升級Astra Trident

Astra Trident

NetApp
January 14, 2026

This PDF was generated from https://docs.netapp.com/zh-tw/trident-2402/trident-managing-k8s/upgrade-trident.html on January 14, 2026. Always check docs.netapp.com for the latest.

目錄

升級Astra Trident . . . 1
- 升級Astra Trident . . . 1
  - 升級前的考量 . . . 1
  - 步驟 1 ：選取版本 . . . 1
  - 步驟 2 ：確定原始安裝方法 . . . 2
  - 步驟 3 ：選擇升級方法 . . . 2
- 與營運者一起升級 . . . 2
  - 瞭解營運商升級工作流程 . . . 2
  - 使用 Trident 運算子或 Helm 升級 Astra Trident 安裝 . . . 3
- 使用tridentctl進行升級 . . . 6

升級Astra Trident

升級Astra Trident

從 24.02 版開始、 Astra Trident 遵循四個月的發行步調、每年提供三個主要版本。每個新版本均以舊版為基礎、並提供新功能、效能增強、錯誤修正及改善功能。我們鼓勵您每年至少升級一次、以善用Astra Trident的新功能。

升級前的考量

升級至最新版Astra Trident時、請考慮下列事項：

- 在指定 Kubernetes 叢集中的所有命名空間中、應該只安裝一個 Astra Trident 執行個體。
- Astra Trident 23.07 及更新版本需要 v1 Volume 快照、不再支援 Alpha 或 beta 快照。
- 如果您在中建立了 Cloud Volumes Service for Google Cloud "CVS服務類型"、您必須更新後端組態才能使用 `standardsw` 或 `zoneredundantstandardsw` 從 Astra Trident 23.01 升級時的服務層級。無法更新 `serviceLevel` 在後端中、可能會導致磁碟區故障。請參閱 "CVS 服務類型範例" 以取得詳細資料。
- 升級時、請務必提供 `parameter.fsType` 在中 `StorageClasses` 由Astra Trident使用。您可以刪除並重新建立 `StorageClasses` 無需中斷既有的磁碟區。
  - 這是強制實施的一項**要求 "安全性內容" 適用於SAN磁碟區。
  - sample INPUT 目錄包含 https://github.com/NetApp/trident/blob/master/trident-installer/sample-input/storage-class-samples/storage-class-basic.yaml.templ 等範例[`storage-class-basic.yaml.templ`^] 和連結： `storage-class-bronze-default.yaml`。
  - 如需詳細資訊、請參閱 "已知問題"。

步驟 1 ：選取版本

Astra Trident版本遵循日期型 `YY.MM` 命名慣例、其中「是」是一年的最後兩位數、「公釐」是月份。DOT版本遵循A `YY.MM.X` 慣例、其中「X」是修補程式層級。您將根據要升級的版本、選擇要升級的版本。

- 您可以直接升級至安裝版本的四個版本範圍內的任何目標版本。例如、您可以直接從 23.01 （或任何 23.01 點版本）升級至 24.02 。
- 如果您要從四個版本的外部版本升級、請執行多步驟升級。請使用的升級指示 "舊版" 您要從升級至最新版本、以符合四個版本的時間範圍。例如、如果您執行 22.01 且想要升級至 24.02 ：
  a. 第一次從 22.01 升級至 23.01 。
  b. 然後從 23.01 升級至 24.02 。

> 在 OpenShift Container Platform 上使用 Trident 運算子進行升級時、您應升級至 Trident 21.01.1 或更新版本。隨21.01.0一起發行的Trident運算子包含已在21.01.1中修正的已知問題。如需詳細資訊、請參閱 "GitHub問題詳細資料"。

步驟 2 ：確定原始安裝方法

若要判斷您原本用來安裝 Astra Trident 的版本：

1. 使用 `kubectl get pods -n trident` 檢查 Pod 。
   - 如果沒有操作員 Pod 、則使用安裝 Astra Trident `tridentctl`。
   - 如果有操作員 Pod 、則使用 Trident 操作員手動或使用 Helm 來安裝 Astra Trident 。
2. 如果有操作員 Pod 、請使用 `kubectl describe torc` 判斷 Astra Trident 是否使用 Helm 安裝。
   - 如果有 Helm 標籤、則使用 Helm 安裝 Astra Trident 。
   - 如果沒有 Helm 標籤、則使用 Trident 運算子手動安裝 Astra Trident 。

