



安裝Trident Protect Trident

NetApp
January 15, 2026

目錄

安裝Trident Protect	1
Trident保護要求	1
Trident Protect Kubernetes 叢集相容性	1
Trident Protect 儲存後端相容性	1
nas-economy 容量要求	2
使用 KubeVirt 虛擬機器保護數據	2
SnapMirror複製的要求	3
安裝並設定Trident Protect	4
安裝Trident Protect	4
安裝Trident Protect CLI 插件	7
安裝Trident Protect CLI 插件	7
查看Trident CLI 外掛程式幫助	9
啟用指令自動補全	9
自訂Trident Protect 安裝	11
指定Trident Protect 容器資源限制	11
自訂安全上下文約束	12
設定其他Trident Protect 舵圖設置	13
將Trident Protect Pod 限制在特定節點上	15

安裝Trident Protect

Trident保護要求

首先，請驗證您的運行環境、應用程式叢集、應用程式和授權是否已準備就緒。確保您的環境符合部署和執行Trident Protect 的這些要求。

Trident Protect Kubernetes 叢集相容性

Trident Protect 與各種完全託管和自架的 Kubernetes 產品相容，包括：

- 亞馬遜彈性 Kubernetes 服務 (EKS)
- Google Kubernetes 引擎 (GKE)
- Microsoft Azure Kubernetes 服務 (AKS)
- 紅帽 OpenShift
- SUSE Rancher
- VMware Tanzu 產品組合
- 上游 Kubernetes



- Trident Protect備份僅支援Linux運算節點。Windows 運算節點不支援備份作業。
- 確保安裝Trident Protect 的叢集已配置正在執行中的快照控制器和相關的 CRD。若要安裝快照控制器，請參閱 ["這些說明"](#)。

Trident Protect 儲存後端相容性

Trident Protect 支援以下儲存後端：

- Amazon FSx for NetApp ONTAP
- Cloud Volumes ONTAP
- ONTAP儲存陣列
- Google Cloud NetApp Volumes
- Azure NetApp Files

請確保您的儲存後端符合以下要求：

- 確保連接到叢集的NetApp儲存裝置使用的是Trident 24.02 或更高版本（建議使用Trident 24.10）。
- 請確保您擁有NetApp ONTAP儲存後端。
- 請確保您已配置用於儲存備份的物件儲存桶。
- 建立您計劃用於應用程式或應用程式資料管理作業的任何應用程式命名空間。Trident Protect 不會為您建立這些命名空間；如果您在自訂資源中指定了不存在的命名空間，則操作將會失敗。

nas-economy 容量要求

Trident Protect 支援對 nas-economy 磁碟區進行備份和復原作業。目前不支援將快照、克隆和SnapMirror複製到 nas-economy 磁碟區。您需要為計劃與Trident Protect 一起使用的每個 nas-economy 磁碟區啟用快照目錄。



某些應用程式與使用快照目錄的磁碟區不相容。對於這些應用，您需要透過在ONTAP儲存系統上執行以下命令來隱藏快照目錄：

```
nfs modify -vserver <svm> -v3-hide-snapshot enabled
```

您可以透過對每個 nas-economy 磁碟區執行以下命令來啟用快照目錄，並將命令替換為 ``<volume-UUID>`` 使用要變更的磁碟區的 UUID：

```
tridentctl update volume <volume-UUID> --snapshot-dir=true --pool-level=true -n trident
```



您可以透過設定Trident後端設定選項，為新磁碟區預設啟用快照目錄。`snapshotDir` 到 `true`。現有捲數不受影響。

使用 KubeVirt 虛擬機器保護數據

Trident Protect 24.10 和 24.10.1 及更高版本在保護運行在 KubeVirt VM 上的應用程式時，行為有所不同。對於這兩個版本，您都可以在資料保護操作期間啟用或停用檔案系統凍結和解凍。



在恢復操作期間，任何 `VirtualMachineSnapshots` 為虛擬機器 (VM) 建立的設定檔無法復原。

Trident Protect 24.10

Trident Protect 24.10 在資料保護作業期間不會自動確保 KubeVirt VM 檔案系統的一致性狀態。如果您想使用Trident Protect 24.10 保護您的 KubeVirt VM 數據，則需要在執行資料保護作業之前手動啟用檔案系統的凍結/解凍功能。這樣可以確保檔案系統處於一致狀態。

