保護 **Kubernetes** 工作負載

NetApp Backup and Recovery

NetApp
June 24, 2026

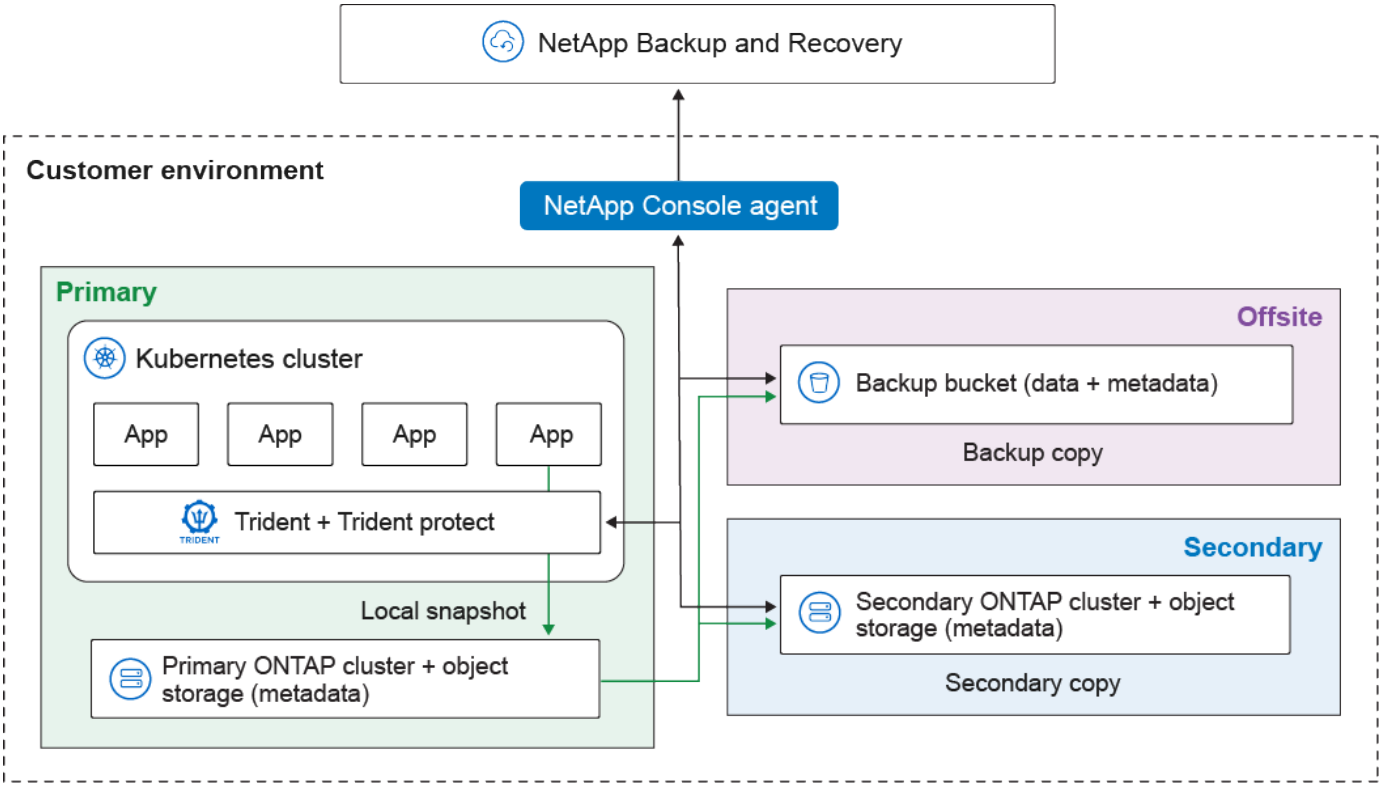
目錄

保護 Kubernetes 工作負載

管理 Kubernetes 工作負載概覽

在 NetApp Backup and Recovery 中管理 Kubernetes 工作負載，讓您能夠在一個平台上發現、管理和保護 Kubernetes 叢集和應用程式。您可以管理託管在 Kubernetes 叢集上的資源和應用程式。還可以建立保護原則並將其與 Kubernetes 應用程式關聯，所有操作均可透過單一介面完成。

下圖展示了 Kubernetes 工作負載的備份和還原的元件和基本架構，以及如何將資料的不同副本儲存在不同位置：



NetApp Backup and Recovery為管理 Kubernetes 工作負載提供了以下優點：

- 單一控制平面，用於保護跨多個 Kubernetes 叢集運行的應用程式。這些應用程式可以包括在 Kubernetes 叢集上執行的容器或虛擬機器。
- 與NetApp SnapMirror本機集成，為所有備份和恢復工作流程提供儲存卸載功能。
- Kubernetes 應用程式的永久增量備份，轉換為更低的復原點目標 (RPO) 和復原時間目標 (RTO)。

您可以完成與管理 Kubernetes 工作負載相關的下列任務：

- "發現 Kubernetes 工作負載"。
- "管理 Kubernetes 集群"。
- "新增和保護 Kubernetes 應用程式"。

- "管理 Kubernetes 應用程式"。
- "恢復 Kubernetes 應用程式"。

探索NetApp Backup and Recovery中的 Kubernetes 工作負載

NetApp Backup and Recovery需要在保護 Kubernetes 工作負載之前發現它們。

所需的**NetApp Console**角色 備份和還原超級管理員。了解詳情"備份和恢復角色和權限"。 "了解所有服務的NetApp Console存取角色" 。

發現 **Kubernetes** 工作負載

在備份和復原清單中，發現您環境中的 Kubernetes 工作負載。新增工作負載會將 Kubernetes 叢集新增至NetApp Backup and Recovery。然後，您可以新增應用程式並保護叢集資源。

> 當您發現目前受 Trident Protect 保護的叢集時，所有與 Trident Protect 一起使用的備份計畫都會在發現過程中停用（Trident Protect 備份計畫與 Backup and Recovery 不相容）。若要保護叢集中的應用程式，"建立新的保護原則"或將應用程式與現有原則建立關聯。之後，您可以視需要移除 Trident Protect 備份計畫。

步驟

1. 執行下列操作之一：
   - 如果您是第一次發現 Kubernetes 工作負載，請在NetApp Backup and Recovery中，在「工作負載」下，選擇「Kubernetes」磁貼。
   - 如果您已經發現 Kubernetes 工作負載，請在NetApp Backup and Recovery中選擇 **Inventory** > **Workloads**，然後選擇 **Discover resources**。
2. 選擇 **Kubernetes** 工作負載類型。
3. 輸入叢集名稱並選擇與叢集一起使用的連接器。
4. 按照出現的命令列說明進行操作：
   - 建立 Trident Protect 命名空間
   - 建立 Kubernetes 機密
   - 新增 Helm 儲存庫
   - 安裝或升級 Trident Protect 和 Trident Protect 連接器

     這些步驟確保NetApp Backup and Recovery可以與叢集互動。
5. 完成這些步驟後，選擇*發現*。

   該集群已新增至清單。
6. 在關聯的 Kubernetes 工作負載中選擇「檢視」以查看該工作負載的應用程式、叢集和命名空間清單。

繼續存取**NetApp Backup and Recovery**儀表板

請依照以下步驟查看NetApp Backup and Recovery儀表板。

1. 從NetApp Console選單中，選擇 保護 > 備份和還原。
2. 選擇一個工作負載圖塊（例如，Microsoft SQL Server）。
3. 從備份和還原選單中，選擇*儀表板*。
4. 檢討資料保護的健康狀況。處於危險中或受保護的工作負載的數量會根據新發現、受保護和備份的工作負載而增加。

   "了解儀表板顯示的內容"。

新增和保護 **Kubernetes** 應用程式

新增和保護 **Kubernetes** 應用程式

NetApp Backup and Recovery 可讓您透過網頁 UI 或套用自訂資源檔案來新增 Kubernetes 應用程式。應用程式可以是以命名空間為基礎，由標準 Kubernetes 資源組成，或是以 VM 為基礎，由一個或多個虛擬機組成。

所需的**NetApp Console**角色

組織管理員或SnapCenter管理員。"了解NetApp Backup and Recovery存取角色" 。 "了解所有服務的NetApp Console存取角色" 。

新增並保護新的 **Kubernetes** 應用程式

保護 Kubernetes 應用程式的第一步是在 NetApp Backup and Recovery 中建立應用程式。建立應用程式後，Backup and Recovery 功能就能辨識 Kubernetes 叢集上正在執行的應用程式。

開始之前

在新增和保護 Kubernetes 應用程式之前，您需要"發現 Kubernetes 工作負載"。

新增基於命名空間的應用程式（**Web UI**）

步驟

1. 在NetApp Backup and Recovery中，選擇 **Inventory**。
2. 在頁面右上角，確保工作負載清單中已選擇 **Kubernetes**。
3. 選取工作負載行項目的 **View** 以檢視 Kubernetes 資源。
4. 選擇“應用程式”標籤。
5. 選擇*建立應用程式*。
6. 輸入應用程式的名稱。
7. 從 **Cluster** 清單中，選擇託管應用程式的叢集。
8. 在 **Filters** 下，選擇 **Namespace** 以按命名空間篩選應用程式。
9. 或者，選擇以下任意欄位來搜尋您想要保護的資源：
   - 關聯的命名空間
   - 資源類型
   - 標籤選擇器
     i. 選擇 **Add cluster-scoped resources** 可新增叢集層級的資源。如果選擇新增，這些資源將在建立應用程式時新增到應用程式中。
     ii. 或者，選擇“搜尋”以根據您的搜尋條件尋找資源。

     Backup and Recovery 不會儲存搜尋參數或結果；這些參數用於在選定的 Kubernetes 叢集中搜尋可以包含在應用程式中的資源。
10. Backup and Recovery 會顯示符合您搜尋條件的資源清單。
11. 如果清單包含您想要保護的資源，請選擇「下一步」。
12. 您也可以在「策略」區域中選擇現有的保護策略來保護應用程式，或建立新策略。如果您未選擇策略，則建立的應用程式將沒有保護策略。您可以稍後"新增保護策略"。
13. 在*Prescripts and postscripts*區域中，啟用並配置您想要在備份操作之前或之後執行的任何prescript或postscript執行掛鉤。若要啟用處方或附言，您必須至少已建立了一個"執行鉤子模板"。
14. 選擇“創建”。

結果

應用程式建立完成後，將顯示在 Kubernetes 清單的 **Applications** 標籤中的應用程式清單中。Backup and Recovery 可根據您的設定保護應用程式，您可以在 **Monitoring** 區域中監控進度。

新增基於虛擬機器的應用程式（**Web UI**）

步驟

1. 在NetApp Backup and Recovery中，選擇 **Inventory**。
2. 在頁面右上角，確保工作負載清單中已選擇 **Kubernetes**。
3. 選取工作負載行項目的 **View** 以檢視 Kubernetes 資源。
4. 選擇“應用程式”標籤。

