安全
Trident

NetApp
January 15, 2026

目錄

安全

安全

請按照此處列出的建議，確保您的Trident安裝安全可靠。

在Trident自身的命名空間中運行它

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隔離，請務必將Trident安裝在其自身的 Kubernetes 命名空間中。(`trident`）。將Trident放在自己的命名空間中，可以確保只有 Kubernetes 管理員才能存取Trident pod 和儲存在命名空間 CRD 物件中的工件（例如後端和 CHAP 金鑰，如果適用）。您應該確保只允許管理員存取Trident命名空間，從而獲得存取權限。 `tridentctl`應用。

使用 CHAP 驗證與ONTAP SAN 後端配合使用

Trident支援基於 CHAP 的ONTAP SAN 工作負載驗證（使用 `ontap-san`和 `ontap-san-economy`司機）。NetApp建議使用Trident的雙向 CHAP 進行主機與儲存後端之間的驗證。

對於使用 SAN 儲存驅動程式的ONTAP後端， Trident可以透過以下方式設定雙向 CHAP 並管理 CHAP 使用者名稱和金鑰： `tridentctl` 。請參閱"準備配置後端ONTAP SAN 驅動程式"了解Trident如何在ONTAP後端設定CHAP。

使用 CHAP 驗證連線NetApp HCI和SolidFire後端

NetApp建議部署雙向 CHAP，以確保主機與NetApp HCI和SolidFire後端之間的驗證。 Trident使用一個包含每個租用戶兩個 CHAP 密碼的秘密物件。安裝Trident後，它會管理 CHAP 金鑰並將其儲存在一個位置。`tridentvolume`對應 PV 的 CR 物件。在建立 PV 時， Trident使用 CHAP 金鑰啟動 iSCSI 會話，並透過 CHAP 與NetApp HCI和SolidFire系統通訊。

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，以保護敏感數據，防止磁碟被盜、退回或重新利用。有關詳細信息，請參閱"配置NetApp卷加密概述"。

- 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。
  - 您可以將 NVE 加密標誌設定為 `""`建立支援 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 NVE 加密標誌設為 。 `false`（後端配置中的預設值）。

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。

- 您可以將 NVE 加密標誌設定為 `true`在Trident後端設定中，可以覆蓋 NAE 加密，並按磁碟區使用特定的加密金鑰。
- 將 NVE 加密標誌設為 `false`在啟用 NAE 的後端上建立啟用 NAE 的磁碟區。您無法透過將 NVE 加密標誌設為 來停用 NAE 加密。 `false`。

- 您可以透過明確設定 NVE 加密標誌，在Trident中手動建立 NVE 磁碟區。 `true`。

有關後端配置選項的更多信息，請參閱：

- "ONTAP SAN 配置選項"
- "ONTAP NAS 設定選項"

Linux 統一金鑰設定 (LUKS)

您可以啟用 Linux 統一密鑰設定 (LUKS) 來加密Trident上的ONTAP SAN 和ONTAP SAN ECONOMY 磁碟區。 Trident支援 LUKS 加密磁碟區的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，這是建議的。"美國國家標準與技術研究院"。

ASA r2 系統不支援 LUKS 加密。有關ASA r2 系統的信息，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）。欲了解更多信息，請訪問"GitLab：加密設定"。
- 出於效能方面的考慮， NetApp建議工作節點支援進階加密標準新指令 (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 的更多信息，請訪問："英特爾：高階加密標準指令集（AES-NI）"。

啟用 LUKS 加密

您可以使用 Linux 統一密鑰設定 (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信息，請參閱"ONTAP SAN 配置選項"。

```
{
  "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`使用 LUKS 加密定義儲存池。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 密碼短語的金鑰。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。

導入 LUKS 磁碟區的後端配置

若要匯入 LUKS 卷，您必須設定 `luksEncryption`到(`true`在後端。這 `luksEncryption`此選項用於告知Trident該磁碟區是否符合 LUKS 標準。(`true`或不符合 LUKS 標準(`false`）如下例所示。

```
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'
```

用於導入 LUKS 磁碟區的 PVC 配置

若要動態匯入 LUKS 卷，請設定註釋 `trident.netapp.io/luksEncryption`到 `true`並在 PVC 中包含一個支援 LUKS 的儲存類，如本例所示。

```
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 密碼短語

您可以輪換 LUKS 密碼短語並確認輪換。

> 在您確認任何磁碟區、快照或金鑰不再引用該密碼短語之前，請勿忘記該密碼短語。如果引用的密碼短語遺失，您可能無法掛載卷，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 密碼短語後建立掛載磁碟區的 pod 時，就會發生 LUKS 密碼短語輪替。當建立新的 pod 時，Trident會將磁碟區上的 LUKS 密碼短語與金鑰中的活動密碼短語進行比較。

- 如果磁碟區上的密碼短語與金鑰中的活動密碼短語不匹配，則會發生輪換。
- 如果磁碟區上的密碼短語與金鑰中的活動密碼短語匹配，則 `previous-luks-passphrase`參數被忽略。

步驟

1. 添加 `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已存在的密碼短語。

體積

```
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`與先前的密碼短語相符。

```
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. 建立一個新的 pod，掛載該磁碟區。這是啟動旋轉所必需的步驟。
5. 確認密碼短語已輪換。