您可以設定Trident Protect 24.10 來管理資料保護作業期間 VM 檔案系統的凍結與解凍。["配置虛擬化"](#)然後使用以下命令：

```
kubectl set env deployment/trident-protect-controller-manager NEPTUNE_VM_FREEZE=true -n trident-protect
```

Trident Protect 24.10.1 及更高版本

從Trident Protect 24.10.1 開始，Trident Protect 會在資料保護作業期間自動凍結和解凍 KubeVirt 檔案系統。您也可以使用以下命令停用此自動行為：

```
kubectl set env deployment/trident-protect-controller-manager NEPTUNE_VM_FREEZE=false -n trident-protect
```

SnapMirror複製的要求

NetApp SnapMirror複製功能可與Trident Protect 搭配使用，適用於下列ONTAP解決方案：

- 本地部署的NetApp FAS、AFF和ASA集群
- NetApp ONTAP Select
- NetApp Cloud Volumes ONTAP
- Amazon FSx for NetApp ONTAP

ONTAP叢集對SnapMirror複製的要求

如果您打算使用SnapMirror複製功能，請確保您的ONTAP叢集符合以下要求：

- * NetApp Trident *：使用ONTAP作為後端服務的來源 Kubernetes 叢集和目標 Kubernetes 叢集上都必須存在NetApp Trident。Trident Protect 支援使用NetApp SnapMirror技術進行複製，該技術使用以下驅動程式支援的儲存類別：
 - ontap-nas：極品飛車
 - ontap-san：iSCSI
 - ontap-san：FC
 - ontap-san：NVMe/TCP（最低要求ONTAP版本 9.15.1）
- 許可證：使用資料保護包的ONTAP SnapMirror非同步許可證必須在來源 ONTAP 叢集和目標ONTAP叢集上啟用。請參閱 ["ONTAP中的SnapMirror許可概述"](#) 了解更多。

從ONTAP 9.10.1 開始，所有許可證均以NetApp許可證文件 (NLF) 的形式交付，這是一個可以啟用多種功能的單一文件。請參閱["ONTAP One 隨附的許可證"](#) 了解更多。



僅支援SnapMirror非同步保護。

SnapMirror複製的對等連線注意事項

如果您打算使用儲存後端對等互連，請確保您的環境符合以下要求：

- 叢集和 SVM：ONTAP儲存後端必須相互連接。請參閱 ["叢集和SVM對等連接概述"](#) 了解更多。



確保兩個ONTAP叢集之間複製關係中使用的 SVM 名稱是唯一的。

- * NetApp Trident和 SVM*：對等遠端 SVM 必須可供目標叢集上的NetApp Trident使用。
- 託管後端：您需要在Trident Protect 中新增和管理ONTAP儲存後端，以建立複製關係。

用於SnapMirror複製的Trident / ONTAP配置

Trident Protect 要求您至少設定一個支援來源叢集和目標叢集複製的儲存後端。如果來源叢集和目標叢集相同，為了獲得最佳彈性，目標應用程式應該使用與來源應用程式不同的儲存後端。

SnapMirror複製的 Kubernetes 叢集要求

請確保您的 Kubernetes 叢集符合以下要求：

- **AppVault** 可存取性：來源叢集和目標叢集都必須具有網路存取權限，才能從 AppVault 讀取資料和寫入數據，以實現應用程式物件複製。
- 網路連線：設定防火牆規則、儲存桶權限和 IP 允許列表，以啟用兩個叢集和 AppVault 之間透過 WAN 進行通訊。



許多企業環境在廣域網路連線中實施嚴格的防火牆策略。在配置複製之前，請先與您的基礎架構團隊確認這些網路需求。

安裝並設定Trident Protect

如果您的環境符合Trident Protect 的要求，您可以依照下列步驟在叢集上安裝Trident Protect。您可以從NetApp取得Trident Protect，或從您自己的私人註冊表中安裝它。如果您的叢集無法存取互聯網，從私有註冊表安裝會很有幫助。

安裝Trident Protect

從NetApp安裝Trident Protect

步驟

1. 新增Trident Helm 倉庫：

```
helm repo add netapp-trident-protect  
https://netapp.github.io/trident-protect-helm-chart
```

2. 使用 Helm 安裝Trident Protect。代替 `<name-of-cluster>` 集群名稱將分配給集群，並用於標識集群的備份和快照：

```
helm install trident-protect netapp-trident-protect/trident-protect  
--set clusterName=<name-of-cluster> --version 100.2506.0 --create  
--namespace --namespace trident-protect
```

