



設定及管理後端 Astra Trident

NetApp
April 18, 2024

目錄

設定及管理後端	1
設定後端	1
Azure NetApp Files	1
設定Cloud Volumes Service 適用於Google Cloud後端的功能	16
設定NetApp HCI 一個不只是功能的SolidFire 後端	31
支援SAN驅動程式ONTAP	37
ASNAS驅動程式ONTAP	59
Amazon FSX for NetApp ONTAP 產品	86
使用kubect建立後端	101
管理後端	107

設定及管理後端

設定後端

後端定義了Astra Trident與儲存系統之間的關係。它告訴Astra Trident如何與該儲存系統通訊、以及Astra Trident如何從該儲存系統配置磁碟區。

Astra Trident會自動從後端提供符合儲存類別所定義需求的儲存資源池。瞭解如何設定儲存系統的後端。

- ["設定Azure NetApp Files 一個靜態後端"](#)
- ["設定Cloud Volumes Service 適用於Google Cloud Platform後端的功能"](#)
- ["設定NetApp HCI 一個不只是功能的SolidFire 後端"](#)
- ["使用ONTAP 功能不一的Cloud Volumes ONTAP NAS驅動程式來設定後端"](#)
- ["使用ONTAP 不支援的Cloud Volumes ONTAP SAN驅動程式來設定後端"](#)
- ["使用Astra Trident搭配Amazon FSX for NetApp ONTAP 解決方案"](#)

Azure NetApp Files

設定Azure NetApp Files 一個靜態後端

您可以將 Azure NetApp Files 設定為 Astra Trident 的後端。您可以使用 Azure NetApp Files 後端連接 NFS 和 SMB 磁碟區。Astra Trident 也支援使用 Azure Kubernetes Services (aks) 叢集的託管身分識別來進行認證管理。

Azure NetApp Files 驅動程式詳細資料

Astra Trident 提供下列 Azure NetApp Files 儲存驅動程式來與叢集通訊。支援的存取模式包括：
ReadWriteOnce (rwo)、*ReadOnlyMany* (ROX)、*_ReadWriteMany* (rwx)、*_ReadWriteOncePod* (RWOP)。

驅動程式	傳輸協定	Volume模式	支援的存取模式	支援的檔案系統
《azure-NetApp-fil形》	NFS 中小企業	檔案系統	Rwo、ROX、rwx、RWOP	nfs、smb

考量

- 此支援服務不支援小於100 GB的磁碟區。Azure NetApp Files如果要求較小的磁碟區、Astra Trident 會自動建立 100-GiB 磁碟區。
- Astra Trident僅支援安裝在Windows節點上執行的Pod上的SMB磁碟區。

管理的身分識別

Astra Trident 支援 ["託管身分識別"](#) 適用於 Azure Kubernetes Services 叢集。若要善用託管身分識別所提供的簡化認證管理功能、您必須具備：

- 使用 aks 部署的 Kubernetes 叢集
- 在 aks Kubernetes 叢集上設定的託管身分識別
- 安裝的 Astra Trident 包含 cloudProvider 以指定 "Azure"。

Trident 運算子

若要使用 Trident 運算子安裝 Astra Trident、請編輯 tridentorchestrator_cr.yaml 以設定 cloudProvider 至 "Azure"。例如：

```
apiVersion: trident.netapp.io/v1
kind: TridentOrchestrator
metadata:
  name: trident
spec:
  debug: true
  namespace: trident
  imagePullPolicy: IfNotPresent
  cloudProvider: "Azure"
```

掌舵

以下範例安裝 Astra Trident Set cloudProvider 使用環境變數將資源移轉至 Azure \$CP：

```
helm install trident trident-operator-100.2402.0.tgz --create
--namespace --namespace <trident-namespace> --set cloudProvider=$CP
```

`tridentctl`

下列範例安裝 Astra Trident 並設定 cloudProvider 標記為 Azure：

```
tridentctl install --cloud-provider="Azure" -n trident
```

雲端身分識別

雲端身分識別可讓 Kubernetes Pod 以工作負載身分驗證來存取 Azure 資源、而非提供明確的 Azure 認證。

若要在 Azure 中使用雲端身分識別、您必須具備：

- 使用 aks 部署的 Kubernetes 叢集
- 在 OKS Kubernetes 叢集上設定的工作負載識別和 oidc-c 發行者
- 安裝的 Astra Trident 包含 cloudProvider 以指定 "Azure" 和 cloudIdentity 指定工作負載身分識別

Trident 運算子

若要使用 Trident 運算子安裝 Astra Trident，請編輯 `tridentorchestrator_cr.yaml` 以設定 `cloudProvider` 至 "Azure" 並設定 `cloudIdentity` 至 `azure.workload.identity/client-id: xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx`。

例如：

```
apiVersion: trident.netapp.io/v1
kind: TridentOrchestrator
metadata:
  name: trident
spec:
  debug: true
  namespace: trident
  imagePullPolicy: IfNotPresent
  cloudProvider: "Azure"
  *cloudIdentity: 'azure.workload.identity/client-id: xxxxxxxx-xxxx-
  xxxx-xxxx-xxxxxxxxxxxx' *
```

掌舵

使用下列環境變數設定 * 雲端供應商 (CP) * 和 * 雲端身分識別 (CI) * 旗標的值：

```
export CP="Azure"
export CI="azure.workload.identity/client-id: xxxxxxxx-xxxx-xxxx-xxxx-
xxxxxxxxxxxx"
```

以下範例安裝 Astra Trident 和 Sets `cloudProvider` 使用環境變數將資源移轉至 Azure `$CP` 並設定 `cloudIdentity` 使用環境變數 `$CI`：

```
helm install trident trident-operator-100.2402.0.tgz --set
cloudProvider=$CP --set cloudIdentity=$CI
```

<code>tridentctl</code>

使用下列環境變數設定 * 雲端供應商 * 和 * 雲端 IDENTITY * 旗標的值：

```
export CP="Azure"
export CI="azure.workload.identity/client-id: xxxxxxxx-xxxx-xxxx-xxxx-
xxxxxxxxxxxx"
```

下列範例安裝 Astra Trident 並設定 `cloud-provider` 標記為 `$CP`` 和 ``cloud-identity` 至 `$CI`：

```
tridentctl install --cloud-provider=$CP --cloud-identity="$CI" -n
trident
```

準備設定Azure NetApp Files 一個功能完善的後端

在您設定Azure NetApp Files 完後端功能之前、您必須確保符合下列要求。

NFS 和 SMB 磁碟區的必要條件

如果您是第一次使用 Azure NetApp Files 、或是在新位置使用、則必須先進行一些初始設定、才能設定 Azure NetApp Files 並建立 NFS Volume 。請參閱 "[Azure：設定Azure NetApp Files 功能以建立NFS Volume](#)"。

若要設定及使用 "[Azure NetApp Files](#)" 後端、您需要下列項目：



- subscriptionID、tenantID、clientID、location`和 `clientSecret 在 AKS 叢集上使用託管身分識別時為選用項目。
- tenantID、clientID`和 `clientSecret 在 AKS 叢集上使用雲端身分識別時為選用項目。

- 容量集區。請參閱 "[Microsoft：為 Azure NetApp Files 建立容量集區](#)"。
- 委派給 Azure NetApp Files 的子網路。請參閱 "[Microsoft：將子網路委派給 Azure NetApp Files](#)"。
- Azure訂閱提供的「SubscriptionID」Azure NetApp Files （含功能不支援的功能）。
- tenantID、clientID`和 `clientSecret 從 "[應用程式註冊](#)" 在Azure Active Directory中、具備Azure NetApp Files 充分的權限執行此功能。應用程式登錄應使用下列其中一項：
 - 擁有者或貢獻者角色 "[由Azure預先定義](#)"。
 - 答 "[自訂貢獻者角色](#)" 在訂購層級 (assignableScopes) 具有下列權限、僅限於Astra Trident所需的權限。建立自訂角色之後、"[使用Azure入口網站指派角色](#)"。

```
{
  "id": "/subscriptions/<subscription-id>/providers/Microsoft.Authorization/roleDefinitions/<role-definition-id>",
  "properties": {
    "roleName": "custom-role-with-limited-perms",
    "description": "custom role providing limited permissions",
    "assignableScopes": [
      "/subscriptions/<subscription-id>"
    ],
    "permissions": [
      {
        "actions": [

"Microsoft.NetApp/netAppAccounts/capacityPools/read",

"Microsoft.NetApp/netAppAccounts/capacityPools/write",

"Microsoft.NetApp/netAppAccounts/capacityPools/volumes/read",

"Microsoft.NetApp/netAppAccounts/capacityPools/volumes/write",

"Microsoft.NetApp/netAppAccounts/capacityPools/volumes/delete",

"Microsoft.NetApp/netAppAccounts/capacityPools/volumes/snapshots/read",

"Microsoft.NetApp/netAppAccounts/capacityPools/volumes/snapshots/write",

"Microsoft.NetApp/netAppAccounts/capacityPools/volumes/snapshots/delete",

"Microsoft.NetApp/netAppAccounts/capacityPools/volumes/MountTargets/read",

          "Microsoft.Network/virtualNetworks/read",

"Microsoft.Network/virtualNetworks/subnets/read",

"Microsoft.Features/featureProviders/subscriptionFeatureRegistrations/read",
```

```

"Microsoft.Features/featureProviders/subscriptionFeatureRegistrations/write",

"Microsoft.Features/featureProviders/subscriptionFeatureRegistrations/delete",

                                "Microsoft.Features/features/read",
                                "Microsoft.Features/operations/read",
                                "Microsoft.Features/providers/features/read",

"Microsoft.Features/providers/features/register/action",

"Microsoft.Features/providers/features/unregister/action",

"Microsoft.Features/subscriptionFeatureRegistrations/read"
                                ],
                                "notActions": [],
                                "dataActions": [],
                                "notDataActions": []
                                }
                                ]
                                }
                                }

```

- Azure location 至少包含一個 ["委派的子網路"](#)。從Trident 22.01起 location 參數是後端組態檔最上層的必填欄位。會忽略虛擬資源池中指定的位置值。
- 以供使用 Cloud Identity 請取得 `client ID` 從 ["使用者指派的託管身分識別"](#) 並在中指定該 ID
azure.workload.identity/client-id: xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx。

SMB磁碟區的其他需求

若要建立 SMB Volume 、您必須具備：

- Active Directory 已設定並連線至 Azure NetApp Files 。請參閱 ["Microsoft：建立及管理 Azure NetApp Files 的 Active Directory 連線"](#)。
- Kubernetes叢集具備Linux控制器節點、以及至少一個執行Windows Server 2019的Windows工作節點。Astra Trident僅支援安裝在Windows節點上執行的Pod上的SMB磁碟區。
- 至少有一個 Astra Trident 秘密、內含您的 Active Directory 認證、以便 Azure NetApp Files 能夠驗證至 Active Directory 。以產生機密 smbcreds：

```

kubectl create secret generic smbcreds --from-literal username=user
--from-literal password='password'

```

- 設定為Windows服務的SCSI Proxy。若要設定 csi-proxy、請參閱 ["GitHub：csi Proxy"](#) 或 ["GitHub：適用於Windows的SCSI Proxy"](#) 適用於Windows上執行的Kubernetes節點。

列舉後端組態選項與範例Azure NetApp Files

瞭解 Azure NetApp Files 的 NFS 和 SMB 後端組態選項、並檢閱組態範例。

後端組態選項

Astra Trident 使用您的後端組態（子網路、虛擬網路、服務層級和位置）、在所要求位置的可用容量集區上建立 Azure NetApp Files Volume、並符合所要求的服務層級和子網路。



Astra Trident不支援手動QoS容量集區。

Azure NetApp Files 後端提供這些組態選項。

參數	說明	預設
「分度」		永遠為1
「storageDriverName」	儲存驅動程式名稱	「Azure - NetApp-Files」
「後端名稱」	自訂名稱或儲存後端	驅動程式名稱+「_」+隨機字元
《訂閱ID》	Azure訂閱的訂閱ID 在 AKS 叢集上啟用受管理的身分識別時為選用項目。	
「TenantId」	應用程式註冊的租戶ID 在 AKS 叢集上使用託管身分識別或雲端身分識別時、為選用項目。	
"clientId"	應用程式註冊的用戶端ID 在 AKS 叢集上使用託管身分識別或雲端身分識別時、為選用項目。	
「客戶機密」	應用程式註冊的用戶端機密 在 AKS 叢集上使用託管身分識別或雲端身分識別時、為選用項目。	
《服務層級》	其中一種是「標準」、「高級」或「超高」	""（隨機）
位置	要建立新磁碟區的Azure位置名稱 在 AKS 叢集上啟用受管理的身分識別時為選用項目。	
"來源群組"	用於篩選已探索資源的資源群組清單	「[]」（無篩選器）
《netappAccounts》	篩選探索資源的NetApp帳戶清單	「[]」（無篩選器）
《容量Pools》	用於篩選已探索資源的容量集區清單	「[]」（無篩選器、隨機）
「虛擬化網路」	具有委派子網路的虛擬網路名稱	"

參數	說明	預設
《Subnet》	委派給「microsoft.Netapp/volumes」的子網路名稱	"
《網路功能》	Volume的vnet功能集可能是 Basic 或 Standard。並非所有地區都提供網路功能、可能必須在訂閱中啟用。指定 networkFeatures 如果未啟用此功能、則會導致磁碟區資源配置失敗。	"
「nfsMountOptions」	精細控制NFS掛載選項。SMB磁碟區已忽略。若要使用NFS 4.1版掛載磁碟區、請包含 nfsvers=4 在以逗號分隔的掛載選項清單中、選擇NFS v4.1。儲存類別定義中設定的掛載選項會覆寫在後端組態中設定的掛載選項。	"nfsvers=3"
《限制Volume大小》	如果要求的磁碟區大小高於此值、則資源配置失敗	""（預設不強制執行）
「DebugTraceFlags」	疑難排解時要使用的偵錯旗標。範例：「{"API":假、「方法」：真、「探索」：true}。除非您正在進行疑難排解並需要詳細的記錄傾印、否則請勿使用此功能。	null
nasType	設定NFS或SMB磁碟區建立。選項包括 nfs、smb 或null。NFS磁碟區的預設值設為null。	nfs



如需網路功能的詳細資訊、請參閱 ["設定Azure NetApp Files 適用於某個聲音量的網路功能"](#)。

必要的權限與資源

如果您在建立 PVC 時收到「找不到容量集區」錯誤、您的應用程式註冊可能沒有相關的必要權限和資源（子網路、虛擬網路、容量集區）。如果啟用偵錯、Astra Trident會記錄在建立後端時探索到的Azure資源。確認使用的角色是否適當。

的值 resourceGroups、netappAccounts、capacityPools、virtualNetwork和 `subnet` 可以使用簡短或完整名稱來指定。在大多數情況下、建議使用完整名稱、因為短名稱可以符合多個名稱相同的資源。

。resourceGroups、netappAccounts和 `capacityPools` 值是篩選器、可將探索到的資源集合限制在此儲存後端可用的資源、並可任意組合指定。完整名稱格式如下：

類型	格式
資源群組	<資源群組>
NetApp帳戶	資源群組//<NetApp帳戶>
容量資源池	資源群組//<NetApp帳戶>/<容量資源池>

類型	格式
虛擬網路	資源群組//<虛擬網路>
子網路	資源群組//<虛擬網路>/<子網路>

Volume資源配置

您可以在組態檔的特殊區段中指定下列選項、以控制預設的Volume資源配置。請參閱 [\[組態範例\]](#) 以取得詳細資料。

參數	說明	預設
「匯出規則」	匯出新磁碟區的規則。 exportRule 必須是以逗號分隔的清單、以CIDR表示法列出所有的IPv4位址或IPv4子網路組合。SMB磁碟區已忽略。	「0.00.0.0/0」
「snapshotDir	控制.snapshot目錄的可見度	"假"
《大小》	新磁碟區的預設大小	100公克
「unixPermissions」	新磁碟區的UNIX權限（4個八進位數字）。SMB磁碟區已忽略。	""（預覽功能、訂閱時需要白名單）

組態範例

下列範例顯示基本組態、讓大部分參數保留預設值。這是定義後端最簡單的方法。

最小組態

這是絕對最低的后端組態。有了這項組態、Astra Trident 會探索您在設定位置中委派給 Azure NetApp Files 的所有 NetApp 帳戶、容量集區和子網路、並隨機將新磁碟區放在其中一個集區和子網路上。因為 nasType 省略 nfs 預設會套用、後端會為NFS磁碟區進行資源配置。

當您剛開始使用 Azure NetApp Files 並試用時、這項組態是理想的選擇、但實際上您會想要為您所配置的磁碟區提供額外的範圍。

```
---
version: 1
storageDriverName: azure-netapp-files
subscriptionID: 9f87c765-4774-fake-ae98-a721add45451
tenantID: 68e4f836-edc1-fake-bff9-b2d865ee56cf
clientID: dd043f63-bf8e-fake-8076-8de91e5713aa
clientSecret: SECRET
location: eastus
```

管理的身分識別

此後端組態已不再如此 subscriptionID、tenantID、clientID 和 clientSecret，使用託管身分識別時為選用功能。

```
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-tbc-anf-1
  namespace: trident
spec:
  version: 1
  storageDriverName: azure-netapp-files
  capacityPools: ["ultra-pool"]
  resourceGroups: ["aks-ami-eastus-rg"]
  netappAccounts: ["smb-na"]
  virtualNetwork: eastus-prod-vnet
  subnet: eastus-anf-subnet
```

雲端身分識別

此後端組態已不再如此 tenantID、clientID 和 clientSecret（使用雲端身分識別時為選用功能）。

```
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-tbc-anf-1
  namespace: trident
spec:
  version: 1
  storageDriverName: azure-netapp-files
  capacityPools: ["ultra-pool"]
  resourceGroups: ["aks-ami-eastus-rg"]
  netappAccounts: ["smb-na"]
  virtualNetwork: eastus-prod-vnet
  subnet: eastus-anf-subnet
  location: eastus
  subscriptionID: 9f87c765-4774-fake-ae98-a721add45451
```

此後端組態可將Volume置於Azure中 `eastus` 位置 `Ultra` 容量資源池：Astra Trident 會自動探索該位置中委派給 Azure NetApp Files 的所有子網路、並隨機在其中一個子網路上放置新的磁碟區。

```
---
version: 1
storageDriverName: azure-netapp-files
subscriptionID: 9f87c765-4774-fake-ae98-a721add45451
tenantID: 68e4f836-edc1-fake-bff9-b2d865ee56cf
clientID: dd043f63-bf8e-fake-8076-8de91e5713aa
clientSecret: SECRET
location: eastus
serviceLevel: Ultra
capacityPools:
- application-group-1/account-1/ultra-1
- application-group-1/account-1/ultra-2
```

此後端組態可進一步將磁碟區放置範圍縮小至單一子網路、並修改部分Volume資源配置預設值。

```
---
version: 1
storageDriverName: azure-netapp-files
subscriptionID: 9f87c765-4774-fake-ae98-a721add45451
tenantID: 68e4f836-edc1-fake-bff9-b2d865ee56cf
clientID: dd043f63-bf8e-fake-8076-8de91e5713aa
clientSecret: SECRET
location: eastus
serviceLevel: Ultra
capacityPools:
- application-group-1/account-1/ultra-1
- application-group-1/account-1/ultra-2
virtualNetwork: my-virtual-network
subnet: my-subnet
networkFeatures: Standard
nfsMountOptions: vers=3,proto=tcp,timeo=600
limitVolumeSize: 500Gi
defaults:
  exportRule: 10.0.0.0/24,10.0.1.0/24,10.0.2.100
  snapshotDir: 'true'
  size: 200Gi
  unixPermissions: '0777'
```

此後端組態可在單一檔案中定義多個儲存集區。當您有多個容量集區支援不同的服務層級、而且想要在Kubernetes中建立代表這些層級的儲存類別時、這很有用。虛擬資源池標籤是用來區分資源池的依據 performance。

```
---
version: 1
storageDriverName: azure-netapp-files
subscriptionID: 9f87c765-4774-fake-ae98-a721add45451
tenantID: 68e4f836-edc1-fake-bff9-b2d865ee56cf
clientID: dd043f63-bf8e-fake-8076-8de91e5713aa
clientSecret: SECRET
location: eastus
resourceGroups:
- application-group-1
networkFeatures: Basic
nfsMountOptions: vers=3,proto=tcp,timeo=600
labels:
  cloud: azure
storage:
- labels:
    performance: gold
    serviceLevel: Ultra
    capacityPools:
    - ultra-1
    - ultra-2
    networkFeatures: Standard
- labels:
    performance: silver
    serviceLevel: Premium
    capacityPools:
    - premium-1
- labels:
    performance: bronze
    serviceLevel: Standard
    capacityPools:
    - standard-1
    - standard-2
```

儲存類別定義

以下內容 StorageClass 定義請參閱上述儲存資源池。

使用的範例定義 `parameter.selector` 欄位

使用 `parameter.selector` 您可以為每個項目指定 `StorageClass` 用於裝載磁碟區的虛擬集區。該磁碟區會在所選的資源池中定義各個層面。

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: gold
provisioner: csi.trident.netapp.io
parameters:
  selector: "performance=gold"
allowVolumeExpansion: true
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: silver
provisioner: csi.trident.netapp.io
parameters:
  selector: "performance=silver"
allowVolumeExpansion: true
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: bronze
provisioner: csi.trident.netapp.io
parameters:
  selector: "performance=bronze"
allowVolumeExpansion: true
```

SMB磁碟區的定義範例

使用 `nasType`、`node-stage-secret-name` 和 `node-stage-secret-namespace`、您可以指定SMB磁碟區、並提供所需的Active Directory認證資料。

預設命名空間的基本組態

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: anf-sc-smb
provisioner: csi.trident.netapp.io
parameters:
  backendType: "azure-netapp-files"
  trident.netapp.io/nasType: "smb"
  csi.storage.k8s.io/node-stage-secret-name: "smbcreds"
  csi.storage.k8s.io/node-stage-secret-namespace: "default"
```