5. 選擇*建立應用程式*。
6. 輸入應用程式的名稱。
7. 從 **Cluster** 清單中，選擇託管應用程式的叢集。
8. 在 **Filters** 下，選擇 **Virtual machines** 以建立基於 VM 的應用程式。
9. 透過選擇命名空間並可選擇性地包含標籤選取器，尋找要新增至應用程式的虛擬機器。

> 如果您從清單中選擇 VM，則應用程式定義是靜態的——之後不會向應用程式新增新的 VM（您需要編輯應用程式才能新增和保護它們）。如果您使用標籤選擇器，則無法選擇個別 VM 或編輯產生的清單，但之後任何與選擇器相符的 VM 都會自動包含在內並受到保護。

選取的虛擬機器會顯示在右側清單中。

10. 如果清單中包含您要保護的 VM，請選取 * 下一步 * 。
11. 您也可以在「策略」區域中選擇現有的保護策略來保護應用程式，或建立新策略。如果您未選擇策略，則建立的應用程式將沒有保護策略。您可以稍後"新增保護策略"。
12. 在*Prescripts and postscripts*區域中，啟用並配置您想要在備份操作之前或之後執行的任何prescript或postscript執行掛鉤。若要啟用處方或附言，您必須至少已建立了一個"執行鉤子模板"。
13. 選擇“創建”。

結果

應用程式已建立並出現在 Kubernetes 清單的 應用程式 標籤中的應用程式清單中。 NetApp Console會根據您的設定啟用對應用程式的保護，並且您可以在備份和還原的*監控*區域中監控進度。

新增基於命名空間的應用程式（**CR**）

步驟

1. 建立目的地應用程式 CR 檔案：
   a. 建立自訂資源（CR）檔案並將其命名為（例如、 `my-app-name.yaml`）。
   b. 設定下列屬性：
      - **metadata.name**：（必填）應用程式自訂資源的名稱。請記住您選擇的名稱，因為保護作業所需的其他 CR 檔案會參照此值。
      - **spec.includedNamespaces**：（必要）使用命名空間和標籤選擇器來指定應用程式使用的命名空間和資源。應用程式命名空間必須包含在此清單中。標籤選擇器是可選的，可用來篩選每個指定命名空間中的資源。
      - **spec.includedClusterScopedResources**：（選用）使用此屬性指定要包含在應用程式定義中的叢集範圍資源。此屬性可讓您根據資源的群組、版本、類型和標籤來選擇這些資源。
        - **groupVersionKind**：（必要）指定叢集範圍資源的 API 群組、版本和類型。
        - **labelSelector**：（可選）依照標籤篩選叢集範圍的資源。
   c. 如有需要，請設定下列註解：
      - **metadata.annotations.protect.trident.netapp.io/skip-vm-freeze**：（可選）此註解僅適用於從虛擬機器定義的應用程式，例如在 KubeVirt 環境中，檔案系統會在快照之前凍結。指定此應用程式在快照期間是否可以寫入檔案系統。如果設定為 true，則應用程式將忽略全域設定，並且可以在快照期間寫入檔案系統。如果設定為 false，則應用程式將忽略全域設定，並且檔案系

統將在快照期間凍結。如果指定了此註解，但應用程式定義中沒有虛擬機器，則該註解將被忽略。如果未指定，則應用程式遵循 "全域檔案系統凍結設定"。

- **protect.trident.netapp.io/protection-command**：（選用）使用此註釋指示 Backup and Recovery 保護或停止保護應用程式。可能的值為 `protect`或 `unprotect`。
- **protect.trident.netapp.io/protection-policy-name**：（選用）使用此註解指定要用於保護此應用程式的 Backup and Recovery 保護原則名稱。此保護原則必須已存在於 Backup and Recovery 中。

如果需要在應用程式建立後套用此註解，可以使用以下命令：

```
kubectl annotate application -n <application CR namespace> <application CR name> protect.trident.netapp.io/skip-vm-freeze="true"
```

+
YAML 範例：

+

```
apiVersion: protect.trident.netapp.io/v1
kind: Application
metadata:
  annotations:
    protect.trident.netapp.io/skip-vm-freeze: "false"
    protect.trident.netapp.io/protection-command: "protect"
    protect.trident.netapp.io/protection-policy-name: "policy-name"
  name: my-app-name
  namespace: my-app-namespace
spec:
  includedNamespaces:
    - namespace: namespace-1
      labelSelector:
        matchLabels:
          app: example-app
    - namespace: namespace-2
      labelSelector:
        matchLabels:
          app: another-example-app
  includedClusterScopedResources:
    - groupVersionKind:
        group: rbac.authorization.k8s.io
        kind: ClusterRole
        version: v1
      labelSelector:
        matchLabels:
          mylabel: test
```

1. （選用）新增篩選條件，包含或排除標記有特定標籤的資源：

   - **resourceFilter.resourceSelectionCriteria**：（篩選必需）使用 `Include`或 `Exclude`來包含或排除在 resourceMatchers 中定義的資源。新增以下 resourceMatchers 參數以定義要包含或排除的資源：

     - **resourceFilter.resourceMatchers**：resourceMatcher 物件的陣列。如果在此陣列中定義多個元素，則它們之間按 OR 運算匹配，每個元素內的欄位（group、kind、version）之間按 AND 運算匹配。

       - **resourceMatchers[].group**：（可選）要篩選的資源群組。
       - **resourceMatchers[].kind**：（可選）要篩選的資源類型。
       - **resourceMatchers[].version**：（可選）要篩選的資源版本。

- **resourceMatchers[].names**：（可選）要過濾的資源的 Kubernetes metadata.name 欄位中的名稱。
- **resourceMatchers[].namespaces**：（可選）要篩選的資源的 Kubernetes metadata.name 欄位中的命名空間。
- **resourceMatchers[].labelSelectors**：（可選）資源在 Kubernetes metadata.name 欄位中定義的標籤選擇器字串 "Kubernetes 說明文件"。例如：`"trident.netapp.io/os=linux"`。

> 當兩者 `resourceFilter` 和 `labelSelector` 同時使用時，`resourceFilter` 首先運行，然後 `labelSelector` 將應用於生成的資源。

例如：

```
spec:
  resourceFilter:
    resourceSelectionCriteria: "Include"
    resourceMatchers:
      - group: my-resource-group-1
        kind: my-resource-kind-1
        version: my-resource-version-1
        names: ["my-resource-names"]
        namespaces: ["my-resource-namespaces"]
        labelSelectors: ["trident.netapp.io/os=linux"]
      - group: my-resource-group-2
        kind: my-resource-kind-2
        version: my-resource-version-2
        names: ["my-resource-names"]
        namespaces: ["my-resource-namespaces"]
        labelSelectors: ["trident.netapp.io/os=linux"]
```

2. 建立與您的環境相符的應用程式 CR 後、套用該 CR。例如：

```
kubectl apply -f my-app-name.yaml
```

在 **NetApp Backup and Recovery** 中建立及管理 **Kubernetes** 備份原則

在 NetApp Backup and Recovery 中，建立您自己的 Kubernetes 備份政策，以管理備份頻率、備份執行時間，以及保留的備份檔案數量。

> 其中一些選項和配置部分並不適用於所有工作負載。

如果從SnapCenter匯入資源，您可能會發現SnapCenter中使用的策略與NetApp Backup and Recovery中使用的

策略存在一些差異。看"SnapCenter與NetApp Backup and Recovery之間的策略差異"。

您可以實現以下與政策相關的目標：

- 建立本機快照策略
- 建立複製到輔助儲存的策略
- 為對象儲存設定建立策略
- 配置進階策略設定
- 編輯原則
- 刪除策略

查看政策

1. 從NetApp Backup and Recovery選單中，選擇 策略。
2. 檢閱原則詳細資料。例如：
    - 工作負載：例如 Microsoft SQL Server、ONTAP Volumes、VMware、KVM、Hyper-V、Oracle Database 或 Kubernetes。
    - 備份類型：例如完整備份和日誌備份。
    - 架構：範例包括本機快照、扇出、級聯、磁碟到磁碟和磁碟到物件儲存。
    - 受保護的資源：顯示該工作負載的總資源中有多少資源受到保護。
    - 勒索軟體保護：顯示策略是否包含本機快照上的快照鎖定、二級儲存上的快照鎖定或物件儲存上的 DataLock 鎖定。

創建策略

您可以建立策略來管理本機快照、複製到二級儲存以及備份到物件儲存。3-2-1 策略的一部分包括建立主儲存系統上的實例、資料庫、應用程式或虛擬機器的快照。

所需的**NetApp Console**角色 儲存檢視器、備份和還原超級管理員、備份和還原備份管理員。了解詳情"備份和恢復角色和權限"。 "了解所有服務的NetApp Console存取角色" 。

開始之前

如果您打算複製到二級儲存並希望在本機快照或遠端ONTAP二級儲存上使用快照鎖定，則首先需要在叢集層級初始化ONTAP合規時脈。這是在政策中啟用快照鎖定的要求。

有關如何執行此操作的說明，請參閱 "在ONTAP中初始化合規性時鐘"。

有關快照鎖定的一般信息，請參閱 "ONTAP中的快照鎖定"。

步驟

1. 從NetApp Backup and Recovery選單中，選擇 策略。
2. 在「策略」頁面中，選擇「建立新策略」。

    此時會顯示「原則」頁面。

3. 在 **Details** 部分輸入資訊：
   - 工作負載類型：選取 **Kubernetes**。
   - 輸入策略名稱。
   - 從*代理*清單中選擇一個控制台代理。
4. 在 **Backup architecture** 區段中輸入資訊。從清單中選擇備份的資料流：
   - **3-2-1 fanout**：主儲存（磁碟）到輔助儲存（磁碟）到雲端（物件儲存）。此配置會在不同的儲存系統中建立多個資料副本，例如 ONTAP 到 ONTAP 以及 ONTAP 到物件儲存。物件存儲可以是雲端超大規模物件存儲，也可以是私有物件存儲。此配置最適合實現最佳資料保護和災難復原。此選項不適用於 Amazon FSx for NetApp ONTAP。
   - 磁碟到磁碟：主儲存設備（磁碟）到次要儲存設備（磁碟）。ONTAP 到 ONTAP 資料保護策略會在兩個 ONTAP 系統之間複寫資料，以確保高可用度和災難恢復。這通常使用 SnapMirror 來實現，它支援同步和非同步複寫。此方法可讓您的資料在不同位置保持更新並可用，以實現強大的資料保護。
   - 磁碟到物件儲存：主要儲存設備（磁碟）到雲端（物件存放區）。這會將資料從 ONTAP 系統複寫到物件儲存系統。這可以是超大規模雲端物件存儲，也可以是私有物件存儲，例如 StorageGRID。此方法非常適合長期資料保留和歸檔。此選項不適用於 Amazon FSx for NetApp ONTAP。
   - 本機快照：選定磁碟區上的本機快照。這會建立執行工作負載之正式作業磁碟區的唯讀時間點複本。您可以使用本機快照從資料遺失或毀損中恢復，也可以建立用於災難恢復的備份。
5. 提供 **Local snapshot settings** 部分的資訊：
   - 選擇“新增計劃”選項來選擇快照計劃或計劃。您最多可以有 5 個時間表。
   - 快照頻率：選擇每小時、每天、每週、每月或每年的頻率。 Kubernetes 工作負載沒有年度頻率。
   - 快照保留：輸入要保留的快照數量。
   - **Provider**：選擇託管 Kubernetes 應用程式資源的儲存提供者，並輸入憑證以向提供者進行身份驗證。
6. 請提供 **Secondary settings** 部分（複製到次要儲存設備）的資訊：
   - 備份：選擇每小時、每天、每週、每月或每年的頻率。
   - 備份目標：選擇二級儲存上用於備份的目標系統。
   - 保留：輸入要保留的快照數量。
   - **Provider**：選擇託管 Kubernetes 應用程式資源的儲存提供者，並輸入憑證以向提供者進行身份驗證。
7. 提供 * 物件儲存設定 * 區段的資訊（備份至物件儲存）：

