



Restaurer les applications

Trident

NetApp
January 15, 2026

Sommaire

- Restaurer les applications 1
 - Restaurez les applications à l'aide de Trident Protect 1
 - Restaurer à partir d'une sauvegarde vers un espace de noms différent 1
 - Restaurer à partir d'une sauvegarde vers l'espace de noms d'origine 4
 - Restaurer à partir d'une sauvegarde vers un cluster différent 7
 - Restaurer à partir d'un instantané vers un espace de noms différent 10
 - Restaurer à partir d'un instantané vers l'espace de noms d'origine 13
 - Vérifier l'état d'une opération de restauration 15
 - Utilisez les paramètres de restauration avancés de Trident Protect 16
 - Annotations et étiquettes d'espace de noms lors des opérations de restauration et de basculement ... 16
 - Champs pris en charge 17
 - Annotations prises en charge 18

Restaurer les applications

Restaurez les applications à l'aide de Trident Protect

Vous pouvez utiliser Trident Protect pour restaurer votre application à partir d'un instantané ou d'une sauvegarde. La restauration à partir d'un instantané existant sera plus rapide lors de la restauration de l'application sur le même cluster.



- Lorsque vous restaurez une application, tous les points d'exécution configurés pour celle-ci sont restaurés avec elle. Si un point d'entrée d'exécution post-restauration est présent, il s'exécute automatiquement dans le cadre de l'opération de restauration.
- La restauration à partir d'une sauvegarde vers un espace de noms différent ou vers l'espace de noms d'origine est prise en charge pour les volumes qtree. Cependant, la restauration à partir d'un instantané vers un espace de noms différent ou vers l'espace de noms d'origine n'est pas prise en charge pour les volumes qtree.
- Vous pouvez utiliser des paramètres avancés pour personnaliser les opérations de restauration. Pour en savoir plus, consultez ["Utilisez les paramètres de restauration avancés de Trident Protect"](#).

Restaurer à partir d'une sauvegarde vers un espace de noms différent

Lorsque vous restaurez une sauvegarde dans un espace de noms différent à l'aide d'une ressource personnalisée BackupRestore, Trident Protect restaure l'application dans un nouvel espace de noms et crée une ressource personnalisée d'application pour l'application restaurée. Pour protéger l'application restaurée, créez des sauvegardes ou des instantanés à la demande, ou établissez un calendrier de protection.



La restauration d'une sauvegarde dans un espace de noms différent contenant des ressources existantes ne modifiera pas les ressources qui partagent les mêmes noms que celles de la sauvegarde. Pour restaurer toutes les ressources de la sauvegarde, supprimez et recréez l'espace de noms cible, ou restaurez la sauvegarde dans un nouvel espace de noms.

Avant de commencer

Assurez-vous que la durée d'expiration du jeton de session AWS est suffisante pour toute opération de restauration S3 de longue durée. Si le jeton expire pendant l'opération de restauration, celle-ci peut échouer.

- Se référer à ["Documentation AWS API"](#) pour plus d'informations sur la vérification de l'expiration du jeton de session actuel.
- Se référer à ["Documentation AWS IAM"](#) pour plus d'informations sur les identifiants avec les ressources AWS.



Lorsque vous restaurez des sauvegardes à l'aide de Kopia comme outil de transfert de données, vous pouvez éventuellement spécifier des annotations dans le CR ou utiliser l'interface de ligne de commande pour contrôler le comportement du stockage temporaire utilisé par Kopia. Se référer à ["Documentation Kopia"](#) pour plus d'informations sur les options que vous pouvez configurer. Utilisez le `tridentctl-protect create --help` commande pour plus d'informations sur la spécification des annotations avec l'interface de ligne de commande Trident Protect.

