安全性
Trident

NetApp
April 08, 2026

目錄

安全性

安全性

請按照此處列出的建議、確保您的 Trident 安裝安全可靠。

在其自己的命名空間中執行 **Trident**

防止應用程式、應用程式管理員、使用者和管理應用程式存取 Trident 物件定義或 Pod 非常重要，以確保可靠的儲存並封鎖潛在的惡意活動。

為了將其他應用程式和使用者與 Trident 分開，請務必將 Trident 安裝在其專屬的 Kubernetes 命名空間中 (`trident`。將 Trident 置於其專屬的命名空間可確保只有 Kubernetes 管理人員才能存取 Trident Pod 以及儲存在命名空間 CRD 物件中的成品（例如後端和 CHAP 密碼，如果適用）。您應該確保僅允許管理員存取 Trident 命名空間，從而存取 `tridentctl` 應用程式。

使用 **CHAP** 驗證與 **ONTAP SAN** 後端

Trident 支援基於 CHAP 的 ONTAP SAN 工作負載驗證（使用 `ontap-san`和 `ontap-san-economy`驅動程式）。NetApp 建議 Trident 在主機和儲存後端之間使用雙向 CHAP 進行身份驗證。

對於使用 SAN 儲存驅動程式的 ONTAP 後端， Trident 可以透過 `tridentctl`設定雙向 CHAP 並管理 CHAP 使用者名稱和金鑰。請參閱"準備使用 ONTAP SAN 驅動程式配置後端"以了解 Trident 在 ONTAP 後端上的 CHAP 設定方式。

在 **NetApp HCI** 和 **SolidFire** 後端使用 **CHAP** 驗證

NetApp 建議部署雙向 CHAP，以確保主機與 NetApp HCI 及 SolidFire 後端之間的驗證。Trident 會使用一個包含每個租戶兩組 CHAP 密碼的 secret 物件。安裝 Trident 時，它會管理 CHAP 機密並將其儲存在對應 PV 的 `tridentvolume` CR 物件中。當你建立 PV 時，Trident 會使用 CHAP 機密來啟動 iSCSI 連線，並透過 CHAP 與 NetApp HCI 及 SolidFire 系統進行通訊。

由 Trident 建立的磁碟區不會與任何磁碟區存取群組關聯。

將 **Trident** 與 **NVE** 和 **NAE** 結合使用

NetApp ONTAP 提供靜態資料加密，以保護磁碟被盜、退回或重新利用時的敏感資料。如需詳細資訊，請參閱 "設定 NetApp Volume Encryption 總覽"。

- 如果後端啟用了 NAE，則在 Trident 中配置的任何磁碟區都會啟用 NAE。
  - 您可以設定 NVE 加密標誌以 `""` 建立啟用 NAE 的磁碟區。
- 如果後端未啟用 NAE，則在 Trident 中配置的任何磁碟區都會啟用 NVE，除非在後端組態中將 NVE 加密旗標設為 `false`（預設值）。

在啟用 NAE 的後端上使用 Trident 建立的磁碟區必須使用 NVE 或 NAE 加密。

- 您可以在 Trident 後端設定中將 NVE 加密標誌設定為 `true`，以覆蓋 NAE 加密並按磁碟區使用特定的加密金鑰。
- 在啟用 NAE 的後端上將 NVE 加密標誌設為 `false`會建立一個啟用 NAE 的磁碟區。您無法將 NVE 加密標誌設為 `false`來停用 NAE 加密。

- 您可以透過明確地將 NVE 加密標誌設為 `true` 來在 Trident 中手動建立 NVE 磁碟區。

如需後端組態選項的詳細資訊，請參閱：

- "ONTAP SAN 組態選項"
- "ONTAP NAS 組態選項"

Linux Unified Key Setup（LUKS）

您可以啟用 Linux Unified Key Setup （LUKS） 來加密 Trident 上的 ONTAP SAN 和 ONTAP SAN ECONOMY 磁碟區。Trident 支援對 LUKS 加密磁碟區進行密碼短語輪換和磁碟區擴展。

