



Ripristina le applicazioni

Trident

NetApp
October 29, 2026

Sommario

- Ripristina le applicazioni 1
 - Ripristina le applicazioni utilizzando Trident Protect 1
 - Ripristina da un backup a un namespace diverso 1
 - Ripristina da un backup nello spazio dei nomi originale 4
 - Ripristina da un backup a un cluster diverso 6
 - Ripristina da uno Snapshot a uno spazio dei nomi diverso 9
 - Ripristina da uno Snapshot allo spazio dei nomi originale 12
 - Verificare lo stato di un'operazione di ripristino 14
 - Utilizza le impostazioni di ripristino avanzate di Trident Protect 15
 - Annotazioni ed etichette dello spazio dei nomi durante le operazioni di ripristino e failover 15
 - Campi supportati 16
 - Annotazioni supportate 17
 - Modifica le risorse durante il ripristino utilizzando risorse personalizzate 17
 - Come funziona la modifica delle risorse 17
 - Risorse supportate 18
 - Operazioni supportate 18

Ripristina le applicazioni

Ripristina le applicazioni utilizzando Trident Protect

Puoi utilizzare Trident Protect per ripristinare la tua applicazione da uno snapshot o da un backup. Il ripristino da uno snapshot esistente sarà più rapido quando si ripristina l'applicazione sullo stesso cluster.



- Quando si ripristina un'applicazione, tutti gli hook di esecuzione configurati per l'applicazione vengono ripristinati con l'app. Se è presente un execution hook post-ripristino, viene eseguito automaticamente come parte dell'operazione di ripristino.
- Il ripristino da un backup a un namespace diverso o al namespace originale è supportato per i volumi qtree. Tuttavia, il ripristino da uno snapshot a un namespace diverso o al namespace originale non è supportato per i volumi qtree.
- È possibile utilizzare le impostazioni avanzate per personalizzare le operazioni di ripristino. Per saperne di più, consulta ["Utilizza le impostazioni di ripristino avanzate di Trident Protect"](#).
- Per ripristinare specifiche macchine virtuali (VM) KubeVirt da un archivio senza ripristinare tutte le risorse dello spazio dei nomi, usa il campo `includedVirtualMachines` su una CR di ripristino. Per le CR supportate, il riferimento ai campi e gli esempi YAML, consulta ["Proteggi le macchine virtuali KubeVirt con Trident Protect"](#).

Ripristina da un backup a un namespace diverso

Quando si ripristina un backup in un namespace diverso utilizzando una BackupRestore CR, Trident Protect ripristina l'applicazione in un nuovo namespace e crea una application CR per l'applicazione ripristinata. Per proteggere l'applicazione ripristinata, crea backup o snapshot on-demand oppure stabilisci una pianificazione di protezione.



- Il ripristino di un backup in un namespace diverso con risorse esistenti non modificherà le risorse che condividono i nomi con quelle nel backup. Per ripristinare tutte le risorse nel backup, eliminare e ricreare il namespace di destinazione o ripristinare il backup in un nuovo namespace.
- Quando si utilizza una CR per ripristinare in un nuovo namespace, è necessario creare manualmente il namespace di destinazione prima di applicare la CR. Trident Protect crea automaticamente gli spazi dei nomi solo quando si utilizza la CLI.

Prima di iniziare

Assicurarsi che la scadenza del token di sessione AWS sia sufficiente per qualsiasi operazione di ripristino s3 di lunga durata. Se il token scade durante l'operazione di ripristino, l'operazione può fallire.

- Fare riferimento a ["Documentazione API AWS"](#) per ulteriori informazioni sulla verifica della scadenza del token della sessione corrente.
- Fare riferimento a ["Documentazione IAM AWS"](#) per ulteriori informazioni sulle credenziali con le risorse AWS.