Utilisez un CR

Étapes

1. Créez le fichier de ressource personnalisée (CR) et nommez-le. `trident-protect-backup-restore-cr.yaml`.
2. Dans le fichier que vous avez créé, configurez les attributs suivants :
 - **metadata.name** : (*Obligatoire*) Le nom de cette ressource personnalisée ; choisissez un nom unique et pertinent pour votre environnement.
 - **spec.appArchivePath** : Chemin d'accès dans AppVault où sont stockés les contenus de sauvegarde. Vous pouvez utiliser la commande suivante pour trouver ce chemin :

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

- **spec.appVaultRef**: (*Obligatoire*) Le nom de l'AppVault où sont stockés les contenus de sauvegarde.
- **spec.namespaceMapping** : Le mappage de l'espace de noms source de l'opération de restauration vers l'espace de noms de destination. Remplacer `my-source-namespace` et `my-destination-namespace` avec des informations provenant de votre environnement.

```
---  
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. (*Facultatif*) Si vous devez sélectionner uniquement certaines ressources de l'application à restaurer, ajoutez un filtrage qui inclut ou exclut les ressources marquées par des étiquettes particulières :



Trident Protect sélectionne automatiquement certaines ressources en raison de leur relation avec les ressources que vous sélectionnez. Par exemple, si vous sélectionnez une ressource de revendication de volume persistant et qu'elle possède un pod associé, Trident Protect restaurera également le pod associé.

- **resourceFilter.resourceSelectionCriteria** : (Requis pour le filtrage) Utiliser `Include` ou `Exclude` inclure ou exclure une ressource définie dans `resourceMatchers`. Ajoutez les paramètres `resourceMatchers` suivants pour définir les ressources à inclure ou à exclure :
 - **resourceFilter.resourceMatchers** : Un tableau d'objets `resourceMatcher`. Si vous définissez plusieurs éléments dans ce tableau, ils correspondent selon une opération OU, et les champs

à l'intérieur de chaque élément (groupe, type, version) correspondent selon une opération ET.

- **resourceMatchers[].group:** (*Optionnel*) Groupe de la ressource à filtrer.
- **resourceMatchers[].kind:** (*Optionnel*) Type de ressource à filtrer.
- **resourceMatchers[].version:** (*Optionnel*) Version de la ressource à filtrer.
- **resourceMatchers[].names:** (*Optionnel*) Noms dans le champ metadata.name de Kubernetes de la ressource à filtrer.
- **resourceMatchers[].namespaces:** (*Optionnel*) Espaces de noms dans le champ metadata.name de Kubernetes de la ressource à filtrer.
- **resourceMatchers[].labelSelectors :** (*Facultatif*) Chaîne de sélection d'étiquette dans le champ metadata.name de la ressource Kubernetes, telle que définie dans le ["Documentation Kubernetes"](#) . Par exemple: "trident.netapp.io/os=linux" .

Par exemple:

```
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. Après avoir rempli le trident-protect-backup-restore-cr.yaml fichier contenant les valeurs correctes, appliquer le CR :

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

Utiliser la CLI

Étapes

1. Restaurez la sauvegarde dans un espace de noms différent, en remplaçant les valeurs entre crochets par les informations de votre environnement. Le namespace-mapping L'argument utilise des espaces de noms séparés par deux-points pour associer les espaces de noms source aux espaces de noms de destination corrects au format source1:dest1, source2:dest2 . Par exemple:

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

Restaurer à partir d'une sauvegarde vers l'espace de noms d'origine

Vous pouvez restaurer une sauvegarde dans l'espace de noms d'origine à tout moment.

Avant de commencer

Assurez-vous que la durée d'expiration du jeton de session AWS est suffisante pour toute opération de restauration S3 de longue durée. Si le jeton expire pendant l'opération de restauration, celle-ci peut échouer.

- Se référer à "[Documentation AWS API](#)" pour plus d'informations sur la vérification de l'expiration du jeton de session actuel.
- Se référer à "[Documentation AWS IAM](#)" pour plus d'informations sur les identifiants avec les ressources AWS.



Lorsque vous restaurez des sauvegardes à l'aide de Kopia comme outil de transfert de données, vous pouvez éventuellement spécifier des annotations dans le CR ou utiliser l'interface de ligne de commande pour contrôler le comportement du stockage temporaire utilisé par Kopia. Se référer à "[Documentation Kopia](#)" pour plus d'informations sur les options que vous pouvez configurer. Utilisez le `tridentctl-protect create --help` commande pour plus d'informations sur la spécification des annotations avec l'interface de ligne de commande Trident Protect.

