



管理後端 Trident

NetApp
January 14, 2026

目錄

管理後端	1
以KECBECVL執行後端管理	1
刪除後端	1
檢視現有的後端	1
更新後端	1
使用tridentctl執行後端管理	2
建立後端	2
刪除後端	2
檢視現有的後端	3
更新後端	3
識別使用後端的儲存類別	3
在後端管理選項之間切換	3
管理後端的選項	4
使用管理 tridentctl`後端 `TridentBackendConfig	4
使用管理 TridentBackendConfig`後端 `tridentctl	9

管理後端

以KECBECVL執行後端管理

瞭解如何使用執行後端管理作業 `kubectl`。

刪除後端

刪除 `TridentBackendConfig` 後、您會指示 `Trident` 刪除 / 保留後端（根據 `deletionPolicy`）。若要刪除後端、請確定已 `deletionPolicy` 設定為刪除。若要僅刪除 `TridentBackendConfig`、請確定已 `deletionPolicy` 設定為保留。這可確保後端仍存在、並可使用進行管理 `tridentctl`。

執行下列命令：

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

`Trident` 不會刪除使用中的 Kubernetes 機密 `TridentBackendConfig`。Kubernetes 使用者負責清除機密。刪除機密時必須小心。只有在後端未使用機密時、才應刪除這些機密。

檢視現有的後端

執行下列命令：

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

您也可以執行 `tridentctl get backend -n trident` 或 `tridentctl get backend -o yaml -n trident` 取得存在的所有後端清單。此清單也會包含使用建立的後端 `tridentctl`。

更新後端

更新後端可能有多種原因：

- 儲存系統的認證資料已變更。若要更新認證、必須更新物件中使用的 Kubernetes Secret `TridentBackendConfig`。`Trident` 會使用提供的最新認證、自動更新後端。執行下列命令以更新 Kubernetes Secret：

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

- 需要 ONTAP 更新參數（例如使用的 SVM 名稱）。
 - 您可以使用下列命令、直接透過 Kubernetes 更新 `TridentBackendConfig` 物件：

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

- 或者、您也可以使用下列命令變更現有的 `TridentBackendConfig` CR：

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



- 如果後端更新失敗、後端仍會繼續維持其最後已知的組態。您可以執行或 `kubectl describe tbc <tbc-name> -n trident` 來檢視記錄、以判斷原因、`kubectl get tbc <tbc-name> -o yaml -n trident`。
- 識別並修正組態檔的問題之後、即可重新執行 `update` 命令。

使用 `tridentctl` 執行後端管理

瞭解如何使用執行後端管理作業 `tridentctl`。

建立後端

建立之後"後端組態檔"，請執行下列命令：

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

如果後端建立失敗、表示後端組態有問題。您可以執行下列命令來檢視記錄、以判斷原因：

```
tridentctl logs -n trident
```

識別並修正組態檔的問題之後、只要再次執行命令即可 `create`。

刪除後端

若要從 `Trident` 刪除後端、請執行下列步驟：

1. 擷取後端名稱：

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

2. 刪除後端：

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



如果 `Trident` 已從這個後端佈建磁碟區和快照、但該後端仍存在、則刪除後端將會阻止新磁碟區由其進行佈建。後端將繼續存在於「刪除」狀態。

檢視現有的後端

若要檢視Trident知道的後端、請執行下列步驟：

- 若要取得摘要、請執行下列命令：

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

- 若要取得所有詳細資料、請執行下列命令：

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

更新後端

建立新的後端組態檔之後、請執行下列命令：

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

如果後端更新失敗、表示後端組態有問題、或是您嘗試了無效的更新。您可以執行下列命令來檢視記錄、以判斷原因：

```
tridentctl logs -n trident
```

識別並修正組態檔的問題之後、只要再次執行命令即可 update。

識別使用後端的儲存類別

這是您可以使用 JSON 來回答的問題類型範例、此問題可 `tridentctl` 為後端物件輸出。這會使用 `jq` 您需要安裝的公用程式。

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

這也適用於使用建立的後端 TridentBackendConfig。

在後端管理選項之間切換

瞭解在 Trident 中管理後端的不同方法。

管理後端的選項

隨着的推出 `TridentBackendConfig`，管理員現在有兩種獨特的方法來管理後端。這會提出下列問題：

- 使用建立的後端是否可以 `tridentctl` 與一起管理 `TridentBackendConfig`？
- 使用建立的後端可以 `TridentBackendConfig` 使用管理 `tridentctl` 嗎？

