Restaure las aplicaciones

Trident

NetApp
February 02, 2026

Tabla de contenidos

Restaure las aplicaciones . . . 1
- Restaurar aplicaciones usando Trident Protect . . . 1
  - Restauración desde un backup a un espacio de nombres diferente . . . 1
  - Restaure desde un backup al espacio de nombres original . . . 5
  - Restauración desde un backup en otro clúster . . . 8
  - Restauración desde una copia snapshot a un espacio de nombres diferente . . . 11
  - Restauración desde una copia Snapshot al espacio de nombres original . . . 14
  - Compruebe el estado de una operación de restauración . . . 17
- Utilice la configuración de restauración avanzada de Trident Protect . . . 17
  - Etiquetas y anotaciones del espacio de nombres durante las operaciones de restauración y conmutación al nodo de respaldo . . . 17
  - Campos admitidos . . . 19
  - Anotaciones admitidas . . . 19

Restaure las aplicaciones

Restaurar aplicaciones usando Trident Protect

Puede utilizar Trident Protect para restaurar su aplicación desde una instantánea o una copia de seguridad. Restaurar desde una instantánea existente será más rápido al restaurar la aplicación en el mismo clúster.

- Al restaurar una aplicación, todos los ganchos de ejecución configurados para la aplicación se restauran con la aplicación. Si hay un enlace de ejecución posterior a la restauración, se ejecuta automáticamente como parte de la operación de restauración.
- Se permite restaurar desde una copia de seguridad a un espacio de nombres diferente o al original en volúmenes qtree. Sin embargo, no se permite restaurar desde una instantánea a un espacio de nombres diferente o al original en volúmenes qtree.
- Puede utilizar la configuración avanzada para personalizar las operaciones de restauración. Para obtener más información, consulte "Utilice la configuración de restauración avanzada de Trident Protect".

Restauración desde un backup a un espacio de nombres diferente

Cuando restaura una copia de seguridad en un espacio de nombres diferente mediante un CR de BackupRestore, Trident Protect restaura la aplicación en un nuevo espacio de nombres y crea un CR de aplicación para la aplicación restaurada. Para proteger la aplicación restaurada, cree copias de seguridad o instantáneas bajo demanda, o establezca un programa de protección.

- Al restaurar un backup en un espacio de nombres diferente con los recursos existentes, no se alterará ningún recurso que comparta los nombres con los que aparecen en el backup. Para restaurar todos los recursos del backup, elimine y vuelva a crear el espacio de nombres objetivo, o restaure el backup en un nuevo espacio de nombres.
- Al utilizar una CR para restaurar a un nuevo espacio de nombres, debe crear manualmente el espacio de nombres de destino antes de aplicar la CR. Trident Protect crea espacios de nombres automáticamente solo cuando se usa la CLI.

Antes de empezar

Asegúrese de que la caducidad del token de sesión de AWS sea suficiente para las operaciones de restauración de S3 que se ejecuten durante mucho tiempo. Si el token caduca durante la operación de restauración, puede fallar la operación.

- Consulte el "Documentación de la API de AWS" para obtener más información sobre la comprobación de la caducidad del token de sesión actual.
- Consulte el documento para "Documentación de AWS IAM" obtener más información acerca de las credenciales con recursos de AWS.

Al restaurar copias de seguridad utilizando Kopia como transportador de datos, puede especificar opcionalmente anotaciones en la CR o usar la CLI para controlar el comportamiento del almacenamiento temporal utilizado por Kopia. Consulte el "Documentación de KOPIA" Para obtener más información sobre las opciones que puede configurar. Utilice el `tridentctl-protect create --help` Comando para obtener más información sobre cómo especificar anotaciones con la CLI de Trident Protect.