Utilisez un CR

Étapes

1. Créez le fichier de ressource personnalisée (CR) et nommez-le. `trident-protect-backup-ipr-cr.yaml`.
2. Dans le fichier que vous avez créé, configurez les attributs suivants :

- **metadata.name** : (*Obligatoire*) Le nom de cette ressource personnalisée ; choisissez un nom unique et pertinent pour votre environnement.
- **spec.appArchivePath** : Chemin d'accès dans AppVault où sont stockés les contenus de sauvegarde. Vous pouvez utiliser la commande suivante pour trouver ce chemin :

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

- **spec.appVaultRef**: (*Obligatoire*) Le nom de l'AppVault où sont stockés les contenus de sauvegarde.

Par exemple:

```
---
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. (*Facultatif*) Si vous devez sélectionner uniquement certaines ressources de l'application à restaurer, ajoutez un filtrage qui inclut ou exclut les ressources marquées par des étiquettes particulières :



Trident Protect sélectionne automatiquement certaines ressources en raison de leur relation avec les ressources que vous sélectionnez. Par exemple, si vous sélectionnez une ressource de revendication de volume persistant et qu'elle possède un pod associé, Trident Protect restaurera également le pod associé.

- **resourceFilter.resourceSelectionCriteria** : (Requis pour le filtrage) Utiliser `Include` ou `Exclude` inclure ou exclure une ressource définie dans `resourceMatchers`. Ajoutez les paramètres `resourceMatchers` suivants pour définir les ressources à inclure ou à exclure :
 - **resourceFilter.resourceMatchers** : Un tableau d'objets `resourceMatcher`. Si vous définissez plusieurs éléments dans ce tableau, ils correspondent selon une opération OU, et les champs à l'intérieur de chaque élément (groupe, type, version) correspondent selon une opération ET.
 - **resourceMatchers[].group**: (*Optionnel*) Groupe de la ressource à filtrer.
 - **resourceMatchers[].kind**: (*Optionnel*) Type de ressource à filtrer.

- **resourceMatchers[].version:** (*Optionnel*) Version de la ressource à filtrer.
- **resourceMatchers[].names:** (*Optionnel*) Noms dans le champ metadata.name de Kubernetes de la ressource à filtrer.
- **resourceMatchers[].namespaces:** (*Optionnel*) Espaces de noms dans le champ metadata.name de Kubernetes de la ressource à filtrer.
- **resourceMatchers[].labelSelectors :** (*Facultatif*) Chaîne de sélection d'étiquette dans le champ metadata.name de la ressource Kubernetes, telle que définie dans le ["Documentation Kubernetes"](#) . Par exemple: "trident.netapp.io/os=linux" .

Par exemple:

```
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. Après avoir rempli le trident-protect-backup-ipr-cr.yaml fichier contenant les valeurs correctes, appliquer le CR :

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

Utiliser la CLI

Étapes

1. Restaurez la sauvegarde dans l'espace de noms d'origine, en remplaçant les valeurs entre crochets par les informations de votre environnement. Le backup L'argument utilise un espace de noms et un nom de sauvegarde au format <namespace>/<name> . Par exemple:

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

Restaurer à partir d'une sauvegarde vers un cluster différent

Vous pouvez restaurer une sauvegarde sur un autre cluster en cas de problème avec le cluster d'origine.



Lorsque vous restaurez des sauvegardes à l'aide de Kopia comme outil de transfert de données, vous pouvez éventuellement spécifier des annotations dans le CR ou utiliser l'interface de ligne de commande pour contrôler le comportement du stockage temporaire utilisé par Kopia. Se référer à "[Documentation Kopia](#)" pour plus d'informations sur les options que vous pouvez configurer. Utilisez le `tridentctl-protect create --help` commande pour plus d'informations sur la spécification des annotations avec l'interface de ligne de commande Trident Protect.

Avant de commencer

Assurez-vous que les conditions préalables suivantes sont remplies :

- Le cluster de destination possède Trident Protect installé.
- Le cluster de destination a accès au chemin du compartiment du même AppVault que le cluster source, où la sauvegarde est stockée.
- Assurez-vous que votre environnement local peut se connecter au compartiment de stockage d'objets défini dans la ressource personnalisée AppVault lors de l'exécution. `tridentctl-protect get appvaultcontent` commande. Si des restrictions réseau empêchent l'accès, exécutez plutôt l'interface de ligne de commande Trident Protect depuis un pod sur le cluster de destination.
- Assurez-vous que la durée de validité du jeton de session AWS est suffisante pour toute opération de restauration de longue durée. Si le jeton expire pendant l'opération de restauration, celle-ci peut échouer.
 - Se référer à "[Documentation AWS API](#)" pour plus d'informations sur la vérification de l'expiration du jeton de session actuel.
 - Se référer à "[Documentation AWS](#)" pour plus d'informations sur les identifiants avec les ressources AWS.