   出現的欄位會根據所選的提供者和架構而有所不同。

   - **Provider**：選擇物件儲存的提供者，並在對應的欄位中輸入憑證（憑證欄位因提供者而異）。
   - 備份目標：選擇已註冊的物件儲存目標。確保目標在您的備份環境中可存取。
   - **IPspace**：選擇用於備份作業的 IP 空間。如果您有多個 IP 空間並想要控制哪一個用於備份，這將非常有用。
   - 計畫設定：選擇為本機快照設定的計畫。您可以刪除計劃，但不能新增計劃，因為計劃是根據本機快照計劃設定的。
   - 保留副本：輸入要保留的快照數量。
   - 運行於：選擇ONTAP傳輸計畫將資料備份到物件儲存。

- 將備份從物件儲存分層到檔案儲存：如果您選擇將備份分層到檔案儲存（例如，AWS Glacier），請選擇層選項和存檔天數。

編輯策略

您可以編輯備份架構、備份頻率、保留原則及原則的其他設定。對於 Kubernetes 工作負載原則，您只能編輯排程和保留設定。

您可以在編輯策略時新增另一個保護級別，但不能刪除保護級別。例如，如果策略僅保護本機快照，則可以將複製新增至輔助儲存或將備份新增至物件儲存。如果您有本機快照和複製，則可以新增物件儲存。但是，如果您有本機快照、複製和物件存儲，則不能刪除其中一個層級。

如果您正在編輯備份到物件儲存的策略，則可以啟用存檔。

如果您從SnapCenter匯入資源，您可能會遇到SnapCenter中使用的政策與NetApp Backup and Recovery中使用的策略之間的一些差異。看"SnapCenter與NetApp Backup and Recovery之間的策略差異"。

所需的**NetApp Console**角色

備份和恢復超級管理員。 "了解所有服務的NetApp Console存取角色" 。

步驟

1. 在NetApp Console中，前往 保護 > 備份和還原。
2. 選擇*政策*選項。
3. 選擇要編輯的策略。
4. 選擇*操作*•••圖標，然後選擇*編輯*。

刪除策略

如果您不再需要某個策略，則可以將其刪除。

您無法刪除與工作負載關聯的策略。

步驟

1. 在控制台中，前往*保護*>*備份和還原*。
2. 選擇*政策*選項。
3. 選擇要刪除的策略。
4. 選擇*操作*•••圖標，然後選擇*刪除*。
5. 確認操作，然後選擇*刪除*。

現在即可使用 **Backup and Recovery Web UI** 備份 **Kubernetes** 應用程式

NetApp Backup and Recovery 可讓您使用 Web 介面手動備份 Kubernetes 應用程式。

所需的**NetApp Console**角色

組織管理員或SnapCenter管理員。"了解NetApp Backup and Recovery存取角色" 。 "了解所有服務的NetApp Console存取角色" 。

現在即可使用 **Web UI** 備份 **Kubernetes** 應用程式

手動建立 Kubernetes 應用程式的備份，為未來的備份和快照建立基線，或確保最新資料受到保護。

步驟

1. 在NetApp Backup and Recovery中，選擇 **Inventory**。
2. 選擇一個 Kubernetes 實例，然後選擇「檢視」以查看與該實例關聯的資源。
3. 選擇“應用程式”標籤。
4. 在應用程式清單中，選擇要備份的應用程式並選擇相關的操作選單。
5. 選擇*立即備份*。
6. 確保選擇了正確的應用程式名稱。
7. 選擇*備份*。

結果

控制台建立應用程式的備份並在備份和還原的*監控*區域中顯示進度。此備份是根據與應用程式關聯的保護策略建立的。

現在可以使用 **Backup and Recovery** 中的自訂資源備份 **Kubernetes** 應用程式

NetApp Backup and Recovery 可讓您使用自訂資源（CR）手動備份 Kubernetes 應用程式。

立即使用自訂資源備份 **Kubernetes** 應用程式

手動建立 Kubernetes 應用程式的備份，為未來的備份和快照建立基線，或確保最新資料受到保護。

如果叢集範圍的資源在應用程式定義中被明確引用，或引用了任何應用程式命名空間，則這些資源將包含在備份、快照或複製中。

開始之前

請確保 AWS 會話令牌的有效期限足以應付任何長時間運行的 s3 備份作業。如果令牌在備份作業期間過期，則操作可能會失敗。

- 有關檢查當前會話令牌過期時間的更多資訊，請參閱 "AWS API 文件"。
- 如需 AWS 資源憑證的詳細資訊，請參閱 "AWS IAM 文件"。

使用自訂資源建立本機快照

若要為您的 Kubernetes 應用程式建立快照並將其儲存在本地，請使用具有特定屬性的 Snapshot 自訂資源。

步驟

1. 建立自訂資源（CR）檔案並將其命名為 `local-snapshot-cr.yaml`。
2. 在您建立的檔案中、設定以下屬性：
   - **metadata.name**：（必填）此自訂資源的名稱；請為您的環境選擇一個唯一且有意義的名稱。
   - **spec.applicationRef**：要建立快照的應用程式的 Kubernetes 名稱。

- **spec.appVaultRef**：（必填）應儲存快照內容（元資料）的 AppVault 名稱。
- **spec.reclaimPolicy**：（選用）定義當快照 CR 刪除時，快照的 AppArchive 會發生什麼情況。這意味著即使設為 `Retain`，快照也會被刪除。有效選項：
  - `Retain` (預設)
  - `Delete`

```
apiVersion: protect.trident.netapp.io/v1
kind: Snapshot
metadata:
  namespace: my-app-namespace
  name: local-snapshot-cr
spec:
  applicationRef: my-application
  appVaultRef: appvault-name
  reclaimPolicy: Retain
```

3. 在 `local-snapshot-cr.yaml` 檔案中填入正確的值後，套用 CR：

```
kubectl apply -f local-snapshot-cr.yaml
```

使用自訂資源將應用程式備份至物件儲存區

建立具有特定屬性的 Backup CR，以將應用程式備份到物件儲存區。

步驟

1. 建立自訂資源（CR）檔案並將其命名為 `object-store-backup-cr.yaml`。
2. 在您建立的檔案中、設定以下屬性：
   - **metadata.name**：（必填）此自訂資源的名稱；請為您的環境選擇一個唯一且有意義的名稱。
   - **spec.applicationRef**：（必需）要備份的應用程式的 Kubernetes 名稱。
   - **spec.appVaultRef**：（必需，與 *spec.appVaultTargetsRef* 互斥）如果您使用相同儲存桶來儲存快照和備份，則這是 AppVault 的名稱，備份內容應儲存在其中。
   - **spec.appVaultTargetsRef**：（必需，與 *spec.appVaultRef* 互斥）如果您使用不同的儲存貯體來儲存快照和備份，則此為應儲存備份內容的 AppVault 名稱。
   - **spec.dataMover**：（選用，從 *Trident Protect* 遷移的叢集必需）字串，指示要用於備份作業的備份工具。如果此叢集已從 Trident Protect 遷移到 Backup and Recovery，則該值區分大小寫，且必須為 `CBS`。
   - **spec.reclaimPolicy**：（選用）定義刪除 Backup CR 時備份內容（中繼資料/磁碟區資料）的處理方式。可能的值：
     - `Delete`
     - `Retain` (預設)

- **spec.cleanupSnapshot**：（必需）確保備份 CR 建立的臨時快照在備份作業完成後不會被刪除。建議值： `false`。

使用同一個儲存貯體儲存快照和備份時的範例 YAML：

```
apiVersion: protect.trident.netapp.io/v1
kind: Backup
metadata:
  namespace: my-app-namespace
  name: my-cr-name
spec:
  applicationRef: my-application
  appVaultRef: appvault-name
  dataMover: CBS
  reclaimPolicy: Retain
  cleanupSnapshot: false
```

使用不同儲存區分別儲存快照和備份時的 YAML 範例：

```
apiVersion: protect.trident.netapp.io/v1
kind: Backup
metadata:
  namespace: my-app-namespace
  name: object-store-backup-cr
spec:
  applicationRef: my-application
  appVaultTargetsRef: appvault-targets-name
  dataMover: CBS
  reclaimPolicy: Retain
  cleanupSnapshot: false
```

3. 在 `object-store-backup-cr.yaml` 檔案中填入正確的值後，套用 CR：

```
kubectl apply -f object-store-backup-cr.yaml
```

使用自訂資源建立 3-2-1 扇出備份

使用 3-2-1 扇出架構進行備份時，備份會同時複製到輔助儲存和物件儲存。若要建立 3-2-1 扇出備份，請建立具有特定屬性的 Backup CR。

步驟

1. 建立自訂資源（CR）檔案並將其命名為 `3-2-1-fanout-backup-cr.yaml`。
2. 在您建立的檔案中、設定以下屬性：

- **metadata.name**：（必填）此自訂資源的名稱；請為您的環境選擇一個唯一且有意義的名稱。
- **spec.applicationRef**：（必需）要備份的應用程式的 Kubernetes 名稱。
- **spec.appVaultTargetsRef**：（必填）應儲存備份內容的 AppVault 名稱。
- **spec.dataMover**：（可選）字串，指示要用於備份作業的備份工具。該值區分大小寫，且必須為 `CBS`。
- **spec.reclaimPolicy**：（選用）定義刪除 Backup CR 時備份內容（中繼資料/磁碟區資料）的處理方式。可能的值：
  - `Delete`
  - `Retain` (預設)
- **spec.cleanupSnapshot**：（必需）確保備份 CR 建立的臨時快照在備份作業完成後不會被刪除。建議值： `false`。
- **spec.replicateSnapshot**：（必要）指示 Backup and Recovery 將快照複製到次要儲存設備。必要值： `true`。
- **spec.replicateSnapshotReclaimPolicy**：（選用）定義刪除複製快照時的處理方式。可能的值：
  - `Delete`
  - `Retain` (預設)

