



## 安裝 Trident Protect Trident

NetApp  
November 14, 2025

# 目錄

安裝 Trident Protect .....	1
Trident 保護需求 .....	1
Trident 保護 Kubernetes 叢集相容性 .....	1
Trident 保護儲存後端相容性 .....	1
NAS 經濟容量需求 .....	2
使用 KubeVirt VM 保護資料 .....	2
SnapMirror 複寫需求 .....	3
安裝及設定 Trident Protect .....	4
安裝 Trident Protect .....	4
安裝 Trident Protect CLI 外掛程式 .....	7
安裝 Trident Protect CLI 外掛程式 .....	7
檢視 Trident CLI 外掛程式說明 .....	9
啟用命令自動完成 .....	9
自訂 Trident Protect 安裝 .....	11
指定 Trident Protect 容器資源限制 .....	11
自訂安全性內容限制 .....	12
設定其他Trident保護 Helm Chart 設定 .....	13
將 Trident 保護 Pod 限制在特定節點 .....	14

# 安裝 Trident Protect

## Trident 保護需求

首先，請確認您的營運環境，應用程式叢集，應用程式和授權是否準備就緒。確保您的環境符合這些需求，以部署及操作 Trident Protect。

### Trident 保護 Kubernetes 叢集相容性

Trident Protect 可與多種完全託管且自我管理的 Kubernetes 產品相容，包括：

- Amazon Elastic Kubernetes Service (EKS)
- Google Kubernetes Engine (GKE)
- Microsoft Azure Kubernetes 服務 (英文)
- Red Hat OpenShift
- SUSE Rancher
- VMware Tanzu 產品組合
- 上游 Kubernetes



- Trident 保護備份僅支援 Linux 計算節點。Windows 運算節點不支援備份作業。
- 請確定您安裝 Trident Protect 的叢集已設定執行中的快照控制器和相關的 CRD。若要安裝快照控制器，請 ["這些指示"](#) 參閱。

### Trident 保護儲存後端相容性

Trident Protect 支援下列儲存設備後端：

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

確保您的儲存後端符合下列需求：

- 確保連接到叢集的 NetApp 儲存裝置使用 Trident 24.02 或更新版本（建議使用 Trident 24.10）。
- 確保您擁有 NetApp ONTAP 儲存後端。
- 請確定您已設定物件儲存貯體以儲存備份。
- 建立您計畫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不存在的命名空間，則作業將會失敗。

## NAS 經濟容量需求

Trident Protect 支援 NAS 經濟型磁碟區的備份與還原作業。目前不支援快照，複製和 SnapMirror 複寫至 NAS 經濟型磁碟區。您需要為打算搭配 Trident Protect 使用的每個 NAS 經濟型磁碟區啟用快照目錄。



某些應用程式與使用 Snapshot 目錄的磁碟區不相容。對於這些應用程式，您需要在 ONTAP 儲存系統上執行下列命令，以隱藏快照目錄：

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

您可以針對每個 NAS 經濟型磁碟區執行下列命令，以您要變更的磁碟區 UUID 取代，來啟用 Snapshot 目錄 <volume-UUID>：

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



您可以將 Trident 後端組態選項設定為，為 true 新的磁碟區預設啟用快照目錄 `snapshotDir`。現有的磁碟區不受影響。

## 使用 KubeVirt VM 保護資料

Trident Protect 在資料保護作業期間為 KubeVirt 虛擬機器提供檔案系統凍結和解凍功能，以確保資料一致性。虛擬機器凍結操作的配置方法和預設行為在 Trident Protect 版本之間有所不同，較新的版本透過 Helm chart 參數提供了簡化的配置。



在恢復操作期間，任何 `VirtualMachineSnapshots` 為虛擬機器 (VM) 所建立的資料不會被復原。

### Trident Protect 25.10 及更新版本

Trident Protect 在資料保護作業期間自動凍結和解凍 KubeVirt 檔案系統，以確保一致性。從 Trident Protect 25.10 開始，您可以使用以下方法停用此行為：`vm.freeze` Helm Chart 安裝過程中的參數。此參數預設啟用。

```
helm install ... --set vm.freeze=false ...
```

## Trident保護 24.10.1 至 25.06

從 Trident Protect 24.10.1 開始，Trident Protect 會在資料保護作業期間，自動凍結和取消凍結 KubeVirt 檔案系統。您也可以使用下列命令停用此自動行為：

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

## Trident Protect 24.10

Trident Protect 24.10 無法在資料保護作業期間，自動確保 KubeVirt VM 檔案系統的狀態一致。如果您想要使用 Trident Protect 24.10 來保護 KubeVirt VM 資料，則必須在資料保護作業之前，手動啟用檔案系統的凍結 / 取消凍結功能。如此可確保檔案系統處於一致的狀態。