Quando si ripristinano i backup utilizzando Kopia come data mover, è possibile specificare facoltativamente annotazioni nel CR o tramite la CLI per controllare il comportamento dello

storage temporaneo utilizzato da Kopia. Consultare il "[Documentazione Kopia](#)" per ulteriori informazioni sulle opzioni che è possibile configurare. Utilizzare il `tridentctl-protect create --help` comando per ulteriori informazioni sulla specifica delle annotazioni con la Trident Protect CLI.

Utilizzare un CR

Passaggi

1. Crea il file custom resource (CR) e assegnagli il nome `trident-protect-backup-restore-cr.yaml`.
2. Nel file che hai creato, configura i seguenti attributi:
 - **metadata.name:** (*Obbligatorio*) Il nome di questa risorsa personalizzata; scegli un nome univoco e sensato per il tuo ambiente.
 - **spec.appArchivePath:** Il percorso all'interno di AppVault in cui sono archiviati i contenuti del backup. È possibile utilizzare il seguente comando per trovare questo percorso:

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

- **spec.appVaultRef:** (*Obbligatorio*) Il nome del AppVault in cui sono archiviati i contenuti del backup.
- **spec.destinationApplicationName:** (*Facoltativo*) Il nome per l'applicazione ripristinata. Se specificato, l'applicazione ripristinata utilizza questo nome. Se non specificato, l'applicazione ripristinata utilizza il nome dell'applicazione di origine.
- **spec.namespaceMapping:** Il mapping dello spazio dei nomi di origine dell'operazione di ripristino allo spazio dei nomi di destinazione. Sostituisci `my-source-namespace` e `my-destination-namespace` con le informazioni del tuo ambiente.

```
---  
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  
  destinationApplicationName: my-new-app-name  
  namespaceMapping: [{"source": "my-source-namespace",  
"destination": "my-destination-namespace"}]
```

3. (*Facoltativo*) Se hai bisogno di selezionare solo determinate risorse dell'applicazione da ripristinare, aggiungi un filtro che includa o escluda le risorse contrassegnate con etichette specifiche:



Trident Protect seleziona automaticamente alcune risorse in base alla loro relazione

con le risorse che selezioni. Ad esempio, se selezioni una risorsa di tipo persistent volume claim e questa ha un pod associato, Trident Protect ripristinerà anche il pod associato.

- **resourceFilter.resourceSelectionCriteria:** (Obbligatorio per il filtraggio) Utilizzare `Include` o `Exclude` per includere o escludere una risorsa definita in `resourceMatchers`. Aggiungi i seguenti parametri `resourceMatchers` per definire le risorse da includere o escludere:
 - **resourceFilter.resourceMatchers:** Un array di `resourceMatcher` oggetti. Se definisci più elementi in questo array, corrispondono come un'operazione OR e i campi all'interno di ciascun elemento (`group`, `kind`, `version`) corrispondono come un'operazione AND.
 - **resourceMatchers[].group:** (*Facoltativo*) Gruppo della risorsa da filtrare.
 - **resourceMatchers[].kind:** (*Facoltativo*) Tipo di risorsa da filtrare.
 - **resourceMatchers[].version:** (*Facoltativo*) Versione della risorsa da filtrare.
 - **resourceMatchers[].names:** (*Facoltativo*) Nomi nel campo `metadata.name` di Kubernetes della risorsa da filtrare.
 - **resourceMatchers[].namespaces:** (*Facoltativo*) Namespace nel campo `metadata.name` di Kubernetes della risorsa da filtrare.
 - **resourceMatchers[].labelSelectors:** (*Facoltativo*) Stringa del selettore di etichetta nel campo `metadata.name` dei metadati Kubernetes della risorsa come definito in ["Documentazione Kubernetes"](#). Ad esempio: `"trident.netapp.io/os=linux"`.