  YAML 範例：

```yaml
apiVersion: protect.trident.netapp.io/v1
kind: Backup
metadata:
  namespace: my-app-namespace
  name: 3-2-1-fanout-backup-cr
spec:
  applicationRef: my-application
  appVaultTargetsRef: appvault-targets-name
  dataMover: CBS
  reclaimPolicy: Retain
  cleanupSnapshot: false
  replicateSnapshot: true
  replicateSnapshotReclaimPolicy: Retain
```

3. 在 `3-2-1-fanout-backup-cr.yaml` 檔案中填入正確的值後，套用 CR：

```
kubectl apply -f 3-2-1-fanout-backup-cr.yaml
```

支援的備份註釋

下表描述了建立備份 CR 時可以使用的註解。

註解	類型	描述	預設值
protect.trident.netapp.io/full-backup	字串	指定備份是否為非增量備份。設定為 `true`可建立非增量備份。最佳實踐是定期執行完整備份，並在兩次完整備份之間執行增量備份，以最大程度地降低還原相關風險。	"false"
protect.trident.netapp.io/snapshot-completion-timeout	字串	允許完成整個快照作業的最長時間。	"60 分鐘"
protect.trident.netapp.io/volume-snapshots-ready-to-use-timeout	字串	磁碟區快照達到可用狀態所允許的最長時間。	"30 分鐘"
protect.trident.netapp.io/volume-snapshots-created-timeout	字串	建立磁碟區快照所允許的最長時間。	"5 分鐘"
protect.trident.netapp.io/pvc-bind-timeout-sec	字串	等待新建立的任何 PersistentVolumeClaims （PVC） 到達該 `Bound` 階段的最大時間 （以秒為單位），超過此時間操作將失敗。	"1200"（20 分鐘）

恢復 **Kubernetes** 應用程式

使用 **Web UI** 還原 **Kubernetes** 應用程式

NetApp Backup and Recovery使您能夠恢復已透過保護策略保護的應用程式。要恢復應用程序，應用程式需要至少有一個可用的恢復點。復原點由本機快照或物件儲存備份（或兩者）組成。您可以使用本機、輔助或物件儲存存檔來還原應用程式。

檢視應用程式還原點的受保護資源

對於使用 Backup and Recovery 保護的每個應用程式、您可以檢視已針對特定還原點備份的資源。

所需的**NetApp Console**角色

Backup and Recovery 檢視器。"了解NetApp Backup and Recovery存取角色". "了解所有服務的NetApp Console存取角色".

步驟

1. 在 NetApp Backup and Recovery 中，選擇 **Inventory** > **Applications**。
2. 從應用程式清單中選擇一個應用程式，然後選取 Actions 圖示 ••• > **View and Restore**。
3. 從還原點清單中選擇一個還原點，然後選擇「操作」圖示 ••• > 檢視資源。

   這裡會顯示資源清單及其詳細資訊。您可以按命名空間或叢集範圍查看資源，也可以將清單下載為 JSON 檔案以供日後審核。

4. 完成後，請選取 **Close**。

恢復 **Kubernetes** 應用程式

您可以從還原點還原基於命名空間或基於 VM 的應用程式，既可以還原所有資源，也可以選擇還原部分資源。

開始之前

如果您要還原使用 Trident Protect 備份的應用程式、請確保來源叢集和目的地叢集上都安裝了 Trident Protect。

所需的**NetApp Console**角色

Backup and Recovery 超級管理員或 Backup and Recovery 還原管理員。"了解NetApp Backup and Recovery 存取角色". "了解所有服務的NetApp Console存取角色".

步驟

1. 在 NetApp Backup and Recovery 選單中，選取 還原。
2. 從清單中選擇一個 Kubernetes 應用程式，然後選擇該應用程式的 **View and Restore** 。

   出現還原點清單。

3. 選擇要使用的還原點的 **Restore** 按鈕。

   Restore data 精靈隨即開始，並顯示「一般設定」頁面。

4. 選擇要從中還原的來源位置。
5. 從*Cluster*清單中選擇目標群集。
6. 選擇還原至原始命名空間或新命名空間。
7. 如果您選擇還原到新的命名空間，請執行以下操作：
   a. 請輸入要使用的目標命名空間或多個命名空間。
   b. 請輸入目標應用程式名稱。
   c. 您也可以選擇「不為已復原的資源建立應用程式」選項，以便在不建立應用程式自訂資源物件的情況下復原資源。這樣可以減少應用程式清單中不必要的條目。
8. 選擇“下一步”。

   此時會出現 *Resource selection* 頁面。

9. 選擇是否要恢復與應用程式相關的所有資源，或使用篩選器選擇要復原的特定資源：

所有資源

a. 選擇*恢復所有資源*。

   當還原基於虛擬機器的應用程式時、Backup and Recovery 會列出還原點中的所有虛擬機器。

b. 選擇“下一步”。

特定命名空間型應用程式資源

a. 選擇 **Selective resources** 並選擇是根據規則還是根據命名空間來篩選所選資源。

資源選擇方法	步驟
使用規則篩選資源	i. 選擇 **Rules** 標籤。 ii. 選擇資源過濾器的行為。如果您選擇“包括”，則會恢復您選擇的資源。如果您選擇“排除”，則您選擇的資源將不會被恢復。 iii. 選擇*新增規則*來新增定義選擇資源的篩選器的規則。您至少需要一條規則來過濾資源。 每個規則都可以根據資源命名空間、標籤、群組、版本和種類等標準進行過濾。 iv. 選擇*儲存*來儲存每條規則。 v. 新增所有需要的規則後，選取 **Show resources** 以查看備份歸檔中符合篩選條件的可用資源。
從清單中手動選擇資源	i. 選擇 **Custom** 選項卡。 ii. 選擇 **Namespace-Scoped** 或 **Cluster-Scoped** 以檢視對應的資源。 備份與還原會列出還原點中的所有資源。 iii. 選擇要在還原作業中包含的資源。

ⓘ 顯示的資源是叢集上目前存在的資源。

b. 完成後，選擇 **Next**。

特定的基於 **VM** 的應用程式資源

a. 選擇*選擇性資源*。

b. 執行下列操作之一：

   - 若要還原整個虛擬機器，請選擇「虛擬機器」索引標籤。

     Backup and Recovery 會列出復原點中的所有虛擬機器。您可以選擇要包含在還原作業中的虛

擬機器。

- 若要還原個別持久性磁碟區聲明，請選擇 **Persistent volume claims** 索引標籤。

  Backup and Recovery 會列出復原點中的所有持久性磁碟區宣告。您可以選擇要在還原作業中包含的持久性磁碟區宣告。

c. 完成後，選擇 **Next**。

此時會顯示「目的地設定」頁面。

10. 展開 **Destination settings** 部分，然後選擇還原至預設儲存類別、不同的儲存類別，或者如果您要還原至不同的叢集，則將儲存類別對應至目的地叢集。
11. 如果選擇還原到不同的儲存類別，請選擇與每個來源儲存類別相符的目標儲存類別。
12. 如果您要還原使用 Trident Protect 建立的備份或快照，則可以選擇查看用作還原作業儲存貯體的 AppVault 詳細資訊。如果您的環境或 AppVault 狀態發生變更，請選取 **Sync App Vault** 以重新整理詳細資訊。

   如果您需要在 Kubernetes 叢集上建立 AppVault 以方便還原使用 Trident Protect 建立的備份或快照，請參閱 "使用 Trident Protect AppVault 物件來管理儲存桶"。

13. （可選）展開 **Restore scripts** 部分，啟用 **Postscript** 選項，選擇將在復原作業完成後執行的執行鉤子範本。如有需要，輸入腳本所需的任何參數，並新增標籤選擇器，以便根據資源標籤篩選資源。
14. （選用）展開 **Resource transformations** 區段，以便在還原程序中新增、移除或修改資源屬性。然後，執行下列動作：

   目前不支援修改 PersistentVolumeClaims 和命名空間。

   a. 啟用 **Resource transformation** 選項，即可對修改器進行變更。
   b. 從 **Template** 清單中選擇一個範本，即可快速套用您常用的修飾符設定。此清單包含常見場景的預定義範本以及您建立的自訂範本。

      在全域 "設定" 區域建立資源轉換範本。

   c. 請透過輸入資源群組、版本、種類和名稱來指定要修改的資源。
   d. 從 **Operation** 清單中選擇操作，以指定您要對資源執行的操作。
   e. 輸入您要變更之特定金鑰的 JSON 路徑。
   f. 如果適用，請輸入要使用的新值。值*欄位僅在執行某些操作（例如*新增*或*替換）時顯示。
   g. 您可以視需要新增更多資源轉換。

15. 完成後,請選擇 **Restore**。

使用自訂資源還原 **Kubernetes** 應用程式

您可以使用自訂資源從快照或備份還原應用程式。如果要將應用程式還原到同一個叢集，從現有快照還原速度會更快。

- 還原應用程式時，所有為該應用程式配置的執行鉤子都會隨應用程式一起還原。如果存在還原後執行鉤子，它會在還原操作過程中自動執行。
- qtree 磁碟區支援從備份還原到不同的命名空間或原始命名空間。但是，qtree 磁碟區不支援從快照還原到不同的命名空間或原始命名空間。
- 您可以使用進階設定來自訂還原操作。如需深入瞭解、請參閱 "使用進階自訂資源還原設定"。

將備份還原至不同的命名空間

當您使用 BackupRestore CR 將備份還原到不同的命名空間時，NetApp Backup and Recovery 會將應用程式還原到新的命名空間，並為還原的應用程式建立一個應用程式 CR。為了保護還原的應用程式，您可以建立隨需備份或快照，或製定保護排程。

- 將備份還原到具有現有資源的不同命名空間不會變更與備份中資源同名的任何資源。若要還原備份中的所有資源，請刪除並重新建立目標命名空間，或將備份還原到新的命名空間。
- 使用 CR 還原到新命名空間時，必須先手動建立目標命名空間，然後再套用 CR。NetApp Backup and Recovery 僅在使用 CLI 時才會自動建立命名空間。

開始之前

請確保 AWS 工作階段權杖的有效期限足以應付任何長時間執行的 s3 還原作業。如果權杖在還原作業期間過期，則作業可能會失敗。

- 有關檢查當前會話令牌過期時間的更多資訊，請參閱 "AWS API 文件"。
- 如需 AWS 資源憑證的詳細資訊，請參閱 "AWS IAM 文件"。

當您使用 Kopia 作為資料移動工具還原備份時，您可以選擇在 CR 中指定註釋，以控制 Kopia 使用的暫存的行為。有關可配置選項的更多資訊，請參閱 "Kopia 說明文件"。

步驟

1. 建立自訂資源（CR）檔案並將其命名為 `trident-protect-backup-restore-cr.yaml`。
2. 在您建立的檔案中、設定以下屬性：
   - **metadata.name**：（必填）此自訂資源的名稱；請為您的環境選擇一個唯一且有意義的名稱。
   - **spec.appArchivePath**：AppVault 內儲存備份內容的路徑。您可以使用以下命令尋找此路徑：