從私人註冊表安裝Trident Protect

如果您的 Kubernetes 叢集無法存取互聯網，您可以從私人鏡像倉庫安裝Trident Protect。在這些範例中，請將括號中的值替換為您環境中的資訊：

步驟

1. 將以下鏡像拉取到本地計算機，更新標籤，然後將其推送到您的私有鏡像倉庫：

```
netapp/controller:25.06.0  
netapp/restic:25.06.0  
netapp/kopia:25.06.0  
netapp/trident-autosupport:25.06.0  
netapp/exehook:25.06.0  
netapp/resourcebackup:25.06.0  
netapp/resourcerestore:25.06.0  
netapp/resourcedelete:25.06.0  
bitnami/kubectl:1.30.2  
kubebuilder/kube-rbac-proxy:v0.16.0
```

例如：

```
docker pull netapp/controller:25.06.0
```

```
docker tag netapp/controller:25.06.0 <private-registry-  
url>/controller:25.06.0
```

```
docker push <private-registry-url>/controller:25.06.0
```

2. 建立Trident Protect 系統命名空間：

```
kubectl create ns trident-protect
```

3. 登入註冊表：

```
helm registry login <private-registry-url> -u <account-id> -p <api-token>
```

4. 建立用於私有註冊表驗證的拉取金鑰：

```
kubectl create secret docker-registry regcred --docker-username=<registry-username> --docker-password=<api-token> -n trident-protect --docker-server=<private-registry-url>
```

5. 新增Trident Helm 倉庫：

```
helm repo add netapp-trident-protect https://netapp.github.io/trident-protect-helm-chart
```

6. 建立一個名為的文件 `protectValues.yaml`。請確保其中包含以下Trident Protect 設定：

```

---
image:
  registry: <private-registry-url>
imagePullSecrets:
  - name: regcred
controller:
  image:
    registry: <private-registry-url>
rbacProxy:
  image:
    registry: <private-registry-url>
crCleanup:
  imagePullSecrets:
    - name: regcred
webhooksCleanup:
  imagePullSecrets:
    - name: regcred

```

7. 使用 Helm 安裝Trident Protect。代替 ``<name_of_cluster>`` 集群名稱將分配給集群，並用於標識集群的備份和快照：

```

helm install trident-protect netapp-trident-protect/trident-protect
--set clusterName=<name_of_cluster> --version 100.2506.0 --create
--namespace --namespace trident-protect -f protectValues.yaml

```

安裝Trident Protect CLI 插件

您可以使用Trident Protect 命令列插件，它是Trident的一個擴充。`tridentctl`用於建立和與Trident Protect 自訂資源 (CR) 互動的實用程式。

安裝Trident Protect CLI 插件

在使用命令列實用程式之前，需要將其安裝在用於存取叢集的電腦上。根據您的機器使用的是 x64 還是ARM CPU，請依照以下步驟操作。

下載適用於 **Linux AMD64 CPU** 的插件

步驟

1. 下載Trident Protect CLI 外掛：

```
curl -L -o tridentctl-protect https://github.com/NetApp/tridentctl-protect/releases/download/25.06.0/tridentctl-protect-linux-amd64
```

下載適用於 **Linux ARM64 CPU** 的插件

步驟

1. 下載Trident Protect CLI 外掛：

```
curl -L -o tridentctl-protect https://github.com/NetApp/tridentctl-protect/releases/download/25.06.0/tridentctl-protect-linux-arm64
```

下載適用於 **Mac AMD64 CPU** 的插件

步驟

1. 下載Trident Protect CLI 外掛：

```
curl -L -o tridentctl-protect https://github.com/NetApp/tridentctl-protect/releases/download/25.06.0/tridentctl-protect-macos-amd64
```

下載適用於 **Mac ARM64 CPU** 的插件

步驟

1. 下載Trident Protect CLI 外掛：

```
curl -L -o tridentctl-protect https://github.com/NetApp/tridentctl-protect/releases/download/25.06.0/tridentctl-protect-macos-arm64
```

1. 啟用插件二進位檔案的執行權限：

```
chmod +x tridentctl-protect
```

2. 將插件二進位檔案複製到 PATH 環境變數中定義的位置。例如， `/usr/bin` 或者 `/usr/local/bin`（您可能需要更高的權限）：

```
cp ./tridentctl-protect /usr/local/bin/
```