Étapes

1. Vérifiez la disponibilité de la ressource personnalisée AppVault sur le cluster de destination à l'aide du plugin CLI Trident Protect :

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



Assurez-vous que l'espace de noms destiné à la restauration de l'application existe sur le cluster de destination.

2. Consultez le contenu de la sauvegarde de l'AppVault disponible sur le cluster de destination :

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

L'exécution de cette commande affiche les sauvegardes disponibles dans AppVault, y compris leurs clusters d'origine, les noms des applications correspondantes, les horodatages et les chemins d'accès aux

archives.

Exemple de résultat :

+-----+-----+-----+-----+				
+-----+-----+-----+-----+				
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. Restaurez l'application sur le cluster de destination en utilisant le nom AppVault et le chemin d'accès à l'archive :

Utilisez un CR

1. Créez le fichier de ressource personnalisée (CR) et nommez-le. `trident-protect-backup-restore-cr.yaml`.
2. Dans le fichier que vous avez créé, configurez les attributs suivants :
 - **metadata.name** : (*Obligatoire*) Le nom de cette ressource personnalisée ; choisissez un nom unique et pertinent pour votre environnement.
 - **spec.appVaultRef** : (*Obligatoire*) Le nom de l'AppVault où sont stockés les contenus de sauvegarde.
 - **spec.appArchivePath** : Chemin d'accès dans AppVault où sont stockés les contenus de sauvegarde. Vous pouvez utiliser la commande suivante pour trouver ce chemin :

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



Si BackupRestore CR n'est pas disponible, vous pouvez utiliser la commande mentionnée à l'étape 2 pour afficher le contenu de la sauvegarde.

- **spec.namespaceMapping** : Le mappage de l'espace de noms source de l'opération de restauration vers l'espace de noms de destination. Remplacer `my-source-namespace` et `my-destination-namespace` avec des informations provenant de votre environnement.

Par exemple:

```
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. Après avoir rempli le `trident-protect-backup-restore-cr.yaml` fichier contenant les valeurs correctes, appliquer le CR :

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

Utiliser la CLI

1. Utilisez la commande suivante pour restaurer l'application, en remplaçant les valeurs entre crochets par les informations provenant de votre environnement. L'argument `namespace-mapping` utilise des

espaces de noms séparés par deux-points pour mapper les espaces de noms sources aux espaces de noms de destination corrects au format `source1:dest1,source2:dest2`. Par exemple:

```
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>
```

Restaurer à partir d'un instantané vers un espace de noms différent

Vous pouvez restaurer des données à partir d'un instantané à l'aide d'un fichier de ressources personnalisé (CR), soit dans un espace de noms différent, soit dans l'espace de noms source d'origine. Lorsque vous restaurez un instantané dans un espace de noms différent à l'aide d'une ressource personnalisée `SnapshotRestore`, Trident Protect restaure l'application dans un nouvel espace de noms et crée une ressource personnalisée d'application pour l'application restaurée. Pour protéger l'application restaurée, créez des sauvegardes ou des instantanés à la demande, ou établissez un calendrier de protection.



`SnapshotRestore` prend en charge `spec.storageClassMapping` cet attribut est valable uniquement lorsque les classes de stockage source et de destination utilisent le même système de stockage dorsal. Si vous tentez de restaurer un `StorageClass` Si le système de stockage utilisé est différent, l'opération de restauration échouera.

Avant de commencer

Assurez-vous que la durée d'expiration du jeton de session AWS est suffisante pour toute opération de restauration S3 de longue durée. Si le jeton expire pendant l'opération de restauration, celle-ci peut échouer.

- Se référer à "[Documentation AWS API](#)" pour plus d'informations sur la vérification de l'expiration du jeton de session actuel.
- Se référer à "[Documentation AWS IAM](#)" pour plus d'informations sur les identifiants avec les ressources AWS.

Utilisez un CR

Étapes

1. Créez le fichier de ressource personnalisée (CR) et nommez-le. `trident-protect-snapshot-restore-cr.yaml`.
2. Dans le fichier que vous avez créé, configurez les attributs suivants :
 - **metadata.name** : (*Obligatoire*) Le nom de cette ressource personnalisée ; choisissez un nom unique et pertinent pour votre environnement.
 - **spec.appVaultRef** : (*Obligatoire*) Le nom de l'AppVault où le contenu de l'instantané est stocké.
 - **spec.appArchivePath** : Le chemin d'accès dans AppVault où le contenu des instantanés est stocké. Vous pouvez utiliser la commande suivante pour trouver ce chemin :

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

- **spec.namespaceMapping** : Le mappage de l'espace de noms source de l'opération de restauration vers l'espace de noms de destination. Remplacer `my-source-namespace` et `my-destination-namespace` avec des informations provenant de votre environnement.