     ```
     kubectl get backups <BACKUP_NAME> -n my-app-namespace -o jsonpath
     ='{.status.appArchivePath}'
     ```

   - **spec.appVaultRef**：（必填）儲存備份內容的 AppVault 名稱。
   - **spec.namespaceMapping**：復原作業的來源命名空間到目標命名空間的對應。請將 `my-source-namespace`和 `my-destination-namespace`替換為您環境中的資訊。

```
apiVersion: protect.trident.netapp.io/v1
kind: BackupRestore
metadata:
  name: my-cr-name
  namespace: my-destination-namespace
spec:
  appArchivePath: my-backup-path
  appVaultRef: appvault-name
  namespaceMapping: [{"source": "my-source-namespace", "destination":
"my-destination-namespace"}]
```

3. （可選）如果您只需要選擇應用程式中的某些資源進行還原，請新增篩選條件，以包含或排除帶有特定標籤的資源：

> Trident Protect 會自動選擇一些資源，因為它們與您選擇的資源有關聯。例如，如果您選擇了持久卷聲明資源，並且它關聯了一個 pod，Trident Protect 也會恢復該關聯 pod。

- **resourceFilter.resourceSelectionCriteria**：（篩選必需）使用 `Include`或 `Exclude`來包含或排除在 resourceMatchers 中定義的資源。新增以下 resourceMatchers 參數以定義要包含或排除的資源：
  - **resourceFilter.resourceMatchers**：resourceMatcher 物件的陣列。如果在此陣列中定義多個元素，則它們之間按 OR 運算匹配，每個元素內的欄位（group、kind、version）之間按 AND 運算匹配。
    - **resourceMatchers[].group**：（可選）要篩選的資源群組。
    - **resourceMatchers[].kind**：（可選）要篩選的資源類型。
    - **resourceMatchers[].version**：（可選）要篩選的資源版本。
    - **resourceMatchers[].names**：（可選）要過濾的資源的 Kubernetes metadata.name 欄位中的名稱。
    - **resourceMatchers[].namespaces**：（可選）要篩選的資源的 Kubernetes metadata.name 欄位中的命名空間。
    - **resourceMatchers[].labelSelectors**：（可選）資源在 Kubernetes metadata.name 欄位中定義的標籤選擇器字串 "Kubernetes 說明文件"。例如：`"trident.netapp.io/os=linux"`。

      例如：

```
spec:
  resourceFilter:
    resourceSelectionCriteria: "Include"
    resourceMatchers:
      - group: my-resource-group-1
        kind: my-resource-kind-1
        version: my-resource-version-1
        names: ["my-resource-names"]
        namespaces: ["my-resource-namespaces"]
        labelSelectors: ["trident.netapp.io/os=linux"]
      - group: my-resource-group-2
        kind: my-resource-kind-2
        version: my-resource-version-2
        names: ["my-resource-names"]
        namespaces: ["my-resource-namespaces"]
        labelSelectors: ["trident.netapp.io/os=linux"]
```

4. 在 `trident-protect-backup-restore-cr.yaml` 檔案中填入正確的值後，套用 CR：

```
kubectl apply -f trident-protect-backup-restore-cr.yaml
```

將備份還原至原始命名空間

您可以隨時將備份還原至原始命名空間。

開始之前

請確保 AWS 工作階段權杖的有效期限足以應付任何長時間執行的 s3 還原作業。如果權杖在還原作業期間過期，則作業可能會失敗。

- 有關檢查當前會話令牌過期時間的更多資訊，請參閱 "AWS API 文件"。
- 如需 AWS 資源憑證的詳細資訊，請參閱 "AWS IAM 文件"。

當您使用 Kopia 作為資料移動工具還原備份時，您可以選擇在 CR 中指定註釋，以控制 Kopia 使用的暫存的行為。有關可配置選項的更多資訊，請參閱 "Kopia 說明文件"。

步驟

1. 建立自訂資源（CR）檔案並將其命名為 `trident-protect-backup-ipr-cr.yaml`。
2. 在您建立的檔案中、設定以下屬性：
   - **metadata.name**：（必填）此自訂資源的名稱；請為您的環境選擇一個唯一且有意義的名稱。
   - **spec.appArchivePath**：AppVault 內儲存備份內容的路徑。您可以使用以下命令尋找此路徑：

```
kubectl get backups <BACKUP_NAME> -n my-app-namespace -o jsonpath
='{.status.appArchivePath}'
```

- **spec.appVaultRef**：（必填）儲存備份內容的 AppVault 名稱。

  例如：

```
apiVersion: protect.trident.netapp.io/v1
kind: BackupInplaceRestore
metadata:
  name: my-cr-name
  namespace: my-app-namespace
spec:
  appArchivePath: my-backup-path
  appVaultRef: appvault-name
```

3. （可選）如果您只需要選擇應用程式中的某些資源進行還原，請新增篩選條件，以包含或排除帶有特定標籤的資源：

   Trident Protect 會自動選擇一些資源，因為它們與您選擇的資源有關聯。例如，如果您選擇了持久卷聲明資源，並且它關聯了一個 pod，Trident Protect 也會恢復該關聯 pod。

   - **resourceFilter.resourceSelectionCriteria**：（篩選必需）使用 `Include`或 `Exclude`來包含或排除在 resourceMatchers 中定義的資源。新增以下 resourceMatchers 參數以定義要包含或排除的資源：
     - **resourceFilter.resourceMatchers**：resourceMatcher 物件的陣列。如果在此陣列中定義多個元素，則它們之間按 OR 運算匹配，每個元素內的欄位（group、kind、version）之間按 AND 運算匹配。
       - **resourceMatchers[].group**：（可選）要篩選的資源群組。
       - **resourceMatchers[].kind**：（可選）要篩選的資源類型。
       - **resourceMatchers[].version**：（可選）要篩選的資源版本。
       - **resourceMatchers[].names**：（可選）要過濾的資源的 Kubernetes metadata.name 欄位中的名稱。
       - **resourceMatchers[].namespaces**：（可選）要篩選的資源的 Kubernetes metadata.name 欄位中的命名空間。
       - **resourceMatchers[].labelSelectors**：（可選）資源在 Kubernetes metadata.name 欄位中定義的標籤選擇器字串 "Kubernetes 說明文件"。例如：`"trident.netapp.io/os=linux"`。

         例如：

```
spec:
  resourceFilter:
    resourceSelectionCriteria: "Include"
    resourceMatchers:
      - group: my-resource-group-1
        kind: my-resource-kind-1
        version: my-resource-version-1
        names: ["my-resource-names"]
        namespaces: ["my-resource-namespaces"]
        labelSelectors: ["trident.netapp.io/os=linux"]
      - group: my-resource-group-2
        kind: my-resource-kind-2
        version: my-resource-version-2
        names: ["my-resource-names"]
        namespaces: ["my-resource-namespaces"]
        labelSelectors: ["trident.netapp.io/os=linux"]
```

4. 在 `trident-protect-backup-ipr-cr.yaml` 檔案中填入正確的值後，套用 CR：

```
kubectl apply -f trident-protect-backup-ipr-cr.yaml
```

將備份還原至不同的叢集

如果原始叢集出現問題、您可以將備份還原至不同的叢集。

- 當您使用 Kopia 作為資料移動工具還原備份時，您可以選擇在 CR 中指定註釋，以控制 Kopia 使用的暫存的行為。有關可配置選項的更多資訊，請參閱 "Kopia 說明文件"。
- 使用 CR 還原到新命名空間時，必須先手動建立目的地命名空間，然後再套用 CR。

開始之前

請確保符合下列先決條件：

- 目標叢集已安裝 Trident Protect。
- 目標叢集可以存取與來源叢集相同的 AppVault 儲存桶路徑，備份檔案就儲存在該路徑中。
- 請確保 AWS 工作階段權杖的有效期限足以應付任何長時間執行的還原作業。如果權杖在還原作業期間過期、作業可能會失敗。
  - 有關檢查當前會話令牌過期時間的更多資訊，請參閱 "AWS API 文件"。
  - 如需 AWS 資源憑證的詳細資訊，請參閱 "AWS 文件"。

步驟

1. 使用 Trident Protect CLI 外掛程式檢查目標叢集上 AppVault CR 的可用性：

```
tridentctl-protect get appvault --context <destination_cluster_name>
```

確保用於應用程式還原的命名空間存在於目的地叢集上。

2. 從目的地叢集檢視可用 AppVault 的備份內容：

```
tridentctl-protect get appvaultcontent <appvault_name> \
--show-resources backup \
--show-paths \
--context <destination_cluster_name>
```

執行此命令將顯示 AppVault 中的可用備份，包括其來源叢集、相應的應用程式名稱、時間戳記和歸檔路徑。

範例輸出：

```
+-------------+-----------+--------+-----------------
+---------------------------+-------------+
|   CLUSTER   |    APP    |  TYPE  |      NAME       |         TIMESTAMP
|    PATH     |
+-------------+-----------+--------+-----------------
+---------------------------+-------------+
| production1 | wordpress | backup | wordpress-bkup-1| 2024-10-30
08:37:40 (UTC)| backuppath1 |
| production1 | wordpress | backup | wordpress-bkup-2| 2024-10-30
08:37:40 (UTC)| backuppath2 |
+-------------+-----------+--------+-----------------
+---------------------------+-------------+
```

3. 使用 AppVault 名稱和歸檔路徑將應用程式還原到目標叢集：
4. 建立自訂資源（CR）檔案並將其命名為 `trident-protect-backup-restore-cr.yaml`。
5. 在您建立的檔案中、設定以下屬性：
   - **metadata.name**：（必填）此自訂資源的名稱；請為您的環境選擇一個唯一且有意義的名稱。
   - **spec.appVaultRef**：（必填）儲存備份內容的 AppVault 名稱。
   - **spec.appArchivePath**：AppVault 內儲存備份內容的路徑。您可以使用以下命令尋找此路徑：

```
kubectl get backups <BACKUP_NAME> -n my-app-namespace -o jsonpath
='{.status.appArchivePath}'
```

如果 BackupRestore CR 不可用，您可以使用步驟 2 中提到的指令來查看備份內容。

- **spec.namespaceMapping**：復原作業的來源命名空間到目標命名空間的對應。請將 `my-source-namespace`和 `my-destination-namespace`替換為您環境中的資訊。

  例如：

```
apiVersion: protect.trident.netapp.io/v1
kind: BackupRestore
metadata:
  name: my-cr-name
  namespace: my-destination-namespace
spec:
  appVaultRef: appvault-name
  appArchivePath: my-backup-path
  namespaceMapping: [{"source": "my-source-namespace", "destination":
"my-destination-namespace"}]
```

6. 在 `trident-protect-backup-restore-cr.yaml` 檔案中填入正確的值後，套用 CR：

```
kubectl apply -f trident-protect-backup-restore-cr.yaml
```