Ad esempio:

```
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. Dopo aver inserito i valori corretti nel file `trident-protect-backup-restore-cr.yaml`, applica il CR:

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

Usa la CLI

Passaggi

1. Ripristina il backup in un namespace diverso, sostituendo i valori tra parentesi con le informazioni del tuo ambiente. L'argomento `namespace-mapping` utilizza namespace separati da due punti per mappare i namespace di origine ai namespace di destinazione corretti nel formato `source1:dest1,source2:dest2`. Ad esempio:

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

Ripristina da un backup nello spazio dei nomi originale

È possibile ripristinare un backup nello spazio dei nomi originale in qualsiasi momento. Quando si esegue un ripristino in-place, Trident Protect gestisce automaticamente le pianificazioni di protezione e le operazioni in corso per impedire punti di ripristino non validi:

- Tutte le pianificazioni di protezione abilitate per l'applicazione vengono disabilitate prima dell'inizio del ripristino. Ciò impedisce che backup o snapshot pianificati vengano eseguiti mentre le risorse dell'applicazione vengono ripristinate.
- Una volta completato correttamente il ripristino, solo le pianificazioni che erano abilitate prima del ripristino vengono riabilite. Le pianificazioni che erano già disabilitate rimangono disabilitate.
- Tutte le operazioni di backup o snapshot in corso vengono annullate prima dell'inizio del ripristino. Se un'operazione non viene annullata entro 5 minuti, il ripristino procede e registra un avviso nello stato CR del ripristino.

Prima di iniziare

Assicurarsi che la scadenza del token di sessione AWS sia sufficiente per qualsiasi operazione di ripristino s3 di lunga durata. Se il token scade durante l'operazione di ripristino, l'operazione può fallire.

- Fare riferimento a ["Documentazione API AWS"](#) per ulteriori informazioni sulla verifica della scadenza del token della sessione corrente.
- Fare riferimento a ["Documentazione IAM AWS"](#) per ulteriori informazioni sulle credenziali con le risorse AWS.



Quando si ripristinano i backup utilizzando Kopia come data mover, è possibile specificare facoltativamente annotazioni nel CR o tramite la CLI per controllare il comportamento dello storage temporaneo utilizzato da Kopia. Consultare il ["Documentazione Kopia"](#) per ulteriori informazioni sulle opzioni che è possibile configurare. Utilizzare il `tridentctl-protect create --help` comando per ulteriori informazioni sulla specifica delle annotazioni con la Trident Protect CLI.

Utilizzare un CR

Passaggi

1. Crea il file custom resource (CR) e assegnagli il nome `trident-protect-backup-ipr-cr.yaml`.

2. Nel file che hai creato, configura i seguenti attributi:

- **metadata.name:** (*Obbligatorio*) Il nome di questa risorsa personalizzata; scegli un nome univoco e sensato per il tuo ambiente.
- **spec.appArchivePath:** Il percorso all'interno di AppVault in cui sono archiviati i contenuti del backup. È possibile utilizzare il seguente comando per trovare questo percorso:

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

- **spec.appVaultRef:** (*Obbligatorio*) Il nome del AppVault in cui sono archiviati i contenuti del backup.

Ad esempio:

```
---  
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. (*Facoltativo*) Se hai bisogno di selezionare solo determinate risorse dell'applicazione da ripristinare, aggiungi un filtro che includa o escluda le risorse contrassegnate con etichette specifiche:



Trident Protect seleziona automaticamente alcune risorse in base alla loro relazione con le risorse che selezioni. Ad esempio, se selezioni una risorsa di tipo persistent volume claim e questa ha un pod associato, Trident Protect ripristinerà anche il pod associato.