您可以將 Trident Protect 24.10 設定為在資料保護作業期間管理 VM 檔案系統的凍結和取消凍結"[設定虛擬化](#)"，然後使用下列命令：

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

## SnapMirror 複寫需求

NetApp SnapMirror 複寫可與 Trident Protect 搭配使用，適用於下列 ONTAP 解決方案：

- 內部部署 NetApp FAS，AFF 和 ASA 叢集
- NetApp ONTAP Select
- NetApp Cloud Volumes ONTAP
- Amazon FSX for NetApp ONTAP 產品

## SnapMirror 複寫的 ONTAP 叢集需求

如果您打算使用 SnapMirror 複寫，請確保 ONTAP 叢集符合下列需求：

- **NetApp Trident**：NetApp Trident 必須存在於使用 ONTAP 作為後端的來源 Kubernetes 叢集和目標 Kubernetes 叢集上。Trident Protect 使用下列驅動程式所支援的儲存類別，以 NetApp SnapMirror 技術支援複寫：
  - ontap-nas：NFS
  - ontap-san：iSCSI
  - ontap-san：足球俱樂部
  - ontap-san：NVMe/TCP（要求最低 ONTAP 版本 9.15.1）
- **\* 授權 \***：使用資料保護套件的 ONTAP SnapMirror 非同步授權必須同時在來源和目的地 ONTAP 叢集上啟用。如需詳細資訊，請參閱 "[SnapMirror授權概述ONTAP](#)"。

從 ONTAP 9.10.1 開始、所有授權都會以 NetApp 授權檔案（NLF）的形式交付、這是一個可啟用多項功能的單一檔案。如需詳細資訊、請參閱 ["ONTAP One 隨附授權"](#)。



僅支援 SnapMirror 非同步保護。

## SnapMirror 複寫的對等考量

如果您計畫使用儲存後端對等，請確保您的環境符合下列需求：

- **\* 叢集與 SVM \***：必須對 ONTAP 儲存設備的後端進行對等處理。如需詳細資訊、請參閱 ["叢集與SVM對等概觀"](#)。



確保兩個 ONTAP 叢集之間複寫關係中使用的 SVM 名稱是唯一的。

- **NetApp Trident 與 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 中新增及管理 ONTAP 儲存後端，才能建立複寫關係。

## 用於 SnapMirror 複寫的 Trident / ONTAP 組態

Trident Protect 要求您至少設定一個儲存後端，以支援來源叢集和目的地叢集的複寫。如果來源叢集和目的地叢集相同、則目的地應用程式應使用不同於來源應用程式的儲存後端、以獲得最佳恢復能力。

## SnapMirror複製的 Kubernetes 叢集要求

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 讀取和寫入應用程式物件複製。
- **網路連線**：設定防火牆規則、儲存桶權限和 IP 允許列表，以實現跨 WAN 的叢集和 AppVault 之間的通訊。



許多企業環境在 WAN 連線中實施嚴格的防火牆策略。在配置複製之前，請與您的基礎設施團隊驗證這些網路需求。

# 安裝及設定 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，或從您自己的私有登錄安裝。如果您的叢集無法存取網際網路，從私有登錄安裝會很有幫助。

## 安裝 Trident Protect

## 安裝 Trident Protect from NetApp

### 步驟

1. 新增Trident Helm儲存庫：

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

2. 使用 Helm 安裝 Trident Protect 。以叢集名稱取代 <name-of-cluster>，該名稱將指派給叢集，用於識別叢集的備份和快照：

```
helm install trident-protect netapp-trident-protect/trident-protect  
--set clusterName=<name-of-cluster> --version 100.2510.0 --create  
--namespace --namespace trident-protect
```

3. （選用）若要啟用偵錯日誌記錄（建議用於故障排除），請使用：

```
helm install trident-protect netapp-trident-protect/trident-protect  
--set clusterName=<name-of-cluster> --set logLevel=debug --version  
100.2510.0 --create-namespace --namespace trident-protect
```

偵錯日誌記錄有助於NetApp支援人員排除故障，而無需變更日誌等級或重現問題。

### 從私有登錄安裝 Trident Protect

如果 Kubernetes 叢集無法存取網際網路，您可以從私有映像登錄安裝 Trident Protect 。在這些範例中，請將方括號中的值取代為環境中的資訊：

### 步驟

1. 將下列影像拉到您的本機電腦，更新標記，然後將它們推送到您的私人登錄：

```
docker.io/netapp/controller:25.10.0  
docker.io/netapp/restic:25.10.0  
docker.io/netapp/kopia:25.10.0  
docker.io/netapp/kopiablockrestore:25.10.0  
docker.io/netapp/trident-autosupport:25.10.0  
docker.io/netapp/exehook:25.10.0  
docker.io/netapp/resourcebackup:25.10.0  
docker.io/netapp/resourcerestore:25.10.0  
docker.io/netapp/resourcedelete:25.10.0  
docker.io/netapp/trident-protect-utils:v1.0.0
```