Utilice un CR

Pasos

1. Cree el archivo de recursos personalizados (CR) y asígnele un nombre `trident-protect-backup-restore-cr.yaml`.
2. En el archivo creado, configure los siguientes atributos:
   - **metadata.name**: (*required*) El nombre de este recurso personalizado; elija un nombre único y sensible para su entorno.
   - **Spec.appArchivePath**: La ruta dentro de AppVault donde se almacena el contenido de la copia de seguridad. Puede utilizar el siguiente comando para buscar esta ruta:

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

   - **Spec.appVaultRef**: (*required*) El nombre del AppVault donde se almacena el contenido de la copia de seguridad.
   - **spec.namespaceMapping**: La asignación del espacio de nombres de origen de la operación de restauración al espacio de nombres de destino. Reemplace `my-source-namespace` y `my-destination-namespace` con la información de su entorno.

     ```
     ---
     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. (*Optional*) Si necesita seleccionar solo ciertos recursos de la aplicación para restaurar, agregue filtros que incluyan o excluyan recursos marcados con etiquetas particulares:

   > Trident Protect selecciona algunos recursos automáticamente debido a su relación con los recursos que usted selecciona. Por ejemplo, si selecciona un recurso de reclamo de volumen persistente y tiene un pod asociado, Trident Protect también restaurará el pod asociado.

   - **ResourceFilter.resourceSelectionCriteria**: (Requerido para filtrar) Usar `Include` o `Exclude` incluir o excluir un recurso definido en resourceMatchers. Agregue los siguientes parámetros resourceMatchers para definir los recursos que se van a incluir o excluir:
     - **ResourceFilter.resourceMatchers**: Una matriz de objetos resourceMatcher. Si define varios elementos en esta matriz, coinciden como una OPERACIÓN OR y los campos dentro de

cada elemento (grupo, tipo, versión) coinciden como una operación AND.

- **ResourceMatchers[].group**: (*Optional*) Grupo del recurso a filtrar.
- **ResourceMatchers[].kind**: (*Optional*) Tipo de recurso a filtrar.
- **ResourceMatchers[].version**: (*Optional*) Versión del recurso que se va a filtrar.
- **ResourceMatchers[].names**: (*Optional*) Nombres en el campo Kubernetes metadata.name del recurso que se va a filtrar.
- **ResourceMatchers[].namespaces**: (*Optional*) Espacios de nombres en el campo Kubernetes metadata.name del recurso que se va a filtrar.
- **ResourceMatchers[].labelSelectors**: (*Optional*) Cadena de selector de etiquetas en el campo Kubernetes metadata.name del recurso tal como se define en el "Documentación de Kubernetes". Por ejemplo `"trident.netapp.io/os=linux"`: .

  Por ejemplo:

```
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. Después de rellenar `trident-protect-backup-restore-cr.yaml` el archivo con los valores correctos, aplique el CR:

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

Utilice la CLI

Pasos

1. Restaure la copia de seguridad en un espacio de nombres diferente, sustituyendo valores entre paréntesis por información de su entorno. El `namespace-mapping` argumento utiliza espacios de nombres separados por dos puntos para asignar espacios de nombres de origen a los espacios de nombres de destino correctos en el formato `source1:dest1,source2:dest2`. Por ejemplo:

```
tridentctl-protect create backuprestore <my_restore_name> \
--backup <backup_namespace>/<backup_to_restore> \
--namespace-mapping <source_to_destination_namespace_mapping> \
-n <application_namespace>
```

Restaure desde un backup al espacio de nombres original

Es posible restaurar un backup en el espacio de nombres original en cualquier momento.

Antes de empezar

Asegúrese de que la caducidad del token de sesión de AWS sea suficiente para las operaciones de restauración de S3 que se ejecuten durante mucho tiempo. Si el token caduca durante la operación de restauración, puede fallar la operación.

- Consulte el "Documentación de la API de AWS" para obtener más información sobre la comprobación de la caducidad del token de sesión actual.
- Consulte el documento para "Documentación de AWS IAM" obtener más información acerca de las credenciales con recursos de AWS.

> Al restaurar copias de seguridad utilizando Kopia como transportador de datos, puede especificar opcionalmente anotaciones en la CR o usar la CLI para controlar el comportamiento del almacenamiento temporal utilizado por Kopia. Consulte el "Documentación de KOPIA" Para obtener más información sobre las opciones que puede configurar. Utilice el `tridentctl-protect create --help` Comando para obtener más información sobre cómo especificar anotaciones con la CLI de Trident Protect.

Utilice un CR

Pasos

1. Cree el archivo de recursos personalizados (CR) y asígnele un nombre `trident-protect-backup-ipr-cr.yaml`.
2. En el archivo creado, configure los siguientes atributos:
   - **metadata.name**: (*required*) El nombre de este recurso personalizado; elija un nombre único y sensible para su entorno.
   - **Spec.appArchivePath**: La ruta dentro de AppVault donde se almacena el contenido de la copia de seguridad. Puede utilizar el siguiente comando para buscar esta ruta:

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

   - **Spec.appVaultRef**: (*required*) El nombre del AppVault donde se almacena el contenido de la copia de seguridad.

     Por ejemplo:

     ```
     ---
     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. (*Optional*) Si necesita seleccionar solo ciertos recursos de la aplicación para restaurar, agregue filtros que incluyan o excluyan recursos marcados con etiquetas particulares:

   > Trident Protect selecciona algunos recursos automáticamente debido a su relación con los recursos que usted selecciona. Por ejemplo, si selecciona un recurso de reclamo de volumen persistente y tiene un pod asociado, Trident Protect también restaurará el pod asociado.

   - **ResourceFilter.resourceSelectionCriteria**: (Requerido para filtrar) Usar `Include` o `Exclude` incluir o excluir un recurso definido en resourceMatchers. Agregue los siguientes parámetros resourceMatchers para definir los recursos que se van a incluir o excluir:
     - **ResourceFilter.resourceMatchers**: Una matriz de objetos resourceMatcher. Si define varios elementos en esta matriz, coinciden como una OPERACIÓN OR y los campos dentro de cada elemento (grupo, tipo, versión) coinciden como una operación AND.
       - **ResourceMatchers[].group**: (*Optional*) Grupo del recurso a filtrar.
       - **ResourceMatchers[].kind**: (*Optional*) Tipo de recurso a filtrar.

- **ResourceMatchers[].version**: (*Optional*) Versión del recurso que se va a filtrar.
- **ResourceMatchers[].names**: (*Optional*) Nombres en el campo Kubernetes metadata.name del recurso que se va a filtrar.
- **ResourceMatchers[].namespaces**: (*Optional*) Espacios de nombres en el campo Kubernetes metadata.name del recurso que se va a filtrar.
- **ResourceMatchers[].labelSelectors**: (*Optional*) Cadena de selector de etiquetas en el campo Kubernetes metadata.name del recurso tal como se define en el "Documentación de Kubernetes". Por ejemplo `"trident.netapp.io/os=linux"`: .

  Por ejemplo:

```
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. Después de rellenar `trident-protect-backup-ipr-cr.yaml` el archivo con los valores correctos, aplique el CR:

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

Utilice la CLI

Pasos

1. Restaure la copia de seguridad en el espacio de nombres original, sustituyendo valores entre paréntesis por información de su entorno. El `backup` argumento utiliza un espacio de nombres y un nombre de copia de seguridad en el formato `<namespace>/<name>`. Por ejemplo:

```
tridentctl-protect create backupinplacerestore <my_restore_name> \
--backup <namespace/backup_to_restore> \
-n <application_namespace>
```

Restauración desde un backup en otro clúster

Puede restaurar un backup a otro clúster si hay un problema con el clúster original.