- **resourceFilter.resourceSelectionCriteria:** (*Obbligatorio per il filtraggio*) Utilizzare `Include` o `Exclude` per includere o escludere una risorsa definita in `resourceMatchers`. Aggiungi i seguenti parametri `resourceMatchers` per definire le risorse da includere o escludere:
 - **resourceFilter.resourceMatchers:** Un array di `resourceMatcher` oggetti. Se definisci più elementi in questo array, corrispondono come un'operazione OR e i campi all'interno di ciascun elemento (`group`, `kind`, `version`) corrispondono come un'operazione AND.
 - **resourceMatchers[].group:** (*Facoltativo*) Gruppo della risorsa da filtrare.
 - **resourceMatchers[].kind:** (*Facoltativo*) Tipo di risorsa da filtrare.
 - **resourceMatchers[].version:** (*Facoltativo*) Versione della risorsa da filtrare.
 - **resourceMatchers[].names:** (*Facoltativo*) Nomi nel campo `metadata.name` di Kubernetes della risorsa da filtrare.

- **resourceMatchers[].namespaces:** (*Facoltativo*) Namespace nel campo metadata.name di Kubernetes della risorsa da filtrare.
- **resourceMatchers[].labelSelectors:** (*Facoltativo*) Stringa del selettore di etichetta nel campo metadata.name dei metadati Kubernetes della risorsa come definito in ["Documentazione Kubernetes"](#). Ad esempio: "trident.netapp.io/os=linux".

Ad esempio:

```
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. Dopo aver inserito i valori corretti nel file trident-protect-backup-ipr-cr.yaml, applica il CR:

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

Usa la CLI

Passaggi

1. Ripristina il backup nello spazio dei nomi originale, sostituendo i valori tra parentesi con le informazioni del tuo ambiente. L'argomento backup utilizza uno spazio dei nomi e un nome di backup nel formato <namespace>/<name>. Ad esempio:

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

Ripristina da un backup a un cluster diverso

È possibile ripristinare un backup su un cluster diverso se si verifica un problema con il cluster originale.



- Quando si ripristinano i backup utilizzando Kopia come data mover, è possibile specificare facoltativamente annotazioni nel CR o tramite la CLI per controllare il comportamento dello storage temporaneo utilizzato da Kopia. Consultare il ["Documentazione Kopia"](#) per ulteriori informazioni sulle opzioni che è possibile configurare. Utilizzare il `tridentctl-protect create --help` comando per ulteriori informazioni sulla specifica delle annotazioni con la Trident Protect CLI.
- Quando si utilizza una CR per ripristinare in un nuovo namespace, è necessario creare manualmente il namespace di destinazione prima di applicare la CR. Trident Protect crea automaticamente gli spazi dei nomi solo quando si utilizza la CLI.

Prima di iniziare

Assicurarsi che siano soddisfatti i seguenti prerequisiti:

- Il cluster di destinazione ha Trident Protect installato.
- Il cluster di destinazione ha accesso al percorso del bucket dello stesso AppVault del cluster di origine, dove è archiviato il backup.
- Assicurati che l'ambiente locale possa connettersi al bucket di storage a oggetti definito nella AppVault CR durante l'esecuzione del comando `tridentctl-protect get appvaultcontent`. Se le restrizioni di rete impediscono l'accesso, esegui la CLI di Trident Protect da un pod sul cluster di destinazione invece.
- Assicurarsi che la scadenza del token di sessione AWS sia sufficiente per qualsiasi operazione di ripristino di lunga durata. Se il token scade durante l'operazione di ripristino, l'operazione può fallire.
 - Fare riferimento a ["Documentazione API AWS"](#) per ulteriori informazioni sulla verifica della scadenza del token della sessione corrente.
 - Fare riferimento a ["Documentazione AWS"](#) per ulteriori informazioni sulle credenziali con le risorse AWS.

Passaggi

1. Verifica che la AppVault CR esista sul cluster di destinazione utilizzando il plugin CLI di Trident Protect:

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



Se il AppVault CR non esiste sul cluster di destinazione, crealo seguendo i passaggi in ["Utilizzare gli oggetti Trident Protect AppVault per gestire i bucket"](#).

2. Visualizza il contenuto del backup disponibile AppVault sul cluster di destinazione e prendi nota `appArchivePath` del backup che desideri ripristinare:

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

L'esecuzione di questo comando visualizza i backup disponibili in AppVault, inclusi i cluster di origine, i nomi delle applicazioni corrispondenti, i timestamp e i percorsi di archivio.

Esempio di output:

```

+-----+-----+-----+-----+
+-----+-----+-----+-----+
|  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. Ripristina l'applicazione nel cluster di destinazione utilizzando il nome AppVault e il percorso di archivio:



Quando si utilizza una CR, assicurarsi che lo spazio dei nomi destinato al ripristino dell'applicazione esista sul cluster di destinazione.