將快照還原至不同的命名空間

您可以使用自訂資源 (CR) 檔案從快照還原資料到不同的命名空間或原始來源命名空間。當您使用 SnapshotRestore CR 將快照還原到不同的命名空間時，NetApp Backup and Recovery 會將應用程式還原到新的命名空間，並為還原的應用程式建立一個應用程式 CR。為了保護還原的應用程式，您可以建立隨需備份或快照，或製定保護排程。

- SnapshotRestore 支援 `spec.storageClassMapping` 屬性，但僅當來源儲存類別和目標儲存類別使用相同的儲存後端時才支援。如果嘗試還原到使用不同儲存後端的 `StorageClass`，則還原操作將會失敗。
- 使用 CR 還原到新命名空間時，必須先手動建立目的地命名空間，然後再套用 CR。

開始之前

請確保 AWS 工作階段權杖的有效期限足以應付任何長時間執行的 s3 還原作業。如果權杖在還原作業期間過期，則作業可能會失敗。

- 有關檢查當前會話令牌過期時間的更多資訊，請參閱 "AWS API 文件"。
- 如需 AWS 資源憑證的詳細資訊，請參閱 "AWS IAM 文件"。

步驟

1. 建立自訂資源（CR）檔案並將其命名為 `trident-protect-snapshot-restore-cr.yaml`。
2. 在您建立的檔案中、設定以下屬性：

- **metadata.name**：（必填）此自訂資源的名稱；請為您的環境選擇一個唯一且有意義的名稱。
- **spec.appVaultRef**：（必填）儲存快照內容的 AppVault 名稱。
- **spec.appArchivePath**：AppVault 內儲存快照內容的路徑。您可以使用以下命令來尋找此路徑：

```
kubectl get snapshots <SNAPHOT_NAME> -n my-app-namespace -o jsonpath
='{.status.appArchivePath}'
```

- **spec.namespaceMapping**：復原作業的來源命名空間到目標命名空間的對應。請將 `my-source-namespace`和 `my-destination-namespace`替換為您環境中的資訊。

```
apiVersion: protect.trident.netapp.io/v1
kind: SnapshotRestore
metadata:
  name: my-cr-name
  namespace: my-app-namespace
spec:
  appVaultRef: appvault-name
  appArchivePath: my-snapshot-path
  namespaceMapping: [{"source": "my-source-namespace", "destination":
"my-destination-namespace"}]
```

3. （可選）如果您只需要選擇應用程式中的某些資源進行還原，請新增篩選條件，以包含或排除帶有特定標籤的資源：

> Trident Protect 會自動選擇一些資源，因為它們與您選擇的資源有關聯。例如，如果您選擇了持久卷聲明資源，並且它關聯了一個 pod，Trident Protect 也會恢復該關聯 pod。

- **resourceFilter.resourceSelectionCriteria**：（篩選必需）使用 `Include`或 `Exclude`來包含或排除在 resourceMatchers 中定義的資源。新增以下 resourceMatchers 參數以定義要包含或排除的資源：
  - **resourceFilter.resourceMatchers**：resourceMatcher 物件的陣列。如果在此陣列中定義多個元素，則它們之間按 OR 運算匹配，每個元素內的欄位（group、kind、version）之間按 AND 運算匹配。
    - **resourceMatchers[].group**：（可選）要篩選的資源群組。
    - **resourceMatchers[].kind**：（可選）要篩選的資源類型。
    - **resourceMatchers[].version**：（可選）要篩選的資源版本。
    - **resourceMatchers[].names**：（可選）要過濾的資源的 Kubernetes metadata.name 欄位中的名稱。
    - **resourceMatchers[].namespaces**：（可選）要篩選的資源的 Kubernetes metadata.name 欄位中的命名空間。
    - **resourceMatchers[].labelSelectors**：（可選）資源在 Kubernetes metadata.name 欄位中定義的標籤選擇器字串 "Kubernetes 說明文件"。例如：`"trident.netapp.io/os=linux"`。

      例如：

```
spec:
  resourceFilter:
    resourceSelectionCriteria: "Include"
    resourceMatchers:
      - group: my-resource-group-1
        kind: my-resource-kind-1
        version: my-resource-version-1
        names: ["my-resource-names"]
        namespaces: ["my-resource-namespaces"]
        labelSelectors: ["trident.netapp.io/os=linux"]
      - group: my-resource-group-2
        kind: my-resource-kind-2
        version: my-resource-version-2
        names: ["my-resource-names"]
        namespaces: ["my-resource-namespaces"]
        labelSelectors: ["trident.netapp.io/os=linux"]
```

4. 在 `trident-protect-snapshot-restore-cr.yaml` 檔案中填入正確的值後，套用 CR：

```
kubectl apply -f trident-protect-snapshot-restore-cr.yaml
```

將快照還原至原始命名空間

您可以隨時將快照還原至原始命名空間。

目前不支援從本機快照對基於 VM 的應用程式進行就地還原（還原至原始命名空間和原始叢集）。

開始之前

請確保 AWS 工作階段權杖的有效期限足以應付任何長時間執行的 s3 還原作業。如果權杖在還原作業期間過期，則作業可能會失敗。

- 有關檢查當前會話令牌過期時間的更多資訊，請參閱 "AWS API 文件"。
- 如需 AWS 資源憑證的詳細資訊，請參閱 "AWS IAM 文件"。

步驟

1. 建立自訂資源（CR）檔案並將其命名為 `trident-protect-snapshot-ipr-cr.yaml`。
2. 在您建立的檔案中、設定以下屬性：
   - **metadata.name**：（必填）此自訂資源的名稱；請為您的環境選擇一個唯一且有意義的名稱。
   - **spec.appVaultRef**：（必填）儲存快照內容的 AppVault 名稱。
   - **spec.appArchivePath**：AppVault 內儲存快照內容的路徑。您可以使用以下命令來尋找此路徑：

```
kubectl get snapshots <SNAPSHOT_NAME> -n my-app-namespace -o
jsonpath='{.status.appArchivePath}'
```

```
apiVersion: protect.trident.netapp.io/v1
kind: SnapshotInplaceRestore
metadata:
  name: my-cr-name
  namespace: my-app-namespace
spec:
  appVaultRef: appvault-name
  appArchivePath: my-snapshot-path
```

3. （可選）如果您只需要選擇應用程式中的某些資源進行還原，請新增篩選條件，以包含或排除帶有特定標籤的資源：

> Trident Protect 會自動選擇一些資源，因為它們與您選擇的資源有關聯。例如，如果您選擇了持久卷聲明資源，並且它關聯了一個 pod，Trident Protect 也會恢復該關聯 pod。

- **resourceFilter.resourceSelectionCriteria**：（篩選必需）使用 `Include`或 `Exclude`來包含或排除在 resourceMatchers 中定義的資源。新增以下 resourceMatchers 參數以定義要包含或排除的資源：
  - **resourceFilter.resourceMatchers**：resourceMatcher 物件的陣列。如果在此陣列中定義多個元素，則它們之間按 OR 運算匹配，每個元素內的欄位（group、kind、version）之間按 AND 運算匹配。
    - **resourceMatchers[].group**：（可選）要篩選的資源群組。
    - **resourceMatchers[].kind**：（可選）要篩選的資源類型。
    - **resourceMatchers[].version**：（可選）要篩選的資源版本。
    - **resourceMatchers[].names**：（可選）要過濾的資源的 Kubernetes metadata.name 欄位中的名稱。
    - **resourceMatchers[].namespaces**：（可選）要篩選的資源的 Kubernetes metadata.name 欄位中的命名空間。
    - **resourceMatchers[].labelSelectors**：（可選）資源在 Kubernetes metadata.name 欄位中定義的標籤選擇器字串 "Kubernetes 說明文件"。例如：`"trident.netapp.io/os=linux"`。

      例如：

```
spec:
  resourceFilter:
    resourceSelectionCriteria: "Include"
    resourceMatchers:
      - group: my-resource-group-1
        kind: my-resource-kind-1
        version: my-resource-version-1
        names: ["my-resource-names"]
        namespaces: ["my-resource-namespaces"]
        labelSelectors: ["trident.netapp.io/os=linux"]
      - group: my-resource-group-2
        kind: my-resource-kind-2
        version: my-resource-version-2
        names: ["my-resource-names"]
        namespaces: ["my-resource-namespaces"]
        labelSelectors: ["trident.netapp.io/os=linux"]
```

4. 在 `trident-protect-snapshot-ipr-cr.yaml` 檔案中填入正確的值後，套用 CR：

```
kubectl apply -f trident-protect-snapshot-ipr-cr.yaml
```

使用進階自訂資源還原設定

您可以利用註釋、命名空間設定和儲存選項等進階設定來自訂還原作業，以滿足您的特定需求。

還原和容錯移轉作業期間的命名空間註釋和標籤

在還原和容錯移轉期間，目的地命名空間標籤和註釋會更新以符合來源：來源中的金鑰會新增至目的地金鑰或覆寫目的地金鑰，而僅存在於目的地中的金鑰則保持不變。

> ⓘ 在 Red Hat OpenShift 中，命名空間註解非常重要，因為它們可以確保還原的 Pod 獲得正確的安全內容限制和權限，使其能夠存取磁碟區並在沒有權限錯誤的情況下執行。如需更多資訊，請參閱 "OpenShift 安全上下文約束文檔"。

設定 Kubernetes 環境變數

```
RESTORE_SKIP_NAMESPACE_ANNOTATIONS
```

在還原或容錯移轉之前，以防止特定目的地命名空間註釋被覆寫。例如：

```
helm upgrade trident-protect -n trident-protect netapp-trident-
protect/trident-protect \
  --set-string
restoreSkipNamespaceAnnotations="{<annotation_key_to_skip_1>,<annotation_k
ey_to_skip_2>}" \
  --reuse-values
```

在還原或容錯移轉期間， `restoreSkipNamespaceAnnotations` 和 `restoreSkipNamespaceLabels` 中指定的任何命名空間註釋和標籤都會從還原或容錯移轉作業中排除。請確保在初始 Helm 安裝期間配置這些設定。若要深入瞭解，請參閱 "設定其他 Trident Protect Helm Chart 設定"。

如果您使用 Helm 並帶有 `--create-namespace`旗標來安裝來源應用程式，Trident Protect 會將名稱標籤複製到目的地命名空間。如果標籤值與來源命名空間名稱相符，則會將其取代為目的地命名空間名稱；否則，則保持不變。

範例

以下範例顯示具有不同標籤和註釋的來源命名空間和目的地命名空間，並顯示作業前後的目的地命名空間，以說明如何新增、合併或覆寫金鑰。

在還原或容錯移轉作業之前

下表說明了復原或容錯移轉作業之前範例來源命名空間和目標命名空間的狀態：

命名空間	註解	標籤
命名空間 ns-1 （來源）	• annotation.one/key: 「updatedvalue」 • annotation.two/key: 「true」	• environment=production • 合規性＝hipaa • 名稱=ns-1
命名空間 ns-2 （目標）	• annotation.one/key: 「true」 • annotation.three/key:「false」	• 角色=資料庫