步驟 3 ：選擇升級方法

一般而言、您應該使用與初始安裝相同的方法進行升級、不過您可以 "在安裝方法之間移動"。有兩種方法可以升級Astra Trident。

- "使用Trident營運者進行升級"

> 我們建議您檢閱 "瞭解營運商升級工作流程" 與操作員一起升級之前。

*

與營運者一起升級

瞭解營運商升級工作流程

在使用 Trident 運算子升級 Astra Trident 之前、您應該先瞭解升級期間發生的背景程序。其中包括 Trident 控制器、控制器 Pod 和節點 Pod 的變更、以及啟用循環更新的節點示範集。

Trident 營運商升級處理

其中一項 "使用 Trident 運算子的優點" 安裝和升級 Astra Trident 是自動處理 Astra Trident 和 Kubernetes 物件、而不會中斷現有的掛載磁碟區。如此一來、 Astra Trident 就能支援零停機的升級、或 "滾動更新 _"。尤其是 Trident 運算子會與 Kubernetes 叢集通訊、以便：

- 刪除並重新建立 Trident Controller 部署和節點示範集。
- 以新版本更換 Trident 控制器 Pod 和 Trident 節點 Pod 。
  - 如果節點未更新、則不會阻止其餘節點更新。
  - 只有執行中 Trident Node Pod 的節點才能裝載磁碟區。

> 如需 Kubernetes 叢集上 Astra Trident 架構的詳細資訊、請參閱 "Astra Trident 架構"。

營運商升級工作流程

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 Astra Trident 版本（版本 *n* ）。
   b. 更新所有 Kubernetes 物件、包括 CRD 、 RBAC 和 Trident SVC 。
   c. 刪除版本 *n* 的 Trident 控制器部署。
   d. 為版本 *n+1* 建立 Trident Controller 部署。
2. * Kubernetes* 為 _n+1 建立 Trident 控制器 Pod 。
3. * Trident 運算子 * ：
   a. 刪除 *n* 的 Trident 節點示範集。操作人員不會等待節點 Pod 終止。
   b. 為 *n+1* 建立 Trident 節點 Demont 。
4. * Kubernetes* 會在未執行 Trident Node Pod 的節點上建立 Trident Node Pod 。如此可確保節點上的任何版本、都不會有超過一個 Trident Node Pod 。

使用 **Trident** 運算子或 **Helm** 升級 **Astra Trident** 安裝

您可以使用 Trident 運算子手動或使用 Helm 來升級 Astra Trident 。您可以從 Trident 運算子安裝升級至其他 Trident 運算子安裝、或從升級 `tridentctl` 安裝至 Trident 運算子版本。檢閱 "選擇升級方法" 在升級 Trident 操作員安裝之前。

升級手動安裝

您可以從叢集範圍的 Trident 運算子安裝升級到另一個叢集範圍的 Trident 運算子安裝。所有Astra Trident版本21.01及更新版本均使用叢集範圍的運算子。

> 若要從使用命名空間範圍運算子（ 20.07 至 20.10 版）安裝的 Astra Trident 進行升級、請使用的升級指示 "您已安裝的版本" Astra Trident 的

關於這項工作

Trident 提供一個套件檔案、可讓您用來安裝運算子、並為 Kubernetes 版本建立相關的物件。

- 對於執行 Kubernetes 1.24 或更早版本的叢集、請使用 "bunder_pre_1_25.yaml"。
- 對於執行 Kubernetes 1.25 或更新版本的叢集、請使用 "bunder_POST_1_25.yaml"。

開始之前

確保您使用的是執行中的 Kubernetes 叢集 "支援的Kubernetes版本"。

步驟

1. 驗證 Astra Trident 版本：

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

2. 刪除用來安裝目前Astra Trident執行個體的Trident運算子。例如、如果您是從 23.07 升級、請執行下列命令：

```
kubectl delete -f 23.07.0/trident-installer/deploy/<bundle.yaml> -n
trident
```