Utilizzare un CR

1. Crea il file custom resource (CR) e assegnagli il nome `trident-protect-backup-restore-cr.yaml`.
2. Nel file che hai creato, configura i seguenti attributi:
 - **metadata.name:** (*Obbligatorio*) Il nome di questa risorsa personalizzata; scegli un nome univoco e sensato per il tuo ambiente.
 - **spec.appVaultRef:** (*Obbligatorio*) Il nome del AppVault in cui sono archiviati i contenuti del backup.
 - **spec.appArchivePath:** (*Obbligatorio*) Il percorso all'interno di AppVault in cui sono archiviati i contenuti del backup. Utilizzare il comando del passaggio 2 per visualizzare i contenuti del backup e trovare `appArchivePath` per il backup che si desidera ripristinare.
 - **spec.destinationApplicationName:** (*Facoltativo*) Il nome per l'applicazione ripristinata. Se specificato, l'applicazione ripristinata utilizza questo nome. Se non specificato, l'applicazione ripristinata utilizza il nome dell'applicazione di origine.
 - **spec.namespaceMapping:** Il mapping dello spazio dei nomi di origine dell'operazione di ripristino allo spazio dei nomi di destinazione. Sostituisci `my-source-namespace` e `my-destination-namespace` con le informazioni del tuo ambiente.

Ad esempio:

```

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
  destinationApplicationName: my-new-app-name
  namespaceMapping: [{"source": "my-source-namespace", "
  destination": "my-destination-namespace"}]
```

3. Dopo aver inserito i valori corretti nel file `trident-protect-backup-restore-cr.yaml`, applica il CR:

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

Usa la CLI

1. Utilizzare il seguente comando per ripristinare l'applicazione, sostituendo i valori tra parentesi con le informazioni del proprio ambiente. L'argomento `namespace-mapping` utilizza namespace separati da due punti per mappare i namespace di origine ai namespace di destinazione corretti nel formato `source1:dest1,source2:dest2`. Ad esempio:

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

Ripristina da uno Snapshot a uno spazio dei nomi diverso

È possibile ripristinare i dati da uno snapshot utilizzando un file di risorsa personalizzata (CR) sia in un namespace diverso che nel namespace di origine. Quando si ripristina uno snapshot in un namespace diverso utilizzando una `SnapshotRestore` CR, Trident Protect ripristina l'applicazione in un nuovo namespace e crea una CR dell'applicazione per l'applicazione ripristinata. Per proteggere l'applicazione ripristinata, crea backup o snapshot on-demand oppure stabilisci una pianificazione di protezione.



- `SnapshotRestore` supporta l' `spec.storageClassMapping` attributo, ma solo quando le classi di archiviazione di origine e destinazione utilizzano lo stesso backend di archiviazione. Se si tenta di eseguire il ripristino su una `StorageClass` che utilizza un backend di archiviazione diverso, l'operazione di ripristino non riuscirà.
- Quando si utilizza una CR per ripristinare in un nuovo namespace, è necessario creare manualmente il namespace di destinazione prima di applicare la CR. Trident Protect crea automaticamente gli spazi dei nomi solo quando si utilizza la CLI.

Prima di iniziare

Assicurarsi che la scadenza del token di sessione AWS sia sufficiente per qualsiasi operazione di ripristino s3 di lunga durata. Se il token scade durante l'operazione di ripristino, l'operazione può fallire.