還原作業後

下表展示了復原或故障轉移作業後範例目標命名空間的狀態。一些鍵已被添加，一些鍵已被覆蓋，並且 `name` 標籤已更新以匹配目標命名空間：

命名空間	註解	標籤
命名空間 ns-2 （目標）	• annotation.one/key: 「updatedvalue」 • annotation.two/key: 「true」 • annotation.three/key:「false」	• 名稱=ns-2 • 合規性＝hipaa • environment=production • 角色=資料庫

支援的欄位

本節說明可用於還原作業的其他欄位。

儲存類別對應

此 `spec.storageClassMapping`屬性定義了從來源應用程式中的儲存類別到目標叢集上新儲存類別的對應。在將應用程式遷移到具有不同儲存類別的叢集之間，或變更 BackupRestore 操作的儲存後端時，可以使用此屬性。

- 範例： *

```
storageClassMapping:
  - destination: "destinationStorageClass1"
    source: "sourceStorageClass1"
  - destination: "destinationStorageClass2"
    source: "sourceStorageClass2"
```

支援的註釋

本節列出系統中用於配置各種行為的支援註解。如果使用者未明確設定註解，系統將使用預設值。

註解	類型	描述	預設值
protect.trident.netapp.io/data-mover-timeout-sec	字串	資料移動器操作允許停止的最長時間（以秒為單位）。	"300"
protect.trident.netapp.io/kopia-content-cache-size-limit-mb	字串	Kopia 內容快取的大小上限（以 MB 為單位）。	"1000"
protect.trident.netapp.io/pvc-bind-timeout-sec	字串	等待所有新建立的 PersistentVolumeClaims（PVC）達到 `Bound`階段的最長時間（以秒為單位），超過此時間操作將會失敗。適用於所有還原 CR 類型（BackupRestore、BackupInplaceRestore、SnapshotRestore、SnapshotInplaceRestore）。如果您的儲存後端或叢集通常需要更多時間，請使用更高的值。	"1200"（20 分鐘）

使用自訂資源還原資源時修改資源

資源轉換功能可讓您在還原資源時進行修改。當還原後的版本需要與原始版本有所不同時，此功能非常有用——例如，在將虛擬機器還原到不同的網路時更改其 IP 位址。您還可以"使用 Web UI 在還原資源時修改資源"。

必要的 **NetApp Console** 角色 Backup and Recovery 超級管理員或 Backup and Recovery 還原管理員。"了解NetApp Backup and Recovery存取角色". "了解所有服務的NetApp Console存取角色".

資源修改的運作方式

`transformations` 欄位位於 `SnapshotRestore`、 `BackupRestore`、 `AppMirrorRelationship` 及其他還原資源中，可讓您在還原過程中修改 Kubernetes 資源。這對於透過變更主機名稱、登錄 URL、資源限制或環境變數，使應用程式或虛擬機器適應新叢集非常有用。

資源轉換使用 "RFC 6902" JSON Patch 操作和 "RFC 6901" JSON Pointer 路徑來定位和修改 Kubernetes 資源中的特定欄位。

以下是包含資源轉換的還原物件的基本結構：

```
apiVersion: protect.trident.netapp.io/v1
kind: SnapshotRestore
metadata:
  name: my-restore
  namespace: target-namespace
spec:
  appVaultRef: my-vault
  appArchivePath: /path/to/snapshot
  namespaceMapping:
    - source: source-ns
      destination: target-ns
  transformations:
    - resource:
        kind: ConfigMap       # Required: resource kind
        group: ""             # Optional: API group (empty for core
resources)
        version: ""           # Optional: API version
        name: ""              # Optional: specific resource name
      operations:
        - op: replace         # Operation type
          path: "/data/key"   # JSON Pointer path
          value: "new-value"  # New value (for add/replace/test)
```

支援的資源

您可以將資源轉換套用至符合下列條件的資源：

- `kind`（必填）：Kubernetes 資源類型（例如 `ConfigMap`、 `Deployment`、 `Pod`）
- `group`（選用）：API 群組（例如， `apps`、 `route.openshift.io`）- 核心資源請省略
- `version`（選用）：API 版本（例如、 `v1`、 `v1beta1`）
- `name`（選用）：僅依名稱套用至特定資源

目前不支援修改 PersistentVolumeClaims 和命名空間。

支援的作業

您可以使用以下操作來修改資源：

- `add`：為資源新增值。
- `copy`：將值從一個路徑複製到另一個路徑。
- `move`：在資源內移動值。
- `remove`：從資源中移除一個值。
- `replace`：替換資源中的值。
- `test`：在執行操作之前先進行測試。

為資源新增值

使用 `add` 操作可在指定路徑新增欄位或值。您可以為物件或陣列新增資料。以下範例為 Deployment 資源新增節點選擇器：

```
transformations:
  - resource:
      kind: Deployment
    operations:
      - op: add
        path: "/spec/template/spec/nodeSelector"
        value:
          "topology.kubernetes.io/zone": "us-east-1a"
          disktype: "ssd"
```

使用以下命令透過命令列執行此轉換：

```
tridentctl-protect --transformation
'apps,v1,Deployment:add{"path":"/spec/template/spec/nodeSelector","value":
{"topology.kubernetes.io/zone":"us-east-1a","disktype":"ssd"}}'
```

複製資源中的值

使用 `copy` 操作可將相同資源內的值從一個路徑複製到另一個路徑。來源值不會改變。以下範例複製 ConfigMap 物件的資料鍵：

```
transformations:
  - resource:
      kind: ConfigMap
    operations:
      - op: copy
        from: "/data/source-key"
        path: "/data/backup-key"
```

使用以下命令透過命令列執行此轉換：

```
tridentctl-protect --transformation
',v1,ConfigMap:copy{"from":"/data/source-key","path":"/data/backup-key"}'
```

在資源內移動值

使用 `move`操作可將相同資源中的值從一個路徑移至另一個路徑。來源路徑將被移除，值將被放置在目標路徑中。以下範例重新命名 ConfigMap 物件的資料鍵：

```
transformations:
  - resource:
      kind: ConfigMap
    operations:
      - op: move
        from: "/data/OLD_KEY"
        path: "/data/NEW_KEY"
```

使用以下命令透過命令列執行此轉換：

```
tridentctl-protect --transformation
',v1,ConfigMap:move{"from":"/data/OLD_KEY","path":"/data/NEW_KEY"}'
```

從資源中移除值

使用 `remove` 操作可刪除指定路徑中的欄位或值。以下範例從 ConfigMap 資源中刪除註解：

```
transformations:
  - resource:
      kind: ConfigMap
    operations:
      - op: remove
        path: "/metadata/annotations/kubectl.kubernetes.io~1last-applied-
configuration"
```

在上述範例的路徑中、 `~1` 是 `/` 的 JSON Pointer 轉義序列。

使用以下命令透過命令列執行此轉換：

```
tridentctl-protect --transformation
',v1,ConfigMap:remove{"path":"/metadata/annotations/kubectl.kubernetes.io~
1last-applied-configuration"}'
```

取代資源中的值

使用 `replace` 操作可取代指定路徑下資源中的現有值。JSON 路徑必須已存在。以下範例更改了 Route 物件的 hostname：

```
transformations:
  - resource:
      kind: Route
      group: route.openshift.io
    operations:
      - op: replace
        path: "/spec/host"
        value: "prod.example.com"
```

使用以下命令透過命令列執行此轉換：

```
tridentctl-protect --transformation
'route.openshift.io,v1,Route:replace{"path":"/spec/host","value":"prod.exa
mple.com"}'
```

測試資源修改

使用 `test`操作來測試指定路徑的值是否符合預期值。如果測試失敗，整個修改將會回復。以下範例僅在 `database-host`為 `environment`時才會更新 `staging`：

```
transformations:
  - resource:
      kind: ConfigMap
    operations:
      - op: test
        path: "/data/environment"
        value: "staging"
      - op: replace
        path: "/data/database-host"
        value: "prod-db.example.com"
```

使用以下命令透過命令列執行此轉換：

```
tridentctl-protect --transformation
',v1,ConfigMap:test{"path":"/data/environment","value":"staging"},replace{
"path":"/data/database-host","value":"prod-db.example.com"}'
```

管理 Kubernetes 集群

NetApp Backup and Recovery使您能夠發現和管理 Kubernetes 集群，以便保護集群託管的資源。

所需的**NetApp Console**角色

組織管理員或SnapCenter管理員。"了解NetApp Backup and Recovery存取角色" 。 "了解所有服務的NetApp Console存取角色" 。

要發現 Kubernetes 集群，請參閱"發現 Kubernetes 工作負載"。

編輯 Kubernetes 叢集訊息

如果需要變更叢集名稱，您可以編輯叢集。

步驟

1. 在NetApp Backup and Recovery中，選擇 **Inventory** > **Clusters**。
2. 在群集清單中，選擇要編輯的群集並選擇相關的操作選單。
3. 選擇*編輯集群*。
4. 對叢集名稱進行任何必要的更改。叢集名稱需要與您在發現過程中使用 Helm 命令的名稱相符。
5. 選擇*完成*。

刪除 Kubernetes 集群

若要停止保護 Kubernetes 集群，請停用保護並刪除相關應用程序，然後從NetApp Backup and Recovery中刪除

該集群。 NetApp Backup and Recovery不會刪除叢集或其資源；它只會從NetApp Console清單中刪除叢集。

步驟

1. 在NetApp Backup and Recovery中，選擇 **Inventory** > **Clusters**。
2. 在群集清單中，選擇要編輯的群集並選擇相關的操作選單。
3. 選擇*刪除群集*。
4. 查看確認對話方塊中的信息，然後選擇*刪除*。

升級 **Trident Protect**

對於執行 Trident Protect 26.05 或更高版本的 Kubernetes 叢集，您可以直接從 NetApp Backup and Recovery 升級 Trident Protect。只有在有新版本可用時，才會顯示升級選項。

步驟

1. 在NetApp Backup and Recovery中，選擇 **Inventory** > **Clusters**。
2. 選擇要升級的 Kubernetes 叢集，然後選取 Actions 圖示 ••• > **Upgrade cluster**。
3. 在升級對話方塊中、請執行下列動作：
   a. 從清單中選擇要升級的版本。
   b. 選擇 **Upgrade** 。

管理 **Kubernetes** 應用程式

NetApp Backup and Recovery讓您能夠取消保護並刪除 Kubernetes 應用程式及相關資源。

所需的**NetApp Console**角色

組織管理員或SnapCenter管理員。"了解NetApp Backup and Recovery存取角色" 。 "了解所有服務的NetApp Console存取角色" 。

取消保護 **Kubernetes** 應用程式

如果您不再需要保護某個應用程序，可以取消保護。當您取消保護應用程式時， NetApp Backup and Recovery 會停止保護該應用程序，但保留所有相關的備份和快照。