- 您也可以選擇將插件二進位檔案複製到您主目錄中的某個位置。在這種情況下，建議確保該位置已新增至您的 PATH 環境變數：

```
cp ./tridentctl-protect ~/bin/
```



將插件複製到 PATH 環境變數中的某個位置，即可透過鍵入以下命令來使用該插件。`tridentctl-protect` 或者 `tridentctl protect` 從任何地點。

查看Trident CLI 外掛程式幫助

您可以使用插件內建的幫助功能來獲得有關插件功能的詳細幫助：

步驟

- 使用幫助功能查看使用指南：

```
tridentctl-protect help
```

啟用指令自動補全

安裝Trident Protect CLI 外掛程式後，您可以為某些指令啟用自動補全功能。

為 **Bash shell** 啟用自動補全功能

步驟

1. 下載完成腳本：

```
curl -L -O https://github.com/NetApp/tridentctl-protect/releases/download/25.06.0/tridentctl-completion.bash
```

2. 在您的使用者目錄下建立一個新目錄，用於存放腳本：

```
mkdir -p ~/.bash/completions
```

3. 將下載的腳本移到 `~/.bash/completions` 目錄：

```
mv tridentctl-completion.bash ~/.bash/completions/
```

4. 將以下行新增至 `~/.bashrc` 您主目錄中的檔案：

```
source ~/.bash/completions/tridentctl-completion.bash
```

為 **Z shell** 啟用自動補全

步驟

1. 下載完成腳本：

```
curl -L -O https://github.com/NetApp/tridentctl-protect/releases/download/25.06.0/tridentctl-completion.zsh
```

2. 在您的使用者目錄下建立一個新目錄，用於存放腳本：

```
mkdir -p ~/.zsh/completions
```

3. 將下載的腳本移到 `~/.zsh/completions` 目錄：

```
mv tridentctl-completion.zsh ~/.zsh/completions/
```

4. 將以下行新增至 `~/.zprofile` 您主目錄中的檔案：

```
source ~/.zsh/completions/tridentctl-completion.zsh
```

結果

下次登入 shell 時，您可以使用 tridentctl-protect 外掛程式進行指令自動補全。

自訂Trident Protect 安裝

您可以自訂Trident Protect 的預設配置，以符合您環境的特定要求。

指定Trident Protect 容器資源限制

安裝Trident Protect 後，您可以使用設定檔來指定Trident Protect 容器的資源限制。設定資源限制可以控制Trident Protect 作業消耗叢集資源的程度。

步驟

1. 建立一個名為的文件 `resourceLimits.yaml`。
2. 根據您的環境需求，在檔案中填入Trident Protect 容器的資源限制選項。

以下範例設定檔顯示了可用設置，並包含每個資源限制的預設值：

```
---
jobResources:
  defaults:
    limits:
      cpu: 8000m
      memory: 10000Mi
      ephemeralStorage: ""
    requests:
      cpu: 100m
      memory: 100Mi
      ephemeralStorage: ""
  resticVolumeBackup:
    limits:
      cpu: ""
      memory: ""
      ephemeralStorage: ""
    requests:
      cpu: ""
      memory: ""
      ephemeralStorage: ""
  resticVolumeRestore:
    limits:
      cpu: ""
```

```

    memory: ""
    ephemeralStorage: ""
  requests:
    cpu: ""
    memory: ""
    ephemeralStorage: ""
  kopiaVolumeBackup:
    limits:
      cpu: ""
      memory: ""
      ephemeralStorage: ""
    requests:
      cpu: ""
      memory: ""
      ephemeralStorage: ""
  kopiaVolumeRestore:
    limits:
      cpu: ""
      memory: ""
      ephemeralStorage: ""
    requests:
      cpu: ""
      memory: ""
      ephemeralStorage: ""

```

3. 應用以下值 `resourceLimits.yaml` 文件：

```

helm upgrade trident-protect -n trident-protect netapp-trident-
protect/trident-protect -f resourceLimits.yaml --reuse-values

```

自訂安全上下文約束

安裝 Trident Protect 後，您可以使用設定檔來修改 Trident Protect 容器的 OpenShift 安全上下文約束 (SCC)。這些約束定義了 Red Hat OpenShift 叢集中 pod 的安全性限制。