- Fare riferimento a "[Documentazione API AWS](#)" per ulteriori informazioni sulla verifica della scadenza del token della sessione corrente.
- Fare riferimento a "[Documentazione IAM AWS](#)" per ulteriori informazioni sulle credenziali con le risorse AWS.

Utilizzare un CR

Passaggi

1. Crea il file custom resource (CR) e assegnagli il nome `trident-protect-snapshot-restore-cr.yaml`.
2. Nel file che hai creato, configura i seguenti attributi:
 - **metadata.name:** (*Obbligatorio*) Il nome di questa risorsa personalizzata; scegli un nome univoco e sensato per il tuo ambiente.
 - **spec.appVaultRef:** (*Obbligatorio*) Il nome del AppVault in cui sono archiviati i contenuti dello snapshot.
 - **spec.appArchivePath:** Il percorso all'interno di AppVault in cui sono archiviati i contenuti dello snapshot. È possibile utilizzare il seguente comando per trovare questo percorso:

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

- **spec.destinationApplicationName:** (*Facoltativo*) Il nome per l'applicazione ripristinata. Se specificato, l'applicazione ripristinata utilizza questo nome. Se non specificato, l'applicazione ripristinata utilizza il nome dell'applicazione di origine.
- **spec.namespaceMapping:** Il mapping dello spazio dei nomi di origine dell'operazione di ripristino allo spazio dei nomi di destinazione. Sostituisci `my-source-namespace` e `my-destination-namespace` con le informazioni del tuo ambiente.

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

3. (*Facoltativo*) Se hai bisogno di selezionare solo determinate risorse dell'applicazione da ripristinare, aggiungi un filtro che includa o escluda le risorse contrassegnate con etichette specifiche:



Trident Protect seleziona automaticamente alcune risorse in base alla loro relazione con le risorse che selezioni. Ad esempio, se selezioni una risorsa di tipo persistent

volume claim e questa ha un pod associato, Trident Protect ripristinerà anche il pod associato.

- **resourceFilter.resourceSelectionCriteria:** (Obbligatorio per il filtraggio) Utilizzare `Include` o `Exclude` per includere o escludere una risorsa definita in `resourceMatchers`. Aggiungi i seguenti parametri `resourceMatchers` per definire le risorse da includere o escludere:
 - **resourceFilter.resourceMatchers:** Un array di `resourceMatcher` oggetti. Se definisci più elementi in questo array, corrispondono come un'operazione OR e i campi all'interno di ciascun elemento (`group`, `kind`, `version`) corrispondono come un'operazione AND.
 - **resourceMatchers[].group:** (*Facoltativo*) Gruppo della risorsa da filtrare.
 - **resourceMatchers[].kind:** (*Facoltativo*) Tipo di risorsa da filtrare.
 - **resourceMatchers[].version:** (*Facoltativo*) Versione della risorsa da filtrare.
 - **resourceMatchers[].names:** (*Facoltativo*) Nomi nel campo `metadata.name` di Kubernetes della risorsa da filtrare.
 - **resourceMatchers[].namespaces:** (*Facoltativo*) Namespace nel campo `metadata.name` di Kubernetes della risorsa da filtrare.
 - **resourceMatchers[].labelSelectors:** (*Facoltativo*) Stringa del selettore di etichetta nel campo `metadata.name` dei metadati Kubernetes della risorsa come definito in ["Documentazione Kubernetes"](#). Ad esempio: `"trident.netapp.io/os=linux"`.