每個命名空間使用不同的機密

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: anf-sc-smb
provisioner: csi.trident.netapp.io
parameters:
  backendType: "azure-netapp-files"
  trident.netapp.io/nasType: "smb"
  csi.storage.k8s.io/node-stage-secret-name: "smbcreds"
  csi.storage.k8s.io/node-stage-secret-namespace: ${pvc.namespace}
```

每個磁碟區使用不同的機密

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: anf-sc-smb
provisioner: csi.trident.netapp.io
parameters:
  backendType: "azure-netapp-files"
  trident.netapp.io/nasType: "smb"
  csi.storage.k8s.io/node-stage-secret-name: ${pvc.name}
  csi.storage.k8s.io/node-stage-secret-namespace: ${pvc.namespace}
```



nasType: smb 支援SMB磁碟區的集區篩選器。nasType: nfs 或 nasType: null NFS集區的篩選器。

建立後端

建立後端組態檔之後、請執行下列命令：

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

如果後端建立失敗、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
```

識別並修正組態檔的問題之後、您可以再次執行create命令。

設定Cloud Volumes Service 適用於Google Cloud後端的功能

瞭解Cloud Volumes Service 解如何使用提供的範例組態、將NetApp for Google Cloud設定為Astra Trident安裝的後端。

Google Cloud 驅動程式詳細資料

Astra Trident 提供 gcp-cvs 與叢集通訊的驅動程式。支援的存取模式包括：*ReadWriteOnce*（*rwo*）、*ReadOnlyMany*（*ROX*）、*_ReadWriteMany*（*rwx*）、*_ReadWriteOncePod*（*RWOP*）。

驅動程式	傳輸協定	Volume模式	支援的存取模式	支援的檔案系統
《GCP—CVS 》	NFS	檔案系統	Rwo、ROX、rwx、RWOP	nfs

深入瞭解Astra Trident對Cloud Volumes Service Google Cloud的支援

Astra Trident可在Cloud Volumes Service 兩個地方建立一個不二的資料區 "服務類型"：

- * CVS效能*：預設的Astra Trident服務類型。這種效能最佳化的服務類型最適合重視效能的正式作業工作負載。CVS效能服務類型是一種硬體選項、可支援最小100 GiB大小的磁碟區。您可以選擇其中一項 "三種服務層級"：
 - standard
 - premium
 - extreme
- * CVS：CVS服務類型提供高分區可用度、但效能等級僅限於中度。CVS服務類型是一種軟體選項、使用儲存資源池來支援小至1 GiB的磁碟區。儲存資源池最多可包含50個磁碟區、其中所有磁碟區都會共用資源池的容量和效能。您可以選擇其中一項 "兩種服務層級"：

- standardsw
- zoneredundantstandardsw

您需要的產品

以設定及使用 "適用於 Google Cloud Cloud Volumes Service" 後端、您需要下列項目：

- Google Cloud帳戶已設定NetApp Cloud Volumes Service 功能
- Google Cloud帳戶的專案編號
- Google Cloud服務帳戶的角色為「netappcloudvolumes.admin」
- API金鑰檔案、供Cloud Volumes Service 您的I方面 帳戶使用

後端組態選項

每個後端都會在單一Google Cloud區域中配置磁碟區。若要在其他區域建立磁碟區、您可以定義其他後端。

參數	說明	預設
「分度」		永遠為1
「storageDriverName」	儲存驅動程式名稱	「GCP-CVS」
「後端名稱」	自訂名稱或儲存後端	驅動程式名稱+「_」+ API 金鑰的一部分
「storageClass」	用於指定CVS服務類型的選用參數。使用 software 可選擇CVS服務類型。否則、Astra Trident會採用CVS 效能服務類型 (hardware) 。	
storagePools	僅限CVS服務類型。選用參數、用於指定用於建立磁碟區的儲存資源池。	
「ProjectNumber」	Google Cloud帳戶專案編號。此值可在Google Cloud 入口網站首頁找到。	
「hostProjectNumber」	如果使用共享VPC網路、則為必要項目。在此案例中、projectNumber 是服務專案、以及 hostProjectNumber 是主機專案。	
《apiRegion》	Astra Trident在Google Cloud區域建立Cloud Volumes Service 了各種不全的功能。建立跨區域Kubernetes叢集時、會在中建立磁碟區 apiRegion 可用於在多個Google Cloud區域的節點上排程的工作負載。跨區域流量會產生額外成本。	
「apiKey」	的Google Cloud服務帳戶API金鑰 netappcloudvolumes.admin 角色：其中包含Google Cloud服務帳戶私密金鑰檔案（逐字複製到後端組態檔）的JSON-格式內容。	
"proxyurl"	Proxy URL（如果需要Proxy伺服器才能連線至CVS帳戶）。Proxy伺服器可以是HTTP Proxy或HTTPS Proxy。對於HTTPS Proxy、會跳過憑證驗證、以允許在Proxy伺服器中使用自我簽署的憑證。不支援已啟用驗證的Proxy伺服器。	

參數	說明	預設
「nfsMountOptions」	精細控制NFS掛載選項。	"nfsves=3"
《限制Volume大小》	如果要求的磁碟區大小高於此值、則資源配置失敗。	""（預設不強制執行）
《服務層級》	適用於新磁碟區的CVS效能或CVS服務層級。CVS的效能值為 standard、premium、或 `extreme`。 CVS值包括 standardsw 或 zoneredundantstandardsw。	CVS效能預設為「標準」。CVS預設為「標準」。
網路	Google Cloud網路用於Cloud Volumes Service 解決資料不整的問題。	"預設"
「DebugTraceFlags」	疑難排解時要使用的偵錯旗標。範例： \{"api":false, "method":true}。除非您正在進行疑難排解並需要詳細的記錄傾印、否則請勿使用此功能。	null
allowedTopologies	若要啟用跨區域存取、您的StorageClass定義適用於 allowedTopologies 必須包含所有區域。例如： - key: topology.kubernetes.io/region values: - us-east1 - europe-west1	

Volume資源配置選項

您可以在中控制預設的Volume資源配置 defaults 組態檔的一節。

參數	說明	預設
「匯出規則」	新磁碟區的匯出規則。必須是以逗號分隔的清單、以CIDR表示法列出所有的IPv4位址或IPv4子網路組合。	「0.0.0.0/0」
「snapshotDir	存取「.snapshot」目錄	"假"
「快照保留區」	保留給快照的磁碟區百分比	""（接受CVS預設值為0）
《大小》	新磁碟區的大小。CVS效能最低為100 GiB。CVS最低為1 GiB。	CVS效能服務類型預設為「100GiB」。CVS服務類型並未設定預設值、但至少需要1 GiB。

CVS效能服務類型範例

下列範例提供CVS效能服務類型的範例組態。

[illegible]

```
token_uri: https://oauth2.googleapis.com/token
auth_provider_x509_cert_url:
https://www.googleapis.com/oauth2/v1/certs
client_x509_cert_url:
https://www.googleapis.com/robot/v1/metadata/x509/cloudvolumes-admin-
sa%40my-gcp-project.iam.gserviceaccount.com
```

範例2：服務層級組態

本範例說明後端組態選項、包括服務層級和Volume預設值。

```
--  
version: 1  
storageDriverName: gcp-cvs  
projectNumber: '012345678901'  
apiRegion: us-west2  
apiKey:  
  type: service_account  
  project_id: my-gcp-project  
  private_key_id: "<id_value>"  
  private_key: |  
    -----BEGIN PRIVATE KEY-----  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8 zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8 zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8 zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8 zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8 zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8 zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8 zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8 zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8 zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8 zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8 zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8 zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8 zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8 zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8 zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8 zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8 zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8 zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8 zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8 zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8 zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8 zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8 zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8 zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8 zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8 zX5ojY9m  
    XsYg6gyxy4zq7OlwWgLwGa==  
    -----END PRIVATE KEY-----  
  client_email: cloudvolumes-admin-sa@my-gcp-  
project.iam.gserviceaccount.com  
  client_id: '123456789012345678901'  
  auth uri: https://accounts.google.com/o/oauth2/auth
```

```
token_uri: https://oauth2.googleapis.com/token
auth_provider_x509_cert_url:
https://www.googleapis.com/oauth2/v1/certs
client_x509_cert_url:
https://www.googleapis.com/robot/v1/metadata/x509/cloudvolumes-admin-
sa%40my-gcp-project.iam.gserviceaccount.com
proxyURL: http://proxy-server-hostname/
nfsMountOptions: vers=3,proto=tcp,timeo=600
limitVolumeSize: 10Ti
serviceLevel: premium
defaults:
  snapshotDir: 'true'
  snapshotReserve: '5'
  exportRule: 10.0.0.0/24,10.0.1.0/24,10.0.2.100
  size: 5Ti
```


範例3：虛擬資源池組態

此範例使用 `storage` 來設定虛擬集區和 `StorageClasses` 請回頭參考。請參閱 [\[儲存類別定義\]](#) 以瞭解如何定義儲存類別。

此處會針對所有設定的虛擬資源池設定特定的預設值 `snapshotReserve 5%`和 `exportRule 至 0.00.0/0`。虛擬資源池是在中定義的 `storage` 區段。每個個別虛擬集區都會定義自己的虛擬集區 `serviceLevel``和某些資源池會覆寫預設值。虛擬資源池標籤是用來區分資源池的依據 ``performance`和 `protection`。

[illegible]

```

XsYg6gyxy4zq70lwWgLwGa==
-----END PRIVATE KEY-----
client_email: cloudvolumes-admin-sa@my-gcp-
project.iam.gserviceaccount.com
client_id: '123456789012345678901'
auth_uri: https://accounts.google.com/o/oauth2/auth
token_uri: https://oauth2.googleapis.com/token
auth_provider_x509_cert_url:
https://www.googleapis.com/oauth2/v1/certs
client_x509_cert_url:
https://www.googleapis.com/robot/v1/metadata/x509/cloudvolumes-admin-
sa%40my-gcp-project.iam.gserviceaccount.com
nfsMountOptions: vers=3,proto=tcp,timeo=600
defaults:
  snapshotReserve: '5'
  exportRule: 0.0.0.0/0
labels:
  cloud: gcp
region: us-west2
storage:
- labels:
  performance: extreme
  protection: extra
  serviceLevel: extreme
  defaults:
    snapshotDir: 'true'
    snapshotReserve: '10'
    exportRule: 10.0.0.0/24
- labels:
  performance: extreme
  protection: standard
  serviceLevel: extreme
- labels:
  performance: premium
  protection: extra
  serviceLevel: premium
  defaults:
    snapshotDir: 'true'
    snapshotReserve: '10'
- labels:
  performance: premium
  protection: standard
  serviceLevel: premium
- labels:
  performance: standard
  serviceLevel: standard

```

儲存類別定義

下列StorageClass定義適用於虛擬集區組態範例。使用 `parameters.selector`、您可以為每個StorageClass指定用於裝載磁碟區的虛擬集區。該磁碟區會在所選的資源池中定義各個層面。

```

apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: cvs-extreme-extra-protection
provisioner: csi.trident.netapp.io
parameters:
  selector: "performance=extreme; protection=extra"
allowVolumeExpansion: true
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: cvs-extreme-standard-protection
provisioner: csi.trident.netapp.io
parameters:
  selector: "performance=premium; protection=standard"
allowVolumeExpansion: true
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: cvs-premium-extra-protection
provisioner: csi.trident.netapp.io
parameters:
  selector: "performance=premium; protection=extra"
allowVolumeExpansion: true
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: cvs-premium
provisioner: csi.trident.netapp.io
parameters:
  selector: "performance=premium; protection=standard"
allowVolumeExpansion: true
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: cvs-standard
provisioner: csi.trident.netapp.io
parameters:
  selector: "performance=standard"

```

```
allowVolumeExpansion: true
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: cvs-extra-protection
provisioner: csi.trident.netapp.io
parameters:
  selector: "protection=extra"
allowVolumeExpansion: true
```

- 第一個StorageClass (cvs-extreme-extra-protection) 對應至第一個虛擬資源池。這是唯一提供極致效能、快照保留率為10%的資源池。
- 最後一個StorageClass (cvs-extra-protection) 撥出提供快照保留10%的任何儲存資源池。Astra Trident決定選取哪個虛擬集區、並確保符合快照保留需求。

CVS服務類型範例

下列範例提供CVS服務類型的範例組態。

範例1：最低組態

這是使用的最低後端組態 storageClass 指定CVS服務類型和預設值 standardsw 服務層級：

```
--  
version: 1  
storageDriverName: gcp-cvs  
projectNumber: '012345678901'  
storageClass: software  
apiRegion: us-east4  
apiKey:  
  type: service_account  
  project_id: my-gcp-project  
  private_key_id: "<id_value>"  
  private_key: |  
    -----BEGIN PRIVATE KEY-----  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8zX5ojY9m  
    znHczZsr rtHisIsAbOguSaPIKeyAZNchRAGz lzZE4jK3bl /qp8B4Kws8zX5ojY9m  
    XsYg6gyxy4zq7OlwWgLwGa==  
    -----END PRIVATE KEY-----  
  client_email: cloudvolumes-admin-sa@my-gcp-  
project.iam.gserviceaccount.com  
  client id: '123456789012345678901'
```

```
auth_uri: https://accounts.google.com/o/oauth2/auth
token_uri: https://oauth2.googleapis.com/token
auth_provider_x509_cert_url:
https://www.googleapis.com/oauth2/v1/certs
client_x509_cert_url:
https://www.googleapis.com/robot/v1/metadata/x509/cloudvolumes-admin-
sa%40my-gcp-project.iam.gserviceaccount.com
serviceLevel: standardsw
```

範例2：儲存資源池組態

此範例後端組態使用 `storagePools` 以設定儲存資源池。

```
---
version: 1
storageDriverName: gcp-cvs
backendName: gcp-std-so-with-pool
projectNumber: '531265380079'
apiRegion: europe-west1
apiKey:
  type: service_account
  project_id: cloud-native-data
  private_key_id: "<id_value>"
  private_key: |-
    -----BEGIN PRIVATE KEY-----
    MIIEvAIBADANBgkqhkiG9w0BAQEFAASCBKYwggSiAgEAAoIBAQDaT+Oui9FBAw19
    L1AGEkrYU5xd9K5NlO5jMkIFND5wCD+Nv+jd1GvtFRLaLK5RvXyF5wzvztmODNS+
    qtScpQ+5cFpQkuGtv9U9+N6qtuVYYO3b504Kp5CtqVPJCgMJaK2j8pZTIqUiMum/
    5/Y9oTbZrjAHSMgJm2nHzFq2X0rqVMaHghI6ATm4DOuWx8XGWKTGIPlc0qPqJlqS
    LLaWOH4VIZQZCAyW5IU9p9CamwqHgdG0uhFNfCgMmED6PBUvVLsLvcq86X+QSWR9k
    ETqElj/sGCenPF7ti1DhGBFafd9hPnxg9PZY29ArEZwY9G/ZjZQX7WPgs0VvxiNR
    DxZRC3GXAgMBAAECggEACn5c59bG/qnVEVI1CwMAalM5M2z09JFhlL1ljKwntNPj
    Vilw2eTW2+UE7HbJru/S7KQgA5Dnn9kvCraEahPRuddUMrD0vG4kTl/IODV6uFuk
    Y0sZfbqd4jMUQ21smvGsqFzwloYWS5qzO1W83ivXH/HW/iqkmY2eW+EPRS/hwSSu
    SscR+SojI7PB0BWSJhlV4yqYf3vcD/D95e12CVHfRCkL85DKumeZ+yHENpiXGZAE
    t8xSs4a50OPm6NHhevCw2a/UQ95/foXNUR450HtbjieJo5o+FF6EYZQGfU2ZHZO8
    37FBKuaJkdGW5xqaI9TL7aqkGkFMF4F2qvOZM+vy8QKBgQD4oVuOkJD1hkTHP86W
    esFlw1kpWyJR9ZA7LI0g/rVpslnX+XdDq0WQf4umDLNau5hYEH9LU6ZSGs1Xk3/B
    NHwR6OXFuqEKNiu83d0zSlHhTy7PZpOZdj5a/vVvQfPDMz7OvsqLRd7YCAbdzuQ0
    +Ahq0Ztwvg0HQ64hdW0ukpYRRwKBgQDgyHj98oqswoYuIa+pPlyS0pPwLmjwKyNm
    /HayzCp+Qjiyy7Tzg8AUqlH1Ou83XbV428jvg7kDhO7PCCKFq+mMmfqHmTpb0Maq
    KpKnZg4ipsqPlyHNNEoRmcailXbwIhCLewMqMrggUiLOmCw4PscL5nK+4GKu2XE1
    jLqjWAZFMQKBgFhkQ9XXRAJ1kR3XpGHOgN890pZOkCVSrqu6aUef/5KY1Fct8ew
    F/+aIxM2iQSVmWQYOvVCnhuY/F2GfAQ7d0om3decuwIOCX/xy7PjHMkLXa2uaZs4
    WR17sLduj62RqXRLX0c0QkwBiNFyHbRcpdkZJQujbYMhBa+7j7SxT4BtAoGAWMWT
    UucocRXZm/pdvz9wteNH3YDwnJLMxm1KC06qMXbBoYrliY4sm3ywJWMC+iCd/H8A
    Gecxd/xVu5mA2L2N3KMq18Zhz8Th0G5DwKyDRJgOQ0Q46yuNXOoYEjlo4Wjyk8Me
    +t1Q8iK98E0UmZnhTgfSpSNElbz2AqnzQ3MN9uECgYAqdvdpVnKGfvdT2ZDjyMoJ
    E89UIC41WjjJGmHsd8W65+3X0RwMzKMT6aZc5tK9J5dHvmWIETnbM+lTImdBbFga
    NWOC6f3r2xbGXHhaWSl+nobpTuvlo56ZRJVvVk7lFMsiddzMuHH8pxfgNJemwA4P
    ThDHCEjv035NNV6KyoO0tA==
    -----END PRIVATE KEY-----
  client_email: cloudvolumes-admin-sa@cloud-native-
  data.iam.gserviceaccount.com
  client_id: '107071413297115343396'
```



```
auth_uri: https://accounts.google.com/o/oauth2/auth
token_uri: https://oauth2.googleapis.com/token
auth_provider_x509_cert_url:
https://www.googleapis.com/oauth2/v1/certs
client_x509_cert_url:
https://www.googleapis.com/robot/v1/metadata/x509/cloudvolumes-admin-
sa%40cloud-native-data.iam.gserviceaccount.com
storageClass: software
zone: europe-west1-b
network: default
storagePools:
- 1bc7f380-3314-6005-45e9-c7dc8c2d7509
serviceLevel: Standardsw
```

接下來呢？

建立後端組態檔之後、請執行下列命令：

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

如果後端建立失敗、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
```

識別並修正組態檔的問題之後、您可以再次執行create命令。

設定NetApp HCI 一個不只是功能的SolidFire 後端

瞭解如何在 Astra Trident 安裝中建立和使用元素後端。

元素驅動程式詳細資料

Astra Trident 提供 `solidfire-san` 用於與叢集通訊的儲存驅動程式。支援的存取模式包括：`ReadWriteOnce`（`rwo`）、`ReadOnlyMany`（`ROX`）、`_ReadWriteMany`（`rwX`）、`_ReadWriteOncePod`（`RWOP`）。

。 `solidfire-san` 儲存驅動程式支援 `file` 和 `block` 磁碟區模式。適用於 `Filesystem` 磁碟區代碼、Astra Trident會建立磁碟區並建立檔案系統。檔案系統類型由StorageClass指定。

驅動程式	傳輸協定	Volume模式	支援的存取模式	支援的檔案系統
「olidfire - san」	iSCSI	區塊	Rwo 、 ROX 、 rwX 、 RWOP	無檔案系統。原始區 塊裝置。

驅動程式	傳輸協定	Volume模式	支援的存取模式	支援的檔案系統
「olidfire - san」	iSCSI	檔案系統	RWO 、 RWOP	《xfs》、《ext3》、 《ext4》

開始之前

在建立元素後端之前、您需要下列項目。

- 支援的儲存系統、可執行Element軟體。
- 提供給NetApp HCI / SolidFire叢集管理員或租戶使用者的認證、以管理磁碟區。
- 您所有的Kubernetes工作節點都應該安裝適當的iSCSI工具。請參閱 ["工作節點準備資訊"](#)。

後端組態選項

如需後端組態選項、請參閱下表：

參數	說明	預設
「分度」		永遠為1
「storageDriverName」	儲存驅動程式名稱	永遠是「solidfire-san」
「後端名稱」	自訂名稱或儲存後端	「S指_」+儲存設備（iSCSI）IP位址SolidFire
端點	MVIP、適用於SolidFire 採用租戶認證的不含用戶身分證明的叢集	
《VIP》	儲存設備（iSCSI）IP位址和連接埠	
《標籤》	套用到磁碟區的任意JSON-格式化標籤集。	「」
《天王名稱》	要使用的租戶名稱（如果找不到、請建立）	
《初始器IFACE》	將iSCSI流量限制在特定的主機介面	「預設」
《UseCHAP》	使用 CHAP 驗證 iSCSI 。Astra Trident 使用 CHAP 。	是的
《存取群組》	要使用的存取群組ID清單	尋找名為「Trident」的存取群組ID
《類型》	QoS規格	
《限制Volume大小》	如果要求的磁碟區大小高於此值、則資源配置失敗	「」（預設不強制執行）
「DebugTraceFlags」	疑難排解時要使用的偵錯旗標。範例： { "API"：假、「方法」：true }	null



除非您正在進行疑難排解並需要詳細的記錄傾印、否則請勿使用「debugTraceFlags」。

範例1：的後端組態 solidfire-san 三種磁碟區類型的驅動程式

此範例顯示使用CHAP驗證的後端檔案、並建立具有特定QoS保證的三種Volume類型模型。您很可能會使用「IOPS」儲存類別參數來定義儲存類別、以使用每個類別。

```
---
version: 1
storageDriverName: solidfire-san
Endpoint: https://<user>:<password>@<mvip>/json-rpc/8.0
SVIP: "<svip>:3260"
TenantName: "<tenant>"
labels:
  k8scluster: dev1
  backend: dev1-element-cluster
UseCHAP: true
Types:
- Type: Bronze
  Qos:
    minIOPS: 1000
    maxIOPS: 2000
    burstIOPS: 4000
- Type: Silver
  Qos:
    minIOPS: 4000
    maxIOPS: 6000
    burstIOPS: 8000
- Type: Gold
  Qos:
    minIOPS: 6000
    maxIOPS: 8000
    burstIOPS: 10000
```

範例2：的後端與儲存類別組態 solidfire-san 驅動程式與虛擬資源池

此範例顯示使用虛擬資源池設定的後端定義檔、以及參照這些資源池的StorageClass。

Astra Trident會在資源配置時、將儲存資源池上的標籤複製到後端儲存LUN。為了方便起見、儲存管理員可以針對每個虛擬資源池定義標籤、並依標籤將磁碟區分組。