使用管理 `tridentctl` 後端 `TridentBackendConfig`

本節說明透過建立物件、直接透過 Kubernetes 介面來管理建立 `TridentBackendConfig` 的後端所需的步驟 `tridentctl`。

這將適用於下列案例：

- 預先存在的後端、因為是使用建立而 `tridentctl` 沒有 `TridentBackendConfig`。
- 使用建立的新後端 `tridentctl`、而其他 `TridentBackendConfig` 物件則存在。

在這兩種情況下、都會繼續出現後端、並在 Trident 排程磁碟區上運作。系統管理員有兩種選擇之一：

- 繼續使用 `tridentctl` 來管理使用它建立的後端。
- 將使用建立的後端連結 `tridentctl` 至新 `TridentBackendConfig` 物件。這樣做將意味着後端將使用而非 `tridentctl` 來管理 `kubectl`。

若要使用管理預先存在的後端 `kubectl`、您必須建立 `TridentBackendConfig` 與現有後端連結的。以下是如何運作的總覽：

1. 建立Kubernetes機密。機密包含 Trident 與儲存叢集 / 服務通訊所需的認證。
2. 建立 `TridentBackendConfig` 物件。其中包含有關儲存叢集 / 服務的詳細資訊、並參考上一步建立的機密。必須注意指定相同的組態參數（例如 `spec.backendName`、`spec.storagePrefix`、`spec.storageDriverName` 等）。`spec.backendName` 必須設定為現有後端的名稱。

步驟0：識別後端

若要建立連結至現有後端的、`TridentBackendConfig` 您必須取得後端組態。在此範例中、假設使用下列Json定義建立後端：

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

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

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

```

{
  "version": 1,
  "storageDriverName": "ontap-nas",
  "managementLIF": "10.10.10.1",
  "dataLIF": "10.10.10.2",
  "backendName": "ontap-nas-backend",
  "svm": "trident_svm",
  "username": "cluster-admin",
  "password": "admin-password",
  "defaults": {
    "spaceReserve": "none",
    "encryption": "false"
  },
  "labels": {
    "store": "nas_store"
  },
  "region": "us_east_1",
  "storage": [
    {
      "labels": {
        "app": "msoffice",
        "cost": "100"
      },
      "zone": "us_east_1a",
      "defaults": {
        "spaceReserve": "volume",
        "encryption": "true",
        "unixPermissions": "0755"
      }
    },
    {
      "labels": {
        "app": "mysqldb",
        "cost": "25"
      },
      "zone": "us_east_1d",
      "defaults": {
        "spaceReserve": "volume",
        "encryption": "false",
        "unixPermissions": "0775"
      }
    }
  ]
}

```

步驟1：建立Kubernetes機密

建立包含後端認證的秘密、如以下範例所示：

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

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

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

步驟 2：建立 TridentBackendConfig CR

下一步是建立一個 TridentBackendConfig CR、該 CR 會自動連結至預先存在的 ontap-nas-backend（如本範例所示）。確保符合下列要求：

- 中定義了相同的後端名稱 `spec.backendName`。
- 組態參數與原始後端相同。
- 虛擬資源池（若有）必須維持與原始後端相同的順序。
- 認證資料是透過Kubernetes Secret提供、而非以純文字提供。

在這種情況下、`TridentBackendConfig`將會如下所示：

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

```

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

```

```

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

```

步驟 3：確認 **CR** 的狀態 `TridentBackendConfig`

建立之後 `TridentBackendConfig`、其階段必須是 `Bound`。它也應反映與現有後端相同的後端名稱和UUID。

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

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

#confirm that no new backends were created (i.e., TridentBackendConfig did not end up creating a new backend)

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

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

後端現在將使用物件完全管理 tbc-ontap-nas-backend TridentBackendConfig。

使用管理 TridentBackendConfig 後端 `tridentctl`

`tridentctl` 可用於列出使用建立的後端 `TridentBackendConfig`。此外，系統管理員也可以選擇透過刪除並確定 `spec.deletionPolicy` 設定為 `retain` 來 `TridentBackendConfig` 完全管理這類後端 `tridentctl`。