在 Trident 中，LUKS 加密磁碟區使用 aes-xts-plain64 密碼和模式，如 "NIST" 所建議。

ASA r2 系統不支援 LUKS 加密。有關 ASA r2 系統的資訊，請參閱"了解 ASA r2 儲存系統"。

開始之前

- 工作節點必須安裝 cryptsetup 2.1 或更高版本（但低於 3.0）。如需更多資訊，請造訪 "Gitlab：cryptsetup"。
- 出於效能考慮，NetApp 建議工作節點支援高級加密標準新指令集（AES-NI）。若要驗證是否支援 AES-NI，請執行下列命令：

```
grep "aes" /proc/cpuinfo
```

如果沒有回傳任何內容，則表示您的處理器不支援 AES-NI。有關 AES-NI 的更多資訊，請造訪："Intel：Advanced Encryption Standard Instructions（AES-NI）"。

啟用 **LUKS** 加密

您可以使用 Linux Unified Key Setup (LUKS) 為 ONTAP SAN 和 ONTAP SAN ECONOMY 磁碟區啟用按磁碟區主機端加密。

步驟

1. 在後端配置中定義 LUKS 加密屬性。有關 ONTAP SAN 後端配置選項的更多資訊,請參閱 "ONTAP SAN 組態選項"。

```
{
  "storage": [
    {
      "labels": {
        "luks": "true"
      },
      "zone": "us_east_1a",
      "defaults": {
        "luksEncryption": "true"
      }
    },
    {
      "labels": {
        "luks": "false"
      },
      "zone": "us_east_1a",
      "defaults": {
        "luksEncryption": "false"
      }
    }
  ]
}
```

2. 使用 `parameters.selector` 來定義使用 LUKS 加密的儲存池。例如：

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: luks
provisioner: csi.trident.netapp.io
parameters:
  selector: "luks=true"
  csi.storage.k8s.io/node-stage-secret-name: luks-${pvc.name}
  csi.storage.k8s.io/node-stage-secret-namespace: ${pvc.namespace}
```

3. 建立一個包含 LUKS 密碼的密鑰。例如：

```
kubectl -n trident create -f luks-pvc1.yaml
apiVersion: v1
kind: Secret
metadata:
  name: luks-pvc1
stringData:
  luks-passphrase-name: A
  luks-passphrase: secretA
```

限制

LUKS 加密磁碟區無法利用 ONTAP 重複資料刪除和壓縮功能。

匯入 LUKS 磁碟區的後端組態

若要匯入 LUKS 磁碟區、您必須在後端將 `luksEncryption`設為(`true。 luksEncryption`選項會告知 Trident 磁碟區是否符合 LUKS 規範(`true`或不符合 LUKS 規範(`false、如下列範例所示。

```
version: 1
storageDriverName: ontap-san
managementLIF: 10.0.0.1
dataLIF: 10.0.0.2
svm: trident_svm
username: admin
password: password
defaults:
  luksEncryption: 'true'
  spaceAllocation: 'false'
  snapshotPolicy: default
  snapshotReserve: '10'
```

用於匯入 LUKS 磁碟區的 PVC 組態

若要動態匯入 LUKS 卷，請將註解 `trident.netapp.io/luksEncryption`設為 `true`並在 PVC 中包含啟用 LUKS 的儲存類，如本範例所示。