在下圖所示的範例後端定義檔中、會針對所有設定的儲存資源池設定特定的預設值 type 銀級。虛擬資源池是在中定義的 storage 區段。在此範例中、有些儲存資源池會自行設定類型、有些資源池則會覆寫上述預設值。

```
---
version: 1
storageDriverName: solidfire-san
Endpoint: https://<user>:<password>@<mvip>/json-rpc/8.0
```

```

SVIP: "<svip>:3260"
TenantName: "<tenant>"
UseCHAP: true
Types:
- Type: Bronze
  Qos:
    minIOPS: 1000
    maxIOPS: 2000
    burstIOPS: 4000
- Type: Silver
  Qos:
    minIOPS: 4000
    maxIOPS: 6000
    burstIOPS: 8000
- Type: Gold
  Qos:
    minIOPS: 6000
    maxIOPS: 8000
    burstIOPS: 10000
type: Silver
labels:
  store: solidfire
  k8scluster: dev-1-cluster
region: us-east-1
storage:
- labels:
    performance: gold
    cost: '4'
  zone: us-east-1a
  type: Gold
- labels:
    performance: silver
    cost: '3'
  zone: us-east-1b
  type: Silver
- labels:
    performance: bronze
    cost: '2'
  zone: us-east-1c
  type: Bronze
- labels:
    performance: silver
    cost: '1'
  zone: us-east-1d

```

下列StorageClass定義是指上述虛擬資源池。使用 `parameters.selector` 欄位中、每個StorageClass會呼叫

哪些虛擬資源池可用於裝載Volume。磁碟區將會在所選的虛擬資源池中定義各個層面。

第一個StorageClass (solidfire-gold-four) 將對應至第一個虛擬資源池。這是唯一提供黃金級效能的資源池。Volume Type QoS 金級。最後一個StorageClass (solidfire-silver) 撥出任何提供銀級效能的儲存資源池。Astra Trident將決定選取哪個虛擬集區、並確保符合儲存需求。

```

apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: solidfire-gold-four
provisioner: csi.trident.netapp.io
parameters:
  selector: "performance=gold; cost=4"
  fsType: "ext4"
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: solidfire-silver-three
provisioner: csi.trident.netapp.io
parameters:
  selector: "performance=silver; cost=3"
  fsType: "ext4"
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: solidfire-bronze-two
provisioner: csi.trident.netapp.io
parameters:
  selector: "performance=bronze; cost=2"
  fsType: "ext4"
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: solidfire-silver-one
provisioner: csi.trident.netapp.io
parameters:
  selector: "performance=silver; cost=1"
  fsType: "ext4"
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: solidfire-silver
provisioner: csi.trident.netapp.io
parameters:
  selector: "performance=silver"
  fsType: "ext4"

```

如需詳細資訊、請參閱

- ["Volume存取群組"](#)


支援SAN驅動程式ONTAP

ONTAP SAN 驅動程式概觀

深入瞭解如何使用ONTAP 支援功能的功能和功能性SAN驅動程式來設定功能性的後端。ONTAP Cloud Volumes ONTAP

ONTAP SAN 驅動程式詳細資料

Astra Trident 提供下列 SAN 儲存驅動程式、可與 ONTAP 叢集通訊。支援的存取模式包括：*ReadWriteOnce*（*rwo*）、*ReadOnlyMany*（*ROX*）、*_ReadWriteMany*（*rw*x）、*_ReadWriteOncePod*（*RWOP*）。



如果您使用 Astra Control 來保護、恢復和移動、請閱讀 [Astra Control 驅動程式相容性](#)。

驅動程式	傳輸協定	Volume模式	支援的存取模式	支援的檔案系統
「ONTAP-SAN」	iSCSI	區塊	Rwo 、 ROX 、 rwx 、 RWOP	無檔案系統；原始區塊裝置
「ONTAP-SAN」	iSCSI	檔案系統	RWO 、 RWOP 檔案系統磁碟區模式中無法使用 Rox 和 rwx 。	《xfs》、《ext3》、《ext 4》
「ONTAP-SAN」	NVMe / TCP 請參閱 NVMe / TCP 的其他考量事項 。	區塊	Rwo 、 ROX 、 rwx 、 RWOP	無檔案系統；原始區塊裝置
「ONTAP-SAN」	NVMe / TCP 請參閱 NVMe / TCP 的其他考量事項 。	檔案系統	RWO 、 RWOP 檔案系統磁碟區模式中無法使用 Rox 和 rwx 。	《xfs》、《ext3》、《ext 4》
《ONTAP-san經濟》	iSCSI	區塊	Rwo 、 ROX 、 rwx 、 RWOP	無檔案系統；原始區塊裝置

驅動程式	傳輸協定	Volume 模式	支援的存取模式	支援的檔案系統
《ONTAP-san經濟》	iSCSI	檔案系統	RWO 、 RWOP 檔案系統磁碟區模式中無法使用 Rox 和 rwx 。	《xfs》、《ext3》、《ext4》

Astra Control 驅動程式相容性

Astra Control 可為使用建立的磁碟區提供無縫保護、災難恢復和移動性（在 Kubernetes 叢集之間移動磁碟區）
ontap-nas、ontap-nas-flexgroup 和 \ontap-san 驅動程式：請參閱 ["Astra Control 複寫先決條件"](#) 以取得詳細資料。



- 使用 ontap-san-economy 只有持續磁碟區使用量計數預期會高於 ["支援的 ONTAP Volume 限制"](#)。
- 使用 ontap-nas-economy 只有持續磁碟區使用量計數預期會高於 ["支援的 ONTAP Volume 限制"](#) 和 ontap-san-economy 無法使用驅動程式。
- 請勿使用 ontap-nas-economy 如果您預期需要資料保護、災難恢復或行動性、

使用者權限

Astra Trident 希望以 ONTAP 支援或 SVM 管理員的身分執行、通常是使用「admin」叢集使用者或「vsadmin」SVM 使用者、或使用具有相同角色的不同名稱使用者。對於 Amazon FSx for NetApp ONTAP 等部署、Astra Trident 預期會使用 ONTAP 叢集「fsxadmin」使用者或「vsadmin」SVM 使用者、或是具有相同角色之不同名稱的使用者、來執行成一個或多個 SVM 管理員。「fsxadmin」使用者是叢集管理使用者的有限替代。



如果您使用「limitAggregateusage」參數、則需要叢集管理權限。使用 Amazon FSx for NetApp ONTAP 支援 Astra Trident 時、「limitAggregateusage」參數無法搭配「vsadmin」和「fsxadmin」使用者帳戶使用。如果您指定此參數、組態作業將會失敗。

雖然可以在 ONTAP 中建立更具限制性的角色、讓 Trident 驅動程式可以使用、但我們不建議這樣做。Trident 的大多數新版本都會呼叫額外的 API、而這些 API 必須納入考量、使升級變得困難且容易出錯。

NVMe / TCP 的其他考量事項

Astra Trident 支援使用的非揮發性記憶體高速（NVMe）傳輸協定 ontap-san 驅動程式包括：

- IPv6
- NVMe 磁碟區的快照和複本
- 調整 NVMe 磁碟區大小
- 匯入 Astra Trident 外部建立的 NVMe Volume、以便 Astra Trident 管理其生命週期
- NVMe 原生多重路徑
- K8s 節點正常或不正常關機（24.02）

Astra Trident 不支援：

- NVMe 原生支援的 DH-HMAC-CHAP

- 裝置對應工具（DM）多重路徑
- LUKS 加密

準備使用ONTAP 支援的SAN驅動程式來設定後端

瞭解使用 ONTAP SAN 驅動程式設定 ONTAP 後端的需求和驗證選項。

需求

對於所有ONTAP 的不支援端點、Astra Trident至少需要指派一個集合體給SVM。

請記住、您也可以執行多個驅動程式、並建立指向一個或多個驅動程式的儲存類別。例如、您可以設定使用「ONTAP-SAN」驅動程式的「SAN開發」類別、以及使用「ONTAP-SAN經濟」類別的「SAN預設」類別。

您所有的Kubernetes工作節點都必須安裝適當的iSCSI工具。請參閱 ["準備工作節點"](#) 以取得詳細資料。

驗證 ONTAP 後端

Astra Trident提供兩種驗證ONTAP 證功能來驗證支援的後端。

- 認證型：ONTAP 對具備所需權限的使用者名稱和密碼。建議使用預先定義的安全登入角色、例如「admin」或「vsadmin」、以確保與ONTAP 各種版本的最大相容性。
- 憑證型：Astra Trident也能ONTAP 使用安裝在後端的憑證與某個叢集進行通訊。在此處、後端定義必須包含用戶端憑證、金鑰及信任的CA憑證（建議使用）的Base64編碼值。

您可以更新現有的後端、以便在認證型和憑證型方法之間移動。不過、一次只支援一種驗證方法。若要切換至不同的驗證方法、您必須從後端組態中移除現有方法。



如果您嘗試同時提供*認證與憑證*、後端建立將會失敗、並在組態檔中提供多種驗證方法。

啟用認證型驗證

Astra Trident需要SVM範圍/叢集範圍管理員的認證資料、才能與ONTAP 該後端進行通訊。建議使用預先定義的標準角色、例如「admin」或「vsadmin」。這可確保與未來ONTAP 的支援版本保持前瞻相容、因為未來的Astra Trident版本可能會使用功能API。您可以建立自訂的安全登入角色、並與Astra Trident搭配使用、但不建議使用。

後端定義範例如下所示：

YAML

```
---
version: 1
backendName: ExampleBackend
storageDriverName: ontap-san
managementLIF: 10.0.0.1
svm: svm_nfs
username: vsadmin
password: password
```

JSON

```
{
  "version": 1,
  "backendName": "ExampleBackend",
  "storageDriverName": "ontap-san",
  "managementLIF": "10.0.0.1",
  "svm": "svm_nfs",
  "username": "vsadmin",
  "password": "password"
}
```

請記住、後端定義是唯一以純文字儲存認證的位置。建立後端之後、使用者名稱/密碼會以Base64編碼、並儲存為Kubernetes機密。建立或更新後端是唯一需要具備認證知識的步驟。因此、這是一項純管理員操作、由Kubernetes /儲存管理員執行。

啟用憑證型驗證

新的和現有的後端可以使用憑證、並與ONTAP 該後端通訊。後端定義需要三個參數。

- 用戶端憑證：用戶端憑證的Base64編碼值。
- 用戶端私密金鑰：關聯私密金鑰的Base64編碼值。
- 信任的CACertificate：受信任CA憑證的Base64編碼值。如果使用信任的CA、則必須提供此參數。如果未使用信任的CA、則可忽略此問題。

典型的工作流程包括下列步驟。

步驟

1. 產生用戶端憑證和金鑰。產生時、請將Common Name (CN) (一般名稱 (CN)) 設定為ONTAP 驗證身分。

```
openssl req -x509 -nodes -days 1095 -newkey rsa:2048 -keyout k8senv.key
-out k8senv.pem -subj "/C=US/ST=NC/L=RTP/O=NetApp/CN=admin"
```

2. 將信任的CA憑證新增ONTAP 至整個叢集。這可能已由儲存管理員處理。如果未使用信任的CA、請忽略。

```
security certificate install -type server -cert-name <trusted-ca-cert-name> -vserver <vserver-name>
ssl modify -vserver <vserver-name> -server-enabled true -client-enabled true -common-name <common-name> -serial <SN-from-trusted-CA-cert> -ca <cert-authority>
```

3. 在ONTAP 支援叢集上安裝用戶端憑證和金鑰（步驟1）。

```
security certificate install -type client-ca -cert-name <certificate-name> -vserver <vserver-name>
security ssl modify -vserver <vserver-name> -client-enabled true
```

4. 確認ONTAP 支援「cert」驗證方法的支援功能。

```
security login create -user-or-group-name admin -application ontapi -authentication-method cert
security login create -user-or-group-name admin -application http -authentication-method cert
```

5. 使用產生的憑證測試驗證。以ONTAP Management LIF IP和SVM名稱取代<SfManagement LIF>和<vserver name>。

```
curl -X POST -Lk https://<ONTAP-Management-LIF>/servlets/netapp.servlets.admin.XMLrequest_filer --key k8senv.key --cert ~/k8senv.pem -d '<?xml version="1.0" encoding="UTF-8"?><netapp xmlns="http://www.netapp.com/filer/admin" version="1.21" vfiler="<vserver-name>"><vserver-get></vserver-get></netapp>'
```

6. 使用Base64編碼憑證、金鑰和信任的CA憑證。

```
base64 -w 0 k8senv.pem >> cert_base64
base64 -w 0 k8senv.key >> key_base64
base64 -w 0 trustedca.pem >> trustedca_base64
```

7. 使用從上一步取得的值建立後端。

```
cat cert-backend.json
{
  "version": 1,
  "storageDriverName": "ontap-san",
  "backendName": "SanBackend",
  "managementLIF": "1.2.3.4",
  "svm": "vserver_test",
  "clientCertificate": "Faaaakkkkeeee...Vaaalllluuuuueeee",
  "clientPrivateKey": "LS0tFaKE...0VaLuES0tLS0K",
  "trustedCACertificate": "QNFinfO...SiqOyN",
  "storagePrefix": "myPrefix_"
}

tridentctl create backend -f cert-backend.json -n trident
+-----+-----+-----+-----+
+-----+-----+
|      NAME      | STORAGE DRIVER |          UUID          |
STATE | VOLUMES |
+-----+-----+-----+-----+
+-----+-----+
| SanBackend | ontap-san      | 586b1cd5-8cf8-428d-a76c-2872713612c1 |
online |         0 |
+-----+-----+-----+-----+
+-----+-----+
```

更新驗證方法或旋轉認證資料

您可以更新現有的後端、以使用不同的驗證方法或旋轉其認證資料。這兩種方法都可行：使用使用者名稱/密碼的後端可更新以使用憑證；使用憑證的後端可更新為使用者名稱/密碼。若要這麼做、您必須移除現有的驗證方法、然後新增驗證方法。然後使用更新的backend.json檔案、其中包含執行「tridentctl後端更新」所需的參數。

```
cat cert-backend-updated.json
{
  "version": 1,
  "storageDriverName": "ontap-san",
  "backendName": "SanBackend",
  "managementLIF": "1.2.3.4",
  "svm": "vserver_test",
  "username": "vsadmin",
  "password": "password",
  "storagePrefix": "myPrefix_"
}

#Update backend with tridentctl
tridentctl update backend SanBackend -f cert-backend-updated.json -n
trident

+-----+-----+-----+
+-----+-----+
|      NAME      | STORAGE DRIVER |          UUID          |
STATE | VOLUMES |
+-----+-----+-----+
+-----+-----+
| SanBackend | ontap-san      | 586b1cd5-8cf8-428d-a76c-2872713612c1 |
online |          9 |
+-----+-----+-----+
+-----+-----+
```



當您旋轉密碼時、儲存管理員必須先更新ONTAP 使用者的密碼（位於BIOS）。接著是後端更新。在循環憑證時、可將多個憑證新增至使用者。然後更新後端以使用新的憑證、之後可從ONTAP 該叢集刪除舊的憑證。

更新後端不會中斷對已建立之磁碟區的存取、也不會影響之後建立的磁碟區連線。成功的後端更新顯示Astra Trident可以與ONTAP 該後端通訊、並處理未來的Volume作業。

使用雙向CHAP驗證連線

Astra Trident可以使用雙向CHAP驗證iSCSI工作階段 `ontap-san` 和 `ontap-san-economy` 驅動程式：這需要啟用 `useCHAP` 選項。設定為時 `true`，Astra Trident 將 SVM 的預設啟動器安全性設定為雙向 CHAP，並從後端檔案設定使用者名稱和密碼。NetApp建議使用雙向CHAP來驗證連線。請參閱下列組態範例：

```

---
version: 1
storageDriverName: ontap-san
backendName: ontap_san_chap
managementLIF: 192.168.0.135
svm: ontap_iscsi_svm
useCHAP: true
username: vsadmin
password: password
chapInitiatorSecret: cl9qxIm36DKyawxy
chapTargetInitiatorSecret: rqxigXgkesIpwxyz
chapTargetUsername: iJF4heBRT0TCwxyz
chapUsername: uh2aNCLSD6cNwxyz

```



「useCHAP」參數是布林選項、只能設定一次。預設值設為假。將其設為true之後、您就無法將其設為假。

除了"useCHAP=true"之外、"chapInitiator Secret (chapInitiator機密)"、"chapTargetInitiatorSecret (chapTargetInitiator機密)"、"chapTargetUsername (chapTargetUsername)"和"chapUsername (chapUsername)"在建立後端之後、可以執行「tridentctl update」來變更機密。

運作方式

儲存管理員將「useCHAP」設定為true、指示Astra Trident在儲存後端上設定CHAP。這包括下列項目：

- 在SVM上設定CHAP：
 - 如果 SVM 的預設啟動器安全性類型為無（預設為「無」） * 且 * 磁碟區中沒有預先存在的 LUN、Astra Trident 將預設安全性類型設為 CHAP 並繼續設定CHAP啟動器和目標使用者名稱和機密。
 - 如果SVM包含LUN、Astra Trident將不會在SVM上啟用CHAP。這可確保不限制對 SVM 上已存在的 LUN 的存取。
- 設定CHAP啟動器和目標使用者名稱和機密；這些選項必須在後端組態中指定（如上所示）。

建立後端之後、Astra Trident會建立對應的「tridentbackend」CRD、並將CHAP機密和使用者名稱儲存為Kubernetes機密。由Astra Trident在此後端上建立的所有PV、都會掛載並附加於CHAP上。

旋轉認證資料並更新後端

您可以更新「backend.json」檔案中的CHAP參數、以更新CHAP認證。這需要更新CHAP機密、並使用「tridentctl update」命令來反映這些變更。



更新後端的CHAP機密時、您必須使用「tridentctl」來更新後端。請勿透過CLI/ONTAP UI更新儲存叢集上的認證資料、因為Astra Trident無法接受這些變更。

```

cat backend-san.json
{
    "version": 1,
    "storageDriverName": "ontap-san",
    "backendName": "ontap_san_chap",
    "managementLIF": "192.168.0.135",
    "svm": "ontap_iscsi_svm",
    "useCHAP": true,
    "username": "vsadmin",
    "password": "password",
    "chapInitiatorSecret": "cl9qxUpDaTeD",
    "chapTargetInitiatorSecret": "rqxigXgkeUpDaTeD",
    "chapTargetUsername": "iJF4heBRT0TCwxyz",
    "chapUsername": "uh2aNCLSD6cNwxyz",
}

./tridentctl update backend ontap_san_chap -f backend-san.json -n trident
+-----+-----+-----+-----+
+-----+-----+
| NAME | STORAGE DRIVER | UUID |
STATE | VOLUMES |
+-----+-----+-----+-----+
+-----+-----+
| ontap_san_chap | ontap-san | aa458f3b-ad2d-4378-8a33-1a472ffbeeb5c |
online | 7 |
+-----+-----+-----+-----+
+-----+-----+

```

現有的連線不會受到影響；如果SVM上的Astra Trident更新認證、它們將繼續保持作用中狀態。新連線將使用更新的認證資料、而現有連線仍保持作用中狀態。中斷舊PV的連線並重新連線、將會使用更新的認證資料。

SAN組態選項與範例ONTAP

瞭解如何在 Astra Trident 安裝中建立及使用 ONTAP SAN 驅動程式。本節提供後端組態範例及將後端對應至 StorageClasses 的詳細資料。

後端組態選項

如需後端組態選項、請參閱下表：

參數	說明	預設
「分度」		永遠為1

參數	說明	預設
「storageDriverName」	儲存驅動程式名稱	ontap-nas、ontap-nas-economy、ontap-nas-flexgroup、ontap-san、ontap-san-economy
「後端名稱」	自訂名稱或儲存後端	驅動程式名稱 + "_" + dataLIF
《馬納格門達利》	<p>叢集或 SVM 管理 LIF 的 IP 位址。</p> <p>您可以指定完整網域名稱 (FQDN)。</p> <p>如果使用 IPv6 旗標安裝 Astra Trident、則可以設定為使用 IPv6 位址。IPv6 位址必須以方括弧定義、例如 [28e8:d9fb:a825:b7bf:69a8:d02f:9e7b:3555]。</p> <p>如需無縫 MetroCluster 之間的互通性、請參閱 MetroCluster 範例。</p>	「10.0.0.1」、"[2001:1234:abcd:::fefo]"
「DataLIF」	<p>傳輸協定LIF的IP位址。</p> <p>請勿指定iSCSI。Astra Trident的用途 "可選擇的LUN對應ONTAP" 探索建立多重路徑工作階段所需的iSCSI LIF。如果發生此情況、將會產生警告 dataLIF 已明確定義。</p> <p>* MetroCluster 請省略。* 請參閱 MetroCluster 範例。</p>	源自SVM
《虛擬機器》	<p>要使用的儲存虛擬機器</p> <p>* MetroCluster 請省略。* 請參閱 MetroCluster 範例。</p>	如果指定SVM "managementLIF"則衍生
《使用CHAP》	使用CHAP驗證iSCSI以供ONTAP 支援不支援的SAN驅動程式使用[布林值]。設定為 true 用於Astra Trident 設定及使用雙向CHAP做為後端SVM的預設驗證。請參閱 " 準備使用ONTAP 支援的SAN驅動程式來設定後端 " 以取得詳細資料。	「假」
《chapInitiator機密》	CHAP啟動器密碼。如果是"useCHAP=true"、則為必要項目	"
《標籤》	套用到磁碟區的任意JSON-格式化標籤集	"
《chapTargetInitiator機密》	CHAP目標啟動器機密。如果是"useCHAP=true"、則為必要項目	"
「chapUsername」	傳入使用者名稱。如果是"useCHAP=true"、則為必要項目	"
《chapTargetUsername》	目標使用者名稱。如果是"useCHAP=true"、則為必要項目	"
「用戶端憑證」	用戶端憑證的Base64編碼值。用於憑證型驗證	"
「clientPrivate Key」	用戶端私密金鑰的Base64編碼值。用於憑證型驗證	"

參數	說明	預設
「可信賴的CACertificate」	受信任CA憑證的Base64編碼值。選用。用於憑證型驗證。	"
《使用者名稱》	與ONTAP 該叢集通訊所需的使用者名稱。用於認證型驗證。	"
密碼	與ONTAP 該叢集通訊所需的密碼。用於認證型驗證。	"
《虛擬機器》	要使用的儲存虛擬機器	如果指定SVM "managementLIF"則衍生
「storagePrefix」	在SVM中配置新磁碟區時所使用的前置碼。稍後無法修改。若要更新此參數、您需要建立新的後端。	trident
「限制Aggregateusage」	如果使用率高於此百分比、則無法進行資源配置。如果您使用Amazon FSX for NetApp ONTAP Sendbackend、請勿指定 limitAggregateUsage。提供的 fsxadmin 和 vsadmin 請勿包含擷取Aggregate使用量所需的權限、並使用Astra Trident加以限制。	""（預設不強制執行）
《限制Volume大小》	如果要求的磁碟區大小高於此值、則資源配置失敗。也會限制其管理的qtree和LUN磁碟區大小上限。	""（預設不會強制執行）
《lunsPerFlexvol》	每FlexVol 個LUN的最大LUN數量、範圍必須在[50、200]	100
「DebugTraceFlags」	疑難排解時要使用的偵錯旗標。例如、 { "api" : false 、 "method" : true} 除非您正在進行疑難排解並需要詳細的記錄傾印、否則請勿使用。	null
《useREST》	使用ONTAP Isrest API的布林參數。技術預覽 useREST 以*技術預覽*的形式提供、建議用於測試環境、而非用於正式作業工作負載。設定為時 true 、Astra Trident將使用ONTAP 靜止API與後端進行通訊。此功能需要ONTAP 使用更新版本的版本。此外ONTAP 、所使用的登入角色必須能夠存取 ontap 應用程式：這是預先定義的 vsadmin 和 cluster-admin 角色： useREST 不支援MetroCluster 使用支援。 useREST 完全符合 NVMe / TCP 的資格。	「假」
sanType	用於選擇 iscsi 適用於 iSCSI 或 nvme 適用於 NVMe / TCP 。	iscsi 如果空白

