



升級Trident Trident

NetApp
January 15, 2026

目錄

升級Trident	1
升級Trident	1
升級前的注意事項	1
步驟 1：選擇版本	1
步驟二：確定原始安裝方法	2
步驟 3：選擇升級方式	2
透過營運商進行升級	2
了解操作員升級工作流程	2
使用Trident Operator 或 Helm 升級Trident安裝	3
使用 tridentctl 進行升級	7

升級Trident

升級Trident

從 24.02 版本開始，Trident 遵循四個月的發布節奏，每年發布三個主要版本。每個新版本都以先前的版本為基礎，並提供新功能、效能增強、錯誤修復和改進。我們建議您至少每年升級一次，以充分利用Trident的新功能。

升級前的注意事項

升級至最新版Trident時，請注意以下事項：

- 在給定的 Kubernetes 叢集的所有命名空間中，應該只安裝一個Trident實例。
- Trident 23.07 及更高版本需要 v1 卷快照，不再支援 alpha 或 beta 快照。
- 如果您在 Google Cloud 中建立了Cloud Volumes Service，"[CVS 服務類型](#)"您必須更新後端配置才能使用 `standardsw` 或者 `zoneredundantstandardsw` 從Trident 23.01 升級時的服務等級。未能更新 `serviceLevel` 後端故障可能導致磁碟區故障。參考 "[CVS 服務類型範例](#)" 了解詳情。
- 升級時，請務必提供以下信息 `parameter.fsType` 在 `StorageClasses` 由Trident使用。您可以刪除並重新創建 `StorageClasses` 在不影響原有流量的情況下。
 - 這是強制執行的要求 "[安全情境](#)"適用於 SAN 卷。
 - <https://github.com/NetApp/trident/tree/master/trident-installer/sample-input> [範例輸入^] 目錄包含範例，例如<https://github.com/NetApp/trident/blob/master/trident-installer/sample-input/storage-class-samples/storage-class-basic.yaml.template> [storage-class-basic.yaml.template ^] 與連結：
`storage-class-bronze-default.yaml`。
 - 更多信息，請參閱"[已知問題](#)"。

步驟 1：選擇版本

Trident 的版本遵循基於日期的命名規則。`YY.MM` 命名規則中，「YY」是年份的最後兩位數字，「MM」是月份。Dot 版本遵循 `YY.MM.X` 按照慣例，其中「X」是補丁等級。您需要根據目前版本選擇要升級到的版本。

- 您可以直接升級到與已安裝版本相差不超過四個版本視窗內的任何目標版本。例如，您可以直接從 24.06（或任何 24.06 的小版本）升級到 25.06。
- 如果您要從四個版本視窗以外的版本升級，請執行多步驟升級。請依照升級說明進行操作。"[早期版本](#)"您正在從目前版本升級到符合四版本發布視窗的最新版本。例如，如果您目前運行的是 23.07 版本，並且想要升級到 25.06 版本：
 - a. 首次升級從 23.07 升級到 24.06。
 - b. 然後從 24.06 升級到 25.06。



在 OpenShift 容器平台上使用Trident運算子進行升級時，應升級至Trident 21.01.1 或更高版本。隨 21.01.0 版本發布的Trident操作符存在一個已知問題，該問題已在 21.01.1 版本中修復。更多詳情請參閱... "[GitHub 上的問題詳情](#)"。

步驟二：確定原始安裝方法

若要確定您最初安裝Trident時所使用的版本：

1. 使用 `kubectl get pods -n trident` 檢查豆莢。
 - 如果沒有操作員艙，Trident是透過以下方式安裝的： `tridentctl`。
 - 如果存在 operator pod，則Trident是透過Trident operator 手動安裝的，或使用 Helm 安裝的。
2. 如果有操作員艙，請使用 `kubectl describe torc` 確定Trident是否使用 Helm 安裝。
 - 如果有 Helm 標籤，表示Trident是使用 Helm 安裝的。
 - 如果沒有 Helm 標籤，則表示Trident是使用Trident操作員手動安裝的。

步驟 3：選擇升級方式

通常情況下，您應該使用與初始安裝相同的方法進行升級，但是您也可以...["安裝方法之間的轉換"](#)。升級Trident有兩種選擇。

- ["使用Trident操作員進行升級"](#)



我們建議您查看["了解操作員升級工作流程"](#)在與運營商升級之前。

*

透過營運商進行升級

了解操作員升級工作流程

在使用Trident操作員升級Trident之前，您應該了解升級期間發生的背景處理程序。這包括對Trident控制器、控制器 Pod 和節點 Pod 以及節點 DaemonSet 的更改，以啟用滾動更新。