```
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: luks-pvc
  namespace: trident
  annotations:
    trident.netapp.io/luksEncryption: "true"
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 1Gi
  storageClassName: luks-sc
```

輪換 LUKS 密碼

您可以輪換 LUKS 密碼並確認輪換。

> 請務必在確認所有磁碟區、快照或金鑰均不再引用該密碼短語後再忘記它。如果引用的密碼短語遺失，您可能無法掛載該磁碟區，且資料將保持加密狀態且無法存取。

關於此任務

當指定新的 LUKS 密碼後建立掛載磁碟區的 Pod 時，就會發生 LUKS 密碼輪替。建立新 Pod 時，Trident 會將磁碟區上的 LUKS 密碼與金鑰中的活動密碼進行比較。

- 如果磁碟區上的密碼與密碼中的作用中密碼不符、就會發生輪替。
- 如果磁碟區上的密碼與密碼中的作用中密碼相符、則會忽略 `previous-luks-passphrase` 參數。

步驟

1. 新增 `node-publish-secret-name`和 `node-publish-secret-namespace` StorageClass 參數。例如：

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: csi-san
provisioner: csi.trident.netapp.io
parameters:
  trident.netapp.io/backendType: "ontap-san"
  csi.storage.k8s.io/node-stage-secret-name: luks
  csi.storage.k8s.io/node-stage-secret-namespace: ${pvc.namespace}
  csi.storage.k8s.io/node-publish-secret-name: luks
  csi.storage.k8s.io/node-publish-secret-namespace: ${pvc.namespace}
```

2. 識別磁碟區或快照上現有的複雜密碼。

磁碟區

```
tridentctl -d get volume luks-pvc1
GET http://127.0.0.1:8000/trident/v1/volume/<volumeID>

...luksPassphraseNames:["A"]
```

快照

```
tridentctl -d get snapshot luks-pvc1
GET http://127.0.0.1:8000/trident/v1/volume/<volumeID>/<snapshotID>

...luksPassphraseNames:["A"]
```

3. 更新磁碟區的 LUKS 金鑰，以指定新的和先前的密碼短語。確保 `previous-luke-passphrase-name`和 `previous-luks-passphrase`與先前的密碼短語一致。

```
apiVersion: v1
kind: Secret
metadata:
  name: luks-pvc1
stringData:
  luks-passphrase-name: B
  luks-passphrase: secretB
  previous-luks-passphrase-name: A
  previous-luks-passphrase: secretA
```

4. 建立新的 pod 並掛載磁碟區。這是啟動輪替所必需的。
5. 確認密碼已輪換。

磁碟區

```
tridentctl -d get volume luks-pvc1
GET http://127.0.0.1:8000/trident/v1/volume/<volumeID>

...luksPassphraseNames:["B"]
```

快照

```
tridentctl -d get snapshot luks-pvc1
GET http://127.0.0.1:8000/trident/v1/volume/<volumeID>/<snapshotID>

...luksPassphraseNames:["B"]
```

結果

當磁碟區和快照上僅傳回新複雜密碼時，複雜密碼就會輪替。

> 如果傳回兩個密碼短語，例如 `luksPassphraseNames: ["B", "A"]`，則輪換尚未完成。您可以觸發一個新的 pod 來嘗試完成輪換。

啟用磁碟區擴充

您可以對 LUKS 加密磁碟區啟用磁碟區擴充。

步驟

1. 啟用 `CSINodeExpandSecret` 功能閘控（beta 1.25+）。詳情請參閱 "Kubernetes 1.25：使用 Secrets 實作 CSI 磁碟區的節點驅動擴展"。
2. 新增 `node-expand-secret-name`和 `node-expand-secret-namespace` StorageClass 參數。例如：