用於資源配置磁碟區的後端組態選項

您可以使用中的這些選項來控制預設資源配置 defaults 組態區段。如需範例、請參閱下列組態範例。

參數	說明	預設
"paceAllocate (配置) "	LUN的空間分配	" 對 "
《保護區》	空間保留模式；「無」（精簡）或「Volume」（粗）	" 無 "
「快照原則」	要使用的Snapshot原則	" 無 "
「qosPolicy」	要指派給所建立磁碟區的QoS原則群組。選擇每個儲存集區/後端的其中一個qosPolicy或adaptiveQosPolicy。搭配Astra Trident使用QoS原則群組需要ONTAP 使用更新版本的版本。我們建議使用非共用的QoS原則群組、並確保原則群組會個別套用至每個組成群組。共享的QoS原則群組將強制所有工作負載的總處理量上限。	"
《adaptiveQosPolicy》	要指派給所建立磁碟區的調適性QoS原則群組。選擇每個儲存集區/後端的其中一個qosPolicy或adaptiveQosPolicy	"
「快照保留區」	保留給快照的磁碟區百分比	「0」如果 snapshotPolicy 為「無」、否則為「」
「PlitOnClone」	建立複本時、從其父複本分割複本	"假"
加密	在新磁碟區上啟用NetApp Volume Encryption (NVE) ；預設為「假」。必須在叢集上授權並啟用NVE、才能使用此選項。如果在後端啟用NAE、則Astra Trident中配置的任何磁碟區都會啟用NAE。如需詳細資訊、請參閱： "Astra Trident如何與NVE和NAE搭配運作" 。	"假"
luksEncryption	啟用LUKS加密。請參閱 "使用Linux統一金鑰設定 (LUKS) " 。 NVMe / TCP 不支援 LUKS 加密。	"
《生態樣式》	新磁碟區的安全樣式	unix
「分層政策」	分層原則以使用「無」	「僅限快照」適用於 ONTAP 9.5 之前的 SVM-DR 組態

Volume資源配置範例

以下是定義預設值的範例：

```

---
version: 1
storageDriverName: ontap-san
managementLIF: 10.0.0.1
svm: trident_svm
username: admin
password: <password>
labels:
  k8scluster: dev2
  backend: dev2-sanbackend
storagePrefix: alternate-trident
debugTraceFlags:
  api: false
  method: true
defaults:
  spaceReserve: volume
  qosPolicy: standard
  spaceAllocation: 'false'
  snapshotPolicy: default
  snapshotReserve: '10'

```



對於使用「ONTAP-san」驅動程式建立的所有磁碟區、Astra Trident會在FlexVol the支援LUN中繼資料的情況下、為該系統的功能增加10%的容量。LUN的配置大小與使用者在PVC中要求的大小完全相同。Astra Trident在FlexVol 整個過程中增加10%的速度（顯示ONTAP 在畫面上可用的尺寸）。使用者現在可以取得所要求的可用容量。此變更也可防止LUN成為唯讀、除非可用空間已充分利用。這不適用於ONTAP-san經濟型。

對於定義「快照保留區」的後端、Astra Trident會計算磁碟區大小、如下所示：

```

Total volume size = [(PVC requested size) / (1 - (snapshotReserve
percentage) / 100)] * 1.1

```

1.1是額外10%的Astra Trident加入FlexVol 到the支援LUN中繼資料的功能。對於「快照保留區」= 5%、而PVC要求= 5GiB、磁碟區總大小為5.79GiB、可用大小為5.5GiB。「volume show (Volume show) 命令應顯示類似以下範例的結果：

Vserver	Volume	Aggregate	State	Type	Size	Available	Used%
		_pvc_89f1c156_3801_4de4_9f9d_034d54c395f4	online	RW	10GB	5.00GB	0%
		_pvc_e42ec6fe_3baa_4af6_996d_134adbbb8e6d	online	RW	5.79GB	5.50GB	0%
		_pvc_e8372153_9ad9_474a_951a_08ae15e1c0ba	online	RW	1GB	511.8MB	0%

3 entries were displayed.

目前、只有調整大小、才能將新計算用於現有的Volume。

最低組態範例

下列範例顯示基本組態、讓大部分參數保留預設值。這是定義後端最簡單的方法。



如果您在 NetApp ONTAP 上搭配 Astra Trident 使用 Amazon FSX 、建議您指定生命的 DNS 名稱、而非 IP 位址。

ONTAP SAN 範例

這是使用的基本組態 `ontap-san` 驅動程式：

```
---
version: 1
storageDriverName: ontap-san
managementLIF: 10.0.0.1
svm: svm_iscsi
labels:
  k8scluster: test-cluster-1
  backend: testcluster1-sanbackend
username: vsadmin
password: <password>
```

ONTAP SAN 經濟效益範例

```
---
version: 1
storageDriverName: ontap-san-economy
managementLIF: 10.0.0.1
svm: svm_iscsi_eco
username: vsadmin
password: <password>
```

MetroCluster 範例

您可以設定後端、避免在切換和切換期間手動更新後端定義 "SVM 複寫與還原"。

若要無縫切換和切換、請使用指定 SVM managementLIF 並省略 dataLIF 和 svm 參數。例如：

```
---
version: 1
storageDriverName: ontap-san
managementLIF: 192.168.1.66
username: vsadmin
password: password
```

憑證型驗證範例

在此基本組態範例中 clientCertificate、clientPrivateKey 和 trustedCACertificate (選用、如果使用信任的CA) 會填入 backend.json 並分別取得用戶端憑證、私密金鑰及信任CA憑證的基礎64編碼值。

```
---
version: 1
storageDriverName: ontap-san
backendName: DefaultSANBackend
managementLIF: 10.0.0.1
svm: svm_iscsi
useCHAP: true
chapInitiatorSecret: cl9qxIm36DKyawxy
chapTargetInitiatorSecret: rqxigXgkesIpwxyz
chapTargetUsername: iJF4heBRT0TCwxyz
chapUsername: uh2aNCLSD6cNwxyz
clientCertificate: ZXR0ZXJwYXB...ICMgJ3BhcGVyc2
clientPrivateKey: vciwKIyAgZG...0cnksIGRlc2NyaX
trustedCACertificate: zcyBbaG...b3Igb3duIGNsYXNz
```

雙向 CHAP 範例

這些範例使用建立後端 `useCHAP` 設定為 `true`。

ONTAP SAN CHAP 範例

```
---
version: 1
storageDriverName: ontap-san
managementLIF: 10.0.0.1
svm: svm_iscsi
labels:
  k8scluster: test-cluster-1
  backend: testcluster1-sanbackend
useCHAP: true
chapInitiatorSecret: cl9qxIm36DKyawxy
chapTargetInitiatorSecret: rqxigXgkesIpwxyz
chapTargetUsername: iJF4heBRT0TCwxyz
chapUsername: uh2aNCLSD6cNwxyz
username: vsadmin
password: <password>
```

ONTAP SAN 經濟 CHAP 範例

```
---
version: 1
storageDriverName: ontap-san-economy
managementLIF: 10.0.0.1
svm: svm_iscsi_eco
useCHAP: true
chapInitiatorSecret: cl9qxIm36DKyawxy
chapTargetInitiatorSecret: rqxigXgkesIpwxyz
chapTargetUsername: iJF4heBRT0TCwxyz
chapUsername: uh2aNCLSD6cNwxyz
username: vsadmin
password: <password>
```

NVMe / TCP 範例

您必須在 ONTAP 後端上設定 NVMe 的 SVM 。這是適用於 NVMe / TCP 的基本後端組態。

```
---
version: 1
backendName: NVMeBackend
storageDriverName: ontap-san
managementLIF: 10.0.0.1
svm: svm_nvme
username: vsadmin
password: password
sanType: nvme
useREST: true
```

虛擬集區的後端範例

在這些後端定義檔案範例中、會針對所有儲存池設定特定的預設值、例如 `spaceReserve` 無、`spaceAllocation` 假、和 `encryption` 錯。虛擬資源池是在儲存區段中定義的。

Astra Trident 會在「意見」欄位中設定資源配置標籤。請在 FlexVol The 過程中提出意見。Astra Trident 會在資源配置時、將虛擬資源池上的所有標籤複製到儲存磁碟區。為了方便起見、儲存管理員可以針對每個虛擬資源池定義標籤、並依標籤將磁碟區分組。

在這些範例中、有些儲存池是自行設定的 `spaceReserve`、`spaceAllocation` 和 `encryption` 值、而某些資源池會覆寫預設值。




```

---
version: 1
storageDriverName: ontap-san
managementLIF: 10.0.0.1
svm: svm_iscsi
useCHAP: true
chapInitiatorSecret: cl9qxIm36DKyawxy
chapTargetInitiatorSecret: rqxigXgkesIpwxyz
chapTargetUsername: iJF4heBRT0TCwxyz
chapUsername: uh2aNCLSD6cNwxyz
username: vsadmin
password: <password>
defaults:
  spaceAllocation: 'false'
  encryption: 'false'
  qosPolicy: standard
labels:
  store: san_store
  kubernetes-cluster: prod-cluster-1
region: us_east_1
storage:
- labels:
  protection: gold
  creditpoints: '40000'
  zone: us_east_1a
  defaults:
    spaceAllocation: 'true'
    encryption: 'true'
    adaptiveQosPolicy: adaptive-extreme
- labels:
  protection: silver
  creditpoints: '20000'
  zone: us_east_1b
  defaults:
    spaceAllocation: 'false'
    encryption: 'true'
    qosPolicy: premium
- labels:
  protection: bronze
  creditpoints: '5000'
  zone: us_east_1c
  defaults:
    spaceAllocation: 'true'
    encryption: 'false'

```

```

---
version: 1
storageDriverName: ontap-san-economy
managementLIF: 10.0.0.1
svm: svm_iscsi_eco
useCHAP: true
chapInitiatorSecret: cl9qxIm36DKyawxy
chapTargetInitiatorSecret: rqxigXgkesIpwxyz
chapTargetUsername: iJF4heBRT0TCwxyz
chapUsername: uh2aNCLSD6cNwxyz
username: vsadmin
password: <password>
defaults:
  spaceAllocation: 'false'
  encryption: 'false'
labels:
  store: san_economy_store
region: us_east_1
storage:
- labels:
  app: oracledb
  cost: '30'
  zone: us_east_1a
  defaults:
    spaceAllocation: 'true'
    encryption: 'true'
- labels:
  app: postgresdb
  cost: '20'
  zone: us_east_1b
  defaults:
    spaceAllocation: 'false'
    encryption: 'true'
- labels:
  app: mysqldb
  cost: '10'
  zone: us_east_1c
  defaults:
    spaceAllocation: 'true'
    encryption: 'false'
- labels:
  department: legal
  creditpoints: '5000'

```

```
zone: us_east_1c
defaults:
  spaceAllocation: 'true'
  encryption: 'false'
```

NVMe / TCP 範例

```
---
version: 1
storageDriverName: ontap-san
sanType: nvme
managementLIF: 10.0.0.1
svm: nvme_svm
username: vsadmin
password: <password>
useREST: true
defaults:
  spaceAllocation: 'false'
  encryption: 'true'
storage:
- labels:
  app: testApp
  cost: '20'
  defaults:
    spaceAllocation: 'false'
    encryption: 'false'
```

將後端對應至StorageClass

下列 StorageClass 定義請參閱 [\[虛擬集區的後端範例\]](#)。使用 `parameters.selector` 欄位中、每個 StorageClass 都會呼叫哪些虛擬集區可用於主控磁碟區。磁碟區將會在所選的虛擬資源池中定義各個層面。

- `protection-gold` StorageClass 會對應至中的第一個虛擬集區 `ontap-san` 後端：這是唯一提供金級保護的集區。

```

apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: protection-gold
provisioner: csi.trident.netapp.io
parameters:
  selector: "protection=gold"
  fsType: "ext4"

```

- ◦ protection-not-gold StorageClass 會對應至中的第二個和第三個虛擬集區 ontap-san 後端：這是唯一提供金級以外保護層級的集區。

```

apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: protection-not-gold
provisioner: csi.trident.netapp.io
parameters:
  selector: "protection!=gold"
  fsType: "ext4"

```

- ◦ app-mysqldb StorageClass 會對應至中的第三個虛擬集區 ontap-san-economy 後端：這是唯一為 mysqldb 類型應用程式提供儲存池組態的集區。

```

apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: app-mysqldb
provisioner: csi.trident.netapp.io
parameters:
  selector: "app=mysqldb"
  fsType: "ext4"

```

- ◦ protection-silver-creditpoints-20k StorageClass 會對應至中的第二個虛擬集區 ontap-san 後端：這是唯一提供銀級保護和 20000 個信用點數的資源池。

```

apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: protection-silver-creditpoints-20k
provisioner: csi.trident.netapp.io
parameters:
  selector: "protection=silver; creditpoints=20000"
  fsType: "ext4"

```

- creditpoints-5k StorageClass 會對應至中的第三個虛擬集區 ontap-san 中的後端和第四個虛擬集區 ontap-san-economy 後端：這是唯一擁有 5000 個信用點數的集區方案。

```

apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: creditpoints-5k
provisioner: csi.trident.netapp.io
parameters:
  selector: "creditpoints=5000"
  fsType: "ext4"

```

- my-test-app-sc StorageClass 會對應至 testAPP 中的虛擬集區 ontap-san 驅動程式搭配 sanType: nvme。這是唯一的集區服務項目 testApp。

```

---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: my-test-app-sc
provisioner: csi.trident.netapp.io
parameters:
  selector: "app=testApp"
  fsType: "ext4"

```

Astra Trident將決定選取哪個虛擬集區、並確保符合儲存需求。

ASNAS驅動程式ONTAP

ONTAP NAS 驅動程式概述

深入瞭解如何使用ONTAP 功能性和功能性NAS驅動程式來設定功能性的後端。ONTAP Cloud Volumes ONTAP

ONTAP NAS 驅動程式詳細資料

Astra Trident 提供下列 NAS 儲存驅動程式、可與 ONTAP 叢集通訊。支援的存取模式包括：*ReadWriteOnce*（*rwo*）、*ReadOnlyMany*（*ROX*）、*_ReadWriteMany*（*rwX*）、*_ReadWriteOncePod*（*RWOP*）。



如果您使用 Astra Control 來保護、恢復和移動、請閱讀 [Astra Control 驅動程式相容性](#)。

驅動程式	傳輸協定	Volume模式	支援的存取模式	支援的檔案系統
「ONTAP-NAS」	NFS 中小企業	檔案系統	Rwo 、 ROX 、 rwX 、 RWOP	" 、 nfs 、 smb
《ONTAP-NANAS經濟》	NFS 中小企業	檔案系統	Rwo 、 ROX 、 rwX 、 RWOP	" 、 nfs 、 smb
「ONTAP-NAA- flexgroup」	NFS 中小企業	檔案系統	Rwo 、 ROX 、 rwX 、 RWOP	" 、 nfs 、 smb

Astra Control 驅動程式相容性

Astra Control可為使用建立的磁碟區提供無縫保護、災難恢復和移動性（在Kubernetes叢集之間移動磁碟區）`ontap-nas`、`ontap-nas-flexgroup`和 `ontap-san` 驅動程式：請參閱 ["Astra Control複寫先決條件"](#) 以取得詳細資料。



- 使用 `ontap-san-economy` 只有持續磁碟區使用量計數預期會高於 ["支援的 ONTAP Volume 限制"](#)。
- 使用 `ontap-nas-economy` 只有持續磁碟區使用量計數預期會高於 ["支援的 ONTAP Volume 限制"](#) 和 `ontap-san-economy` 無法使用驅動程式。
- 請勿使用 `ontap-nas-economy` 如果您預期需要資料保護、災難恢復或行動性、

使用者權限

Astra Trident希望以ONTAP 支援的形式執行、通常是以支援的方式執行 `admin` 叢集使用者或 `vsadmin` SVM使用者、或具有相同角色之不同名稱的使用者。

對於Amazon FSX for NetApp ONTAP 支援的NetApp功能、Astra Trident預期會以ONTAP 使用叢集的形式執行、以執行支援或SVM管理員的身分 `fsxadmin` 使用者或 `vsadmin` SVM使用者、或具有相同角色之不同名稱的使用者。。 `fsxadmin` 使用者是叢集管理使用者的有限替代。



如果您使用「`limitAggregateusage`」參數、則需要叢集管理權限。使用Amazon FSx for NetApp ONTAP 支援Astra Trident時、「`limitAggregateusage`」參數無法搭配「`vsadmin`」和「`fsxadmin`」使用者帳戶使用。如果您指定此參數、組態作業將會失敗。

雖然可以在 ONTAP 中建立更具限制性的角色、讓 Trident 驅動程式可以使用、但我們不建議這樣做。Trident的大多數新版本都會呼叫額外的API、而這些API必須納入考量、使升級變得困難且容易出錯。

準備使用ONTAP 不含NAS的驅動程式來設定後端

瞭解使用 ONTAP NAS 驅動程式設定 ONTAP 後端的需求、驗證選項和匯出原則。

需求

- 對於所有ONTAP 的不支援端點、Astra Trident至少需要指派一個集合體給SVM。
- 您可以執行多個驅動程式、並建立指向其中一個或另一個的儲存類別。例如、您可以設定使用的Gold類別 `ontap-nas` 驅動程式和銅級、使用 `ontap-nas-economy` 。
- 您所有的Kubernetes工作節點都必須安裝適當的NFS工具。請參閱 ["請按這裡"](#) 以取得更多詳細資料。
- Astra Trident僅支援安裝在Windows節點上執行的Pod上的SMB磁碟區。請參閱 [準備配置SMB磁碟區](#) 以取得詳細資料。

驗證 ONTAP 後端

Astra Trident提供兩種驗證ONTAP 證功能來驗證支援的後端。

- 認證型：此模式需要對 ONTAP 後端擁有足夠的權限。建議您使用與預先定義的安全登入角色相關聯的帳戶、例如 `admin` 或 `vsadmin` 以確保與ONTAP 更新版本的最大相容性。
- 憑證型：此模式需要在後端安裝憑證、Astra Trident 才能與 ONTAP 叢集通訊。在此處、後端定義必須包含用戶端憑證、金鑰及信任的CA憑證（建議使用）的Base64編碼值。

您可以更新現有的後端、以便在認證型和憑證型方法之間移動。不過、一次只支援一種驗證方法。若要切換至不同的驗證方法、您必須從後端組態中移除現有方法。



如果您嘗試同時提供*認證與憑證*、後端建立將會失敗、並在組態檔中提供多種驗證方法。

啟用認證型驗證

Astra Trident需要SVM範圍/叢集範圍管理員的認證資料、才能與ONTAP 該後端進行通訊。建議使用預先定義的標準角色、例如「`admin`」或「`vsadmin`」。這可確保與未來ONTAP 的支援版本保持前瞻相容、因為未來的Astra Trident版本可能會使用功能API。您可以建立自訂的安全登入角色、並與Astra Trident搭配使用、但不建議使用。

後端定義範例如下所示：

YAML

```
---
version: 1
backendName: ExampleBackend
storageDriverName: ontap-nas
managementLIF: 10.0.0.1
dataLIF: 10.0.0.2
svm: svm_nfs
username: vsadmin
password: password
```

JSON

```
{
  "version": 1,
  "backendName": "ExampleBackend",
  "storageDriverName": "ontap-nas",
  "managementLIF": "10.0.0.1",
  "dataLIF": "10.0.0.2",
  "svm": "svm_nfs",
  "username": "vsadmin",
  "password": "password"
}
```

請記住、後端定義是唯一以純文字儲存認證的位置。建立後端之後、使用者名稱/密碼會以Base64編碼、並儲存為Kubernetes機密。建立/更新後端是唯一需要知道認證資料的步驟。因此、這是一項純管理員操作、由Kubernetes /儲存管理員執行。

啟用憑證型驗證

新的和現有的後端可以使用憑證、並與ONTAP 該後端通訊。後端定義需要三個參數。

- 用戶端憑證：用戶端憑證的Base64編碼值。
- 用戶端私密金鑰：關聯私密金鑰的Base64編碼值。
- 信任的CACertificate：受信任CA憑證的Base64編碼值。如果使用信任的CA、則必須提供此參數。如果未使用信任的CA、則可忽略此問題。

典型的工作流程包括下列步驟。

步驟

1. 產生用戶端憑證和金鑰。產生時、請將Common Name (CN) (一般名稱 (CN)) 設定為ONTAP 驗證身分。


```
openssl req -x509 -nodes -days 1095 -newkey rsa:2048 -keyout k8senv.key  
-out k8senv.pem -subj "/C=US/ST=NC/L=RTP/O=NetApp/CN=vsadmin"
```

2. 將信任的CA憑證新增ONTAP 至整個叢集。這可能已由儲存管理員處理。如果未使用信任的CA、請忽略。

```
security certificate install -type server -cert-name <trusted-ca-cert-  
name> -vserver <vserver-name>  
ssl modify -vserver <vserver-name> -server-enabled true -client-enabled  
true -common-name <common-name> -serial <SN-from-trusted-CA-cert> -ca  
<cert-authority>
```

3. 在ONTAP 支援叢集上安裝用戶端憑證和金鑰（步驟1）。

```
security certificate install -type client-ca -cert-name <certificate-  
name> -vserver <vserver-name>  
security ssl modify -vserver <vserver-name> -client-enabled true
```

4. 確認ONTAP 支援「cert」驗證方法的支援功能。

```
security login create -user-or-group-name vsadmin -application ontapi  
-authentication-method cert -vserver <vserver-name>  
security login create -user-or-group-name vsadmin -application http  
-authentication-method cert -vserver <vserver-name>
```

5. 使用產生的憑證測試驗證。以ONTAP Management LIF IP和SVM名稱取代<SfManagement LIF>和<vserver name>。您必須確保LIF的服務原則設定為「預設資料管理」。

```
curl -X POST -Lk https://<ONTAP-Management-  
LIF>/servlets/netapp.servlets.admin.XMLrequest_filer --key k8senv.key  
--cert ~/k8senv.pem -d '<?xml version="1.0" encoding="UTF-8"?><netapp  
xmlns="http://www.netapp.com/filer/admin" version="1.21"  
vfiler="<vserver-name>"><vserver-get></vserver-get></netapp>'
```