```
---  
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. (*Facultatif*) Si vous devez sélectionner uniquement certaines ressources de l'application à restaurer, ajoutez un filtrage qui inclut ou exclut les ressources marquées par des étiquettes particulières :



Trident Protect sélectionne automatiquement certaines ressources en raison de leur relation avec les ressources que vous sélectionnez. Par exemple, si vous sélectionnez une ressource de revendication de volume persistant et qu'elle possède un pod associé, Trident Protect restaurera également le pod associé.

- **resourceFilter.resourceSelectionCriteria** : (Requis pour le filtrage) Utiliser `Include` ou `Exclude` inclure ou exclure une ressource définie dans `resourceMatchers`. Ajoutez les paramètres `resourceMatchers` suivants pour définir les ressources à inclure ou à exclure :
 - **resourceFilter.resourceMatchers** : Un tableau d'objets `resourceMatcher`. Si vous définissez plusieurs éléments dans ce tableau, ils correspondent selon une opération OU, et les champs à l'intérieur de chaque élément (groupe, type, version) correspondent selon une opération ET.

- **resourceMatchers[].group:** (*Optionnel*) Groupe de la ressource à filtrer.
- **resourceMatchers[].kind:** (*Optionnel*) Type de ressource à filtrer.
- **resourceMatchers[].version:** (*Optionnel*) Version de la ressource à filtrer.
- **resourceMatchers[].names:** (*Optionnel*) Noms dans le champ metadata.name de Kubernetes de la ressource à filtrer.
- **resourceMatchers[].namespaces:** (*Optionnel*) Espaces de noms dans le champ metadata.name de Kubernetes de la ressource à filtrer.
- **resourceMatchers[].labelSelectors :** (*Facultatif*) Chaîne de sélection d'étiquette dans le champ metadata.name de la ressource Kubernetes, telle que définie dans le ["Documentation Kubernetes"](#) . Par exemple: "trident.netapp.io/os=linux" .

Par exemple:

```
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. Après avoir rempli le trident-protect-snapshot-restore-cr.yaml fichier contenant les valeurs correctes, appliquer le CR :

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

Utiliser la CLI

Étapes

1. Restaurez l'instantané dans un espace de noms différent, en remplaçant les valeurs entre crochets par les informations de votre environnement.
 - Le snapshot L'argument utilise un espace de noms et un nom d'instantané au format <namespace>/<name> .
 - Le namespace-mapping L'argument utilise des espaces de noms séparés par deux-points pour associer les espaces de noms source aux espaces de noms de destination corrects au format

```
source1:dest1,source2:dest2.
```

Par exemple:

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

Restaurer à partir d'un instantané vers l'espace de noms d'origine

Vous pouvez restaurer un instantané dans l'espace de noms d'origine à tout moment.

Avant de commencer

Assurez-vous que la durée d'expiration du jeton de session AWS est suffisante pour toute opération de restauration S3 de longue durée. Si le jeton expire pendant l'opération de restauration, celle-ci peut échouer.

- Se référer à "[Documentation AWS API](#)" pour plus d'informations sur la vérification de l'expiration du jeton de session actuel.
- Se référer à "[Documentation AWS IAM](#)" pour plus d'informations sur les identifiants avec les ressources AWS.

Utilisez un CR

Étapes

1. Créez le fichier de ressource personnalisée (CR) et nommez-le. `trident-protect-snapshot-ipr-cr.yaml`.
2. Dans le fichier que vous avez créé, configurez les attributs suivants :
 - **metadata.name** : (*Obligatoire*) Le nom de cette ressource personnalisée ; choisissez un nom unique et pertinent pour votre environnement.
 - **spec.appVaultRef** : (*Obligatoire*) Le nom de l'AppVault où le contenu de l'instantané est stocké.
 - **spec.appArchivePath** : Le chemin d'accès dans AppVault où le contenu des instantanés est stocké. Vous pouvez utiliser la commande suivante pour trouver ce chemin :