步驟

1. 建立一個名為的文件 `sccconfig.yaml`。
2. 在文件中新增 SCC 選項，並根據您的環境需求修改參數。

以下範例顯示了 SCC 選項的參數預設值：

```
scc:
  create: true
  name: trident-protect-job
  priority: 1
```

下表描述了SCC選項的參數：

範圍	描述	預設
創造	確定是否可以建立 SCC 資源。僅當 `scc.create` 設定為 `true` Helm 安裝過程會辨識 OpenShift 環境。如果不是在 OpenShift 上運行，或者如果 `scc.create` 設定為 `false` 不會創建 SCC 資源。	真的
姓名	指定 SCC 的名稱。	三叉戟保護工作
優先事項	確定 SCC 的優先順序。優先順序較高的 SCC 會優先於優先順序較低的 SCC 進行評估。	1

3. 應用以下值 `sccconfig.yaml` 文件：

```
helm upgrade trident-protect netapp-trident-protect/trident-protect -f
sccconfig.yaml --reuse-values
```

這將用指定的值替換預設值。`sccconfig.yaml` 文件。

設定其他Trident Protect 舵圖設置

您可以自訂AutoSupport設定和命名空間過濾以滿足您的特定要求。下表描述了可用的配置參數：

範圍	類型	描述
自動支援代理	細繩	為NetApp AutoSupport連線配置代理 URL。使用此功能透過代理伺服器路由支援包上傳。例子： http://my.proxy.url 。
自動支援.不安全	布林值	設定為“跳過AutoSupport代理連接的 TLS 驗證” true。僅用於不安全的代理連線。(預設: false)
自動支援已啟用	布林值	啟用或停用每日Trident Protect AutoSupport組合包上傳。設定為 false 每日定時上傳功能已停用，但您仍可以手動產生支援包。(預設: `true`)

範圍	類型	描述
恢復跳過命名空間註釋	細繩	若要從備份和復原作業中排除的命名空間註解的逗號分隔清單。允許您根據註釋過濾命名空間。
restoreSkipNamespaceLabels	細繩	若要從備份和復原作業中排除的命名空間標籤的逗號分隔清單。允許您根據標籤過濾命名空間。

您可以使用 YAML 設定檔或命令列標誌來設定這些選項：

使用 **YAML** 文件

步驟

1. 建立一個設定檔並將其命名為 `values.yaml`。
2. 在您建立的文件中，新增您想要自訂的設定選項。

```
autoSupport:
  enabled: false
  proxy: http://my.proxy.url
  insecure: true
restoreSkipNamespaceAnnotations: "annotation1,annotation2"
restoreSkipNamespaceLabels: "label1,label2"
```

3. 填寫完後 `values.yaml` 使用正確的值建立配置文件，並套用該設定檔：

```
helm upgrade trident-protect -n trident-protect netapp-trident-protect/trident-protect -f values.yaml --reuse-values
```

使用 **CLI** 標誌

步驟

1. 使用以下命令： `--set` 用於指定個別參數的標誌：

```
helm upgrade trident-protect -n trident-protect netapp-trident-protect/trident-protect \
  --set autoSupport.enabled=false \
  --set autoSupport.proxy=http://my.proxy.url \
  --set restoreSkipNamespaceAnnotations="annotation1,annotation2" \
  --set restoreSkipNamespaceLabels="label1,label2" \
  --reuse-values
```

將Trident Protect Pod 限制在特定節點上

您可以使用 Kubernetes nodeSelector 節點來選擇約束，根據節點標籤來控制哪些節點有資格執行Trident Protect pod。預設情況下，Trident Protect 僅限於執行 Linux 的節點。您可以根據需要進一步自訂這些限制條件。

步驟

1. 建立一個名為的文件 `nodeSelectorConfig.yaml`。
2. 在檔案中新增 `nodeSelector` 選項，並修改檔案以新增或變更節點標籤，從而根據您的環境需求進行限制。例如，以下檔案包含預設的作業系統限制，但同時也針對特定區域和應用程式名稱：

```
nodeSelector:  
  kubernetes.io/os: linux  
  region: us-west  
  app.kubernetes.io/name: mysql
```

3. 應用以下值 ``nodeSelectorConfig.yaml`` 文件：

```
helm upgrade trident-protect -n trident-protect netapp-trident-  
protect/trident-protect -f nodeSelectorConfig.yaml --reuse-values
```

這將預設限制替換為您在設定中指定的限制。``nodeSelectorConfig.yaml`` 文件。

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。