6. 使用Base64編碼憑證、金鑰和信任的CA憑證。

```
base64 -w 0 k8senv.pem >> cert_base64  
base64 -w 0 k8senv.key >> key_base64  
base64 -w 0 trustedca.pem >> trustedca_base64
```

7. 使用從上一步取得的值建立後端。

```

cat cert-backend-updated.json
{
  "version": 1,
  "storageDriverName": "ontap-nas",
  "backendName": "NasBackend",
  "managementLIF": "1.2.3.4",
  "dataLIF": "1.2.3.8",
  "svm": "vserver_test",
  "clientCertificate": "Faaaakkkkeeee...Vaaalllluuuuueeee",
  "clientPrivateKey": "LS0tFaKE...0VaLuES0tLS0K",
  "storagePrefix": "myPrefix_"
}

#Update backend with tridentctl
tridentctl update backend NasBackend -f cert-backend-updated.json -n
trident

+-----+-----+-----+-----+
+-----+-----+
|      NAME      | STORAGE DRIVER |          UUID          |
STATE | VOLUMES |
+-----+-----+-----+-----+
+-----+-----+
| NasBackend | ontap-nas      | 98e19b74-aec7-4a3d-8dcf-128e5033b214 |
online |          9 |
+-----+-----+-----+-----+
+-----+-----+

```

更新驗證方法或旋轉認證資料

您可以更新現有的後端、以使用不同的驗證方法或旋轉其認證資料。這兩種方法都可行：使用使用者名稱/密碼的後端可更新以使用憑證；使用憑證的後端可更新為使用者名稱/密碼。若要這麼做、您必須移除現有的驗證方法、然後新增驗證方法。然後使用更新的backend.json檔案、其中包含要執行的必要參數 `tridentctl update backend`。

```
cat cert-backend-updated.json
{
  "version": 1,
  "storageDriverName": "ontap-nas",
  "backendName": "NasBackend",
  "managementLIF": "1.2.3.4",
  "dataLIF": "1.2.3.8",
  "svm": "vserver_test",
  "username": "vsadmin",
  "password": "password",
  "storagePrefix": "myPrefix_"
}

#Update backend with tridentctl
tridentctl update backend NasBackend -f cert-backend-updated.json -n
trident

+-----+-----+-----+
+-----+-----+
|      NAME      | STORAGE DRIVER |          UUID          |
STATE | VOLUMES |
+-----+-----+-----+
+-----+-----+
| NasBackend | ontap-nas      | 98e19b74-aec7-4a3d-8dcf-128e5033b214 |
online |          9 |
+-----+-----+-----+
+-----+-----+
```



當您旋轉密碼時、儲存管理員必須先更新ONTAP 使用者的密碼（位於BIOS）。接著是後端更新。在循環憑證時、可將多個憑證新增至使用者。然後更新後端以使用新的憑證、之後可從ONTAP 該叢集刪除舊的憑證。

更新後端不會中斷對已建立之磁碟區的存取、也不會影響之後建立的磁碟區連線。成功的後端更新顯示Astra Trident可以與ONTAP 該後端通訊、並處理未來的Volume作業。

管理NFS匯出原則

Astra Trident使用NFS匯出原則來控制其所配置之磁碟區的存取。

使用匯出原則時、Astra Trident提供兩種選項：

- Astra Trident可動態管理匯出原則本身；在此作業模式中、儲存管理員會指定代表可接受IP位址的CIDR區塊清單。Astra Trident會自動將這些範圍內的節點IP新增至匯出原則。或者、如果未指定CIDR、則會將節點上找到的任何全域範圍單點傳送IP新增至匯出原則。
- 儲存管理員可以建立匯出原則、並手動新增規則。除非在組態中指定不同的匯出原則名稱、否則Astra Trident會使用預設的匯出原則。

Astra Trident 提供動態管理 ONTAP 後端匯出原則的能力。這可讓儲存管理員為工作節點IP指定允許的位址空間、而非手動定義明確的規則。它可大幅簡化匯出原則管理；修改匯出原則不再需要在儲存叢集上進行手動介入。此外、這有助於限制只有在指定範圍內有IP的工作者節點才能存取儲存叢集、以支援精細且自動化的管理。



使用動態匯出原則時、請勿使用網路位址轉譯（NAT）。使用 NAT 時、儲存控制器會看到前端 NAT 位址、而非實際 IP 主機位址、因此在匯出規則中找不到相符項目時、就會拒絕存取。

範例

必須使用兩種組態選項。以下是後端定義範例：

```
---
version: 1
storageDriverName: ontap-nas
backendName: ontap_nas_auto_export
managementLIF: 192.168.0.135
svm: svm1
username: vsadmin
password: password
autoExportCIDRs:
- 192.168.0.0/24
autoExportPolicy: true
```



使用此功能時、您必須確保SVM中的根連接點具有先前建立的匯出原則、並具有允許節點CIDR區塊（例如預設匯出原則）的匯出規則。請務必遵循 NetApp 建議的最佳實務做法、將 SVM 專門用於 Astra Trident。

以下是使用上述範例說明此功能的運作方式：

- 將「AutoExportPolicy」設為「true」。這表示Astra Trident將為「svm1」SVM建立匯出原則、並使用「AutoExportCIDR」位址區塊來處理新增和刪除規則的作業。例如、在SVM上、UUID為403b5326-8482-40dB/96d0-d83fb3f4daec和「autodportPolicy」設為「true」的後端會建立名為「trident 403b5326-8482-40dB/96d0-d83fb3f4daec」的匯出原則。
- 「AutoExportCIDR」包含位址區塊清單。此欄位為選用欄位、預設為「0.00.0.0/0」、「：/0」。如果未定義、Astra Trident會新增在工作者節點上找到的所有全域範圍單點傳送位址。

在此範例中、會提供「192 · 168 · 0 · 0/24」位址空間。這表示、屬於此位址範圍的Kubernetes節點IP將新增至Astra Trident所建立的匯出原則。當Astra Trident登錄其執行的節點時、會擷取節點的IP位址、並對照「AutoExportCIDR」中提供的位址區塊進行檢查。篩選IP之後、Astra Trident會針對所探索的用戶端IP建立匯出原則規則、並針對所識別的每個節點建立一個規則。

您可以在建立後端後、更新「AutoExportPolicy」和「AutoExportCTR」。您可以為自動管理或刪除現有CIDR的後端附加新的CIDR。刪除CIDR時請務必謹慎、以確保不會中斷現有的連線。您也可以選擇停用後端的「autodportPolicy」、然後回到手動建立的匯出原則。這需要在後端組態中設定「exportPolicy」參數。

Astra Trident建立或更新後端之後、您可以使用「tridentctl」或對應的「tridentbackend」CRD來檢查後端：

```
./tridentctl get backends ontap_nas_auto_export -n trident -o yaml
items:
- backendUUID: 403b5326-8482-40db-96d0-d83fb3f4daec
  config:
    aggregate: ""
    autoExportCIDRs:
    - 192.168.0.0/24
    autoExportPolicy: true
    backendName: ontap_nas_auto_export
    chapInitiatorSecret: ""
    chapTargetInitiatorSecret: ""
    chapTargetUsername: ""
    chapUsername: ""
    dataLIF: 192.168.0.135
    debug: false
    debugTraceFlags: null
    defaults:
      encryption: "false"
      exportPolicy: <automatic>
      fileType: ext4
```

當節點新增至Kubernetes叢集並向Astra Trident控制器註冊時、現有後端的匯出原則會更新（前提是它們位於後端的「AutoExportCIDR」中指定的位址範圍內）。

移除節點時、Astra Trident會檢查所有線上的後端、以移除節點的存取規則。Astra Trident將此節點IP從託管後端的匯出原則中移除、可防止惡意掛載、除非叢集中的新節點重複使用此IP。

對於先前現有的後端、請使用更新後端 `tridentctl update backend` 將確保Astra Trident自動管理匯出原則。如此將會建立以後端 UUID 命名的新匯出原則、並在重新掛載後端上的磁碟區時、使用新建立的匯出原則。



刪除具有自動管理匯出原則的後端、將會刪除動態建立的匯出原則。如果重新建立後端、則會將其視為新的後端、並導致建立新的匯出原則。

如果即時節點的IP位址已更新、您必須重新啟動節點上的Astra Trident Pod。Astra Trident接著會更新其管理的後端匯出原則、以反映此IP變更。

準備配置SMB磁碟區

只需稍加準備、您就可以使用來配置 SMB 磁碟區 `ontap-nas` 驅動程式：



您必須在 SVM 上同時設定 NFS 和 SMB/CIFS 通訊協定、才能建立 `ontap-nas-economy` 適用於內部部署 ONTAP 的 SMB Volume。若未設定上述任一種通訊協定、將導致 SMB 磁碟區建立失敗。

開始之前

在配置 SMB 磁碟區之前、您必須具備下列項目。

- Kubernetes叢集具備Linux控制器節點、以及至少一個執行Windows Server 2019的Windows工作節點。Astra Trident僅支援安裝在Windows節點上執行的Pod上的SMB磁碟區。
- 至少有一個Astra Trident機密、其中包含您的Active Directory認證資料。以產生機密 `smbcreds`：

```
kubectl create secret generic smbcreds --from-literal username=user
--from-literal password='password'
```

- 設定為Windows服務的SCSI Proxy。若要設定 `csi-proxy`、請參閱 ["GitHub：csi Proxy"](#) 或 ["GitHub：適用於Windows的SCSI Proxy"](#) 適用於Windows上執行的Kubernetes節點。

步驟

1. 對於內部部署 ONTAP、您可以選擇性地建立 SMB 共用、或是 Astra Trident 可以為您建立一個。



Amazon FSX for ONTAP 需要 SMB 共享。

您可以使用兩種方式之一來建立SMB管理共用區 ["Microsoft管理主控台"](#) 共享資料夾嵌入式管理單元或使用ONTAP CLI。若要使用ONTAP CLI建立SMB共用：

- a. 如有必要、請建立共用的目錄路徑結構。

◦ `vserver cifs share create` 命令會在共用建立期間檢查-path選項中指定的路徑。如果指定的路徑不存在、則命令會失敗。

- b. 建立與指定SVM相關的SMB共用區：

```
vserver cifs share create -vserver vserver_name -share-name
share_name -path path [-share-properties share_properties,...]
[other_attributes] [-comment text]
```

- c. 確認共用區已建立：

```
vserver cifs share show -share-name share_name
```



請參閱 ["建立SMB共用區"](#) 以取得完整詳細資料。

2. 建立後端時、您必須設定下列項目以指定SMB Volume。如需ONTAP 所有的FSXfor Sendbackend組態選項、請參閱 ["FSX提供ONTAP 各種組態選項和範例"](#)。

參數	說明	範例
smbShare	<p>您可以指定下列其中一項：使用 Microsoft 管理主控台或 ONTAP CLI 建立的 SMB 共用名稱；允許 Astra Trident 建立 SMB 共用的名稱；或將參數保留空白以防止共用磁碟區存取。</p> <p>對於內部部署 ONTAP，此參數為選用項目。</p> <p>Amazon FSX 需要此參數才能支援 ONTAP 後端、且不可為空白。</p>	smb-share
nasType	*必須設定為 smb.*如果為null、則預設為 nfs。	smb
《生態樣式》	新磁碟區的安全樣式。必須設定為 ntfs 或 mixed 適用於 SMB 磁碟區。	ntfs 或 mixed 適用於SMB磁碟區
「unixPermissions」	新磁碟區的模式。SMB磁碟區*必須保留為空白。*	"

列舉NAS組態選項與範例ONTAP

瞭解如何在 Astra Trident 安裝中建立及使用 ONTAP NAS 驅動程式。本節提供後端組態範例及將後端對應至 StorageClasses 的詳細資料。

後端組態選項

如需後端組態選項、請參閱下表：

參數	說明	預設
「分度」		永遠為1
「storageDriverName」	儲存驅動程式名稱	「ONTAP - NAS」、「ONTAP - NAS - 經濟」、「ONTAP - NAS - Flexgroup」、「ONTAP - SAN」、「ONTAP - SAN 經濟」
「後端名稱」	自訂名稱或儲存後端	驅動程式名稱 + "_" + dataLIF
《馬納格門達利》	<p>叢集或SVM管理LIF的IP位址</p> <p>您可以指定完整網域名稱（FQDN）。</p> <p>如果使用 IPv6 旗標安裝 Astra Trident，則可以設定為使用 IPv6 位址。IPv6 位址必須以方括弧定義、例如 [28e8:d9fb:a825:b7bf:69a8:d02f:9e7b:3555]。</p> <p>如需無縫 MetroCluster 之間的互通性、請參閱 MetroCluster 範例。</p>	「10.0.0.1」、「[2001:1234:abcd:::fefo]」

參數	說明	預設
「DataLIF」	<p>傳輸協定LIF的IP位址。</p> <p>我們建議具體說明 dataLIF。如果未提供、Astra Trident會從SVM擷取資料lifs。您可以指定要用於NFS掛載作業的完整網域名稱（FQDN）、讓您建立循環配置資源DNS、以便在多個資料生命期之間達到負載平衡。</p> <p>可在初始設定之後變更。請參閱。</p> <p>如果使用 IPv6 旗標安裝 Astra Trident、則可以設定為使用 IPv6 位址。IPv6 位址必須以方括弧定義、例如 [28e8:d9fb:a825:b7bf:69a8:d02f:9e7b:3555]。</p> <p>* MetroCluster 請省略。* 請參閱 MetroCluster 範例。</p>	指定位址或從SVM衍生（若未指定）（不建議使用）
《虛擬機器》	<p>要使用的儲存虛擬機器</p> <p>* MetroCluster 請省略。* 請參閱 MetroCluster 範例。</p>	如果指定SVM "managementLIF"則衍生
「AutoExportPolicy」	啟用自動匯出原則建立及更新[布林值]。使用 autoExportPolicy 和 autoExportCIDRs 選項：Astra Trident可自動管理匯出原則。	錯
《AutoExportCIDR》（自動匯出CTR）	<p>篩選 Kubernetes 節點 IP 的 CIDR 清單、以對抗時間 autoExportPolicy 已啟用。</p> <p>使用 autoExportPolicy 和 autoExportCIDRs 選項：Astra Trident可自動管理匯出原則。</p>	["0.0.0/0"、":/0"]
《標籤》	套用到磁碟區的任意JSON-格式化標籤集	"
「用戶端憑證」	用戶端憑證的Base64編碼值。用於憑證型驗證	"
「clientPrivate Key」	用戶端私密金鑰的Base64編碼值。用於憑證型驗證	"
「可信賴的CACertificate」	受信任CA憑證的Base64編碼值。選用。用於憑證型驗證	"
《使用者名稱》	連線至叢集/ SVM的使用者名稱。用於認證型驗證	
密碼	連線至叢集/ SVM的密碼。用於認證型驗證	
「storagePrefix」	在SVM中配置新磁碟區時所使用的前置碼。設定後無法更新	" Trident "
「限制Aggregateusage」	如果使用率高於此百分比、則無法進行資源配置。*不適用於Amazon FSX for ONTAP Sfor Sfor *	""（預設不強制執行）
《限制Volume大小》	如果要求的磁碟區大小高於此值、則資源配置失敗。也會限制其管理的qtree和LUN、以及的磁碟區大小上限 qtreesPerFlexvol 選項可自訂每FlexVol 個支援區的配額樹數上限。	""（預設不會強制執行）

參數	說明	預設
《lunsPerFlexvol》	每FlexVol 個LUN的最大LUN數量、範圍必須在[50、200]	"100"
「DebugTraceFlags」	疑難排解時要使用的偵錯旗標。例如、 { "api" : false 、 "method" : true} 請勿使用 debugTraceFlags 除非您正在疑難排解並需要詳細的記錄傾印。	null
nasType	設定NFS或SMB磁碟區建立。選項包括 nfs、 smb 或null。NFS磁碟區的預設值設為null。	nfs
「nfsMountOptions」	以逗號分隔的NFS掛載選項清單。Kubernetes持續磁碟區的掛載選項通常會在儲存類別中指定、但如果儲存類別中未指定掛載選項、則Astra Trident會改回使用儲存後端組態檔中指定的掛載選項。如果儲存類別或組態檔中未指定掛載選項、Astra Trident將不會在相關的持續磁碟區上設定任何掛載選項。	"
"qtreesPerFlexvol"	每FlexVol 個邊的最大qtree數、必須在範圍內[50、300]	"200"
smbShare	您可以指定下列其中一項：使用 Microsoft 管理主控台或 ONTAP CLI 建立的 SMB 共用名稱；允許 Astra Trident 建立 SMB 共用的名稱；或將參數保留空白以防止共用磁碟區存取。 對於內部部署 ONTAP、此參數為選用項目。 Amazon FSX 需要此參數才能支援 ONTAP 後端、且不可為空白。	smb-share
《useREST》	使用ONTAP Isrest API的布林參數。技術預覽 useREST 以*技術預覽*的形式提供、建議用於測試環境、而非用於正式作業工作負載。設定為時 true、Astra Trident將使用ONTAP 靜止API與後端進行通訊。此功能需要ONTAP 使用更新版本的版本。此外ONTAP、所使用的登入角色必須能夠存取 ontap 應用程式：這是預先定義的 vsadmin 和 cluster-admin 角色： useREST 不支援MetroCluster 使用支援。	錯

用於資源配置磁碟區的後端組態選項

您可以使用中的這些選項來控制預設資源配置 defaults 組態區段。如需範例、請參閱下列組態範例。

參數	說明	預設
"paceAllocate (配置)"	LUN的空間分配	" 對 "
《保護區》	空間保留模式；「無」（精簡）或「Volume」（粗）	" 無 "
「快照原則」	要使用的Snapshot原則	" 無 "

參數	說明	預設
「qosPolicy」	要指派給所建立磁碟區的QoS原則群組。選擇每個儲存集區/後端的其中一個qosPolicy或adaptiveQosPolicy	"
《adaptiveQosPolicy》	要指派給所建立磁碟區的調適性QoS原則群組。選擇每個儲存集區/後端的其中一個qosPolicy或adaptiveQosPolicy。不受ONTAP-NAS-經濟支援。	"
「快照保留區」	保留給快照的磁碟區百分比	「0」如果 snapshotPolicy 為「無」、否則為「」
「PlitOnClone」	建立複本時、從其父複本分割複本	"假"
加密	在新磁碟區上啟用NetApp Volume Encryption (NVE) ；預設為「假」。必須在叢集上授權並啟用NVE、才能使用此選項。如果在後端啟用NAE、則Astra Trident中配置的任何磁碟區都會啟用NAE。如需詳細資訊、請參閱： "Astra Trident如何與NVE和NAE搭配運作" 。	"假"
「分層政策」	分層原則以使用「無」	「僅限快照」適用於 ONTAP 9.5 之前的 SVM-DR 組態
「unixPermissions」	新磁碟區的模式	"777" 表示 NFS 磁碟區；SMB 磁碟區為空的（不適用）
「snapshotDir」	控制對的存取 .snapshot 目錄	"假"
「匯出政策」	要使用的匯出原則	"預設"
《生態樣式》	新磁碟區的安全樣式。NFS支援 mixed 和 unix 安全樣式；SMB支援 mixed 和 ntfs 安全樣式：	NFS預設為 unix。SMB預設為 ntfs。



搭配Astra Trident使用QoS原則群組需要ONTAP 使用更新版本的版本。建議使用非共用的QoS原則群組、並確保原則群組會個別套用至每個組成群組。共享的QoS原則群組將強制所有工作負載的總處理量上限。

Volume資源配置範例

以下是定義預設值的範例：

```

---
version: 1
storageDriverName: ontap-nas
backendName: customBackendName
managementLIF: 10.0.0.1
dataLIF: 10.0.0.2
labels:
  k8scluster: dev1
  backend: dev1-nasbackend
svm: trident_svm
username: cluster-admin
password: <password>
limitAggregateUsage: 80%
limitVolumeSize: 50Gi
nfsMountOptions: nfsvers=4
debugTraceFlags:
  api: false
  method: true
defaults:
  spaceReserve: volume
  qosPolicy: premium
  exportPolicy: myk8scluster
  snapshotPolicy: default
  snapshotReserve: '10'

```

對於「ONTAP-NAS」和「ONTAP-NAS-flexGroups」、Astra Trident現在採用新的計算方法、確保FlexVol 利用snapshotReserve百分比和PVC正確調整其規模。當使用者要求使用PVCs時、Astra Trident會FlexVol 使用新的計算方式、建立原始的包含更多空間的候選區。此計算可確保使用者在永久虛擬磁碟中獲得所要求的可寫入空間、且空間不得小於所要求的空間。在v21.07之前、當使用者要求使用PVC（例如5GiB）、快照保留區達到50%時、他們只能獲得2.5GiB的可寫入空間。這是因為使用者要求的是整個磁碟區、而「快照保留區」則是其中的一個百分比。使用Trident 21.07時、使用者要求的是可寫入空間、而Astra Trident則將「快照保留區」編號定義為整個Volume的百分比。這不適用於「ONTAP-NAS-經濟」。請參閱下列範例以瞭解此功能的運作方式：

計算方式如下：

```

Total volume size = (PVC requested size) / (1 - (snapshotReserve
percentage) / 100)

```

對於snapshotReserve = 50%、而PVC要求= 5GiB、磁碟區總大小為 $2/0.5 = 10\text{GiB}$ 、可用大小為5GiB、這是使用者在PVC要求中要求的大小。「volume show (Volume show) 命令應顯示類似以下範例的結果：

Vserver	Volume	Aggregate	State	Type	Size	Available	Used%
		_pvc_89f1c156_3801_4de4_9f9d_034d54c395f4	online	RW	10GB	5.00GB	0%
		_pvc_e8372153_9ad9_474a_951a_08ae15e1c0ba	online	RW	1GB	511.8MB	0%

2 entries were displayed.