Trident操作員升級處理

眾多之一["使用Trident運算子的好處"](#)安裝和升級Trident 的過程是自動處理Trident和 Kubernetes 對象，而不會中斷現有的已掛載磁碟區。這樣一來，Trident就能支援零停機時間的升級，或者說，["捲動更新"](#)。具體來說，Trident運算子與 Kubernetes 叢集通訊以：

- 刪除並重新建立Trident Controller 部署和節點 DaemonSet。
- 將Trident Controller Pod 和Trident Node Pod 替換為新版本。
 - 如果某個節點沒有更新，並不妨礙其他節點的更新。
 - 只有運行了Trident Node Pod 的節點才能掛載磁碟區。



有關 Kubernetes 叢集上Trident架構的更多信息，請參閱：["Trident架構"](#)。

操作員升級工作流程

當您使用Trident運算子啟動升級時：

1. Trident運算子：
 - a. 偵測目前安裝的Trident版本（版本 n ）。
 - b. 更新所有 Kubernetes 對象，包括 CRD、RBAC 和Trident SVC。
 - c. 刪除版本為 n 的Trident Controller 部署。
 - d. 建立版本 $n+1$ 的Trident Controller 部署。
2. **Kubernetes** 為 $n+1$ 建立Trident Controller Pod。
3. Trident運算子：
 - a. 刪除 n 的Trident節點 DaemonSet。操作員不會等待節點 Pod 終止。
 - b. 為 $n+1$ 建立Trident節點守護程式集。
4. **Kubernetes** 在未執行Trident Node Pod n 的節點上建立Trident Node Pod。這樣可以確保一個節點上永遠不會有超過一個Trident Node Pod，無論版本如何。

使用Trident Operator 或 Helm 升級Trident安裝

您可以使用Trident Operator 手動升級Trident，也可以使用 Helm 進行升級。您可以從一個Trident Operator 安裝升級到另一個Trident Operator 安裝，或從一個 Trident Operator 安裝升級到另一個 Trident Operator 安裝。`tridentctl`安裝到Trident操作員版本。審查["選擇升級方式"](#)在升級Trident操作程序之前。

升級手動安裝

您可以從叢集範圍的Trident操作員安裝升級到另一個叢集範圍的Trident操作員安裝。所有Trident版本都使用叢集範圍的運算元。



若要從使用命名空間作用域運算子安裝的Trident（版本 20.07 至 20.10）進行升級，請使用下列升級說明：["您已安裝的版本"](#)Trident。

關於此任務

Trident提供了一個捆綁文件，您可以使用該文件安裝 Operator 並為您的 Kubernetes 版本建立關聯物件。

- 對於運行 Kubernetes 1.24 的集群，請使用["bundle_pre_1_25.yaml"](#)。
- 對於運行 Kubernetes 1.25 或更高版本的集群，請使用["bundle_post_1_25.yaml"](#)。

開始之前

請確保您使用的是正在運行的 Kubernetes 叢集。["支援的 Kubernetes 版本"](#)。

步驟

1. 請驗證您的Trident版本：

```
./tridentctl -n trident version
```

- 更新 `operator.yaml`，`tridentorchestrator_cr.yaml`，和 `post_1_25_bundle.yaml` 使用要升級到的版本（例如 25.06）的登錄和映像路徑，以及正確的金鑰。
- 刪除用於安裝目前Trident實例的Trident操作員。例如，如果您從 25.02 升級，請執行以下命令：

```
kubectl delete -f 25.02.0/trident-installer/deploy/<bundle.yaml> -n trident
```

- 如果您使用自訂方式進行了初始安裝 `TridentOrchestrator` 屬性，您可以編輯 `TridentOrchestrator` 用於修改安裝參數的物件。這可能包括為離線模式指定鏡像Trident和 CSI 映像登錄、啟用偵錯日誌或指定映像拉取金鑰所做的變更。
- 使用適用於您環境的正確 `bundle.yaml` 檔案安裝Trident，其中 `<bundle.yaml>` 是 `bundle_pre_1_25.yaml` 或者 `bundle_post_1_25.yaml` 根據您的 Kubernetes 版本。例如，如果您正在安裝Trident 25.06.0，請執行下列命令：

```
kubectl create -f 25.06.0/trident-installer/deploy/<bundle.yaml> -n trident
```

- 編輯三叉戟項圈以包含圖像 25.06.0。

升級 Helm 安裝

您可以升級Trident Helm 安裝。



當已安裝Trident的Kubernetes 叢集從 1.24 版本升級到 1.25 或更高版本時，必須更新 `values.yaml` 檔案進行設定。`excludePodSecurityPolicy` 到 `true` 或添加 `--set excludePodSecurityPolicy=true` 到 `helm upgrade` 升級叢集前需要執行此命令。

如果您已將 Kubernetes 叢集從 1.24 升級到 1.25，但未升級Trident helm，則 helm 升級將失敗。若要成功升級Helm，請先執行以下步驟：