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: luks
provisioner: csi.trident.netapp.io
parameters:
  selector: "luks=true"
  csi.storage.k8s.io/node-stage-secret-name: luks-${pvc.name}
  csi.storage.k8s.io/node-stage-secret-namespace: ${pvc.namespace}
  csi.storage.k8s.io/node-expand-secret-name: luks-${pvc.name}
  csi.storage.k8s.io/node-expand-secret-namespace: ${pvc.namespace}
allowVolumeExpansion: true
```

結果

啟動線上儲存擴充功能時，kubelet 會將對應的憑證傳遞給驅動程式。

Kerberos 傳輸中加密

使用 Kerberos 傳輸中加密，您可以對託管叢集和儲存後端之間的流量啟用加密，從而提高資料存取安全性。

Trident 支援以 ONTAP 作為儲存後端的 Kerberos 加密：

- 內部部署 **ONTAP** - Trident 支援透過 NFSv3 和 NFSv4 連線，從 Red Hat OpenShift 和上游 Kubernetes 叢集到內部部署 ONTAP 磁碟區的 Kerberos 加密。

您可以建立、刪除、調整大小、建立快照、複製、唯讀複製和匯入使用 NFS 加密的磁碟區。

使用內部部署 **ONTAP** 磁碟區設定傳輸中 **Kerberos** 加密

您可以為託管叢集和本機 ONTAP 儲存後端之間的儲存流量啟用 Kerberos 加密。

> 對於使用本機 ONTAP 儲存後端的 NFS 流量，僅支援使用 `ontap-nas` 儲存驅動程式進行 Kerberos 加密。

開始之前

- 請確保您可以使用 `tridentctl` 公用程式。
- 請確保您擁有 ONTAP 儲存後端的管理員存取權限。
- 請確保您知道要從 ONTAP 儲存後端共用的磁碟區名稱。
- 請確保您已設定 ONTAP 儲存虛擬機器以支援 NFS 磁碟區的 Kerberos 加密。請參閱 "在資料 LIF 上啟用 Kerberos"以取得相關說明。
- 請確保所有與 Kerberos 加密一起使用的 NFSv4 磁碟區都已正確設定。請參閱 NetApp NFSv4 域配置部分（第 13 頁） "NetApp NFSv4 增強功能與最佳實務指南"。

新增或修改 **ONTAP** 匯出原則

您需要為現有的 ONTAP 匯出策略新增規則，或建立新的匯出策略，以支援對 ONTAP 儲存 VM 根磁碟區以及與上游 Kubernetes 叢集共用的任何 ONTAP 磁碟區進行 Kerberos 加密。您新增的匯出策略規則或建立的新匯出策略需要支援以下存取協定和存取權限：

存取通訊協定

使用 NFS、NFSv3 和 NFSv4 存取協定設定匯出原則。

存取詳細資料

您可以根據磁碟區的需求、配置三種不同版本的 Kerberos 加密之一：

- **Kerberos 5** -（驗證和加密）
- **Kerberos 5i** -（驗證和加密，具有身分保護功能）
- **Kerberos 5p** -（驗證和加密，提供身分和隱私保護）

使用適當的存取權限設定 ONTAP 匯出原則規則。例如，如果叢集將使用 Kerberos 5i 和 Kerberos 5p 加密混合掛載 NFS 磁碟區，請使用下列存取設定：

類型	唯讀存取	讀取 / 寫入存取權	超級使用者存取
UNIX	已啟用	已啟用	已啟用
Kerberos 5i	已啟用	已啟用	已啟用

類型	唯讀存取	讀取 / 寫入存取權	超級使用者存取
Kerberos 5p	已啟用	已啟用	已啟用

有關如何建立 ONTAP 匯出原則和匯出原則規則的資訊，請參閱下列文件：

- "建立匯出原則"
- "將規則新增至匯出原則"

建立儲存後端

您可以建立包含 Kerberos 加密功能的 Trident 儲存後端組態。

關於此任務

建立配置 Kerberos 加密的儲存後端設定檔時，可以使用 `spec.nfsMountOptions` 參數指定三種不同版本的 Kerberos 加密之一：

- `spec.nfsMountOptions: sec=krb5` （驗證與加密）
- `spec.nfsMountOptions: sec=krb5i`（驗證與加密及身分保護）
- `spec.nfsMountOptions: sec=krb5p`（驗證與加密，具備身分與隱私保護）

只能指定一個 Kerberos 等級。如果在參數清單中指定多個 Kerberos 加密等級，則僅使用第一個選項。

步驟

1. 在託管叢集上，使用以下範例建立儲存後端組態檔。將方括號 <> 中的值替換為您環境中的資訊：