在升級Astra Trident時、先前安裝的現有後端會按照上述說明來配置磁碟區。對於在升級之前建立的磁碟區、您應該調整其磁碟區大小、以便觀察變更。例如、較早前使用「快照保留區=50」的2GiB室早導致磁碟區提供1GiB的可寫入空間。例如、將磁碟區大小調整為3GiB、可讓應用程式在6 GiB磁碟區上擁有3GiB的可寫入空間。

最低組態範例

下列範例顯示基本組態、讓大部分參數保留預設值。這是定義後端最簡單的方法。



如果您在NetApp ONTAP 支援Trident的NetApp支援上使用Amazon FSX、建議您指定lifs的DNS名稱、而非IP位址。

ONTAP NAS 經濟效益範例

```
---
version: 1
storageDriverName: ontap-nas-economy
managementLIF: 10.0.0.1
dataLIF: 10.0.0.2
svm: svm_nfs
username: vsadmin
password: password
```

ONTAP NAS FlexGroup 範例

```
---
version: 1
storageDriverName: ontap-nas-flexgroup
managementLIF: 10.0.0.1
dataLIF: 10.0.0.2
svm: svm_nfs
username: vsadmin
password: password
```

MetroCluster 範例

您可以設定後端、避免在切換和切換期間手動更新後端定義 "SVM 複寫與還原"。

若要無縫切換和切換、請使用指定 SVM managementLIF 並省略 dataLIF 和 svm 參數。例如：

```
---
version: 1
storageDriverName: ontap-nas
managementLIF: 192.168.1.66
username: vsadmin
password: password
```

SMB Volume 範例

```
---
version: 1
backendName: ExampleBackend
storageDriverName: ontap-nas
managementLIF: 10.0.0.1
nasType: smb
securityStyle: ntfs
unixPermissions: ""
dataLIF: 10.0.0.2
svm: svm_nfs
username: vsadmin
password: password
```

憑證型驗證範例

這是最小的後端組態範例。clientCertificate、clientPrivateKey`和`trustedCACertificate（選用、如果使用信任的CA）會填入 backend.json 並分別取得用戶端憑證、私密金鑰及信任CA憑證的基礎64編碼值。

```
---
version: 1
backendName: DefaultNASBackend
storageDriverName: ontap-nas
managementLIF: 10.0.0.1
dataLIF: 10.0.0.15
svm: nfs_svm
clientCertificate: ZXR0ZXJwYXB...ICMgJ3BhcGVyc2
clientPrivateKey: vciwKIyAgZG...0cnksIGRlc2NyaX
trustedCACertificate: zcyBbaG...b3Igb3duIGNsYXNz
storagePrefix: myPrefix_
```

自動匯出原則範例

本範例說明如何指示Astra Trident使用動態匯出原則來自動建立及管理匯出原則。這對的運作方式相同 ontap-nas-economy 和 ontap-nas-flexgroup 驅動程式：

```
---
version: 1
storageDriverName: ontap-nas
managementLIF: 10.0.0.1
dataLIF: 10.0.0.2
svm: svm_nfs
labels:
  k8scluster: test-cluster-east-1a
  backend: test1-nasbackend
autoExportPolicy: true
autoExportCIDRs:
- 10.0.0.0/24
username: admin
password: password
nfsMountOptions: nfsvers=4
```

IPv6 位址範例

此範例顯示 managementLIF 使用 IPv6 位址。

```
---
version: 1
storageDriverName: ontap-nas
backendName: nas_ipv6_backend
managementLIF: "[5c5d:5edf:8f:7657:bef8:109b:1b41:d491]"
labels:
  k8scluster: test-cluster-east-1a
  backend: test1-ontap-ipv6
svm: nas_ipv6_svm
username: vsadmin
password: password
```

Amazon FSX for ONTAP 使用 SMB Volume 範例

- smbShare 使用 SMB 磁碟區的 ONTAP 需要 FSX 參數。

```
---
version: 1
backendName: SMBBackend
storageDriverName: ontap-nas
managementLIF: example.mgmt.fqdn.aws.com
nasType: smb
dataLIF: 10.0.0.15
svm: nfs_svm
smbShare: smb-share
clientCertificate: ZXR0ZXJwYXB...ICMgJ3BhcGVyc2
clientPrivateKey: vciwKIyAgZG...0cnksIGRlc2NyaX
trustedCACertificate: zcyBbaG...b3Igb3duIGNsYXNz
storagePrefix: myPrefix_
```

虛擬集區的后端範例

在下面顯示的后端定義檔案範例中、會針對所有儲存池設定特定的預設值、例如 spaceReserve 無、spaceAllocation 假、和 encryption 錯。虛擬資源池是在儲存區段中定義的。

Astra Trident 會在「意見」欄位中設定資源配置標籤。註解是在的 FlexVol 上設定 ontap-nas 或 FlexGroup 支援 ontap-nas-flexgroup。Astra Trident 會在資源配置時、將虛擬資源池上的所有標籤複製到儲存磁碟區。為了方便起見、儲存管理員可以針對每個虛擬資源池定義標籤、並依標籤將磁碟區分組。

在這些範例中、有些儲存池是自行設定的 `spaceReserve`、`spaceAllocation` 和 `encryption` 值、而某些資源池會覆寫預設值。


```

---
version: 1
storageDriverName: ontap-nas
managementLIF: 10.0.0.1
svm: svm_nfs
username: admin
password: <password>
nfsMountOptions: nfsvers=4
defaults:
  spaceReserve: none
  encryption: 'false'
  qosPolicy: standard
labels:
  store: nas_store
  k8scluster: prod-cluster-1
region: us_east_1
storage:
- labels:
  app: msoffice
  cost: '100'
  zone: us_east_1a
  defaults:
    spaceReserve: volume
    encryption: 'true'
    unixPermissions: '0755'
    adaptiveQosPolicy: adaptive-premium
- labels:
  app: slack
  cost: '75'
  zone: us_east_1b
  defaults:
    spaceReserve: none
    encryption: 'true'
    unixPermissions: '0755'
- labels:
  department: legal
  creditpoints: '5000'
  zone: us_east_1b
  defaults:
    spaceReserve: none
    encryption: 'true'
    unixPermissions: '0755'
- labels:

```

```
    app: wordpress
    cost: '50'
    zone: us_east_1c
    defaults:
      spaceReserve: none
      encryption: 'true'
      unixPermissions: '0775'
- labels:
    app: mysqlldb
    cost: '25'
    zone: us_east_1d
    defaults:
      spaceReserve: volume
      encryption: 'false'
      unixPermissions: '0775'
```

```
---
version: 1
storageDriverName: ontap-nas-flexgroup
managementLIF: 10.0.0.1
svm: svm_nfs
username: vsadmin
password: <password>
defaults:
  spaceReserve: none
  encryption: 'false'
labels:
  store: flexgroup_store
  k8scluster: prod-cluster-1
region: us_east_1
storage:
- labels:
  protection: gold
  creditpoints: '50000'
  zone: us_east_1a
  defaults:
    spaceReserve: volume
    encryption: 'true'
    unixPermissions: '0755'
- labels:
  protection: gold
  creditpoints: '30000'
  zone: us_east_1b
  defaults:
    spaceReserve: none
    encryption: 'true'
    unixPermissions: '0755'
- labels:
  protection: silver
  creditpoints: '20000'
  zone: us_east_1c
  defaults:
    spaceReserve: none
    encryption: 'true'
    unixPermissions: '0775'
- labels:
  protection: bronze
  creditpoints: '10000'
  zone: us_east_1d
```

```
defaults:  
  spaceReserve: volume  
  encryption: 'false'  
  unixPermissions: '0775'
```

```
---
version: 1
storageDriverName: ontap-nas-economy
managementLIF: 10.0.0.1
svm: svm_nfs
username: vsadmin
password: <password>
defaults:
  spaceReserve: none
  encryption: 'false'
labels:
  store: nas_economy_store
region: us_east_1
storage:
- labels:
  department: finance
  creditpoints: '6000'
  zone: us_east_1a
  defaults:
    spaceReserve: volume
    encryption: 'true'
    unixPermissions: '0755'
- labels:
  protection: bronze
  creditpoints: '5000'
  zone: us_east_1b
  defaults:
    spaceReserve: none
    encryption: 'true'
    unixPermissions: '0755'
- labels:
  department: engineering
  creditpoints: '3000'
  zone: us_east_1c
  defaults:
    spaceReserve: none
    encryption: 'true'
    unixPermissions: '0775'
- labels:
  department: humanresource
  creditpoints: '2000'
  zone: us_east_1d
  defaults:
```

```
spaceReserve: volume
encryption: 'false'
unixPermissions: '0775'
```

將後端對應至StorageClass

請參閱下列 StorageClass 定義 [\[虛擬集區的後端範例\]](#)。使用 `parameters.selector` 欄位中、每個 StorageClass 都會呼叫哪些虛擬集區可用於主控磁碟區。磁碟區將會在所選的虛擬資源池中定義各個層面。

- `protection-gold` StorageClass 會對應至中的第一個和第二個虛擬集區 `ontap-nas-flexgroup` 後端：這是唯一提供金級保護的資源池。

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: protection-gold
provisioner: csi.trident.netapp.io
parameters:
  selector: "protection=gold"
  fsType: "ext4"
```

- `protection-not-gold` StorageClass 會對應至中的第三和第四個虛擬集區 `ontap-nas-flexgroup` 後端：這是唯一提供金級以外保護層級的資源池。

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: protection-not-gold
provisioner: csi.trident.netapp.io
parameters:
  selector: "protection!=gold"
  fsType: "ext4"
```

- `app-mysqldb` StorageClass 會對應至中的第四個虛擬集區 `ontap-nas` 後端：這是唯一為 `mysqldb` 類型應用程式提供儲存池組態的集區。

```

apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: app-mysqldb
provisioner: csi.trident.netapp.io
parameters:
  selector: "app=mysqldb"
  fsType: "ext4"

```

- `tprotection-silver-creditpoints-20k` StorageClass 會對應至中的第三個虛擬集區 `ontap-nas-flexgroup` 後端：這是唯一提供銀級保護和 20000 個信用點數的資源池。

```

apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: protection-silver-creditpoints-20k
provisioner: csi.trident.netapp.io
parameters:
  selector: "protection=silver; creditpoints=20000"
  fsType: "ext4"

```

- `creditpoints-5k` StorageClass 會對應至中的第三個虛擬集區 `ontap-nas` 後端和中的第二個虛擬集區 `ontap-nas-economy` 後端：這是唯一擁有 5000 個信用點數的集區方案。

```

apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: creditpoints-5k
provisioner: csi.trident.netapp.io
parameters:
  selector: "creditpoints=5000"
  fsType: "ext4"

```

Astra Trident將決定選取哪個虛擬集區、並確保符合儲存需求。

更新 dataLIF 初始組態之後

您可以在初始組態後變更資料LIF、方法是執行下列命令、以更新資料LIF提供新的後端Json檔案。

```

tridentctl update backend <backend-name> -f <path-to-backend-json-file-with-updated-dataLIF>

```



如果將PVCS附加至一或多個Pod、您必須關閉所有對應的Pod、然後將其重新啟動、新的資料LIF才會生效。

Amazon FSX for NetApp ONTAP 產品

使用Astra Trident搭配Amazon FSX for NetApp ONTAP 解決方案

"Amazon FSX for NetApp ONTAP 產品" 是完全託管的AWS服務、可讓客戶啟動及執行採用NetApp ONTAP 資訊儲存作業系統的檔案系統。FSX for ONTAP VMware可讓您運用熟悉的NetApp功能、效能和管理功能、同時充分發揮儲存AWS資料的簡易性、敏捷度、安全性和擴充性。FSX for ONTAP Sfor支援ONTAP Isf供 檔案系統功能和管理API。

總覽

檔案系統是Amazon FSX的主要資源、類似ONTAP 於內部部署的一個叢集。在每個SVM中、您可以建立一個或多個磁碟區、這些磁碟區是儲存檔案系統中檔案和資料夾的資料容器。有了Amazon FSX for NetApp ONTAP 的功能、Data ONTAP 即可在雲端以託管檔案系統的形式提供支援。新的檔案系統類型稱為* NetApp ONTAP Sing*。

使用Astra Trident搭配Amazon FSX for NetApp ONTAP 供應NetApp時、您可以確保在Amazon Elastic Kubernetes Service (EKS) 中執行的Kubernetes叢集、能夠配置區塊和檔案以ONTAP 支援的持續磁碟區。

考量

- SMB Volume :
 - 使用支援SMB磁碟區 `ontap-nas` 僅限驅動程式。
 - Astra Trident EKS 附加元件不支援 SMB Volume 。
 - Astra Trident僅支援安裝在Windows節點上執行的Pod上的SMB磁碟區。
- 在 Astra Trident 24.02 之前、Trident 無法刪除在 Amazon FSX 檔案系統上建立且已啟用自動備份的磁碟區。若要在 Astra Trident 24.02 或更新版本中避免此問題、請指定 `fsxFilesystemID`、AWS `apiRegion`、AWS `apiKey` 和 AWS `secretKey` 在 AWS FSX for ONTAP 的後端組態檔案中。



如果您要指定 Astra Trident 的 IAM 角色、則可以省略指定 `apiRegion`、`apiKey` 和 `secretKey` 明確地將欄位移至 Astra Trident。如需詳細資訊、請參閱 "[FSX提供ONTAP 各種組態選項和範例](#)"。

適用於 ONTAP 驅動程式詳細資料的 FSX

您可以ONTAP 使用下列驅動程式、將Astra Trident與Amazon FSX for NetApp整合：

- 「ONTAP-san」：每個提供的PV都是其專屬Amazon FSX for NetApp ONTAP 的LUN。
- 「ONTAP-san經濟型」：每個配置的PV都是LUN、每個Amazon FSX for NetApp ONTAP 的LUN數量可設定。
- 「ONTAP-NAS」：每個提供的PV都是適用於NetApp ONTAP 的完整Amazon FSX。
- 「ONTAP-NAS-EAS」：每個提供的PV都是qtree、每個Amazon FSX的NetApp ONTAP 功能是可設定的配額

樹數。

- 「ONTAP-NAS-Flexgroup」：每個提供的PV都是適用於NetApp ONTAP FlexGroup 的完整Amazon FSX。

如需驅動程式詳細資料、請參閱 "[NAS 驅動程式](#)" 和 "[SAN 驅動程式](#)"。

驗證

Astra Trident提供兩種驗證模式。

- 憑證型：Astra Trident會使用SVM上安裝的憑證、與FSX檔案系統上的SVM進行通訊。
- 認證型：您可以將「fsxadmin」使用者用於您的檔案系統、或是為SVM設定的「vsadmin」使用者。



Astra Trident希望以 vsadmin SVM使用者或具有相同角色之不同名稱的使用者。適用於NetApp ONTAP 的Amazon FSX具備以下功能 fsxadmin 使用者只能有限地取代ONTAP 此功能 admin 叢集使用者：強烈建議使用 vsadmin 使用Astra Trident。

您可以更新後端以在認證型和憑證型方法之間移動。不過、如果您嘗試提供*認證資料和認證*、後端建立將會失敗。若要切換至不同的驗證方法、您必須從後端組態中移除現有方法。

如需啟用驗證的詳細資訊、請參閱您的驅動程式類型驗證：

- "[ASNAS驗證ONTAP](#)"
- "[支援SAN驗證ONTAP](#)"

EKS 的雲端身分識別

雲端身分識別可讓 Kubernetes Pod 以 AWS IAM 角色的身分驗證、而非提供明確的 AWS 認證、來存取 AWS 資源。

若要在 AWS 中使用雲端身分識別、您必須具備：

- 使用 EKS 部署的 Kubernetes 叢集
- 安裝的 Astra Trident 包含 cloudProvider 指定 "AWS" 和 cloudIdentity 指定 AWS IAM 角色。

Trident 運算子

若要使用 Trident 運算子安裝 Astra Trident、請編輯 `tridentorchestrator_cr.yaml` 以設定 `cloudProvider` 至 "AWS" 並設定 `cloudIdentity` 至 AWS IAM 角色。

例如：

```
apiVersion: trident.netapp.io/v1
kind: TridentOrchestrator
metadata:
  name: trident
spec:
  debug: true
  namespace: trident
  imagePullPolicy: IfNotPresent
  cloudProvider: "AWS"
  cloudIdentity: "'eks.amazonaws.com/role-arn:
arn:aws:iam::123456:role/astratrident-role'"
```

掌舵

使用下列環境變數設定 * 雲端供應商 * 和 * 雲端 IDENTITY * 旗標的值：

```
export CP="AWS"
export CI="'eks.amazonaws.com/role-arn:
arn:aws:iam::123456:role/astratrident-role'"
```

以下範例安裝 Astra Trident 和 Sets `cloudProvider` 至 AWS 使用環境變數 `$CP` 並使用環境變數設定 `'cloudIdentity'` `$CI`：

```
helm install trident trident-operator-100.2402.0.tgz --set
cloudProvider=$CP --set cloudIdentity=$CI
```

<code>tridentctl</code>

使用下列環境變數設定 * 雲端供應商 * 和 * 雲端 IDENTITY * 旗標的值：

```
export CP="AWS"
export CI="'eks.amazonaws.com/role-arn:
arn:aws:iam::123456:role/astratrident-role'"
```

下列範例安裝 Astra Trident 並設定 `cloud-provider` 標記為 `$CP` 和 ``cloud-identity` 至 `$CI`：

```
tridentctl install --cloud-provider=$CP --cloud-identity="$CI" -n
trident
```

如需詳細資訊、請參閱

- ["Amazon FSX for NetApp ONTAP 的支援文件"](#)
- ["Amazon FSX for NetApp ONTAP 的部落格文章"](#)

整合Amazon FSX for NetApp ONTAP 功能

您可以將Amazon FSX for NetApp ONTAP 的支援文件系統與Astra Trident整合、以確保在Amazon Elastic Kubernetes Service (EKS) 中執行的Kubernetes叢集能夠配置區塊並以ONTAP 支援的方式歸檔持續Volume。

需求

此外 ["Astra Trident的需求"](#)、若要將FSXfor ONTAP 支援與Astra Trident整合、您需要：

- 現有的Amazon EKS叢集或自行管理的Kubernetes叢集、已安裝「kubectll」。
- 可從叢集工作節點存取的現有 Amazon FSX for NetApp ONTAP 檔案系統和儲存虛擬機器 (SVM)。
- 已準備好的工作節點 ["NFS或iSCSI"](#)。



請務必遵循Amazon Linux和Ubuntu所需的節點準備步驟 ["Amazon機器映像"](#) (AMIs)、視您的EKS AMI類型而定。

- Astra Trident僅支援安裝在Windows節點上執行的Pod上的SMB磁碟區。請參閱 [準備配置SMB磁碟區](#) 以取得詳細資料。

整合SAN和NAS驅動程式ONTAP



如果您要設定SMB磁碟區、則必須閱讀 [準備配置SMB磁碟區](#) 在建立後端之前。

步驟

1. 使用其中一項部署Astra Trident ["部署方法"](#)。
2. 收集SVM管理LIF DNS名稱。例如、使用AWS CLI尋找 `DNSName` 輸入 `Endpoints → Management` 執行下列命令之後：

```
aws fsx describe-storage-virtual-machines --region <file system region>
```

3. 建立及安裝的憑證 ["NAS後端驗證"](#) 或 ["SAN後端驗證"](#)。



您可以使用SSH從任何位置登入檔案系統（例如安裝憑證）、而該SSH可連至檔案系統。使用「fsxadmin」使用者、您在建立檔案系統時設定的密碼、以及「AWS FSx file-systems」中的管理DNS名稱。

4. 使用您的憑證和管理LIF的DNS名稱建立後端檔案、如下例所示：

YAML

```
version: 1
storageDriverName: ontap-san
backendName: customBackendName
managementLIF: svm-XXXXXXXXXXXXXXXXXXXX.fs-XXXXXXXXXXXXXXXXXXXX.fsx.us-
east-2.aws.internal
svm: svm01
clientCertificate: ZXR0ZXJwYXB...ICMgJ3BhcGVyc2
clientPrivateKey: vciwKIyAgZG...0cnksIGRlc2NyaX
trustedCACertificate: zcyBbaG...b3Igb3duIGNsYXNz
```

JSON

```
{
  "version": 1,
  "storageDriverName": "ontap-san",
  "backendName": "customBackendName",
  "managementLIF": "svm-XXXXXXXXXXXXXXXXXXXX.fs-
XXXXXXXXXXXXXXXXXXXX.fsx.us-east-2.aws.internal",
  "svm": "svm01",
  "clientCertificate": "ZXR0ZXJwYXB...ICMgJ3BhcGVyc2",
  "clientPrivateKey": "vciwKIyAgZG...0cnksIGRlc2NyaX",
  "trustedCACertificate": "zcyBbaG...b3Igb3duIGNsYXNz"
}
```

或者、您也可以使用儲存在 AWS Secret Manager 中的 SVM 認證（使用者名稱和密碼）來建立後端檔案、如以下範例所示：

YAML

```
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-tbc-ontap-nas
spec:
  version: 1
  storageDriverName: ontap-nas
  backendName: tbc-ontap-nas
  svm: svm-name
  aws:
    fsxFileSystemID: fs-xxxxxxxxxx
  managementLIF:
  credentials:
    name: "arn:aws:secretsmanager:us-west-2:xxxxxxxx:secret:secret-
name"
    type: awsarn
```

JSON

```
{
  "apiVersion": "trident.netapp.io/v1",
  "kind": "TridentBackendConfig",
  "metadata": {
    "name": "backend-tbc-ontap-nas"
  },
  "spec": {
    "version": 1,
    "storageDriverName": "ontap-nas",
    "backendName": "tbc-ontap-nas",
    "svm": "svm-name",
    "aws": {
      "fsxFileSystemID": "fs-xxxxxxxxxx"
    },
    "managementLIF": null,
    "credentials": {
      "name": "arn:aws:secretsmanager:us-west-
2:xxxxxxxx:secret:secret-name",
      "type": "awsarn"
    }
  }
}
```

如需建立後端的相關資訊、請參閱下列連結：

- ["使用ONTAP NetApp NAS驅動程式設定後端"](#)
- ["使用ONTAP SAN驅動程式設定後端"](#)

準備配置SMB磁碟區

您可以使用來配置SMB磁碟區 `ontap-nas` 驅動程式：完成之前 [整合SAN和NAS驅動程式ONTAP](#) 完成下列步驟。

開始之前

在您使用配置 SMB 磁碟區之前、請先使用 `ontap-nas` 驅動程式、您必須具備下列項目。

- Kubernetes叢集具備Linux控制器節點、以及至少一個執行Windows Server 2019的Windows工作節點。Astra Trident僅支援安裝在Windows節點上執行的Pod上的SMB磁碟區。
- 至少有一個Astra Trident機密、其中包含您的Active Directory認證資料。以產生機密 `smbcreds`：

```
kubectl create secret generic smbcreds --from-literal username=user  
--from-literal password='password'
```

- 設定為Windows服務的SCSI Proxy。若要設定 `csi-proxy`、請參閱 ["GitHub：csi Proxy"](#) 或 ["GitHub：適用於Windows的SCSI Proxy"](#) 適用於Windows上執行的Kubernetes節點。

步驟

1. 建立SMB共用區。您可以使用兩種方式之一來建立SMB管理共用區 ["Microsoft管理主控台"](#) 共享資料夾嵌入式管理單元或使用ONTAP CLI。若要使用ONTAP CLI建立SMB共用：