- 從下列位置安裝 `helm-mapkubeapis` 插件 <https://github.com/helm/helm-mapkubeapis>。
- 在Trident安裝所在的命名空間中，對Trident版本執行一次試運行。這裡列出了需要清理的資源。

```
helm mapkubeapis --dry-run trident --namespace trident
```

- 使用 Helm 執行完整運行以進行清理。

```
helm mapkubeapis trident --namespace trident
```

步驟

1. 如果你"使用 Helm 安裝了Trident"。你可以使用 `helm upgrade trident netapp-trident/trident-operator --version 100.2506.0` 一步即可升級。如果您沒有新增 Helm 倉庫或無法使用它進行升級：
 - a. 從此處下載最新版Trident"[GitHub 上的 Assets 部分](#)"。
 - b. 使用 `helm upgrade` 其中命令 `trident-operator-25.06.0.tgz` 反映您想要升級到的版本。

```
helm upgrade <name> trident-operator-25.06.0.tgz
```



如果在初始安裝期間設定了自訂選項（例如，為Trident和 CSI 映像指定私人映像登錄），請附加以下內容：`helm upgrade` 命令使用 `--set` 確保將這些選項包含在升級命令中，否則這些值將重設為預設值。

2. 跑步 `helm list` 確認圖表和應用程式版本均已升級。跑步 `tridentctl logs` 查看所有調試資訊。

從 `tridentctl` 安裝到Trident操作員

您可以從下列位置升級到最新版本的Trident運算子：`tridentctl` 安裝。現有的後端和PVC將自動可用。



在切換安裝方法之前，請先查看"[安裝方法之間的轉換](#)"。

步驟

1. 下載最新版的Trident。

```
# Download the release required [25.06.0]
mkdir 25.06.0
cd 25.06.0
wget
https://github.com/NetApp/trident/releases/download/v25.06.0/trident-
installer-25.06.0.tar.gz
tar -xf trident-installer-25.06.0.tar.gz
cd trident-installer
```

2. 創建 `tridentorchestrator` 來自清單文件的 CRD。

```
kubectl create -f
deploy/crds/trident.netapp.io_tridentorchestrators_crd_post1.16.yaml
```

3. 在相同命名空間中部署叢集範圍的操作符。

```
kubectl create -f deploy/<bundle-name.yaml>
```

```
serviceaccount/trident-operator created
clusterrole.rbac.authorization.k8s.io/trident-operator created
clusterrolebinding.rbac.authorization.k8s.io/trident-operator created
deployment.apps/trident-operator created
podsecuritypolicy.policy/tridentoperatorpods created
```

```
#Examine the pods in the Trident namespace
```

NAME	READY	STATUS	RESTARTS	AGE
trident-controller-79df798bdc-m79dc	6/6	Running	0	150d
trident-node-linux-xrst8	2/2	Running	0	150d
trident-operator-5574dbbc68-nthjv	1/1	Running	0	1m30s

4. 創建一個 `TridentOrchestrator` 安裝Trident的 CR。

```
cat deploy/crds/tridentorchestrator_cr.yaml
apiVersion: trident.netapp.io/v1
kind: TridentOrchestrator
metadata:
  name: trident
spec:
  debug: true
  namespace: trident
```

```
kubectl create -f deploy/crds/tridentorchestrator_cr.yaml
```

```
#Examine the pods in the Trident namespace
```

NAME	READY	STATUS	RESTARTS	AGE
trident-csi-79df798bdc-m79dc	6/6	Running	0	1m
trident-csi-xrst8	2/2	Running	0	1m
trident-operator-5574dbbc68-nthjv	1/1	Running	0	5m41s

5. 確認Trident已升級至預期版本。

```
kubectl describe torc trident | grep Message -A 3
```

```
Message:          Trident installed
Namespace:        trident
Status:           Installed
Version:          v25.06.0
```


使用 **tridentctl** 進行升級

您可以使用以下方法輕鬆升級現有的Trident安裝 **tridentctl**。

關於此任務

解除安裝並重新安裝Trident相當於升級。卸載Trident時，Trident部署使用的持久卷聲明 (PVC) 和持久卷 (PV) 不會被刪除。在Trident離線期間，已設定的 PV 仍將可用；Trident恢復線上後，將為在此期間建立的任何 PVC 設定磁碟區。

開始之前

審查["選擇升級方式"](#)升級前使用 **tridentctl**。

步驟

1. 運行卸載命令 `tridentctl` 移除與Trident關聯的所有資源，但保留 CRD 和相關物件。

```
./tridentctl uninstall -n <namespace>
```

2. 重新安裝Trident。參考["使用 tridentctl 安裝Trident"](#)。



請勿中斷升級過程。確保安裝程式運行完成。

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。