```
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. (*Facultatif*) Si vous devez sélectionner uniquement certaines ressources de l'application à restaurer, ajoutez un filtrage qui inclut ou exclut les ressources marquées par des étiquettes particulières :



Trident Protect sélectionne automatiquement certaines ressources en raison de leur relation avec les ressources que vous sélectionnez. Par exemple, si vous sélectionnez une ressource de revendication de volume persistant et qu'elle possède un pod associé, Trident Protect restaurera également le pod associé.

- **resourceFilter.resourceSelectionCriteria** : (Requis pour le filtrage) Utiliser `Include` ou `Exclude` inclure ou exclure une ressource définie dans `resourceMatchers`. Ajoutez les paramètres `resourceMatchers` suivants pour définir les ressources à inclure ou à exclure :
 - **resourceFilter.resourceMatchers** : Un tableau d'objets `resourceMatcher`. Si vous définissez plusieurs éléments dans ce tableau, ils correspondent selon une opération OU, et les champs à l'intérieur de chaque élément (groupe, type, version) correspondent selon une opération ET.
 - **resourceMatchers[].group** : (*Optionnel*) Groupe de la ressource à filtrer.
 - **resourceMatchers[].kind** : (*Optionnel*) Type de ressource à filtrer.
 - **resourceMatchers[].version** : (*Optionnel*) Version de la ressource à filtrer.
 - **resourceMatchers[].names** : (*Optionnel*) Noms dans le champ `metadata.name` de Kubernetes de la ressource à filtrer.

- **resourceMatchers[].namespaces**: (*Optionnel*) Espaces de noms dans le champ metadata.name de Kubernetes de la ressource à filtrer.
- **resourceMatchers[].labelSelectors** : (*Facultatif*) Chaîne de sélection d'étiquette dans le champ metadata.name de la ressource Kubernetes, telle que définie dans le ["Documentation Kubernetes"](#) . Par exemple: "trident.netapp.io/os=linux" .

Par exemple:

```
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. Après avoir rempli le trident-protect-snapshot-ipr-cr.yaml fichier contenant les valeurs correctes, appliquer le CR :

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

Utiliser la CLI

Étapes

1. Restaurez l'instantané dans l'espace de noms d'origine, en remplaçant les valeurs entre crochets par les informations de votre environnement. Par exemple:

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

Vérifier l'état d'une opération de restauration

Vous pouvez utiliser la ligne de commande pour vérifier l'état d'une opération de restauration : en cours, terminée ou ayant échoué.

Étapes

1. Utilisez la commande suivante pour récupérer l'état de l'opération de restauration, en remplaçant les valeurs entre parenthèses par les informations provenant de votre environnement :

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

Utilisez les paramètres de restauration avancés de Trident Protect

Vous pouvez personnaliser les opérations de restauration à l'aide de paramètres avancés tels que les annotations, les paramètres d'espace de noms et les options de stockage pour répondre à vos besoins spécifiques.

Annotations et étiquettes d'espace de noms lors des opérations de restauration et de basculement

Lors des opérations de restauration et de basculement, les étiquettes et annotations de l'espace de noms de destination sont mises en correspondance avec celles de l'espace de noms source. Les étiquettes ou annotations de l'espace de noms source qui n'existent pas dans l'espace de noms de destination sont ajoutées, et toutes les étiquettes ou annotations existantes sont écrasées pour correspondre à la valeur de l'espace de noms source. Les étiquettes ou annotations qui existent uniquement dans l'espace de noms de destination restent inchangées.



Si vous utilisez Red Hat OpenShift, il est important de noter le rôle essentiel des annotations d'espace de noms dans les environnements OpenShift. Les annotations d'espace de noms garantissent que les pods restaurés adhèrent aux autorisations et aux configurations de sécurité appropriées définies par les contraintes de contexte de sécurité (SCC) OpenShift et peuvent accéder aux volumes sans problèmes d'autorisation. Pour plus d'informations, veuillez consulter le ["Documentation sur les contraintes de contexte de sécurité d'OpenShift"](#).

Vous pouvez empêcher l'écrasement de certaines annotations dans l'espace de noms de destination en définissant la variable d'environnement Kubernetes. `RESTORE_SKIP_NAMESPACE_ANNOTATIONS` avant d'effectuer l'opération de restauration ou de basculement. Par exemple:

```
helm upgrade trident-protect --set  
restoreSkipNamespaceAnnotations=<annotation_key_to_skip_1>,<annotation_key  
_to_skip_2> --reuse-values
```



Lors d'une opération de restauration ou de basculement, toutes les annotations et étiquettes d'espace de noms spécifiées dans `restoreSkipNamespaceAnnotations` et `restoreSkipNamespaceLabels` sont exclues de l'opération de restauration ou de basculement. Assurez-vous que ces paramètres sont configurés lors de l'installation initiale de Helm. Pour en savoir plus, consultez ["Configurer les options de filtrage AutoSupport et d'espace de noms"](#).

Si vous avez installé l'application source à l'aide de Helm avec le `--create-namespace` Le drapeau, un traitement spécial est accordé au `name` Légende. Lors du processus de restauration ou de basculement, Trident Protect copie cette étiquette dans l'espace de noms de destination, mais met à jour la valeur avec la valeur de l'espace de noms de destination si la valeur de la source correspond à l'espace de noms source. Si cette valeur ne correspond pas à l'espace de noms source, elle est copiée dans l'espace de noms de destination sans modification.

Exemple

L'exemple suivant présente un espace de noms source et un espace de noms de destination, chacun avec des annotations et des étiquettes différentes. Vous pouvez voir l'état de l'espace de noms de destination avant et après l'opération, et comment les annotations et les étiquettes sont combinées ou écrasées dans l'espace de noms de destination.

Avant l'opération de restauration ou de basculement

Le tableau suivant illustre l'état des espaces de noms source et de destination de l'exemple avant l'opération de restauration ou de basculement :

Espace de noms	Annotations	Étiquettes
Espace de noms ns-1 (source)	• <code>annotation.one/key: "updatedvalue"</code> • <code>annotation.two/key: "true"</code>	• <code>environnement=production</code> • <code>conformité=hippaa</code> • <code>nom=ns-1</code>
Espace de noms ns-2 (destination)	• <code>annotation.one/key: "true"</code> • <code>annotation.three/key: "false"</code>	• <code>rôle=base de données</code>

Après l'opération de restauration

Le tableau suivant illustre l'état de l'espace de noms de destination d'exemple après l'opération de restauration ou de basculement. Des touches ont été ajoutées, d'autres ont été écrasées, et le `name` L'étiquette a été mise à jour pour correspondre à l'espace de noms de destination :

Espace de noms	Annotations	Étiquettes
Espace de noms ns-2 (destination)	• <code>annotation.one/key: "updatedvalue"</code> • <code>annotation.two/key: "true"</code> • <code>annotation.three/key: "false"</code>	• <code>nom=ns-2</code> • <code>conformité=hippaa</code> • <code>environnement=production</code> • <code>rôle=base de données</code>

Champs pris en charge

Cette section décrit les champs supplémentaires disponibles pour les opérations de restauration.

Mappage des classes de stockage

Le `spec.storageClassMapping` L'attribut définit un mappage entre une classe de stockage présente dans l'application source et une nouvelle classe de stockage sur le cluster cible. Vous pouvez l'utiliser lors de la

migration d'applications entre des clusters avec différentes classes de stockage ou lors du changement du backend de stockage pour les opérations BackupRestore.

Exemple:

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

Annotations prises en charge

Cette section répertorie les annotations prises en charge pour configurer différents comportements du système. Si aucune annotation n'est explicitement définie par l'utilisateur, le système utilisera la valeur par défaut.

Annotation	Type	Description	Valeur par défaut
protect.trident.netapp.io/data-mover-timeout-sec	chaîne	Le temps maximal (en secondes) pendant lequel l'opération de déplacement de données peut être bloquée.	"300"