- a. 如有必要、請建立共用的目錄路徑結構。

◦ `vserver cifs share create` 命令會在共用建立期間檢查-path選項中指定的路徑。如果指定的路徑不存在、則命令會失敗。

- b. 建立與指定SVM相關的SMB共用區：

```
vserver cifs share create -vserver vserver_name -share-name  
share_name -path path [-share-properties share_properties,...]  
[other_attributes] [-comment text]
```

- c. 確認共用區已建立：

```
vserver cifs share show -share-name share_name
```



請參閱 ["建立SMB共用區"](#) 以取得完整詳細資料。

2. 建立後端時、您必須設定下列項目以指定SMB Volume。如需ONTAP 所有的FSXfor Sendbackend組態選項、請參閱 ["FSX提供ONTAP 各種組態選項和範例"](#)。

參數	說明	範例
smbShare	您可以指定下列其中一項：使用 Microsoft 管理主控台或 ONTAP CLI 建立的 SMB 共用名稱、或是允許 Astra Trident 建立 SMB 共用的名稱。 ONTAP 後端的 Amazon FSX 需要此參數。	smb-share
nasType	*必須設定為 smb.*如果為null、則預設為 nfs。	smb
《生態樣式》	新磁碟區的安全樣式。必須設定為 ntfs 或 mixed 適用於 SMB 磁碟區。	ntfs 或 mixed 適用於SMB磁碟區
「unixPermissions」	新磁碟區的模式。SMB磁碟區*必須保留為空白。*	"

FSX提供ONTAP 各種組態選項和範例

深入瞭解Amazon FSX for ONTAP Sfor Sf。本節提供後端組態範例。

後端組態選項

如需後端組態選項、請參閱下表：

參數	說明	範例
「分度」		永遠為1
「storageDriverName」	儲存驅動程式名稱	ontap-nas、ontap-nas-economy、ontap-nas-flexgroup、ontap-san、ontap-san-economy
「後端名稱」	自訂名稱或儲存後端	驅動程式名稱+ 「_」 + dataLIF
《馬納格門達利》	叢集或SVM管理LIF的IP位址 您可以指定完整網域名稱（FQDN）。 如果使用 IPv6 旗標安裝 Astra Trident、則可以設定為使用 IPv6 位址。IPv6位址必須以方括弧來定義、例如[28e8：d9fb：a825：b7bf：69a8：d02f：9e7b：3555]。	「10.0.0.1」、 「[2001:1234:abcd:::fefo]」

參數	說明	範例
「DataLIF」	<p>傳輸協定LIF的IP位址。</p> <p>不適用NAS驅動程式：建議您指定dataLIF ONTAP。如果未提供、Astra Trident會從SVM擷取資料lifs。您可以指定要用於NFS掛載作業的完整網域名稱（FQDN）、讓您建立循環配置資源DNS、以便在多個資料生命期之間達到負載平衡。可在初始設定之後變更。請參閱。</p> <p>《SAN驅動程式：請勿指定用於iSCSI》ONTAP。Astra Trident使用ONTAP「選擇性LUN地圖」來探索建立多重路徑工作階段所需的iSCSI lifs。如果明確定義dataLIF、就會產生警告。</p> <p>如果使用 IPv6 旗標安裝 Astra Trident、則可以設定為使用 IPv6 位址。IPv6位址必須以方括弧來定義、例如[28e8:d9fb:a825:b7bf:69a8:d02f:9e7b:3555]。</p>	
「AutoExportPolicy」	啟用自動匯出原則建立及更新[布林值]。使用 autoExportPolicy 和 autoExportCIDRs 選項：Astra Trident可自動管理匯出原則。	「假」
《AutoExportCIDR》（自動匯出CTR）	<p>篩選 Kubernetes 節點 IP 的 CIDR 清單、以對抗時間 autoExportPolicy 已啟用。</p> <p>使用 autoExportPolicy 和 autoExportCIDRs 選項：Astra Trident可自動管理匯出原則。</p>	「[「0.00.0.0/0」、「:/0」]」
《標籤》	套用到磁碟區的任意JSON-格式化標籤集	"
「用戶端憑證」	用戶端憑證的Base64編碼值。用於憑證型驗證	"
「clientPrivate Key」	用戶端私密金鑰的Base64編碼值。用於憑證型驗證	"
「可信賴的CACertificate」	受信任CA憑證的Base64編碼值。選用。用於憑證型驗證。	"
《使用者名稱》	連線至叢集或SVM的使用者名稱。用於認證型驗證。例如、vsadmin。	
密碼	連線至叢集或SVM的密碼。用於認證型驗證。	

參數	說明	範例
《虛擬機器》	要使用的儲存虛擬機器	指定SVM管理LIF時衍生。
「toragePrefix」	在SVM中配置新磁碟區時所使用的前置碼。無法在建立後修改。若要更新此參數、您需要建立新的後端。	trident
「限制Aggregateusage」	*請勿指定Amazon FSX for NetApp ONTAP 的功能。*所提供的 fsxadmin 和 vsadmin 請勿包含擷取Aggregate使用量所需的權限、並使用Astra Trident加以限制。	請勿使用。
《限制Volume大小》	如果要求的磁碟區大小高於此值、則資源配置失敗。也會限制其管理的qtree和LUN、以及的磁碟區大小上限 qtreesPerFlexvol 選項可自訂每FlexVol 個支援區的配額樹數上限。	「」（預設不強制執行）
《lunsPerFlexvol》	每FlexVol 個LUN的最大LUN數量、範圍必須為[50、200]。僅限SAN。	100
「DebugTraceFlags」	疑難排解時要使用的偵錯旗標。範例： {"API":假、「method」:true} 不使用 debugTraceFlags 除非您正在疑難排解並需要詳細的記錄傾印。	null
「nfsMountOptions」	以逗號分隔的NFS掛載選項清單。Kubernetes持續磁碟區的掛載選項通常會在儲存類別中指定、但如果儲存類別中未指定掛載選項、則Astra Trident會改回使用儲存後端組態檔中指定的掛載選項。如果儲存類別或組態檔中未指定掛載選項、Astra Trident將不會在相關的持續磁碟區上設定任何掛載選項。	"
nasType	設定NFS或SMB磁碟區建立。選項包括 nfs、smb 或 null。*必須設定為 `smb` 對於SMB Volume。*設定為null、預設為NFS Volume。	nfs
"qtreesPerFlexvol"	每FlexVol 個邊的最大qtree數、必須在範圍內[50、300]	200
smbShare	您可以指定下列其中一項：使用 Microsoft 管理主控台或 ONTAP CLI 建立的 SMB 共用名稱、或是允許 Astra Trident 建立 SMB 共用的名稱。 ONTAP 後端的 Amazon FSX 需要此參數。	smb-share

參數	說明	範例
《useREST》	使用ONTAP Isrest API的布林參數。技術預覽 useREST 以*技術預覽*的形式提供、建議用於測試環境、而非用於正式作業工作負載。設定為true、Astra Trident將使用ONTAP 靜止API與後端進行通訊。此功能需要ONTAP 使用更新版本的版本。此外ONTAP 、所使用的登入角色必須能夠存取 ontap 應用程式：這是預先定義的 vsadmin 和 cluster-admin 角色：	「假」
aws	您可以在 AWS FSX for ONTAP 的組態檔中指定下列項目： - fsxFilesystemID：指定 AWS FSX 檔案系統的 ID 。 - apiRegion：AWS API 區域名稱。 - apikey：AWS API 金鑰。 - secretKey：AWS 秘密金鑰。	"" "" ""
credentials	指定要儲存在 AWS Secret Manager 中的 FSX SVM 認證。 - name：機密的 Amazon 資源名稱（ARN）、其中包含 SVM 的認證。 - type：設為 awsarn。 請參閱 "建立 AWS Secrets Manager 密碼" 以取得更多資訊。	

更新 dataLIF 初始組態之後

您可以在初始組態後變更資料LIF、方法是執行下列命令、以更新資料LIF提供新的後端Json檔案。

```
tridentctl update backend <backend-name> -f <path-to-backend-json-file-with-updated-dataLIF>
```



如果將PVCS附加至一或多個Pod、您必須關閉所有對應的Pod、然後將其重新啟動、新的資料LIF才會生效。

用於資源配置磁碟區的後端組態選項

您可以使用中的這些選項來控制預設資源配置 defaults 組態區段。如需範例、請參閱下列組態範例。

參數	說明	預設
"paceAllocate (配置) "	LUN的空間分配	"真的"

參數	說明	預設
《保護區》	空間保留模式；「無」（精簡）或「Volume」（完整）	無
「快照原則」	要使用的Snapshot原則	無
「qosPolicy」	要指派給所建立磁碟區的QoS原則群組。選擇每個儲存集區或後端的其中一個qosPolicy或adaptiveQosPolicy。搭配Astra Trident使用QoS原則群組需要ONTAP 使用更新版本的版本。我們建議使用非共用的QoS原則群組、並確保原則群組會個別套用至每個組成群組。共享的QoS原則群組將強制所有工作負載的總處理量上限。	「」
《adaptiveQosPolicy》	要指派給所建立磁碟區的調適性QoS原則群組。選擇每個儲存集區或後端的其中一個qosPolicy或adaptiveQosPolicy。不受ONTAP-NAS-經濟支援。	「」
「快照保留區」	保留給快照「0」的磁碟區百分比	如果 snapshotPolicy 是 none、else 「」
「PlitOnClone」	建立複本時、從其父複本分割複本	「假」
加密	在新磁碟區上啟用NetApp Volume Encryption（NVE）；預設為「假」。必須在叢集上授權並啟用NVE、才能使用此選項。如果在後端啟用NAE、則Astra Trident中配置的任何磁碟區都會啟用NAE。如需詳細資訊、請參閱： "Astra Trident如何與NVE和NAE搭配運作" 。	「假」
luksEncryption	啟用LUKS加密。請參閱 "使用Linux 統一金鑰設定（LUKS）" 。僅限SAN。	"
「分層政策」	要使用的分層原則 none	snapshot-only 適用於 ONTAP 9.5 之前的 SVM-DR 組態
「unixPermissions」	新磁碟區的模式。如果是 SMB 磁碟區、請保留空白。	「」
《生態樣式》	新磁碟區的安全樣式。NFS支援mixed和unix安全樣式；SMB支援mixed和ntfs安全樣式：	NFS預設為unix。SMB預設為ntfs。

組態範例

SMB 磁碟區的儲存類別組態

使用 `nasType`、`node-stage-secret-name` 和 `node-stage-secret-namespace`、您可以指定 SMB 磁碟區、並提供所需的 Active Directory 認證資料。使用支援 SMB 磁碟區 `ontap-nas` 僅限驅動程式。

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: nas-smb-sc
provisioner: csi.trident.netapp.io
parameters:
  backendType: "ontap-nas"
  trident.netapp.io/nasType: "smb"
  csi.storage.k8s.io/node-stage-secret-name: "smbcreds"
  csi.storage.k8s.io/node-stage-secret-namespace: "default"
```

搭配加密管理程式的 AWS FSX for ONTAP 組態

```
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-tbc-ontap-nas
spec:
  version: 1
  storageDriverName: ontap-nas
  backendName: tbc-ontap-nas
  svm: svm-name
  aws:
    fsxFileSystemID: fs-xxxxxxxxxx
  managementLIF:
  credentials:
    name: "arn:aws:secretsmanager:us-west-2:xxxxxxx:secret:secret-name"
    type: awsarn
```

在 EKS 叢集上設定 Astra Trident EKS 附加版本 23.10

Astra Trident 簡化了在 Kubernetes 進行 NetApp ONTAP 儲存管理的 Amazon FSX、讓開發人員和管理員能夠專注於應用程式部署。Astra Trident EKS 附加元件包含最新的安全性修補程式、錯誤修正程式、並經過 AWS 驗證、可與 Amazon EKS 搭配使用。EKS 附加元件可讓您持續確保 Amazon EKS 叢集安全穩定、並減少安裝、設定及更新附加元件所需的

工作量。

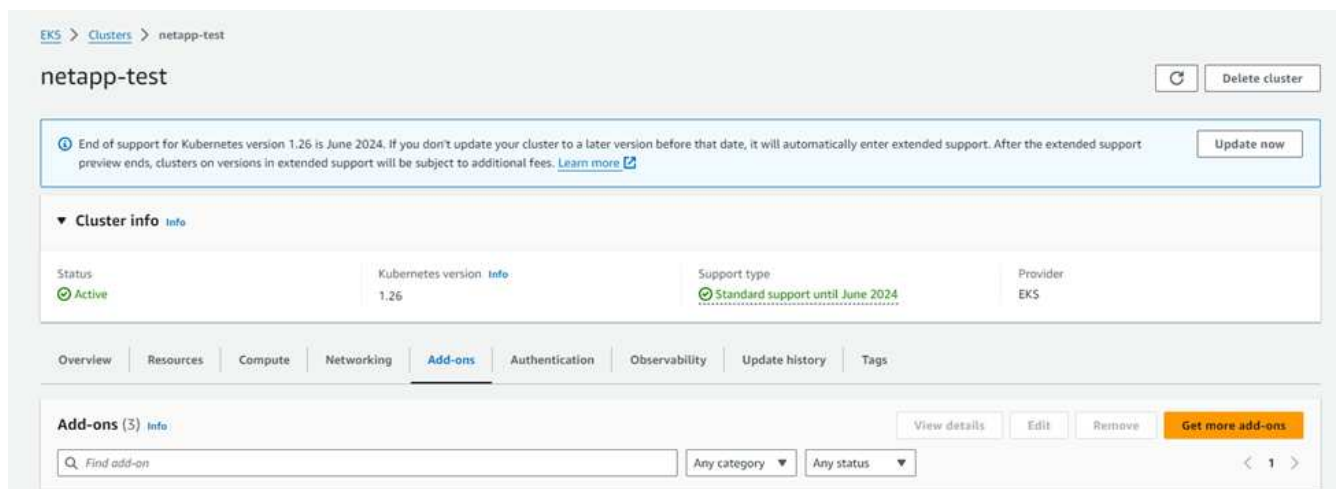
先決條件

在設定 AWS EKS 的 Astra Trident 附加元件之前、請確定您具有下列項目：

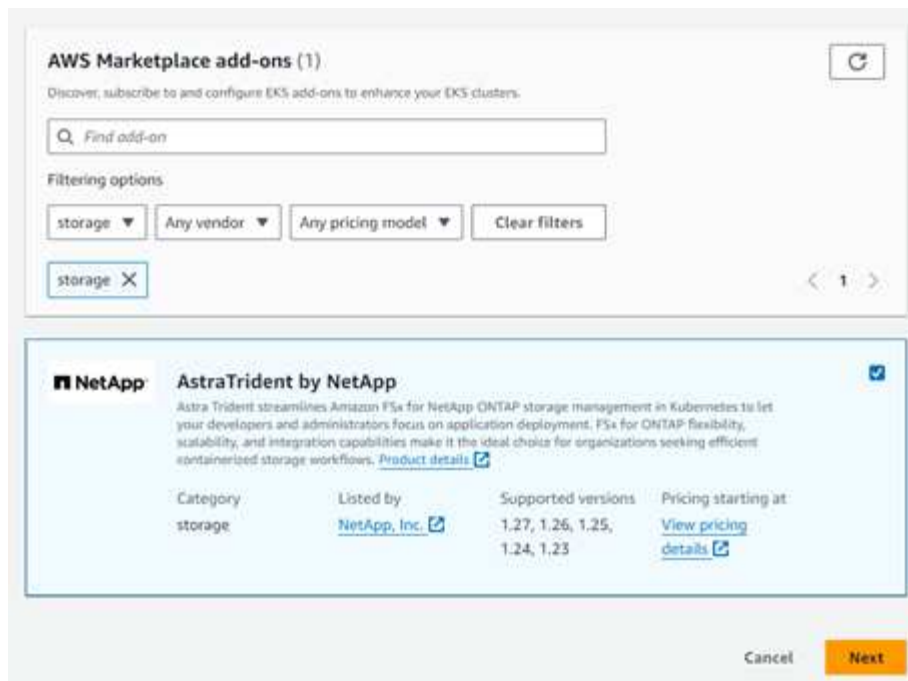
- 具有附加訂閱的 Amazon EKS 叢集帳戶
- AWS 對 AWS 市場的權限：
"aws-marketplace:ViewSubscriptions",
"aws-marketplace:Subscribe",
"aws-marketplace:Unsubscribe"
- AMI 類型：Amazon Linux 2 （AL2_x86_64）或 Amazon Linux 2 ARM （AL2_ARM_64）
- 節點類型：AMD 或 ARM
- 現有的 Amazon FSX for NetApp ONTAP 檔案系統

步驟

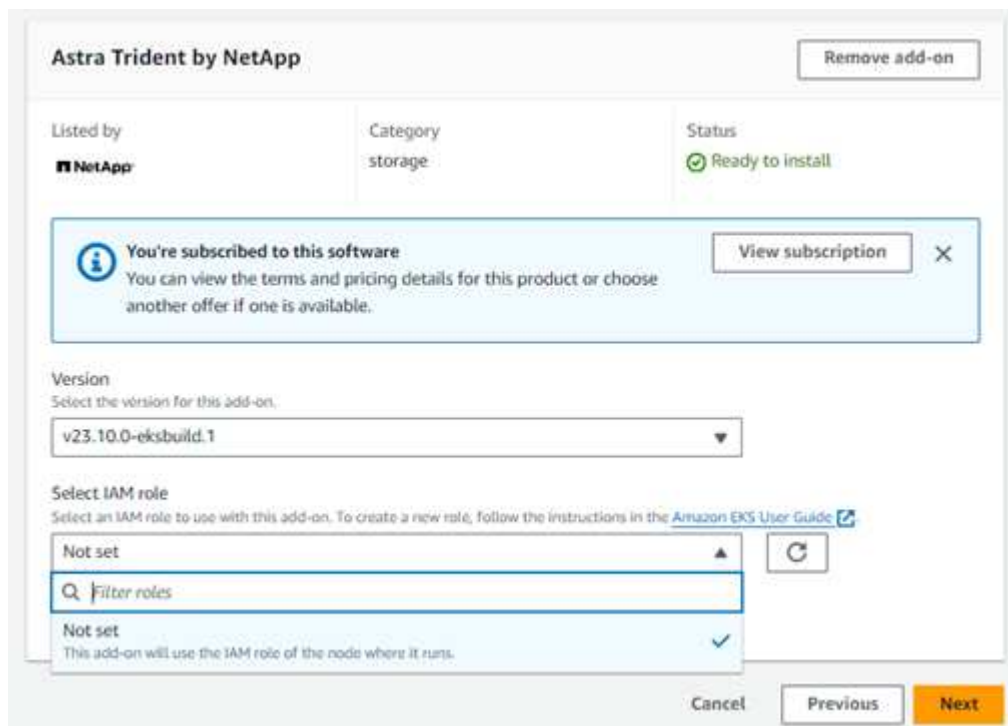
1. 在您的 EKS Kubernetes 叢集上、瀏覽至 * 附加元件 * 索引標籤。



2. 前往 * AWS Marketplace 附加元件 * 並選擇 _storage 類別。



3. 找到 * > NetApp * 的「答案」、然後選取 Astra Trident 附加元件的核取方塊。
4. 選擇所需版本的附加元件。



5. 選取 IAM 角色選項以從節點繼承。
6. 視需要設定任何選用設定、然後選取 * 下一步 * 。

Review and add

Step 1: Select add-ons Edit

Selected add-ons

Find add-on

Add-on name	Type	Status
netapp_trident-operator	storage	Ready to install

Step 2: Configure selected add-ons settings Edit

Selected add-ons version

Add-on name	Version	IAM role
netapp_trident-operator	v23.10.0-eksbuild.1	Inherit from node

Cancel Previous **Create**

7. 選擇* Create（建立）。
8. 確認附加元件的狀態為 *Active*。

Add-ons (1) View details Edit Remove Get more add-ons

Find add-on Any category Any status

Category	Status	Version	IAM role	Created by
storage	Active	v23.10.0-eksbuild.1	Inherited from node	NetApp, Inc.

使用 CLI 安裝 / 解除安裝 Astra Trident EKS 附加元件

使用 CLI 安裝 Astra Trident EKS 附加元件：

下列命令範例會安裝 Astra Trident EKS 附加元件：

```
eksctl create addon --cluster K8s-arm --name netapp_trident-operator --version v23.10.0-eksbuild.1
eksctl create addon --cluster K8s-arm --name netapp_trident-operator --version v23.10.0-eksbuild.1 （使用專用版本）
```

使用 CLI 解除安裝 Astra Trident EKS 附加元件：

下列命令會解除安裝 Astra Trident EKS 附加元件：

```
eksctl delete addon --cluster K8s-arm --name netapp_trident-operator
```

使用kubecI建立後端

後端定義了Astra Trident與儲存系統之間的關係。它告訴Astra Trident如何與該儲存系統通訊、以及Astra Trident如何從該儲存系統配置磁碟區。安裝Astra Trident之後、下一步是建立後端。「TridentBackendConfig」自訂資源定義（CRD）可讓您直接透過Kubernetes介

面建立及管理Trident後端。您可以使用「kubectl」或Kubernetes發佈的等效CLI工具來執行此作業。

TridentBackendConfig

「TridentBackendConfig」（「tbc」、「tbconfig」、「tbackendconfig」）是前端、名稱為CRD、可讓您使用「kubectl」來管理Astra Trident後端。Kubernetes與儲存管理員現在可以直接透過Kubernetes CLI建立及管理後端、而不需要使用專屬的命令列公用程式（「tridentctl」）。

建立「TridentBackendConfig」物件之後、會發生下列情況：

- Astra Trident會根據您提供的組態自動建立後端。內部代表的是「TridentBackend」（「tbc」、「tridentbackend」）、CR。
- 《TridentBackendConfig》與由Astra Trident所建立的《TridentBackend》有獨特的關聯。

每個「TridentBackendConfig」都有一對一的對應、並有「TridentBackend」。前者是提供給使用者設計及設定後端的介面、後者是Trident代表實際後端物件的方式。



「TridentBackend」CRS是由Astra Trident自動建立的。您*不應該*修改這些項目。如果您想要更新後端、請修改「TridentBackendConfig」物件來執行此動作。

請參閱下列範例、以瞭解「TridentBackendConfig」CR的格式：