例如：

```
docker pull docker.io/netapp/controller:25.10.0
```

```
docker tag docker.io/netapp/controller:25.10.0 <private-registry-url>/controller:25.10.0
```

```
docker push <private-registry-url>/controller:25.10.0
```



要取得 Helm Chart，首先需要在可以存取網路的電腦上下載 Helm Chart。helm pull trident-protect --version 100.2510.0 --repo <https://netapp.github.io/trident-protect-helm-chart> 然後複製結果 `trident-protect-100.2510.0.tgz` 將檔案複製到您的離線環境並進行安裝 helm install trident-protect ./trident-protect-100.2510.0.tgz 而不是在最後一步使用存儲庫引用。

## 2. 建立 Trident Protect 系統命名空間：

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

## 3. 登入登錄：

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

## 4. 建立用於私人登錄驗證的拉出密碼：

```
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儲存庫：

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

## 6. 建立名為的檔案 protectValues.yaml。請確定其中包含下列 Trident Protect 設定：



```
---
imageRegistry: <private-registry-url>
imagePullSecrets:
  - name: regcred
```



這 `imageRegistry` 和 `imagePullSecrets` 這些值適用於所有組件影像，包括 `resourcebackup` 和 `resourcerestore`。如果您將鏡像推送到註冊表中的特定儲存庫路徑（例如，`example.com:443/my-repo`），請在登錄欄位中包含完整路徑。這將確保所有圖像都從此處提取。`<private-registry-url>/<image-name>:<tag>`。

7. 使用 Helm 安裝 Trident Protect。以叢集名稱取代 `<name_of_cluster>`，該名稱將指派給叢集，用於識別叢集的備份和快照：

```
helm install trident-protect netapp-trident-protect/trident-protect
--set clusterName=<name_of_cluster> --version 100.2510.0 --create
--namespace --namespace trident-protect -f protectValues.yaml
```

8. （選用）若要啟用偵錯日誌記錄（建議用於故障排除），請使用：

```
helm install trident-protect netapp-trident-protect/trident-protect
--set clusterName=<name-of-cluster> --set logLevel=debug --version
100.2510.0 --create-namespace --namespace trident-protect -f
protectValues.yaml
```

偵錯日誌記錄有助於NetApp支援人員排除故障，而無需變更日誌等級或重現問題。



有關其他 Helm Chart 設定選項，包括AutoSupport設定和命名空間過濾，請參閱 "[自訂 Trident Protect 安裝](#)"。

## 安裝 Trident Protect CLI 外掛程式

您可以使用 Trident Protect 命令列外掛程式（Trident 公用程式的延伸 `tridentctl`）來建立自訂資源（CRS），並與 Trident 互動。

### 安裝 Trident Protect CLI 外掛程式

在使用命令列公用程式之前，您必須先將其安裝在用來存取叢集的機器上。根據您的機器使用的是 x64 或 ARM CPU，請遵循下列步驟。

下載適用於 **Linux AMD64 CPU** 的外掛程式

步驟

1. 下載 Trident Protect CLI 外掛程式：

```
curl -L -o tridentctl-protect https://github.com/NetApp/tridentctl-protect/releases/download/25.10.0/tridentctl-protect-linux-amd64
```

下載適用於 **Linux ARM64 CPU** 的外掛程式

步驟

1. 下載 Trident Protect CLI 外掛程式：

```
curl -L -o tridentctl-protect https://github.com/NetApp/tridentctl-protect/releases/download/25.10.0/tridentctl-protect-linux-arm64
```

下載適用於 **Mac AMD64 CPU** 的外掛程式

步驟

1. 下載 Trident Protect CLI 外掛程式：

```
curl -L -o tridentctl-protect https://github.com/NetApp/tridentctl-protect/releases/download/25.10.0/tridentctl-protect-macos-amd64
```

下載 **Mac ARM64 CPU** 的外掛程式

步驟

1. 下載 Trident Protect CLI 外掛程式：

```
curl -L -o tridentctl-protect https://github.com/NetApp/tridentctl-protect/releases/download/25.10.0/tridentctl-protect-macos-arm64
```

1. 啟用外掛程式二進位檔的執行權限：

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

2. 將外掛程式二進位檔複製到路徑變數中定義的位置。例如， `/usr/bin` 或 `/usr/local/bin`（您可能需要提升的 Privileges）：

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

- 您也可以選擇將外掛程式二進位檔複製到主目錄中的某個位置。在這種情況下，建議您確保位置是 PATH 變數的一部分：

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



將外掛程式複製到 PATH 變數中的某個位置，可讓您輸入或 `tridentctl protect`` 從任何位置使用外掛程式 ``tridentctl-protect``。

## 檢視 **Trident CLI** 外掛程式說明

您可以使用內建的外掛程式說明功能，取得外掛程式功能的詳細說明：

### 步驟

- 使用說明功能檢視使用指南：