protect.trident.netapp.io/kopia-content-cache-size-limit-mb	chaîne	La limite de taille maximale (en mégaoctets) pour le cache de contenu Kopia.	"1000"

Informations sur le copyright

Copyright © 2026 NetApp, Inc. Tous droits réservés. Imprimé aux États-Unis. Aucune partie de ce document protégé par copyright ne peut être reproduite sous quelque forme que ce soit ou selon quelque méthode que ce soit (graphique, électronique ou mécanique, notamment par photocopie, enregistrement ou stockage dans un système de récupération électronique) sans l'autorisation écrite préalable du détenteur du droit de copyright.

Les logiciels dérivés des éléments NetApp protégés par copyright sont soumis à la licence et à l'avis de non-responsabilité suivants :

CE LOGICIEL EST FOURNI PAR NETAPP « EN L'ÉTAT » ET SANS GARANTIES EXPRESSES OU TACITES, Y COMPRIS LES GARANTIES TACITES DE QUALITÉ MARCHANDE ET D'ADÉQUATION À UN USAGE PARTICULIER, QUI SONT EXCLUES PAR LES PRÉSENTES. EN AUCUN CAS NETAPP NE SERA TENU POUR RESPONSABLE DE DOMMAGES DIRECTS, INDIRECTS, ACCESSOIRES, PARTICULIERS OU EXEMPLAIRES (Y COMPRIS L'ACHAT DE BIENS ET DE SERVICES DE SUBSTITUTION, LA PERTE DE JOUISSANCE, DE DONNÉES OU DE PROFITS, OU L'INTERRUPTION D'ACTIVITÉ), QUELLES QU'EN SOIENT LA CAUSE ET LA DOCTRINE DE RESPONSABILITÉ, QU'IL S'AGISSE DE RESPONSABILITÉ CONTRACTUELLE, STRICTE OU DÉLICTELLE (Y COMPRIS LA NÉGLIGENCE OU AUTRE) DÉCOULANT DE L'UTILISATION DE CE LOGICIEL, MÊME SI LA SOCIÉTÉ A ÉTÉ INFORMÉE DE LA POSSIBILITÉ DE TELS DOMMAGES.

NetApp se réserve le droit de modifier les produits décrits dans le présent document à tout moment et sans préavis. NetApp décline toute responsabilité découlant de l'utilisation des produits décrits dans le présent document, sauf accord explicite écrit de NetApp. L'utilisation ou l'achat de ce produit ne concède pas de licence dans le cadre de droits de brevet, de droits de marque commerciale ou de tout autre droit de propriété intellectuelle de NetApp.

Le produit décrit dans ce manuel peut être protégé par un ou plusieurs brevets américains, étrangers ou par une demande en attente.

LÉGENDE DE RESTRICTION DES DROITS : L'utilisation, la duplication ou la divulgation par le gouvernement sont sujettes aux restrictions énoncées dans le sous-paragraphe (b)(3) de la clause Rights in Technical Data-Noncommercial Items du DFARS 252.227-7013 (février 2014) et du FAR 52.227-19 (décembre 2007).

Les données contenues dans les présentes se rapportent à un produit et/ou service commercial (tel que défini par la clause FAR 2.101). Il s'agit de données propriétaires de NetApp, Inc. Toutes les données techniques et tous les logiciels fournis par NetApp en vertu du présent Accord sont à caractère commercial et ont été exclusivement développés à l'aide de fonds privés. Le gouvernement des États-Unis dispose d'une licence limitée irrévocable, non exclusive, non cessible, non transférable et mondiale. Cette licence lui permet d'utiliser uniquement les données relatives au contrat du gouvernement des États-Unis d'après lequel les données lui ont été fournies ou celles qui sont nécessaires à son exécution. Sauf dispositions contraires énoncées dans les présentes, l'utilisation, la divulgation, la reproduction, la modification, l'exécution, l'affichage des données sont interdits sans avoir obtenu le consentement écrit préalable de NetApp, Inc. Les droits de licences du Département de la Défense du gouvernement des États-Unis se limitent aux droits identifiés par la clause 252.227-7015(b) du DFARS (février 2014).

Informations sur les marques commerciales

NETAPP, le logo NETAPP et les marques citées sur le site <http://www.netapp.com/TM> sont des marques déposées ou des marques commerciales de NetApp, Inc. Les autres noms de marques et de produits sont des marques commerciales de leurs propriétaires respectifs.