```
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-tbc-ontap-san
spec:
  version: 1
  backendName: ontap-san-backend
  storageDriverName: ontap-san
  managementLIF: 10.0.0.1
  dataLIF: 10.0.0.2
  svm: trident_svm
  credentials:
    name: backend-tbc-ontap-san-secret
```

您也可以查看中的範例 ["Trident安裝程式"](#) 所需儲存平台/服務的範例組態目錄。

◦ spec 採用後端特定的組態參數。在此範例中、後端使用 ontap-san 儲存驅動程式、並使用此處列出的組態參數。如需所需儲存驅動程式的組態選項清單、請參閱 ["儲存驅動程式的後端組態資訊"](#)。

在《TridentBackendConfig》（CRR）中新推出的「sPEC」一節也包含「認證」和「刪除原則」欄位：

- 「認證資料」：此參數為必填欄位、包含用於驗證儲存系統/服務的認證資料。此設定為使用者建立的Kubernetes Secret。認證資料無法以純文字格式傳遞、因此會產生錯誤。
- 「刪除原則」：此欄位可定義刪除「TridentBackendConfig」時應發生的情況。可能需要兩種可能的值之一

:

- 「刪除」：這會同時刪除「TridentBackendConfig」和相關後端。這是預設值。
- 「保留」：刪除「TridentBackendConfig」（TridentBackendConfig）CR時、後端定義仍會存在、並可使用「tridentctl」進行管理。將刪除原則設為「保留」可讓使用者降級至較早版本（21.04之前）、並保留建立的後端。此欄位的值可在建立「TridentBackendConfig」之後更新。



後端名稱是使用「sPEC.backendName」來設定。如果未指定、則會將後端名稱設為「TridentBackendConfig」物件（metadata.name）的名稱。建議使用「sPEC.backendName」明確設定後端名稱。



使用「tridentctl」建立的後端沒有關聯的「TridentBackendConfig」物件。您可以建立「TridentBackendConfig」（TridentBackendConfig）的CR、選擇以「kubectll」管理這類後端。必須謹慎指定相同的組態參數（例如「s.pec.backendName」、「sec.storagePrefix」、「sPEec.storageDriverName」等）。Astra Trident會自動將新建立的「TridentBackendConfig」連結至預先存在的後端。

步驟總覽

若要使用「kubectll」建立新的後端、您應該執行下列動作：

1. 建立 "**Kubernetes機密**"。此機密包含Astra Trident與儲存叢集/服務通訊所需的認證資料。
2. 建立「TridentBackendConfig」物件。其中包含有關儲存叢集/服務的詳細資訊、並參考上一步建立的機密。

建立後端之後、您可以使用「kubectl Get tbc <tbc-name>-n <trident命名空間>」來觀察其狀態、並收集其他詳細資料。

步驟1：建立Kubernetes機密

建立包含後端存取認證的秘密。這是每個儲存服務/平台所獨有的功能。範例如下：

```
kubectll -n trident create -f backend-tbc-ontap-san-secret.yaml
apiVersion: v1
kind: Secret
metadata:
  name: backend-tbc-ontap-san-secret
type: Opaque
stringData:
  username: cluster-admin
  password: t@Ax@7q(>
```

下表摘要說明每個儲存平台的機密必須包含的欄位：

儲存平台機密欄位說明	秘密	欄位說明
Azure NetApp Files	ClientID	應用程式註冊的用戶端ID

儲存平台機密欄位說明	秘密	欄位說明
適用於 GCP Cloud Volumes Service	Private金鑰ID	私密金鑰的ID。GCP服務帳戶API金鑰的一部分、具有CVS管理員角色
適用於 GCP Cloud Volumes Service	Private金鑰	私密金鑰：GCP服務帳戶API金鑰的一部分、具有CVS管理員角色
元素（NetApp HCI / SolidFire）	端點	MVIP、適用於SolidFire 採用租戶認證的不含用戶身分證明的叢集
ONTAP	使用者名稱	連線至叢集/ SVM的使用者名稱。用於認證型驗證
ONTAP	密碼	連線至叢集/ SVM的密碼。用於認證型驗證
ONTAP	用戶端權限金鑰	用戶端私密金鑰的Base64編碼值。用於憑證型驗證
ONTAP	chap使用 者名稱	傳入使用者名稱。如果useCHAP=true則需要。適用於「ONTAP-SAN」和「ONTAP-san經濟」
ONTAP	chapInitiator機密	CHAP啟動器密碼。如果useCHAP=true則需要。適用於「ONTAP-SAN」和「ONTAP-san經濟」
ONTAP	chapTargetUsername	目標使用者名稱。如果useCHAP=true則需要。適用於「ONTAP-SAN」和「ONTAP-san經濟」
ONTAP	chapTargetInitiator機密	CHAP目標啟動器機密。如果useCHAP=true則需要。適用於「ONTAP-SAN」和「ONTAP-san經濟」

在此步驟中建立的機密會參照下一步所建立之「TridentBackendConfig」物件的「sapec.ecent」欄位。

步驟2：建立 TridentBackendConfig CR

您現在可以建立「TridentBackendConfig」的CR了。在此範例中、使用「ONTAP-SAN」驅動程式的後端是使用「TridentBackendConfig」物件建立、如下所示：

```
kubectl -n trident create -f backend-tbc-ontap-san.yaml
```

```
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-tbc-ontap-san
spec:
  version: 1
  backendName: ontap-san-backend
  storageDriverName: ontap-san
  managementLIF: 10.0.0.1
  dataLIF: 10.0.0.2
  svm: trident_svm
  credentials:
    name: backend-tbc-ontap-san-secret
```

步驟3：確認的狀態 TridentBackendConfig CR

現在您已經建立了「TridentBackendConfig」（TridentBackendConfig）CR、您就可以驗證其狀態。請參閱下列範例：

```
kubectl -n trident get tbc backend-tbc-ontap-san
```

NAME	BACKEND	NAME	BACKEND	UUID
PHASE	STATUS			
backend-tbc-ontap-san	ontap-san-backend			8d24fce7-6f60-4d4a-8ef6-bab2699e6ab8
Bound	Success			

已成功建立後端、並連結至「TridentBackendConfig」CR。

階段可以採用下列其中一個值：

- Bound：TridentBackendConfig CR與後端相關聯、且後端包含 configRef 設定為 TridentBackendConfig CR 的 uid。
- 《Unbound》：使用「」表示。「TridentBackendConfig」物件不會繫結至後端。根據預設、所有新建立的「TridentBackendConfig」CRS均處於此階段。階段變更之後、就無法再恢復為Unbound（未綁定）。
- Deleting：TridentBackendConfig CR 的 deletionPolicy 已設定為刪除。當 TridentBackendConfig 系統會刪除CR、並轉換為「刪除」狀態。
 - 如果後端不存在持續磁碟區宣告（PVCS）、刪除「TridentBackendConfig」（TridentBackendConfig）會導致Astra Trident刪除後端、以及刪除「TridentBackendConfig」（TridentBackendConfig）。
 - 如果後端上有一個或多個PVCS、則會進入刪除狀態。隨後、「TridentBackendConfig」CR也會進入刪除階段。只有刪除所有的PVCS之後、才會刪除後端和「TridentBackendConfig」。
- 「遺失」：與「TridentBackendConfig」CR相關的後端意外或刻意刪除、而「TridentBackendConfig」CR

仍有刪除後端的參考資料。無論「刪除原則」值為何、「TridentBackendConfig」CR仍可刪除。

- Unknown：Astra Trident無法判斷與相關聯的後端狀態或存在 TridentBackendConfig CR.例如、如果API伺服器沒有回應或是 tridentbackends.trident.netapp.io CRD遺失。這可能需要介入。

在此階段、成功建立後端！還有多種作業可以額外處理、例如 "[後端更新和後端刪除](#)"。

（選用）步驟4：取得更多詳細資料

您可以執行下列命令來取得有關後端的詳細資訊：

```
kubectl -n trident get tbc backend-tbc-ontap-san -o wide
```

NAME	BACKEND NAME	BACKEND UUID	
PHASE	STATUS	STORAGE DRIVER	DELETION POLICY
backend-tbc-ontap-san	ontap-san-backend	8d24fce7-6f60-4d4a-8ef6-bab2699e6ab8	Bound Success ontap-san delete

此外、您也可以取得「TridentBackendConfig」的YAML/Json傾印。

```
kubectl -n trident get tbc backend-tbc-ontap-san -o yaml
```

```

apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  creationTimestamp: "2021-04-21T20:45:11Z"
  finalizers:
  - trident.netapp.io
  generation: 1
  name: backend-tbc-ontap-san
  namespace: trident
  resourceVersion: "947143"
  uid: 35b9d777-109f-43d5-8077-c74a4559d09c
spec:
  backendName: ontap-san-backend
  credentials:
    name: backend-tbc-ontap-san-secret
  managementLIF: 10.0.0.1
  dataLIF: 10.0.0.2
  storageDriverName: ontap-san
  svm: trident_svm
  version: 1
status:
  backendInfo:
    backendName: ontap-san-backend
    backendUUID: 8d24fce7-6f60-4d4a-8ef6-bab2699e6ab8
  deletionPolicy: delete
  lastOperationStatus: Success
  message: Backend 'ontap-san-backend' created
  phase: Bound

```

《backendInfo》 (backendInfo) 包含了針對「TridentBackendConfig」 (TridentBackendConfig) 的CR而建立的後端「backendName」 (背端名稱) 和「backendUUID」 (背端UUID)。「lastoperationStatus」欄位代表上次執行的「TridentBackendConfig」 (TridentBackendConfig) CR狀態、可由使用者觸發 (例如、使用者在「show」中變更內容)、或由Astra Trident觸發 (例如、Astra Trident重新啟動期間)。可能是「成功」或「失敗」。「階段」代表「TridentBackendConfig」與後端之間的關係狀態。在上述範例中、「階段」具有界限值、這表示「TridentBackendConfig」CR與後端相關聯。

您可以執行「`kubectl -n triident描述tbc <tbc-cr-name>`」命令、以取得事件記錄的詳細資料。



您無法使用「tridentctl」來更新或刪除包含相關「TridentBackendConfig」物件的後端。若要瞭解在「tridentctl」和「TridentBackendConfig」之間切換的步驟、[請參閱此處](#)。

管理後端

以KECBECVL執行後端管理

瞭解如何使用「kubectl」來執行後端管理作業。

刪除後端

刪除「TridentBackendConfig」（TridentBackendConfig）之後、即指示Astra Trident刪除/保留後端（根據「刪除原則」）。若要刪除後端、請確定「刪除原則」已設定為刪除。如果只要刪除「TridentBackendConfig」、請確定「刪除原則」已設定為保留。這可確保後端仍存在、並可使用「tridentctl」進行管理。

執行下列命令：

```
kubectl delete tbc <tbc-name> -n trident
```

Astra Trident並不會刪除「TridentBackendConfig」所使用的Kubernetes Secrets。Kubernetes使用者負責清除機密。刪除機密時必須小心。只有在後端未使用機密時、才應刪除這些機密。

檢視現有的後端

執行下列命令：

```
kubectl get tbc -n trident
```

您也可以執行「tridentctl Get backend -n trident」或「tridentctl Get backend -o yaml -n trident」、以取得所有後端的清單。這份清單也會包含以「tridentctl」建立的後端。

更新後端

更新後端可能有多種原因：

- 儲存系統的認證資料已變更。若要更新認證資料、必須更新「TridentBackendConfig」物件中使用的Kubernetes Secret。Astra Trident會自動以提供的最新認證資料更新後端。執行下列命令以更新Kubernetes Secret：

```
kubectl apply -f <updated-secret-file.yaml> -n trident
```

- 需要ONTAP 更新參數（例如使用的SVM名稱）。
 - 您可以更新 TridentBackendConfig 使用下列命令直接透過 Kubernetes 執行物件：

```
kubectl apply -f <updated-backend-file.yaml>
```

- 或者、您也可以變更現有的 TridentBackendConfig 使用下列命令的 CR：

```
kubectl edit tbc <tbc-name> -n trident
```



- 如果後端更新失敗、後端仍會繼續維持其最後已知的組態。您可以執行「`kubectl Get tbc <tbc-name>-o yaml -n trident`」或「`kubectl 描述 tbc <tbc-name>-n trident`」來檢視記錄以判斷原因。
- 識別並修正組態檔的問題之後、即可重新執行update命令。

使用tridentctl執行後端管理

瞭解如何使用「tridentctl」來執行後端管理作業。

建立後端

建立之後 "後端組態檔"，執行下列命令：

```
tridentctl create backend -f <backend-file> -n trident
```

如果後端建立失敗、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 -n trident
```

識別並修正組態檔的問題之後、您只需再次執行「create」命令即可。

刪除後端

若要從Astra Trident刪除後端、請執行下列步驟：

1. 擷取後端名稱：

```
tridentctl get backend -n trident
```

2. 刪除後端：

```
tridentctl delete backend <backend-name> -n trident
```



如果Astra Trident已從這個後端配置磁碟區和快照、但該後端仍存在、則刪除後端會使新的磁碟區無法由其進行資源配置。後端將繼續處於「刪除」狀態、而Trident將繼續管理這些磁碟區和快照、直到它們被刪除為止。

檢視現有的後端

若要檢視Trident知道的後端、請執行下列步驟：

- 若要取得摘要、請執行下列命令：

```
tridentctl get backend -n trident
```

- 若要取得所有詳細資料、請執行下列命令：

```
tridentctl get backend -o json -n trident
```

更新後端

建立新的後端組態檔之後、請執行下列命令：

```
tridentctl update backend <backend-name> -f <backend-file> -n trident
```

如果後端更新失敗、表示後端組態有問題、或是您嘗試了無效的更新。您可以執行下列命令來檢視記錄、以判斷原因：

```
tridentctl logs -n trident
```

識別並修正組態檔的問題之後、您只需再次執行「update」命令即可。

識別使用後端的儲存類別

這是您可以用Json回答的問題類型範例、其中的「tridentctl」會輸出後端物件。這會使用您需要安裝的「jq」公用程式。

```
tridentctl get backend -o json | jq '[.items[] | {backend: .name, storageClasses: [.storage[].storageClasses]|unique}]'
```

這也適用於使用「TridentBackendConfig」建立的後端。

在後端管理選項之間切換

瞭解Astra Trident管理後端的不同方法。

管理後端的選項

隨之推出「TridentBackendConfig」管理員現在有兩種獨特的後端管理方法。這會提出下列問題：

- 使用「tridentctl」建立的後端、是否能以「TridentBackendConfig」來管理？
- 使用「TridentBackendConfig」建立的後端、是否可以使用「tridentctl」來管理？

管理 tridentctl 後端使用 TridentBackendConfig

本節說明透過Kubernetes介面建立「TridentBackendConfig」物件、直接透過「tridentctl」建立的後端管理所需的步驟。

這將適用於下列案例：

- 沒有的既有後端 TridentBackendConfig 因為它們是使用建立的 tridentctl。
- 使用「tridentctl」建立的新後端、而其他「TridentBackendConfig」物件則存在。

在這兩種情況下、後端仍會繼續存在、Astra Trident排程磁碟區會繼續運作。系統管理員有兩種選擇之一：

- 繼續使用「tridentctl」來管理使用它建立的後端。
- 將使用「tridentctl」建立的後端連結至新的「TridentBackendConfig」物件。這樣做將意味着後端將使用“kubedl”而不是“tridentctl”來管理。

若要使用「kubedl」管理預先存在的後端、您需要建立連結至現有後端的「TridentBackendConfig」。以下是如何運作的總覽：

1. 建立Kubernetes機密。此機密包含Astra Trident與儲存叢集/服務通訊所需的認證資料。
2. 建立「TridentBackendConfig」物件。其中包含有關儲存叢集/服務的詳細資訊、並參考上一步建立的機密。必須謹慎指定相同的組態參數（例如「s.pec.backendName」、「sec.storagePrefix」、「sPEec.storageDriverName」等）。必須將「Pec.backendName」設定為現有後端的名稱。

步驟0：識別後端

以建立 TridentBackendConfig 若要連結至現有的後端、您必須取得後端組態。在此範例中、假設使用下列Json定義建立後端：

```
tridentctl get backend ontap-nas-backend -n trident
+-----+-----+
+-----+-----+-----+-----+
|          NAME          | STORAGE DRIVER |          UUID          |
| STATE  | VOLUMES |          |
+-----+-----+-----+-----+
+-----+-----+-----+-----+
| ontap-nas-backend    | ontap-nas      | 52f2eb10-e4c6-4160-99fc-96b3be5ab5d7 |
| online |      25 |          |
+-----+-----+-----+-----+
+-----+-----+-----+-----+

cat ontap-nas-backend.json

{
  "version": 1,
```

```

"storageDriverName": "ontap-nas",
"managementLIF": "10.10.10.1",
"dataLIF": "10.10.10.2",
"backendName": "ontap-nas-backend",
"svm": "trident_svm",
"username": "cluster-admin",
"password": "admin-password",

"defaults": {
  "spaceReserve": "none",
  "encryption": "false"
},
"labels":{"store":"nas_store"},
"region": "us_east_1",
"storage": [
  {
    "labels":{"app":"msoffice", "cost":"100"},
    "zone":"us_east_1a",
    "defaults": {
      "spaceReserve": "volume",
      "encryption": "true",
      "unixPermissions": "0755"
    }
  },
  {
    "labels":{"app":"mysqldb", "cost":"25"},
    "zone":"us_east_1d",
    "defaults": {
      "spaceReserve": "volume",
      "encryption": "false",
      "unixPermissions": "0775"
    }
  }
]
}

```

步驟1：建立Kubernetes機密

建立包含後端認證的秘密、如以下範例所示：

```
cat tbc-ontap-nas-backend-secret.yaml

apiVersion: v1
kind: Secret
metadata:
  name: ontap-nas-backend-secret
type: Opaque
stringData:
  username: cluster-admin
  password: admin-password

kubectl create -f tbc-ontap-nas-backend-secret.yaml -n trident
secret/backend-tbc-ontap-san-secret created
```

步驟2：建立 TridentBackendConfig CR

下一步是建立一個「TridentBackendConfig」（TridentBackendConfig）CR、它會自動連結至現有的「ONTAP-NAS-backend」（如本範例所示）。確保符合下列要求：

- 相同的後端名稱是在「s.pec.backendName」中定義。
- 組態參數與原始後端相同。
- 虛擬資源池（若有）必須維持與原始後端相同的順序。
- 認證資料是透過Kubernetes Secret提供、而非以純文字提供。

在這種情況下、「TridentBackendConfig」將會如下所示：

```

cat backend-tbc-ontap-nas.yaml
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: tbc-ontap-nas-backend
spec:
  version: 1
  storageDriverName: ontap-nas
  managementLIF: 10.10.10.1
  dataLIF: 10.10.10.2
  backendName: ontap-nas-backend
  svm: trident_svm
  credentials:
    name: mysecret
  defaults:
    spaceReserve: none
    encryption: 'false'
  labels:
    store: nas_store
  region: us_east_1
  storage:
  - labels:
      app: msoffice
      cost: '100'
      zone: us_east_1a
      defaults:
        spaceReserve: volume
        encryption: 'true'
        unixPermissions: '0755'
  - labels:
      app: mysqldb
      cost: '25'
      zone: us_east_1d
      defaults:
        spaceReserve: volume
        encryption: 'false'
        unixPermissions: '0775'

kubectl create -f backend-tbc-ontap-nas.yaml -n trident
tridentbackendconfig.trident.netapp.io/tbc-ontap-nas-backend created

```

步驟3：確認的狀態 TridentBackendConfig **CR**

在建立「TridentBackendConfig」之後、其階段必須是「綁定」。它也應反映與現有後端相同的後端名稱和UUID。

```
kubectl get tbc tbc-ontap-nas-backend -n trident
```

NAME	BACKEND NAME	BACKEND UUID
tbc-ontap-nas-backend	ontap-nas-backend	52f2eb10-e4c6-4160-99fc-96b3be5ab5d7
Bound	Success	

#confirm that no new backends were created (i.e., TridentBackendConfig did not end up creating a new backend)

```
tridentctl get backend -n trident
```

NAME	STORAGE DRIVER	UUID
ontap-nas-backend	ontap-nas	52f2eb10-e4c6-4160-99fc-96b3be5ab5d7
online	25	

現在可以使用「tbc-ontap-nas-backend」「TridentBackendConfig」物件來完全管理後端。

管理 TridentBackendConfig 後端使用 tridentctl

可以使用「tridentctl」來列出使用「TridentBackendConfig」建立的後端。此外、系統管理員也可以刪除「TridentBackendConfig」、並確定「pec.deletionPolicy」設為「效能」、藉此選擇透過「tridentctl」來完全管理此類後端。

步驟0：識別後端

例如、假設使用「TridentBackendConfig」建立下列後端：

```
kubectl get tbc backend-tbc-ontap-san -n trident -o wide
```

NAME	BACKEND NAME	BACKEND UUID
backend-tbc-ontap-san	ontap-san-backend	81abcb27-ea63-49bb-b606-0a5315ac5f82

```
tridentctl get backend ontap-san-backend -n trident
```

NAME	STORAGE DRIVER	UUID
ontap-san-backend	ontap-san	81abcb27-ea63-49bb-b606-0a5315ac5f82

從輸出中可以看出這一點 TridentBackendConfig 已成功建立並繫結至後端 [觀察後端的 UUID] 。

步驟1：確認 deletionPolicy 設為 retain

讓我們來看看「改革政策」的價值。這需要設定為「維護」。這將確保刪除「TridentBackendConfig」(TridentBackendConfig) 的CR時、後端定義仍會存在、而且可以使用「tridentctl」進行管理。

```
kubectl get tbc backend-tbc-ontap-san -n trident -o wide
```

NAME	BACKEND NAME	BACKEND UUID
backend-tbc-ontap-san	ontap-san-backend	81abcb27-ea63-49bb-b606-0a5315ac5f82

```
# Patch value of deletionPolicy to retain
kubectl patch tbc backend-tbc-ontap-san --type=merge -p
'{"spec":{"deletionPolicy":"retain"}}' -n trident
tridentbackendconfig.trident.netapp.io/backend-tbc-ontap-san patched

#Confirm the value of deletionPolicy
kubectl get tbc backend-tbc-ontap-san -n trident -o wide
```

NAME	BACKEND NAME	BACKEND UUID
backend-tbc-ontap-san	ontap-san-backend	81abcb27-ea63-49bb-b606-0a5315ac5f82



除非將「刪除原則」設定為「需要」、否則請勿繼續下一步。

步驟2：刪除 TridentBackendConfig CR

最後一個步驟是刪除「TridentBackendConfig」（TridentBackendConfig）。確認「刪除原則」設為「保留」之後、您可以繼續刪除：

```
kubectl delete tbc backend-tbc-ontap-san -n trident
tridentbackendconfig.trident.netapp.io "backend-tbc-ontap-san" deleted

tridentctl get backend ontap-san-backend -n trident
+-----+-----+
+-----+-----+-----+-----+
|          NAME          | STORAGE DRIVER |                      UUID                      |
| STATE  | VOLUMES |                      |
+-----+-----+-----+-----+
+-----+-----+-----+-----+
| ontap-san-backend | ontap-san      | 81abcb27-ea63-49bb-b606-0a5315ac5f82 |
| online |      33 |                      |
+-----+-----+-----+-----+
+-----+-----+-----+-----+
```

刪除「TridentBackendConfig」物件之後、Astra Trident便會移除該物件、而不會實際刪除後端本身。

版權資訊

Copyright © 2024 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。