namespace
spec:
  appArchivePath: my-backup-path
  appVaultRef: appvault-name
```

3. (*Optional*) Falls Sie nur bestimmte Ressourcen der Anwendung für die Wiederherstellung auswählen möchten, fügen Sie Filter hinzu, die Ressourcen mit bestimmten Bezeichnungen ein- oder ausschließen:

   Trident Protect wählt bestimmte Ressourcen automatisch aus, weil sie mit den von Ihnen ausgewählten Ressourcen in Beziehung stehen. Wenn Sie beispielsweise eine Ressource für einen persistenten Volume-Claim auswählen und diese einen zugehörigen Pod hat, wird Trident Protect auch den zugehörigen Pod wiederherstellen.

   - **resourceFilter.resourceSelectionCriteria**: (Für die Filterung erforderlich) Verwenden Sie `Include` oder `Exclude`, um eine in resourceMatchers definierte Ressource ein- oder auszuschließen. Fügen Sie die folgenden resourceMatchers-Parameter hinzu, um die Ressourcen zu definieren, die ein- oder auszuschließen sind:
     - **resourceFilter.resourceMatchers**: Ein Array von resourceMatcher-Objekten. Wenn Sie mehrere Elemente in diesem Array definieren, werden diese mit einer ODER-Verknüpfung verglichen und die Felder innerhalb jedes Elements (group, kind, version) werden mit einer UND-Verknüpfung verglichen.

- **resourceMatchers[].group**: (*Optional*) Gruppe der zu filternden Ressource.
- **resourceMatchers[].kind**: (*Optional*) Art der zu filternden Ressource.
- **resourceMatchers[].version**: (*Optional*) Version der zu filternden Ressource.
- **resourceMatchers[].names**: (*Optional*) Namen im Kubernetes metadata.name-Feld der Ressource, die gefiltert werden soll.
- **resourceMatchers[].namespaces**: (*Optional*) Namespaces im Kubernetes metadata.name-Feld der Ressource, die gefiltert werden soll.
- **resourceMatchers[].labelSelectors**: (*Optional*) Label-Selektorzeichenfolge im Kubernetes-Metadatenfeld name der Ressource, wie definiert in der "Kubernetes-Dokumentation". Zum Beispiel: `"trident.netapp.io/os=linux"`.

  Beispiel:

```
spec:
  resourceFilter:
    resourceSelectionCriteria: "Include"
    resourceMatchers:
      - group: my-resource-group-1
        kind: my-resource-kind-1
        version: my-resource-version-1
        names: ["my-resource-names"]
        namespaces: ["my-resource-namespaces"]
        labelSelectors: ["trident.netapp.io/os=linux"]
      - group: my-resource-group-2
        kind: my-resource-kind-2
        version: my-resource-version-2
        names: ["my-resource-names"]
        namespaces: ["my-resource-namespaces"]
        labelSelectors: ["trident.netapp.io/os=linux"]
```

4. Nachdem Sie die `trident-protect-backup-ipr-cr.yaml` Datei mit den korrekten Werten gefüllt haben, wenden Sie die CR an:

```
kubectl apply -f trident-protect-backup-ipr-cr.yaml
```

Verwenden Sie die Befehlszeile

Schritte

1. Stellen Sie die Sicherung im ursprünglichen Namensraum wieder her, indem Sie die Werte in Klammern durch Informationen aus Ihrer Umgebung ersetzen. Das `backup` Argument verwendet einen Namensraum und einen Sicherungsnamen im Format `<namespace>/<name>`. Beispiel:

```
tridentctl-protect create backupinplacerestore <my_restore_name> \
--backup <namespace/backup_to_restore> \
-n <application_namespace>
```

Wiederherstellung aus einem Backup auf einem anderen Cluster

Sie können ein Backup auf einem anderen Cluster wiederherstellen, wenn es ein Problem mit dem ursprünglichen Cluster gibt.

- Wenn Sie Backups mit Kopia als Data Mover wiederherstellen, können Sie optional Anmerkungen in der CR oder über die CLI angeben, um das Verhalten des von Kopia verwendeten temporären Speichers zu steuern. Weitere Informationen über die Optionen, die Sie konfigurieren können, finden Sie in der "Kopia-Dokumentation". Verwenden Sie den `tridentctl-protect create --help`-Befehl, um weitere Informationen zum Angeben von Anmerkungen mit der Trident Protect CLI zu erhalten.
- Wenn Sie eine CR zur Wiederherstellung in einem neuen Namespace verwenden, müssen Sie den Ziel-Namespace manuell erstellen, bevor Sie die CR anwenden. Trident Protect erstellt Namespaces automatisch nur bei Verwendung der CLI.

Bevor Sie beginnen

Stellen Sie sicher, dass die folgenden Voraussetzungen erfüllt sind:

- Auf dem Ziel-Cluster ist Trident Protect installiert.
- Der Ziel-Cluster hat Zugriff auf den Bucket-Pfad desselben AppVault wie der Quell-Cluster, in dem die Sicherung gespeichert ist.
- Stellen Sie sicher, dass Ihre lokale Umgebung eine Verbindung zum im AppVault CR definierten Objektspeicher-Bucket herstellen kann, wenn Sie den `tridentctl-protect get appvaultcontent` Befehl ausführen. Wenn Netzwerkbeschränkungen den Zugriff verhindern, führen Sie die Trident Protect CLI stattdessen innerhalb eines Pods auf dem Ziel-Cluster aus.
- Stellen Sie sicher, dass die Gültigkeitsdauer des AWS-Sitzungstokens für alle länger dauernden Wiederherstellungsvorgänge ausreichend ist. Wenn das Token während des Wiederherstellungsvorgangs abläuft, kann der Vorgang fehlschlagen.
  - Weitere Informationen zum Prüfen des Ablaufs des aktuellen Sitzungstokens finden Sie in der "AWS API-Dokumentation".
  - Weitere Informationen zu Anmeldeinformationen für AWS-Ressourcen finden Sie in der "AWS-Dokumentation".

Schritte

1. Überprüfen Sie, ob die AppVault CR auf dem Ziel-Cluster mithilfe des Trident Protect CLI-Plugins vorhanden ist:

```
tridentctl-protect get appvault --context <destination_cluster_name>
```

Falls die AppVault CR auf dem Ziel-Cluster nicht existiert, erstellen Sie sie gemäß den Schritten in "Verwenden Sie Trident Protect AppVault-Objekte, um Buckets zu verwalten".

2. Sehen Sie sich die Sicherungsinhalte des verfügbaren AppVault auf dem Ziel-Cluster an und notieren Sie sich `appArchivePath` der Sicherung, die Sie wiederherstellen möchten:

```
tridentctl-protect get appvaultcontent <appvault_name> \
--show-resources backup \
--show-paths \
--context <destination_cluster_name>
```

Durch Ausführen dieses Befehls werden die verfügbaren Backups im AppVault angezeigt, einschließlich ihrer Ursprungscluster, entsprechenden Anwendungsnamen, Zeitstempel und Archivpfade.

Beispielausgabe:

```
+-------------+-----------+--------+-----------------
+--------------------------+-------------+
|   CLUSTER   |    APP    |  TYPE  |      NAME       |         TIMESTAMP
|    PATH     |
+-------------+-----------+--------+-----------------
+--------------------------+-------------+
| production1 | wordpress | backup | wordpress-bkup-1| 2024-10-30
08:37:40 (UTC)| backuppath1 |
| production1 | wordpress | backup | wordpress-bkup-2| 2024-10-30
08:37:40 (UTC)| backuppath2 |
+-------------+-----------+--------+-----------------
+--------------------------+-------------+
```

3. Stellen Sie die Anwendung im Ziel-Cluster mithilfe des AppVault-Namens und des Archivpfads wieder her:

Bei Verwendung eines CR muss sichergestellt werden, dass der für die Anwendungswiederherstellung vorgesehene Namespace auf dem Ziel-Cluster existiert.

Verwenden Sie einen CR

1. Erstellen Sie die benutzerdefinierte Ressourcendatei (CR) und benennen Sie sie `trident-protect-backup-restore-cr.yaml`.
2. Konfigurieren Sie in der von Ihnen erstellten Datei die folgenden Attribute:
   - **metadata.name**: (*Erforderlich*) Der Name dieser benutzerdefinierten Ressource; wählen Sie einen eindeutigen und sinnvollen Namen für Ihre Umgebung.
   - **spec.appVaultRef**: (*Erforderlich*) Der Name des AppVault, in dem die Sicherungsinhalte gespeichert sind.
   - **spec.appArchivePath**: (*Erforderlich*) Der Pfad innerhalb von AppVault, in dem die Sicherungsinhalte gespeichert sind. Verwenden Sie den Befehl aus Schritt 2, um die Sicherungsinhalte anzuzeigen und `appArchivePath`für die Sicherung zu finden, die Sie wiederherstellen möchten.
   - **spec.destinationApplicationName**: (*Optional*) Der Name für die wiederhergestellte Anwendung. Falls angegeben, verwendet die wiederhergestellte Anwendung diesen Namen. Falls nicht angegeben, verwendet die wiederhergestellte Anwendung den Namen der Quellanwendung.
   - **spec.namespaceMapping**: Die Zuordnung des Quell-Namespace des Wiederherstellungsvorgangs zum Ziel-Namespace. Ersetzen Sie `my-source-namespace` und `my-destination-namespace` durch Informationen aus Ihrer Umgebung.

     Beispiel:

```
apiVersion: protect.trident.netapp.io/v1
kind: BackupRestore
metadata:
  name: my-cr-name
  namespace: my-destination-namespace
spec:
  appVaultRef: appvault-name
  appArchivePath: my-backup-path
  destinationApplicationName: my-new-app-name
  namespaceMapping: [{"source": "my-source-namespace", "
destination": "my-destination-namespace"}]
```

3. Nachdem Sie die `trident-protect-backup-restore-cr.yaml` Datei mit den korrekten Werten gefüllt haben, wenden Sie die CR an:

```
kubectl apply -f trident-protect-backup-restore-cr.yaml
```

Verwenden Sie die Befehlszeile

1. Verwenden Sie den folgenden Befehl, um die Anwendung wiederherzustellen, und ersetzen Sie die Werte in Klammern durch Informationen aus Ihrer Umgebung. Das Argument namespace-mapping verwendet durch Doppelpunkte getrennte Namensräume, um Quell-Namensräume den korrekten Ziel-Namensräumen im Format Quelle1:Ziel1,Quelle2:Ziel2 zuzuordnen. Beispiel:

```
tridentctl-protect create backuprestore <restore_name> \
--namespace-mapping <source_to_destination_namespace_mapping> \
--appvault <appvault_name> \
--path <backup_path> \
--destination-app-name <custom_app_name> \
--context <destination_cluster_name> \
-n <application_namespace>
```

Wiederherstellung aus einem Snapshot in einen anderen Namespace

Sie können Daten aus einem Snapshot mithilfe einer benutzerdefinierten Ressource (CR) entweder in einem anderen Namespace oder im ursprünglichen Quell-Namespace wiederherstellen. Wenn Sie einen Snapshot mithilfe einer SnapshotRestore CR in einem anderen Namespace wiederherstellen, stellt Trident Protect die Anwendung in einem neuen Namespace wieder her und erstellt eine Anwendungs-CR für die wiederhergestellte Anwendung. Um die wiederhergestellte Anwendung zu schützen, erstellen Sie bedarfsgesteuerte Backups oder Snapshots oder legen Sie einen Schutzzeitplan fest.

- SnapshotRestore unterstützt das `spec.storageClassMapping` Attribut, jedoch nur, wenn die Quell- und Ziel-Speicherklassen dasselbe Speicher-Backend verwenden. Wenn Sie versuchen, auf eine `StorageClass` wiederherzustellen, die ein anderes Speicher-Backend verwendet, schlägt der Wiederherstellungsvorgang fehl.
- Wenn Sie eine CR zur Wiederherstellung in einem neuen Namespace verwenden, müssen Sie den Ziel-Namespace manuell erstellen, bevor Sie die CR anwenden. Trident Protect erstellt Namespaces automatisch nur bei Verwendung der CLI.

Bevor Sie beginnen

Stellen Sie sicher, dass die Gültigkeitsdauer des AWS-Sitzungstokens für alle länger dauernden s3-Wiederherstellungsvorgänge ausreichend ist. Wenn das Token während des Wiederherstellungsvorgangs abläuft, kann der Vorgang fehlschlagen.

- Weitere Informationen zum Prüfen des Ablaufs des aktuellen Sitzungstokens finden Sie in der "AWS API-Dokumentation".
- Weitere Informationen zu Anmeldeinformationen für AWS-Ressourcen finden Sie in der "AWS IAM-Dokumentation".

Verwenden Sie einen CR

Schritte

1. Erstellen Sie die benutzerdefinierte Ressourcendatei (CR) und benennen Sie sie `trident-protect-snapshot-restore-cr.yaml`.
2. Konfigurieren Sie in der von Ihnen erstellten Datei die folgenden Attribute:
   - **metadata.name**: (*Erforderlich*) Der Name dieser benutzerdefinierten Ressource; wählen Sie einen eindeutigen und sinnvollen Namen für Ihre Umgebung.
   - **spec.appVaultRef**: (*Erforderlich*) Der Name des AppVault, in dem die Snapshot-Inhalte gespeichert sind.
   - **spec.appArchivePath**: Der Pfad innerhalb von AppVault, in dem die Snapshot-Inhalte gespeichert sind. Sie können den folgenden Befehl verwenden, um diesen Pfad zu finden:

     ```
     kubectl get snapshots <SNAPHOT_NAME> -n my-app-namespace -o
     jsonpath='{.status.appArchivePath}'
     ```

   - **spec.destinationApplicationName**: (*Optional*) Der Name für die wiederhergestellte Anwendung. Falls angegeben, verwendet die wiederhergestellte Anwendung diesen Namen. Falls nicht angegeben, verwendet die wiederhergestellte Anwendung den Namen der Quellanwendung.
   - **spec.namespaceMapping**: Die Zuordnung des Quell-Namespace des Wiederherstellungsvorgangs zum Ziel-Namespace. Ersetzen Sie `my-source-namespace` und `my-destination-namespace` durch Informationen aus Ihrer Umgebung.

     ```
     ---
     apiVersion: protect.trident.netapp.io/v1
     kind: SnapshotRestore
     metadata:
       name: my-cr-name
       namespace: my-app-namespace
     spec:
       appVaultRef: appvault-name
       appArchivePath: my-snapshot-path
       namespaceMapping: [{"source": "my-source-namespace",
     "destination": "my-destination-namespace"}]
     ```

3. (*Optional*) Falls Sie nur bestimmte Ressourcen der Anwendung für die Wiederherstellung auswählen möchten, fügen Sie Filter hinzu, die Ressourcen mit bestimmten Bezeichnungen ein- oder ausschließen:

   > Trident Protect wählt bestimmte Ressourcen automatisch aus, weil sie mit den von Ihnen ausgewählten Ressourcen in Beziehung stehen. Wenn Sie beispielsweise eine Ressource für einen persistenten Volume-Claim auswählen und diese einen zugehörigen Pod hat, wird Trident Protect auch den zugehörigen Pod wiederherstellen.

   - **resourceFilter.resourceSelectionCriteria**: (Für die Filterung erforderlich) Verwenden Sie

`Include` oder `Exclude`, um eine in resourceMatchers definierte Ressource ein- oder auszuschließen. Fügen Sie die folgenden resourceMatchers-Parameter hinzu, um die Ressourcen zu definieren, die ein- oder auszuschließen sind:

- **resourceFilter.resourceMatchers**: Ein Array von resourceMatcher-Objekten. Wenn Sie mehrere Elemente in diesem Array definieren, werden diese mit einer ODER-Verknüpfung verglichen und die Felder innerhalb jedes Elements (group, kind, version) werden mit einer UND-Verknüpfung verglichen.
  - **resourceMatchers[].group**: (*Optional*) Gruppe der zu filternden Ressource.
  - **resourceMatchers[].kind**: (*Optional*) Art der zu filternden Ressource.
  - **resourceMatchers[].version**: (*Optional*) Version der zu filternden Ressource.
  - **resourceMatchers[].names**: (*Optional*) Namen im Kubernetes metadata.name-Feld der Ressource, die gefiltert werden soll.
  - **resourceMatchers[].namespaces**: (*Optional*) Namespaces im Kubernetes metadata.name-Feld der Ressource, die gefiltert werden soll.
  - **resourceMatchers[].labelSelectors**: (*Optional*) Label-Selektorzeichenfolge im Kubernetes-Metadatenfeld name der Ressource, wie definiert in der "Kubernetes-Dokumentation". Zum Beispiel: `"trident.netapp.io/os=linux"`.

    Beispiel:

```
spec:
  resourceFilter:
    resourceSelectionCriteria: "Include"
    resourceMatchers:
      - group: my-resource-group-1
        kind: my-resource-kind-1
        version: my-resource-version-1
        names: ["my-resource-names"]
        namespaces: ["my-resource-namespaces"]
        labelSelectors: ["trident.netapp.io/os=linux"]
      - group: my-resource-group-2
        kind: my-resource-kind-2
        version: my-resource-version-2
        names: ["my-resource-names"]
        namespaces: ["my-resource-namespaces"]
        labelSelectors: ["trident.netapp.io/os=linux"]
```

4. Nachdem Sie die `trident-protect-snapshot-restore-cr.yaml` Datei mit den korrekten Werten gefüllt haben, wenden Sie die CR an:

```
kubectl apply -f trident-protect-snapshot-restore-cr.yaml
```

Verwenden Sie die Befehlszeile

Schritte

1. Stellen Sie den Snapshot in einem anderen Namespace wieder her, wobei Sie die Werte in Klammern durch Informationen aus Ihrer Umgebung ersetzen.

   - Das `snapshot` Argument verwendet einen Namespace und einen Snapshot-Namen im Format `<namespace>/<name>`.
   - Das `namespace-mapping` Argument verwendet durch Doppelpunkte getrennte Namensräume, um Quell-Namensräume den korrekten Ziel-Namensräumen im Format `source1:dest1,source2:dest2` zuzuordnen.

     Beispiel:

```
tridentctl-protect create snapshotrestore <my_restore_name> \
--snapshot <namespace/snapshot_to_restore> \
--namespace-mapping <source_to_destination_namespace_mapping> \
--destination-app-name <custom_app_name> \
-n <application_namespace>
```

Wiederherstellung aus einem Snapshot in den ursprünglichen Namensraum

Sie können einen Snapshot jederzeit im ursprünglichen Namensraum wiederherstellen. Wenn Sie eine Wiederherstellung vor Ort durchführen, verwaltet Trident Protect automatisch Schutzzeitpläne und laufende Vorgänge, um ungültige Wiederherstellungspunkte zu verhindern:

- Alle für die Anwendung aktivierten Schutzzeitpläne werden vor Beginn der Wiederherstellung deaktiviert. Dadurch wird verhindert, dass geplante Sicherungen oder Snapshots ausgeführt werden, während die Anwendungsressourcen wiederhergestellt werden.
- Nach erfolgreicher Wiederherstellung werden nur die vor der Wiederherstellung aktivierten Zeitpläne wieder aktiviert. Bereits deaktivierte Zeitpläne bleiben deaktiviert.
- Laufende Sicherungs- oder Snapshot-Vorgänge werden vor Beginn der Wiederherstellung abgebrochen. Wird ein Vorgang nicht innerhalb von 5 Minuten abgebrochen, wird die Wiederherstellung fortgesetzt und eine Warnung im Wiederherstellungs-CR-Status protokolliert.

Bevor Sie beginnen

Stellen Sie sicher, dass die Gültigkeitsdauer des AWS-Sitzungstokens für alle länger dauernden s3-Wiederherstellungsvorgänge ausreichend ist. Wenn das Token während des Wiederherstellungsvorgangs abläuft, kann der Vorgang fehlschlagen.

- Weitere Informationen zum Prüfen des Ablaufs des aktuellen Sitzungstokens finden Sie in der "AWS API-Dokumentation".
- Weitere Informationen zu Anmeldeinformationen für AWS-Ressourcen finden Sie in der "AWS IAM-Dokumentation".

Verwenden Sie einen CR

Schritte

1. Erstellen Sie die benutzerdefinierte Ressourcendatei (CR) und benennen Sie sie `trident-protect-snapshot-ipr-cr.yaml`.
2. Konfigurieren Sie in der von Ihnen erstellten Datei die folgenden Attribute:
   - **metadata.name**: (*Erforderlich*) Der Name dieser benutzerdefinierten Ressource; wählen Sie einen eindeutigen und sinnvollen Namen für Ihre Umgebung.
   - **spec.appVaultRef**: (*Erforderlich*) Der Name des AppVault, in dem die Snapshot-Inhalte gespeichert sind.
   - **spec.appArchivePath**: Der Pfad innerhalb von AppVault, in dem die Snapshot-Inhalte gespeichert sind. Sie können den folgenden Befehl verwenden, um diesen Pfad zu finden:

     ```
     kubectl get snapshots <SNAPSHOT_NAME> -n my-app-namespace -o jsonpath='{.status.appArchivePath}'
     ```

     ```
     ---
     apiVersion: protect.trident.netapp.io/v1
     kind: SnapshotInplaceRestore
     metadata:
       name: my-cr-name
       namespace: my-app-namespace
     spec:
       appVaultRef: appvault-name
       appArchivePath: my-snapshot-path
     ```

3. (*Optional*) Falls Sie nur bestimmte Ressourcen der Anwendung für die Wiederherstellung auswählen möchten, fügen Sie Filter hinzu, die Ressourcen mit bestimmten Bezeichnungen ein- oder ausschließen:

   Trident Protect wählt bestimmte Ressourcen automatisch aus, weil sie mit den von Ihnen ausgewählten Ressourcen in Beziehung stehen. Wenn Sie beispielsweise eine Ressource für einen persistenten Volume-Claim auswählen und diese einen zugehörigen Pod hat, wird Trident Protect auch den zugehörigen Pod wiederherstellen.

   - **resourceFilter.resourceSelectionCriteria**: (Für die Filterung erforderlich) Verwenden Sie `Include` oder `Exclude`, um eine in resourceMatchers definierte Ressource ein- oder auszuschließen. Fügen Sie die folgenden resourceMatchers-Parameter hinzu, um die Ressourcen zu definieren, die ein- oder auszuschließen sind:
     - **resourceFilter.resourceMatchers**: Ein Array von resourceMatcher-Objekten. Wenn Sie mehrere Elemente in diesem Array definieren, werden diese mit einer ODER-Verknüpfung verglichen und die Felder innerhalb jedes Elements (group, kind, version) werden mit einer UND-Verknüpfung verglichen.
       - **resourceMatchers[].group**: (*Optional*) Gruppe der zu filternden Ressource.
       - **resourceMatchers[].kind**: (*Optional*) Art der zu filternden Ressource.

- **resourceMatchers[].version**: (*Optional*) Version der zu filternden Ressource.
- **resourceMatchers[].names**: (*Optional*) Namen im Kubernetes metadata.name-Feld der Ressource, die gefiltert werden soll.
- **resourceMatchers[].namespaces**: (*Optional*) Namespaces im Kubernetes metadata.name-Feld der Ressource, die gefiltert werden soll.
- **resourceMatchers[].labelSelectors**: (*Optional*) Label-Selektorzeichenfolge im Kubernetes-Metadatenfeld name der Ressource, wie definiert in der "Kubernetes-Dokumentation". Zum Beispiel: `"trident.netapp.io/os=linux"`.

  Beispiel:

```
spec:
  resourceFilter:
    resourceSelectionCriteria: "Include"
    resourceMatchers:
      - group: my-resource-group-1
        kind: my-resource-kind-1
        version: my-resource-version-1
        names: ["my-resource-names"]
        namespaces: ["my-resource-namespaces"]
        labelSelectors: ["trident.netapp.io/os=linux"]
      - group: my-resource-group-2
        kind: my-resource-kind-2
        version: my-resource-version-2
        names: ["my-resource-names"]
        namespaces: ["my-resource-namespaces"]
        labelSelectors: ["trident.netapp.io/os=linux"]
```

4. Nachdem Sie die `trident-protect-snapshot-ipr-cr.yaml` Datei mit den korrekten Werten gefüllt haben, wenden Sie die CR an:

```
kubectl apply -f trident-protect-snapshot-ipr-cr.yaml
```

Verwenden Sie die Befehlszeile

Schritte

1. Stellen Sie den Snapshot im ursprünglichen Namespace wieder her, indem Sie die Werte in Klammern durch Informationen aus Ihrer Umgebung ersetzen. Beispiel:

```
tridentctl-protect create snapshotinplacerestore <my_restore_name> \
--snapshot <namespace/snapshot_to_restore> \
-n <application_namespace>
```

Überprüfen Sie den Status eines Wiederherstellungsvorgangs

Sie können die Befehlszeile verwenden, um den Status eines Wiederherstellungsvorgangs zu überprüfen, der gerade läuft, abgeschlossen ist oder fehlgeschlagen ist.

Schritte

1. Verwenden Sie den folgenden Befehl, um den Status des Wiederherstellungsvorgangs abzurufen, und ersetzen Sie die Werte in Klammern durch Informationen aus Ihrer Umgebung:

```
kubectl get backuprestore -n <namespace_name> <my_restore_cr_name> -o
jsonpath='{.status}'
```

Verwenden Sie die erweiterten Trident Protect-Wiederherstellungseinstellungen

Sie können Wiederherstellungsvorgänge mithilfe erweiterter Einstellungen wie Annotationen, Namespace-Einstellungen und Speicheroptionen an Ihre spezifischen Anforderungen anpassen.

Namespace-Annotationen und -Labels während Wiederherstellungs- und Failover-Operationen

Während Wiederherstellungs- und Failover-Vorgängen werden die Labels und Annotationen im Ziel-Namensraum so angepasst, dass sie den Labels und Annotationen im Quell-Namensraum entsprechen. Labels oder Annotationen aus dem Quell-Namensraum, die im Ziel-Namensraum nicht existieren, werden hinzugefügt, und alle Labels oder Annotationen, die bereits vorhanden sind, werden überschrieben, um dem Wert aus dem Quell-Namensraum zu entsprechen. Labels oder Annotationen, die nur im Ziel-Namensraum existieren, bleiben unverändert.

> Wenn Sie Red Hat OpenShift verwenden, ist es wichtig, die entscheidende Rolle von Namespace-Annotationen in OpenShift-Umgebungen zu beachten. Namespace-Annotationen stellen sicher, dass wiederhergestellte Pods die entsprechenden Berechtigungen und Sicherheitskonfigurationen einhalten, die durch OpenShift Security Context Constraints (SCCs) definiert sind, und ohne Berechtigungsprobleme auf Volumes zugreifen können. Weitere Informationen finden Sie unter "OpenShift security context constraints Dokumentation".

Sie können verhindern, dass bestimmte Annotationen im Ziel-Namespace überschrieben werden, indem Sie die Kubernetes-Umgebungsvariable `RESTORE_SKIP_NAMESPACE_ANNOTATIONS` festlegen, bevor Sie die Wiederherstellungs- oder Failover-Operation durchführen. Beispiel:

```
helm upgrade trident-protect -n trident-protect netapp-trident-
protect/trident-protect \
  --set-string
restoreSkipNamespaceAnnotations="{<annotation_key_to_skip_1>,<annotation_k
ey_to_skip_2>}" \
  --reuse-values
```

Bei der Durchführung einer Wiederherstellungs- oder Failover-Operation werden alle in `restoreSkipNamespaceAnnotations` und `restoreSkipNamespaceLabels` angegebenen Namespace-Annotationen und -Labels von der Wiederherstellungs- oder Failover-Operation ausgeschlossen. Stellen Sie sicher, dass diese Einstellungen während der initialen Helm-Installation konfiguriert sind. Weitere Informationen finden Sie unter "Konfigurieren Sie zusätzliche Trident Protect Helm-Chart-Einstellungen".

Wenn Sie die Quellanwendung mit Helm mit dem `--create-namespace` Flag installiert haben, wird dem `name` Label-Schlüssel eine besondere Behandlung zuteil. Während des Wiederherstellungs- oder Failover-Prozesses kopiert Trident Protect dieses Label in den Ziel-Namespace, aktualisiert jedoch den Wert auf den Wert des Ziel-Namespaces, wenn der Wert aus der Quelle mit dem Quell-Namespace übereinstimmt. Wenn dieser Wert nicht mit dem Quell-Namespace übereinstimmt, wird er unverändert in den Ziel-Namespace kopiert.

Beispiel

Das folgende Beispiel zeigt einen Quell- und einen Ziel-Namensraum mit jeweils unterschiedlichen Annotationen und Labels. Sie können den Zustand des Ziel-Namensraums vor und nach der Operation sehen und erkennen, wie die Annotationen und Labels im Ziel-Namensraum kombiniert oder überschrieben werden.

Vor dem Wiederherstellungs- oder Failover-Vorgang

Die folgende Tabelle veranschaulicht den Zustand der Beispiel-Quell- und Ziel-Namespaces vor der Wiederherstellungs- oder Failover-Operation:

Namensraum	Anmerkungen	Etiketten
Namespace ns-1 (Quelle)	• annotation.one/key: "updatedvalue" • annotation.two/key: "true"	• environment=production • compliance=hipaa • name=ns-1
Namespace ns-2 (Ziel)	• annotation.one/key: "true" • annotation.three/key: "false"	• role=database

Nach dem Wiederherstellungsvorgang

Die folgende Tabelle veranschaulicht den Zustand des Beispiel-Ziel-Namespace nach der Wiederherstellung oder dem Failover. Einige Schlüssel wurden hinzugefügt, einige wurden überschrieben, und das `name` Label wurde aktualisiert, um dem Ziel-Namespace zu entsprechen:

Namensraum	Anmerkungen	Etiketten
Namespace ns-2 (Ziel)	• annotation.one/key: "updatedvalue" • annotation.two/key: "true" • annotation.three/key: "false"	• name=ns-2 • compliance=hipaa • environment=production • role=database

Unterstützte Felder

In diesem Abschnitt werden zusätzliche Felder beschrieben, die für Wiederherstellungsvorgänge zur Verfügung stehen.

Speicherklassenzuordnung

Das `spec.storageClassMapping` Attribut definiert eine Zuordnung von einer Speicherklasse in der Quellanwendung zu einer neuen Speicherklasse im Zielcluster. Sie können dies verwenden, wenn Sie Anwendungen zwischen Clustern mit unterschiedlichen Speicherklassen migrieren oder das Speicher-Backend für BackupRestore-Operationen ändern.

Beispiel:

```
storageClassMapping:
  - destination: "destinationStorageClass1"
    source: "sourceStorageClass1"
  - destination: "destinationStorageClass2"
    source: "sourceStorageClass2"
```

Unterstützte Annotationen

Dieser Abschnitt listet die unterstützten Annotationen zur Konfiguration verschiedener Verhaltensweisen im System auf. Wenn eine Annotation nicht explizit vom Benutzer festgelegt wird, verwendet das System den Standardwert.

Anmerkung	Typ	Beschreibung	Standardwert
protect.trident.netapp.io/data-mover-timeout-sec	Zeichenkette	Die maximal zulässige Zeit (in Sekunden), in der der Datenübertragungsvorgang angehalten werden darf.	"300"
protect.trident.netapp.io/kopia-content-cache-size-limit-mb	Zeichenkette	Die maximale Größenbeschränkung (in Megabytes) für den Kopia-Inhaltscache.	"1000"
protect.trident.netapp.io/pvc-bind-timeout-sec	Zeichenkette	Maximale Zeit (in Sekunden), die auf neu erstellte PersistentVolumeClaims (PVCs) gewartet wird, um die `Bound` Phase zu erreichen, bevor der Vorgang fehlschlägt. Gilt für alle Restore-CR-Typen (BackupRestore, BackupInplaceRestore, SnapshotRestore, SnapshotInplaceRestore). Verwenden Sie einen höheren Wert, wenn Ihr Storage-Backend oder Cluster häufig mehr Zeit benötigt.	"1200" (20 Minuten)

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 IRGENDEINER 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.