```
tridentctl-protect help
```

## 啟用命令自動完成

安裝 Trident Protect CLI 外掛程式之後，您可以啟用某些命令的自動完成功能。

## 啟用 **Bash Shell** 的自動完成功能

### 步驟

1. 建立完成腳本：

```
tridentctl-protect completion bash > tridentctl-completion.bash
```

2. 在主目錄中建立新目錄以包含指令碼：

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

3. 將下載的指令碼移至 `~/.bash/completions` 目錄：

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

4. 將下列行新增至 `~/.bashrc` 主目錄中的檔案：

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

## 啟用 **Z Shell** 的自動完成功能

### 步驟

1. 建立完成腳本：

```
tridentctl-protect completion zsh > tridentctl-completion.zsh
```

2. 在主目錄中建立新目錄以包含指令碼：

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

3. 將下載的指令碼移至 `~/.zsh/completions` 目錄：

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

4. 將下列行新增至 `~/.zprofile` 主目錄中的檔案：

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

結果

下次登入 Shell 時，您可以將命令自動完成功能與 tridentctl-Protect 外掛程式搭配使用。

## 自訂 Trident Protect 安裝

您可以自訂 Trident Protect 的預設組態，以符合您環境的特定需求。

### 指定 Trident Protect 容器資源限制

安裝 Trident Protect 之後，您可以使用組態檔來指定 Trident Protect 容器的資源限制。設定資源限制可讓您控制 Trident Protect 作業消耗多少叢集資源。

步驟

1. 建立名為的檔案 `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。

以下範例組態檔顯示可用的設定，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. 套用檔案中的值 resourceLimits.yaml：

```

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

```

## 自訂安全性內容限制

安裝 Trident Protect 之後，您可以使用組態檔來修改 Trident Protect 容器的 OpenShift 安全性內容限制（SCC）。這些限制定義了 Red Hat OpenShift 叢集中 Pod 的安全性限制。

### 步驟

1. 建立名為的檔案 sccconfig.yaml。
2. 將 SCC 選項新增至檔案，並根據環境需求修改參數。

以下範例顯示 SCC 選項參數的預設值：

```

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

```

下表說明 SCC 選項的參數：

參數	說明	預設
建立	決定是否可以建立 SCC 資源。只有在設定為 true 且 Helm 安裝程序識別 OpenShift 環境時，才會建立 SCC 資源 `scc.create`。如果未在 OpenShift 上操作，或如果設為 false，則 `scc.create` 不會建立任何 SCC 資源。	是的
名稱	指定 SCC 的名稱。	Trident 保護工作
優先順序	定義 SCC 的優先順序。優先順序值較高的 SCC 會在值較低之前進行評估。	1

### 3. 套用檔案中的值 sccconfig.yaml：

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

這會將預設值取代為檔案中指定的值 sccconfig.yaml。

## 設定其他Trident保護 Helm Chart 設定

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：

參數	類型	說明
自動支援代理	字串	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： <a href="http://my.proxy.url">http://my.proxy.url</a> 。
自動支援.不安全	布林值	設定為時跳過AutoSupport代理連線的 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保護AutoSupport捆綁包上傳。當設定為 false，計劃的每日上傳已停用，但您仍然可以手動產生支援包。(預設: true)
恢復跳過命名空間註釋	字串	若要從備份和復原作業中排除的命名空間註釋的逗號分隔清單。允許您根據註釋過濾命名空間。
restoreSkipNamespaceLabels	字串	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. 建立設定檔並命名 `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. 填充後 `'values.yaml'` 具有正確值的文件，應用設定檔：

```
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'` 標誌來指定單一參數：

```
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** 保護 **Pod** 限制在特定節點

您可以使用 Kubernetes `nodeSelector` 節點選擇限制，根據節點標籤來控制哪些節點符合執行 Trident Protect Pod 的資格。根據預設，Trident Protect 僅限於執行 Linux 的節點。您可以根據自己的需求，進一步自訂這些限制。

##### 步驟

1. 建立名為的檔案 `nodeSelectorConfig.yaml`。
2. 將 `nodeSelector` 選項新增至檔案，並修改檔案以新增或變更節點標籤，以根據環境需求加以限制。例如，



下列檔案包含預設的作業系統限制，但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. 套用檔案中的值 `nodeSelectorConfig.yaml`：

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

這會將預設限制取代為您在檔案中指定的限制 `nodeSelectorConfig.yaml`。

## 版權資訊

Copyright © 2025 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。