體積

```
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密碼短語時，密碼短語就會輪替。

例如，如果傳回兩個密碼短語。 `luksPassphraseNames: ["B", "A"]`旋轉不完整。您可以觸發一個新的機艙來嘗試完成輪換。

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`功能門（beta 1.25+）。參考 "Kubernetes 1.25：使用金鑰實作 CSI 磁碟區的節點驅動擴展" 了解詳情。
2. 添加 `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 飛行中加密

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：

- 本地**ONTAP** - Trident支援透過 NFSv3 和 NFSv4 連線對來自 Red Hat OpenShift 和上游 Kubernetes 叢集的本機ONTAP磁碟區進行 Kerberos 加密。

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

配置本地**ONTAP**磁碟區的飛行中 **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。

僅使用以下方式支援對本地ONTAP儲存後端的 NFS 流量進行 Kerberos 加密： `ontap-nas`儲存驅動程式。

開始之前

- 確保您可以訪問 `tridentctl`公用事業。
- 請確保您擁有ONTAP儲存後端的管理員權限。
- 請確保您知道要從ONTAP儲存後端共用的磁碟區的名稱。
- 請確保您已準備好ONTAP儲存 VM，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 網域設定部分（第 13 頁）。 "NetApp NFSv4 增強功能與最佳實務指南" 。

新增或修改**ONTAP**匯出策略

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您建立的新匯出策略需要支援以下存取協定和存取權限：

訪問協定

配置導出策略，支援 NFS、NFSv3 和 NFSv4 存取協定。

訪問詳情

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

配置ONTAP導出策略規則，使其具有適當的存取權限。例如，如果叢集將使用 Kerberos 5i 和 Kerberos 5p 混合加密方式掛載 NFS 卷，請使用下列存取設定：

類型	只讀存取權限	讀/寫權限	超級使用者存取權限
UNIX	已啟用	已啟用	已啟用
Kerberos 5i	已啟用	已啟用	已啟用
Kerberos 5p	已啟用	已啟用	已啟用

有關如何建立ONTAP匯出策略和匯出策略規則，請參閱以下文件：

- "建立導出策略"
- "在匯出策略中新增規則"

建立儲存後端

您可以建立包含 Kerberos 加密功能的Trident儲存後端設定。

關於此任務

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

- `spec.nfsMountOptions: sec=krb5`（身份驗證和加密）
- `spec.nfsMountOptions: sec=krb5i`（身份驗證和加密，以及身份保護）
- `spec.nfsMountOptions: sec=krb5p`（身份驗證和加密，以及身分和隱私保護）

只能指定一個 Kerberos 等級。如果在參數清單中指定多個 Kerberos 加密級別，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設定檔。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. 使用上一步建立的設定檔建立後端：

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

如果後端建立失敗，則後端配置存在問題。您可以透過執行以下命令查看日誌以確定原因：

```
tridentctl logs
```

在您發現並修正設定檔中的問題後，您可以再次執行建立命令。

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，可以使用下列方式指定三種不同版本的 Kerberos 加密之一： `mountOptions`範圍：

- `mountOptions: sec=krb5`（身份驗證和加密）
- `mountOptions: sec=krb5i`（身份驗證和加密，以及身份保護）
- `mountOptions: sec=krb5p`（身份驗證和加密，以及身分和隱私保護）

只能指定一個 Kerberos 等級。如果在參數清單中指定多個 Kerberos 加密級別，則僅使用第一個選項。如果在儲存後端設定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 對象，請參考下列範例：

```
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
```

供應量

建立儲存後端和儲存類別之後，現在可以設定捲了。有關說明，請參閱 "提供一定量"。

使用Azure NetApp Files磁碟區設定飛行中 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 Red Hat OpenShift 叢集上啟用Trident 。
- 確保您可以訪問 `tridentctl`公用事業。
- 請確保您已按照以下說明準備好Azure NetApp Files儲存後端以進行 Kerberos 加密：注意相關要求並按照說明進行操作。 "Azure NetApp Files文檔" 。
- 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 網域設定部分（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儲存後端設定。

關於此任務

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

- 使用*儲存後端等級* `spec.kerberos`場地
- 使用*虛擬池層級* `spec.storage.kerberos`場地

在虛擬池層級定義配置時，使用儲存類別中的標籤選擇池。

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`（身份驗證和加密）
- `kerberos: sec=krb5i`（身份驗證和加密，以及身份保護）
- `kerberos: sec=krb5p`（身份驗證和加密，以及身分和隱私保護）

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池層級），使用下列範例之一建立儲存後端設定檔。將方括號 <> 中的值替換為您環境中的資訊：

儲存後端範例

```
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. 使用上一步建立的設定檔建立後端：

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

如果後端建立失敗，則後端配置存在問題。您可以透過執行以下命令查看日誌以確定原因：

```
tridentctl logs
```

在您發現並修正設定檔中的問題後，您可以再次執行建立命令。

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 對象，請參考下列範例：

```
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
```

供應量

建立儲存後端和儲存類別之後，現在可以設定捲了。有關說明，請參閱 "提供一定量"。