```
apiVersion: v1
kind: Secret
metadata:
  name: backend-ontap-nas-secret
type: Opaque
stringData:
  clientID: <CLIENT_ID>
  clientSecret: <CLIENT_SECRET>
---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-ontap-nas
spec:
  version: 1
  storageDriverName: "ontap-nas"
  managementLIF: <STORAGE_VM_MGMT_LIF_IP_ADDRESS>
  dataLIF: <PROTOCOL_LIF_FQDN_OR_IP_ADDRESS>
  svm: <STORAGE_VM_NAME>
  username: <STORAGE_VM_USERNAME_CREDENTIAL>
  password: <STORAGE_VM_PASSWORD_CREDENTIAL>
  nasType: nfs
  nfsMountOptions: ["sec=krb5i"] #can be krb5, krb5i, or krb5p
  qtreesPerFlexvol:
  credentials:
    name: backend-ontap-nas-secret
```

2. 使用上一步建立的組態檔建立後端：

```
tridentctl create backend -f <backend-configuration-file>
```

如果後端建立失敗，則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
```

在您識別並修正組態檔的問題後、您可以再次執行 create 命令。

建立儲存類別

您可以建立儲存類別來配置具有 Kerberos 加密的磁碟區。

關於此任務

建立儲存類別物件時，可以使用 `mountOptions` 參數指定三種不同版本的 Kerberos 加密之一：

- `mountOptions: sec=krb5`（驗證與加密）
- `mountOptions: sec=krb5i`（驗證與加密及身分保護）
- `mountOptions: sec=krb5p`（驗證與加密，具備身分與隱私保護）

只能指定一個 Kerberos 加密等級。如果在參數清單中指定了多個 Kerberos 加密等級，則僅使用第一個選項。如果在儲存後端組態中指定的加密等級與在儲存類別物件中指定的加密等級不同，則以儲存類別物件為準。

步驟

1. 建立一個 StorageClass Kubernetes 物件，請參考以下範例：

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: ontap-nas-sc
provisioner: csi.trident.netapp.io
mountOptions:
  - sec=krb5i #can be krb5, krb5i, or krb5p
parameters:
  backendType: ontap-nas
  storagePools: ontapnas_pool
  trident.netapp.io/nasType: nfs
allowVolumeExpansion: true
```

2. 建立儲存類別：

```
kubectl create -f sample-input/storage-class-ontap-nas-sc.yaml
```

3. 請確保已建立儲存類別：

```
kubectl get sc ontap-nas-sc
```

您應該會看到類似以下內容的輸出：

```
NAME          PROVISIONER             AGE
ontap-nas-sc    csi.trident.netapp.io   15h
```

配置磁碟區

建立儲存後端和儲存類別後，即可配置磁碟區。有關說明，請參閱 "配置磁碟區"。

使用 **Azure NetApp Files** 磁碟區設定傳輸中 **Kerberos** 加密

您可以為託管叢集與單一 Azure NetApp Files 儲存後端或 Azure NetApp Files 儲存後端虛擬池之間的儲存流量啟用 Kerberos 加密。

開始之前

- 請確保已在受管理的 Red Hat OpenShift 叢集上啟用 Trident。
- 請確保您可以使用 `tridentctl` 公用程式。
- 請確保您已為 Kerberos 加密準備好 Azure NetApp Files 儲存後端，請注意需求並遵循 "Azure NetApp Files 文件"中的說明。
- 請確保所有與 Kerberos 加密一起使用的 NFSv4 磁碟區都已正確設定。請參閱 NetApp NFSv4 域配置部分（第 13 頁） "NetApp NFSv4 增強功能與最佳實務指南"。

建立儲存後端

您可以建立包含 Kerberos 加密功能的 Azure NetApp Files 儲存後端組態。

關於此任務

建立配置 Kerberos 加密的儲存後端組態檔時，您可以將其定義為套用於下列兩個層級之一：

- 使用 `spec.kerberos`欄位的 **storage backend level**
- 使用 `spec.storage.kerberos`欄位的*虛擬資源池層級*

在虛擬資源池層級定義組態時，會使用儲存類別中的標籤來選取資源池。

無論在哪個級別、您都可以指定三種不同版本的 Kerberos 加密之一：

- `kerberos: sec=krb5` （驗證與加密）
- `kerberos: sec=krb5i`（驗證與加密及身分保護）
- `kerberos: sec=krb5p`（驗證與加密，具備身分與隱私保護）

步驟

1. 在託管叢集上，根據您需要定義儲存後端的位置（儲存後端等級或虛擬資源池等級），使用下列其中一個範例建立儲存後端組態檔。將方括號 <> 中的值替換為您環境中的資訊：

儲存後端層級範例