3. 如果您使用自訂初始安裝 `TridentOrchestrator` 屬性、您可以編輯 `TridentOrchestrator` 物件以修改安裝參數。這可能包括針對離線模式指定鏡射Trident和csi映像登錄、啟用偵錯記錄或指定映像提取機密所做的變更。
4. 使用適用於您環境的正確套件 YAML 檔案（其中包含 <bundle.yaml> ）來安裝 Astra Trident `bundle_pre_1_25.yaml` 或 `bundle_post_1_25.yaml` 以 Kubernetes 版本為基礎。例如、如果您要安裝 Astra Trident 24.02 、請執行下列命令：

```
kubectl create -f 24.02.0/trident-installer/deploy/<bundle.yaml> -n
trident
```

升級 **Helm** 安裝

您可以升級 Astra Trident Helm 安裝。

> 將Kubernetes叢集從1.24升級至1.25或更新版本、且已安裝Astra Trident時、您必須更新vales.yaml才能設定 `excludePodSecurityPolicy` 至 `true` 或新增 `--set excludePodSecurityPolicy=true` 至 `helm upgrade` 命令、然後才能升級叢集。

步驟

1. 如果您 "使用 Helm 安裝 Astra Trident"、您可以使用 `helm upgrade trident netapp-trident/trident-operator --version 100.2402.0` 只需一步即可升級。如果您未新增 Helm repo 或無法使用它來升級：
   a. 從下載最新的 Astra Trident 版本 "GitHub的_Assets區段"。
   b. 使用 `helm upgrade` 命令位置 `trident-operator-24.02.0.tgz` 反映您要升級的版本。

```
helm upgrade <name> trident-operator-24.02.0.tgz
```

> 如果您在初始安裝期間設定自訂選項（例如指定 Trident 和 CSI 映像的私有、鏡射登錄）、請附加 `helm upgrade` 命令使用 `--set` 為了確保升級命令中包含這些選項、否則這些值會重設為預設值。

2. 執行 `helm list` 以確認圖表和應用程式版本均已升級。執行 `tridentctl logs` 以檢閱任何偵錯訊息。

從升級 `tridentctl` 安裝至 **Trident** 操作員

您可以從升級至最新版的Trident運算子 `tridentctl` 安裝：現有的後端和 PVC 將會自動提供使用。

在安裝方法之間切換之前、請參閱 "在安裝方法之間移動"。

步驟

1. 下載最新的Astra Trident版本。

```
# Download the release required [24.020.0]
mkdir 24.02.0
cd 24.02.0
wget
https://github.com/NetApp/trident/releases/download/v24.02.0/trident-
installer-24.02.0.tar.gz
tar -xf trident-installer-24.02.0.tar.gz
cd trident-installer
```

2. 建立 `tridentorchestrator` 資訊清單中的CRD。

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

3. 在同一個命名空間中部署叢集範圍的運算子。

```
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. 建立 `TridentOrchestrator` 用於安裝Astra 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 已升級至所需版本。

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

Message:                Trident installed
Namespace:              trident
Status:                 Installed
Version:                v24.02.0
```

使用tridentctl進行升級

您可以使用輕鬆升級現有的Astra Trident安裝 `tridentctl`。

關於這項工作

解除安裝和重新安裝Astra Trident可做為升級。當您解除安裝Trident時、不會刪除由Astra Trident部署所使用的持續磁碟區宣告（PVc）和持續磁碟區（PV）。當Astra Trident離線時、已配置的PV仍可繼續使用、而Astra Trident會在任何建立於過渡期間的永久虛電路恢復上線後、為其配置磁碟區。

開始之前

檢閱 "選擇升級方法" 使用升級之前 `tridentctl`。

步驟

1. 在中執行解除安裝命令 `tridentctl` 移除與 Astra Trident 相關的所有資源、但 CRD 和相關物件除外。

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

2. 重新安裝 Astra Trident 。請參閱 "使用tridentctl安裝Astra Trident"。

請勿中斷升級程序。確保安裝程式執行完成。

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 保留隨時變更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。