- Al restaurar copias de seguridad utilizando Kopia como transportador de datos, puede especificar opcionalmente anotaciones en la CR o usar la CLI para controlar el comportamiento del almacenamiento temporal utilizado por Kopia. Consulte el "Documentación de KOPIA" Para obtener más información sobre las opciones que puede configurar. Utilice el `tridentctl-protect create --help` Comando para obtener más información sobre cómo especificar anotaciones con la CLI de Trident Protect.
- Al utilizar una CR para restaurar a un nuevo espacio de nombres, debe crear manualmente el espacio de nombres de destino antes de aplicar la CR. Trident Protect crea espacios de nombres automáticamente solo cuando se usa la CLI.

Antes de empezar

Asegúrese de que se cumplen los siguientes requisitos previos:

- El clúster de destino tiene Trident Protect instalado.
- El clúster de destino tiene acceso a la ruta de bloque de la misma AppVault que el clúster de origen, en la que se almacena el backup.
- Asegúrese de que su entorno local pueda conectarse al bucket de almacenamiento de objetos definido en el CR de AppVault al ejecutar el proceso. `tridentctl-protect get appvaultcontent` dominio. Si las restricciones de red impiden el acceso, ejecute la CLI de Trident Protect desde dentro de un pod en el clúster de destino.
- Asegúrese de que la caducidad del token de sesión de AWS sea suficiente para las operaciones de restauración que se ejecuten durante mucho tiempo. Si el token caduca durante la operación de restauración, puede fallar la operación.
  - Consulte el "Documentación de la API de AWS" para obtener más información sobre la comprobación de la caducidad del token de sesión actual.
  - Consulte el documento para "Documentación de AWS" obtener más información acerca de las credenciales con recursos de AWS.

Pasos

1. Verifique la disponibilidad de AppVault CR en el clúster de destino mediante el complemento CLI de Trident Protect:

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

   Asegúrese de que el espacio de nombres destinado para la restauración de la aplicación exista en el clúster de destino.

2. Visualice el contenido de las copias de seguridad del AppVault disponible desde el clúster de destino:

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

Al ejecutar este comando, se muestran las copias de seguridad disponibles en AppVault, incluidos sus clústeres de origen, los nombres de aplicaciones correspondientes, las marcas de tiempo y las rutas de archivo.

Ejemplo de salida:

```
+-------------+-----------+--------+-----------------
+--------------------------+-------------+
|   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. Restaure la aplicación en el clúster de destino mediante el nombre de AppVault y la ruta de archivo:

Utilice un CR

1. Cree el archivo de recursos personalizados (CR) y asígnele un nombre `trident-protect-backup-restore-cr.yaml`.
2. En el archivo creado, configure los siguientes atributos:
   - **metadata.name**: (*required*) El nombre de este recurso personalizado; elija un nombre único y sensible para su entorno.
   - **Spec.appVaultRef**: (*required*) El nombre del AppVault donde se almacena el contenido de la copia de seguridad.
   - **Spec.appArchivePath**: La ruta dentro de AppVault donde se almacena el contenido de la copia de seguridad. Puede utilizar el siguiente comando para buscar esta ruta:

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

     Si BackupRestore CR no está disponible, puede usar el comando mencionado en el paso 2 para ver el contenido de la copia de seguridad.

   - **spec.namespaceMapping**: La asignación del espacio de nombres de origen de la operación de restauración al espacio de nombres de destino. Reemplace `my-source-namespace` y `my-destination-namespace` con la información de su entorno.

     Por ejemplo:

```
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"}]
```

3. Después de rellenar `trident-protect-backup-restore-cr.yaml` el archivo con los valores correctos, aplique el CR:

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

Utilice la CLI

1. Utilice el siguiente comando para restaurar la aplicación, sustituyendo valores entre paréntesis por información de su entorno. El argumento de asignación de espacio de nombres utiliza espacios de

nombres separados por dos puntos para asignar espacios de nombres de origen a los espacios de nombres de destino correctos con el formato source1:DEST1,source2:DEST2. Por ejemplo:

```
tridentctl-protect create backuprestore <restore_name> \
--namespace-mapping <source_to_destination_namespace_mapping> \
--appvault <appvault_name> \
--path <backup_path> \
--context <destination_cluster_name> \
-n <application_namespace>
```

Restauración desde una copia snapshot a un espacio de nombres diferente

Puede restaurar datos desde una instantánea utilizando un archivo de recursos personalizados (CR) ya sea en un espacio de nombres diferente o en el espacio de nombres de origen original. Cuando restaura una instantánea a un espacio de nombres diferente mediante un CR de SnapshotRestore, Trident Protect restaura la aplicación en un nuevo espacio de nombres y crea un CR de aplicación para la aplicación restaurada. Para proteger la aplicación restaurada, cree copias de seguridad o instantáneas bajo demanda, o establezca un programa de protección.

- SnapshotRestore admite el `spec.storageClassMapping` atributo, pero solo cuando las clases de almacenamiento de origen y destino utilizan el mismo backend de almacenamiento. Si intenta restaurar a un `StorageClass` que utiliza un backend de almacenamiento diferente, la operación de restauración fallará.
- Al utilizar una CR para restaurar a un nuevo espacio de nombres, debe crear manualmente el espacio de nombres de destino antes de aplicar la CR. Trident Protect crea espacios de nombres automáticamente solo cuando se usa la CLI.

Antes de empezar

Asegúrese de que la caducidad del token de sesión de AWS sea suficiente para las operaciones de restauración de S3 que se ejecuten durante mucho tiempo. Si el token caduca durante la operación de restauración, puede fallar la operación.

- Consulte el "Documentación de la API de AWS" para obtener más información sobre la comprobación de la caducidad del token de sesión actual.
- Consulte el documento para "Documentación de AWS IAM" obtener más información acerca de las credenciales con recursos de AWS.

Utilice un CR

Pasos

1. Cree el archivo de recursos personalizados (CR) y asígnele un nombre `trident-protect-snapshot-restore-cr.yaml`.
2. En el archivo creado, configure los siguientes atributos:
   - **metadata.name**: (*required*) El nombre de este recurso personalizado; elija un nombre único y sensible para su entorno.
   - **Spec.appVaultRef**: (*required*) El nombre del AppVault donde se almacena el contenido de la instantánea.
   - **Spec.appArchivePath**: La ruta dentro de AppVault donde se almacena el contenido de la instantánea. Puede utilizar el siguiente comando para buscar esta ruta:

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

   - **spec.namespaceMapping**: La asignación del espacio de nombres de origen de la operación de restauración al espacio de nombres de destino. Reemplace `my-source-namespace` y `my-destination-namespace` con la información de su entorno.

     ```
     ---
     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. (*Optional*) Si necesita seleccionar solo ciertos recursos de la aplicación para restaurar, agregue filtros que incluyan o excluyan recursos marcados con etiquetas particulares:

   > Trident Protect selecciona algunos recursos automáticamente debido a su relación con los recursos que usted selecciona. Por ejemplo, si selecciona un recurso de reclamo de volumen persistente y tiene un pod asociado, Trident Protect también restaurará el pod asociado.

   - **ResourceFilter.resourceSelectionCriteria**: (Requerido para filtrar) Usar `Include` o `Exclude` incluir o excluir un recurso definido en resourceMatchers. Agregue los siguientes parámetros resourceMatchers para definir los recursos que se van a incluir o excluir:
     - **ResourceFilter.resourceMatchers**: Una matriz de objetos resourceMatcher. Si define varios elementos en esta matriz, coinciden como una OPERACIÓN OR y los campos dentro de

cada elemento (grupo, tipo, versión) coinciden como una operación AND.

- **ResourceMatchers[].group**: (*Optional*) Grupo del recurso a filtrar.
- **ResourceMatchers[].kind**: (*Optional*) Tipo de recurso a filtrar.
- **ResourceMatchers[].version**: (*Optional*) Versión del recurso que se va a filtrar.
- **ResourceMatchers[].names**: (*Optional*) Nombres en el campo Kubernetes metadata.name del recurso que se va a filtrar.
- **ResourceMatchers[].namespaces**: (*Optional*) Espacios de nombres en el campo Kubernetes metadata.name del recurso que se va a filtrar.
- **ResourceMatchers[].labelSelectors**: (*Optional*) Cadena de selector de etiquetas en el campo Kubernetes metadata.name del recurso tal como se define en el "Documentación de Kubernetes". Por ejemplo `"trident.netapp.io/os=linux"`: .

  Por ejemplo:

```
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. Después de rellenar `trident-protect-snapshot-restore-cr.yaml` el archivo con los valores correctos, aplique el CR:

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

Utilice la CLI

Pasos

1. Restaure la instantánea en un espacio de nombres diferente, reemplazando los valores entre paréntesis por información de su entorno.

   - El `snapshot` argumento utiliza un espacio de nombres y un nombre de instantánea en el formato `<namespace>/<name>`.
   - El `namespace-mapping` argumento utiliza espacios de nombres separados por dos puntos para

asignar espacios de nombres de origen a los espacios de nombres de destino correctos en el formato `source1:dest1,source2:dest2`.

Por ejemplo:

```
tridentctl-protect create snapshotrestore <my_restore_name> \
--snapshot <namespace/snapshot_to_restore> \
--namespace-mapping <source_to_destination_namespace_mapping> \
-n <application_namespace>
```

Restauración desde una copia Snapshot al espacio de nombres original

Es posible restaurar una copia de Snapshot en el espacio de nombres original en cualquier momento.

Antes de empezar

Asegúrese de que la caducidad del token de sesión de AWS sea suficiente para las operaciones de restauración de S3 que se ejecuten durante mucho tiempo. Si el token caduca durante la operación de restauración, puede fallar la operación.

- Consulte el "Documentación de la API de AWS" para obtener más información sobre la comprobación de la caducidad del token de sesión actual.
- Consulte el documento para "Documentación de AWS IAM" obtener más información acerca de las credenciales con recursos de AWS.

Utilice un CR

Pasos

1. Cree el archivo de recursos personalizados (CR) y asígnele un nombre `trident-protect-snapshot-ipr-cr.yaml`.
2. En el archivo creado, configure los siguientes atributos:
   - **metadata.name**: (*required*) El nombre de este recurso personalizado; elija un nombre único y sensible para su entorno.
   - **Spec.appVaultRef**: (*required*) El nombre del AppVault donde se almacena el contenido de la instantánea.
   - **Spec.appArchivePath**: La ruta dentro de AppVault donde se almacena el contenido de la instantánea. Puede utilizar el siguiente comando para buscar esta ruta:

```
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. (*Optional*) Si necesita seleccionar solo ciertos recursos de la aplicación para restaurar, agregue filtros que incluyan o excluyan recursos marcados con etiquetas particulares:

> Trident Protect selecciona algunos recursos automáticamente debido a su relación con los recursos que usted selecciona. Por ejemplo, si selecciona un recurso de reclamo de volumen persistente y tiene un pod asociado, Trident Protect también restaurará el pod asociado.

   - **ResourceFilter.resourceSelectionCriteria**: (Requerido para filtrar) Usar `Include` o `Exclude` incluir o excluir un recurso definido en resourceMatchers. Agregue los siguientes parámetros resourceMatchers para definir los recursos que se van a incluir o excluir:
     - **ResourceFilter.resourceMatchers**: Una matriz de objetos resourceMatcher. Si define varios elementos en esta matriz, coinciden como una OPERACIÓN OR y los campos dentro de cada elemento (grupo, tipo, versión) coinciden como una operación AND.
       - **ResourceMatchers[].group**: (*Optional*) Grupo del recurso a filtrar.
       - **ResourceMatchers[].kind**: (*Optional*) Tipo de recurso a filtrar.
       - **ResourceMatchers[].version**: (*Optional*) Versión del recurso que se va a filtrar.
       - **ResourceMatchers[].names**: (*Optional*) Nombres en el campo Kubernetes

metadata.name del recurso que se va a filtrar.

- **ResourceMatchers[].namespaces**: (*Optional*) Espacios de nombres en el campo Kubernetes metadata.name del recurso que se va a filtrar.
- **ResourceMatchers[].labelSelectors**: (*Optional*) Cadena de selector de etiquetas en el campo Kubernetes metadata.name del recurso tal como se define en el "Documentación de Kubernetes". Por ejemplo `"trident.netapp.io/os=linux"`: .

  Por ejemplo:

```
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. Después de rellenar `trident-protect-snapshot-ipr-cr.yaml` el archivo con los valores correctos, aplique el CR:

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

Utilice la CLI

Pasos

1. Restaure la instantánea en el espacio de nombres original, reemplazando los valores entre paréntesis por información de su entorno. Por ejemplo:

```
tridentctl-protect create snapshotinplacerestore <my_restore_name> \
--snapshot <namespace/snapshot_to_restore> \
-n <application_namespace>
```

Compruebe el estado de una operación de restauración

Puede usar la línea de comandos para comprobar el estado de una operación de restauración en curso, que se completó o con errores.

Pasos

1. Utilice el siguiente comando para recuperar el estado de la operación de restauración, sustituyendo valores de entre corchetes con información de su entorno:

```
kubectl get backuprestore -n <namespace_name> <my_restore_cr_name> -o
jsonpath='{.status}'
```

Utilice la configuración de restauración avanzada de Trident Protect

Puede personalizar las operaciones de restauración utilizando configuraciones avanzadas, como anotaciones, configuraciones de espacios de nombres y opciones de almacenamiento para satisfacer sus requisitos específicos.

Etiquetas y anotaciones del espacio de nombres durante las operaciones de restauración y conmutación al nodo de respaldo

Durante las operaciones de restauración y conmutación al nodo de respaldo, se realizan etiquetas y anotaciones en el espacio de nombres de destino que coincidan con las etiquetas y anotaciones en el espacio de nombres de origen. Se añaden etiquetas o anotaciones del espacio de nombres origen que no existen en el espacio de nombres destino, y las etiquetas o anotaciones que ya existan se sobrescriben para que coincidan con el valor del espacio de nombres origen. Las etiquetas o anotaciones que sólo existen en el espacio de nombres de destino permanecen sin cambios.

> Si utiliza Red Hat OpenShift, es importante tener en cuenta el papel fundamental que desempeñan las anotaciones de espacios de nombres en los entornos OpenShift. Las anotaciones de espacio de nombres garantizan que los pods restaurados cumplan con los permisos y las configuraciones de seguridad adecuados definidos por las restricciones de contexto de seguridad (SCC) de OpenShift y puedan acceder a los volúmenes sin problemas de permisos. Para obtener más información, consulte la"Documentación de restricciones de contexto de seguridad de OpenShift" .

Puede evitar que se sobrescriban anotaciones específicas en el espacio de nombres de destino mediante el establecimiento de la variable de entorno de Kubernetes `RESTORE_SKIP_NAMESPACE_ANNOTATIONS` antes de llevar a cabo la operación de restauración o conmutación por error. Por ejemplo:

```
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
```

Al realizar una operación de restauración o conmutación por error, se tendrán en cuenta las anotaciones y etiquetas de espacio de nombres especificadas en `restoreSkipNamespaceAnnotations` y `restoreSkipNamespaceLabels` quedan excluidos de la operación de restauración o conmutación por error. Asegúrese de que estos ajustes se configuren durante la instalación inicial de Helm. Para obtener más información, consulte "Configurar ajustes adicionales del gráfico de timón Trident Protect".