步驟0：識別後端

例如、假設下列後端是使用建立的 TridentBackendConfig：

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

NAME	BACKEND NAME	BACKEND UUID
backend-tbc-ontap-san	ontap-san-backend	81abcb27-ea63-49bb-b606-0a5315ac5f82

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

NAME	STORAGE DRIVER	UUID
ontap-san-backend	ontap-san	81abcb27-ea63-49bb-b606-0a5315ac5f82

從輸出中、會看到 `TridentBackendConfig` 已成功建立並繫結至後端 [觀察後端的 UUID] 。

步驟 1：確認 `deletionPolicy` 設定為 `retain`

讓我們來看看的價值 `deletionPolicy`。這需要設為 `retain`。如此可確保刪除 CR 時 `TridentBackendConfig`、後端定義仍會存在、並可透過進行管理 `tridentctl`。

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

NAME	BACKEND NAME	BACKEND UUID
backend-tbc-ontap-san	ontap-san-backend	81abcb27-ea63-49bb-b606-0a5315ac5f82

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

#Confirm the value of deletionPolicy
kubectl get tbc backend-tbc-ontap-san -n trident -o wide
```

NAME	BACKEND NAME	BACKEND UUID
backend-tbc-ontap-san	ontap-san-backend	81abcb27-ea63-49bb-b606-0a5315ac5f82



除非設定為 `retain`、否則請勿繼續執行下一個步驟 `deletionPolicy`。

步驟 2：刪除 TridentBackendConfig CR

最後一步是刪除 TridentBackendConfig CR。確認設定為 retain 之後，deletionPolicy，您可以繼續刪除：

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

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

刪除物件後 TridentBackendConfig、Trident 只是將其移除、而不會實際刪除後端本身。

版權資訊

Copyright © 2026 NetApp, Inc. 版權所有。台灣印製。非經版權所有人事先書面同意，不得將本受版權保護文件的任何部分以任何形式或任何方法（圖形、電子或機械）重製，包括影印、錄影、錄音或儲存至電子檢索系統中。

由 NetApp 版權資料衍伸之軟體必須遵守下列授權和免責聲明：

此軟體以 NETAPP「原樣」提供，不含任何明示或暗示的擔保，包括但不限於有關適售性或特定目的適用性之擔保，特此聲明。於任何情況下，就任何已造成或基於任何理論上責任之直接性、間接性、附隨性、特殊性、懲罰性或衍生性損害（包括但不限於替代商品或服務之採購；使用、資料或利潤上的損失；或企業營運中斷），無論是在使用此軟體時以任何方式所產生的契約、嚴格責任或侵權行為（包括疏忽或其他）等方面，NetApp 概不負責，即使已被告知有前述損害存在之可能性亦然。

NetApp 保留隨時變本文所述之任何產品的權利，恕不另行通知。NetApp 不承擔因使用本文所述之產品而產生的責任或義務，除非明確經過 NetApp 書面同意。使用或購買此產品並不會在依據任何專利權、商標權或任何其他 NetApp 智慧財產權的情況下轉讓授權。

本手冊所述之產品受到一項（含）以上的美國專利、國外專利或申請中專利所保障。

有限權利說明：政府機關的使用、複製或公開揭露須受 DFARS 252.227-7013（2014 年 2 月）和 FAR 52.227-19（2007 年 12 月）中的「技術資料權利 - 非商業項目」條款 (b)(3) 小段所述之限制。

此處所含屬於商業產品和 / 或商業服務（如 FAR 2.101 所定義）的資料均為 NetApp, Inc. 所有。根據本協議提供的所有 NetApp 技術資料和電腦軟體皆屬於商業性質，並且完全由私人出資開發。美國政府對於該資料具有非專屬、非轉讓、非轉授權、全球性、有限且不可撤銷的使用權限，僅限於美國政府為傳輸此資料所訂合約所允許之範圍，並基於履行該合約之目的方可使用。除非本文另有規定，否則未經 NetApp Inc. 事前書面許可，不得逕行使用、揭露、重製、修改、履行或展示該資料。美國政府授予國防部之許可權利，僅適用於 DFARS 條款 252.227-7015(b)（2014 年 2 月）所述權利。

商標資訊

NETAPP、NETAPP 標誌及 <http://www.netapp.com/TM> 所列之標章均為 NetApp, Inc. 的商標。文中所涉及的所有其他公司或產品名稱，均為其各自所有者的商標，不得侵犯。