在應用程式的保護操作仍在進行時，您無法取消其保護。您可以等待操作完成，或作為變通方法，刪除還原點正在執行的保護操作正在使用。然後您就可以取消應用程式的保護。

步驟

1. 在NetApp Backup and Recovery中，選擇 **Inventory**。
2. 選擇一個 Kubernetes 實例，然後選擇「檢視」以查看與該實例關聯的資源。
3. 選擇“應用程式”標籤。
4. 在應用程式清單中，選擇要取消保護的應用程式並選擇相關的操作選單。
5. 選擇*取消保護*。

6. 閱讀通知，準備好後，選擇*取消保護*。

刪除 **Kubernetes** 應用程式

刪除不再需要的應用程式。 NetApp Backup and Recovery停止保護並刪除已刪除應用程式的所有備份和快照。

步驟

1. 在NetApp Backup and Recovery中，選擇 **Inventory**。
2. 選擇一個 Kubernetes 實例，然後選擇「檢視」以查看與該實例關聯的資源。
3. 選擇“應用程式”標籤。
4. 在應用程式清單中，選擇要刪除的應用程式並選擇相關的操作選單。
5. 選擇*刪除*。
6. 啟用*刪除快照和備份*以刪除應用程式的所有快照和備份。

   您將無法再使用這些快照和備份還原應用程式。

7. 確認操作並選擇*刪除*。

刪除 **Kubernetes** 應用程式的還原點

如果您需要取消應用程式的保護，並且目前正在執行保護操作，則可能需要刪除該應用程式的還原點。

步驟

1. 在 NetApp Backup and Recovery 選單中，選取 還原。
2. 從清單中選擇一個 Kubernetes 應用程式，然後選擇該應用程式的 **View and Restore** 。

   出現還原點清單。

3. 選擇您需要刪除的復原點，然後選取 Actions 圖示 ••• > **Delete recovery point** 將其刪除。

管理適用於 **Kubernetes** 工作負載的**NetApp Backup and Recovery**執行掛鉤模板

執行鉤子是一種自訂操作，它與託管 Kubernetes 應用程式中的資料保護操作一起運行。例如，透過使用執行掛鉤在快照之前暫停資料庫事務並在之後恢復它們來建立應用程式一致的快照。建立執行鉤子模板時，指定鉤子類型、要執行的腳本以及目標容器的過濾器。使用模板將執行掛鉤連結到您的應用程式。

NetApp Backup and Recovery 會在資料保護期間，為像 KubeVirt 這樣的應用程式凍結和解除凍結檔案系統。您可以透過 Trident Protect 文件，全域或針對特定應用程式停用此行為：

- 若要為所有應用程式停用此行為，請參閱 "使用 KubeVirt 虛擬機器保護數據"。
- 若要針對特定應用程式停用此行為，請參閱 "定義應用程式"。

所需的**NetApp Console**角色

組織管理員或SnapCenter管理員。"了解NetApp Backup and Recovery存取角色" 。 "了解所有服務的NetApp Console存取角色" 。

執行鉤子的類型

NetApp Backup and Recovery根據運行時間支援以下類型的執行掛鉤：

- 預快照
- 快照後
- 預備份
- 備份後
- 恢復後

執行順序

當執行資料保護操作時，執行掛鉤事件會依照下列順序發生：

1. 任何適用的自訂預操作執行掛鉤都在適當的容器上運行。您可以建立多個自訂預操作掛鉤，但它們的執行順序無法保證或配置。
2. 如果適用，則會發生檔案系統凍結。
3. 執行資料保護操作。
4. 如果適用，凍結的檔案系統將被解凍。
5. NetApp Backup and Recovery在適當的容器上執行任何適用的自訂操作前執行掛鉤。您可以建立多個自訂後操作掛鉤，但它們的執行順序無法保證或配置。

如果建立多個相同類型的鉤子，則無法保證它們的執行順序。不同類型的鉤子總是按照指定的順序運行。例如，以下是具有所有不同類型鉤子的配置的執行順序：

1. 快照前鉤子執行
2. 快照後鉤子執行
3. 執行備份前掛鉤
4. 執行備份後鉤子

在生產中啟用執行掛鉤腳本之前對其進行測試。使用“kubectl exec”測試腳本，然後透過將應用程式複製到臨時命名空間並還原來驗證快照和備份。

如果快照前執行鉤子新增、變更或刪除 Kubernetes 資源，則這些變更將包含在快照或備份以及任何後續復原作業中。

關於自訂執行鉤子的重要說明

在為您的應用程式規劃執行掛鉤時，請考慮以下事項。

- 執行鉤子必須使用腳本來執行操作。許多執行鉤子可以引用同一個腳本。

- 執行鉤子需要以可執行shell腳本的格式編寫。
- 腳本大小限制為 96KB。
- 執行掛鉤設定和任何符合條件用於確定哪些掛鉤適用於快照、備份或還原操作。

執行掛鉤可以減少或停用應用程式功能。讓您的自訂鉤子盡快運行。如果您啟動具有相關執行掛鉤的備份或快照操作，但隨後取消它，則如果備份或快照操作已經開始，則仍允許掛鉤運行。這意味著備份後執行掛鉤中使用的邏輯不能假定備份已完成。

執行鉤子過濾器

當您為應用程式新增或編輯執行掛鉤時，您可以向執行掛鉤添加過濾器來管理該掛鉤將匹配哪些容器。過濾器對於在所有容器上使用相同容器鏡像但可能將每個鏡像用於不同目的的應用程式（例如 Elasticsearch）很有用。過濾器可讓您建立執行掛鉤在某些（但不一定是所有）相同的容器上運行的場景。如果為單一執行掛鉤建立多個篩選器，它們將透過邏輯 AND 運算子組合在一起。每個執行掛鉤最多可以有 10 個活動過濾器。

新增到執行掛鉤的每個過濾器都使用正規表示式來匹配叢集中的容器。當鉤子與容器匹配時，鉤子將在該容器上運行其關聯的腳本。過濾器的正規表示式使用正規表示式 2 (RE2) 語法，該語法不支援建立從符合清單中排除容器的過濾器。有關NetApp Backup and Recovery在執行鉤子過濾器中支援的正規表示式的語法的信息，請參見 "正規表示式 2 (RE2) 語法支持"。

如果將命名空間過濾器新增至在復原或複製作業後執行的執行掛鉤，且復原或複製來源和目標位於不同的命名空間中，則命名空間篩選器僅適用於目標命名空間。

執行鉤子範例

訪問 "NetApp Verda GitHub 項目"下載流行應用程式（如 Apache Cassandra 和 Elasticsearch）的真實執行掛鉤。您還可以查看範例並獲得建立自己的自訂執行掛鉤的想法。

Verda GitHub 程式碼庫中的腳本按原樣提供，不受 NetApp 官方支援。只有 Trident Protect 和 Backup and Recovery 中的執行掛鉤框架才受 NetApp 官方支援。

建立執行鉤子模板

您可以建立自訂執行掛鉤模板，用於在應用程式上執行資料保護操作之前或之後執行操作。

您在此處建立的範本僅可用於保護 Kubernetes 工作負載。

步驟

1. 在控制台中，前往*保護*>*備份和還原*。
2. 選擇“設定”標籤。
3. 展開*執行鉤子模板*部分。
4. 選擇*建立執行鉤子模板*。
5. 輸入執行掛鉤的名稱。
6. （可選）選擇一種鉤子類型。例如，還原後鉤子會在還原作業完成後執行。
7. 在 **Script** 文字方塊中，輸入要作為執行掛鉤範本的一部分執行的可執行 shell 腳本。或者，您可以選擇“上傳

腳本”來上傳腳本檔案。

8. 選擇“創建”。

建立模板後，它將出現在*執行掛鉤模板*部分的模板清單中。

在 NetApp Backup and Recovery 中建立及管理 Kubernetes 工作負載的保護報告

在 NetApp Backup and Recovery 中，為 Kubernetes 工作負載建立保護報告，以檢視保護狀態與詳細資訊，包括成功與失敗備份的數量、備份類型、叢集健康資訊等。

必要的 **NetApp Console** 角色 Backup and Recovery 超級管理員、Backup and Recovery 備份管理員或 Backup and Recovery 還原管理員。深入瞭解"備份和恢復角色和權限"。 "了解所有服務的NetApp Console存取角色"

建立保護報告

建立保護報告以檢視叢集的保護狀態。

步驟

1. 從NetApp Backup and Recovery選單中，選擇 報告 選項。
2. 選擇*建立報告*。
3. 輸入報告範圍詳細資料：
   - 報告名稱：輸入報告的唯一名稱。
   - 報告類型：選擇按帳戶或按工作負載產生報告（從清單中選擇 Kubernetes）。
   - **Select cluster**：如果您按工作負載選擇，請從清單中選擇要為其產生報告的 cluster，然後選擇 **Accept**。選擇 **Select all** 可產生所有 cluster 的報告。
4. 輸入報告範圍：選擇您希望報告包含過去一天、過去 7 天、過去 30 天、過去一個季度或過去一年的資料。
5. 輸入報表配置詳細資料：選擇報表是僅執行一次還是排程定期產生報表。對於排程報表，請選擇重複頻率並選取開始日期。
   a. 輸入電子郵件傳送詳細資料：（僅適用於排程報告）如果您希望透過電子郵件傳送報告，請輸入一個或多個應接收排程報告的電子郵件地址。

   在設定頁面配置電子郵件通知。有關配置電子郵件通知的詳細信息，請參閱"配置設定"。
6. 選擇“創建”。

下載保護報告

下載產生的保護報告，報告格式可以是 JSON 檔案或 PDF 文件，以便查看和分享。

步驟

1. 從NetApp Backup and Recovery選單中，選擇 報告 選項。
2. 在 **Reports** 頁面上，選擇 **Reports** 選單，即可查看產生的保護報告清單。

3. 若要下載報告，請選擇「操作」圖示 ••• > 下載。
    - 選擇 **Download JSON** 以下載 JSON 格式的報表。
    - 選擇 **Download PDF** 將報告下載為 PDF 文件。

檢視保護報告

可在 NetApp Backup and Recovery 內快速檢視保護報告的互動式詳細資訊。您可以查看作業摘要資訊、資料保護狀態、組態詳細資料等。

步驟

1. 從NetApp Backup and Recovery選單中，選擇 報告 選項。
2. 在 **Reports** 頁面上，選擇 **Reports** 選單，即可查看產生的保護報告清單。
3. 若要查看報告，請選擇「操作」圖示 ••• > 查看報告。

    報告詳細資料隨即顯示。

刪除保護報告

當您不再需要保護報告時，請將其刪除。

步驟

1. 從NetApp Backup and Recovery選單中，選擇 報告 選項。
2. 在 **Reports** 頁面上，選擇 **Reports** 選單，即可查看產生的保護報告清單。
3. 若要刪除報告，請選擇 Actions 圖示 ••• > **Delete** 。
4. 選擇 **Delete** 以確認操作。