Si instalaste la aplicación de origen usando Helm con el `--create-namespace` bandera, se le da un trato especial a la `name` Clave de etiqueta. Durante el proceso de restauración o conmutación por error, Trident Protect copia esta etiqueta en el espacio de nombres de destino, pero actualiza el valor al valor del espacio de nombres de destino si el valor del origen coincide con el espacio de nombres de origen. Si este valor no coincide con el espacio de nombres de origen, se copia al espacio de nombres de destino sin cambios.

Ejemplo

El siguiente ejemplo presenta un espacio de nombres de origen y destino, cada uno con anotaciones y etiquetas diferentes. Puede ver el estado del espacio de nombres de destino antes y después de la operación, así como cómo las anotaciones y etiquetas se combinan o sobrescriben en el espacio de nombres de destino.

Antes de la operación de restauración o conmutación por error

En la siguiente tabla se muestra el estado del ejemplo de espacios de nombres de origen y destino antes de la operación de restauración o conmutación por error:

Espacio de nombres	Anotaciones	Etiquetas
Espacio de nombres ns-1 (origen)	• anotación.uno/clave: “updatedvalue” • anotación.dos/clave: “verdadero”	• entorno=producción • cumplimiento=hipaa • name=ns-1
Espacio de nombres ns-2 (destino)	• anotación.uno/tecla: “verdadero” • anotación.tres/clave: “falso”	• role=base de datos

Después de la operación de restauración

En la siguiente tabla se muestra el estado del espacio de nombres de destino de ejemplo después de la operación de restauración o conmutación por error. Se han agregado algunas claves, algunas se han sobrescrito y la `name` etiqueta se ha actualizado para que coincida con el espacio de nombres de destino:

Espacio de nombres	Anotaciones	Etiquetas
Espacio de nombres ns-2 (destino)	• anotación.uno/clave: “updatedvalue” • anotación.dos/clave: “verdadero” • anotación.tres/clave: “falso”	• name=ns-2 • cumplimiento=hipaa • entorno=producción • role=base de datos

Campos admitidos

Esta sección describe campos adicionales disponibles para operaciones de restauración.

Mapeo de clases de almacenamiento

El `spec.storageClassMapping` El atributo define una asignación de una clase de almacenamiento presente en la aplicación de origen a una nueva clase de almacenamiento en el clúster de destino. Puede usar esto al migrar aplicaciones entre clústeres con diferentes clases de almacenamiento o al cambiar el backend de almacenamiento para operaciones de BackupRestore.

Ejemplo:

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

Anotaciones admitidas

Esta sección enumera las anotaciones compatibles para configurar diversos comportamientos en el sistema. Si el usuario no configura una anotación explícitamente, el sistema utilizará el valor predeterminado.

Anotación	Tipo	Descripción	Valor predeterminado
protect.trident.netapp.io/tiempo-de-espera-del-transportador-de-datos-sec	cadena	El tiempo máximo (en segundos) permitido para que la operación de transferencia de datos permanezca detenida.	"300"
protect.trident.netapp.io/kopia-content-cache-size-limit-mb	cadena	El límite de tamaño máximo (en megabytes) para el caché de contenido de Kopia.	"1000"

Anotación	Tipo	Descripción	Valor predeterminado
protect.trident.netapp.io/pvc-bind-timeout-sec	cadena	Tiempo máximo (en segundos) que se debe esperar para que cualquier PersistentVolumeClaim (PVC) recién creado llegue al `Bound` fase antes de que fallen las operaciones. Se aplica a todos los tipos de restauración de CR (BackupRestore, BackupInplaceRestore, SnapshotRestore, SnapshotInplaceRestore). Utilice un valor más alto si su backend de almacenamiento o clúster requiere con frecuencia más tiempo.	"1200" (20 minutos)