```
apiVersion: v1
kind: Secret
metadata:
  name: backend-tbc-secret
type: Opaque
stringData:
  clientID: <CLIENT_ID>
  clientSecret: <CLIENT_SECRET>

---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-tbc
spec:
  version: 1
  storageDriverName: azure-netapp-files
  subscriptionID: <SUBSCRIPTION_ID>
  tenantID: <TENANT_ID>
  location: <AZURE_REGION_LOCATION>
  serviceLevel: Standard
  networkFeatures: Standard
  capacityPools: <CAPACITY_POOL>
  resourceGroups: <RESOURCE_GROUP>
  netappAccounts: <NETAPP_ACCOUNT>
  virtualNetwork: <VIRTUAL_NETWORK>
  subnet: <SUBNET>
  nasType: nfs
  kerberos: sec=krb5i #can be krb5, krb5i, or krb5p
  credentials:
    name: backend-tbc-secret
```

虛擬資源池層級範例

```
---
apiVersion: v1
kind: Secret
metadata:
  name: backend-tbc-secret
type: Opaque
stringData:
  clientID: <CLIENT_ID>
  clientSecret: <CLIENT_SECRET>

---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-tbc
spec:
  version: 1
  storageDriverName: azure-netapp-files
  subscriptionID: <SUBSCRIPTION_ID>
  tenantID: <TENANT_ID>
  location: <AZURE_REGION_LOCATION>
  serviceLevel: Standard
  networkFeatures: Standard
  capacityPools: <CAPACITY_POOL>
  resourceGroups: <RESOURCE_GROUP>
  netappAccounts: <NETAPP_ACCOUNT>
  virtualNetwork: <VIRTUAL_NETWORK>
  subnet: <SUBNET>
  nasType: nfs
  storage:
    - labels:
        type: encryption
      kerberos: sec=krb5i #can be krb5, krb5i, or krb5p
  credentials:
    name: backend-tbc-secret
```

2. 使用上一步建立的組態檔建立後端：

```
tridentctl create backend -f <backend-configuration-file>
```

如果後端建立失敗，則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
```

在您識別並修正組態檔的問題後、您可以再次執行 create 命令。

建立儲存類別

您可以建立儲存類別來配置具有 Kerberos 加密的磁碟區。

步驟

1. 建立一個 StorageClass Kubernetes 物件，請參考以下範例：

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: sc-nfs
provisioner: csi.trident.netapp.io
parameters:
  backendType: azure-netapp-files
  trident.netapp.io/nasType: nfs
  selector: type=encryption
```

2. 建立儲存類別：

```
kubectl create -f sample-input/storage-class-sc-nfs.yaml
```

3. 請確保已建立儲存類別：

```
kubectl get sc -sc-nfs
```

您應該會看到類似以下內容的輸出：

```
NAME          PROVISIONER             AGE
sc-nfs        csi.trident.netapp.io   15h
```

配置磁碟區

建立儲存後端和儲存類別後，即可配置磁碟區。有關說明，請參閱 "配置磁碟區"。