Ad esempio:

```
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. Dopo aver inserito i valori corretti nel file `trident-protect-snapshot-restore-cr.yaml`, applica il CR:

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

Usa la CLI

Passaggi

1. Ripristina l'istantanea in uno spazio dei nomi diverso, sostituendo i valori tra parentesi con le informazioni del tuo ambiente.

- L'`snapshot` argomento utilizza uno spazio dei nomi e un nome di snapshot nel formato `/`.
- L'argomento `namespace-mapping` utilizza namespace separati da due punti per mappare i namespace di origine ai namespace di destinazione corretti nel formato `source1:dest1,source2:dest2`.

Ad esempio:

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

Ripristina da uno Snapshot allo spazio dei nomi originale

È possibile ripristinare uno snapshot nello spazio dei nomi originale in qualsiasi momento. Quando si esegue un ripristino in-place, Trident Protect gestisce automaticamente le pianificazioni di protezione e le operazioni in corso per impedire punti di ripristino non validi:

- Tutte le pianificazioni di protezione abilitate per l'applicazione vengono disabilitate prima dell'inizio del ripristino. Ciò impedisce che backup o snapshot pianificati vengano eseguiti mentre le risorse dell'applicazione vengono ripristinate.
- Una volta completato correttamente il ripristino, solo le pianificazioni che erano abilitate prima del ripristino vengono riabilite. Le pianificazioni che erano già disabilitate rimangono disabilitate.
- Tutte le operazioni di backup o snapshot in corso vengono annullate prima dell'inizio del ripristino. Se un'operazione non viene annullata entro 5 minuti, il ripristino procede e registra un avviso nello stato CR del ripristino.

Prima di iniziare

Assicurarsi che la scadenza del token di sessione AWS sia sufficiente per qualsiasi operazione di ripristino s3 di lunga durata. Se il token scade durante l'operazione di ripristino, l'operazione può fallire.

- Fare riferimento a "[Documentazione API AWS](#)" per ulteriori informazioni sulla verifica della scadenza del token della sessione corrente.
- Fare riferimento a "[Documentazione IAM AWS](#)" per ulteriori informazioni sulle credenziali con le risorse AWS.

Utilizzare un CR

Passaggi

1. Crea il file custom resource (CR) e assegnagli il nome `trident-protect-snapshot-ipr-cr.yaml`.

2. Nel file che hai creato, configura i seguenti attributi:

- **metadata.name:** (*Obbligatorio*) Il nome di questa risorsa personalizzata; scegli un nome univoco e sensato per il tuo ambiente.
- **spec.appVaultRef:** (*Obbligatorio*) Il nome del AppVault in cui sono archiviati i contenuti dello snapshot.
- **spec.appArchivePath:** Il percorso all'interno di AppVault in cui sono archiviati i contenuti dello snapshot. È possibile utilizzare il seguente comando per trovare questo percorso:

```
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
```

+ . (*Facoltativo*) Se hai bisogno di selezionare solo determinate risorse dell'applicazione da ripristinare, aggiungi un filtro che includa o escluda le risorse contrassegnate con etichette specifiche:

+ **NOTA:** Trident Protect seleziona automaticamente alcune risorse in base alla loro relazione con le risorse che selezioni. Ad esempio, se selezioni una risorsa di tipo persistent volume claim e questa ha un pod associato, Trident Protect ripristinerà anche il pod associato.

+ * **resourceFilter.resourceSelectionCriteria:** (*Obbligatorio per il filtraggio*) Usa `Include` o `Exclude` per includere o escludere una risorsa definita in `resourceMatchers`. Aggiungi i seguenti parametri `resourceMatchers` per definire le risorse da includere o escludere: **resourceFilter.resourceMatchers:** Un array di oggetti `resourceMatcher`. Se definisci più elementi in questo array, corrispondono come un'operazione OR e i campi all'interno di ciascun elemento (`group`, `kind`, `version`) corrispondono come un'operazione AND. * `resourceMatchers[].group:` (*Opzionale*) Gruppo della risorsa da filtrare. * `resourceMatchers[].kind:` (*Opzionale*) Tipo della risorsa da filtrare. * `resourceMatchers[].version:` (*Opzionale*) Versione della risorsa da filtrare. * `resourceMatchers[].names:` (*Opzionale*) Nomi nel campo `metadata.name` di Kubernetes della risorsa da filtrare. * `resourceMatchers[].namespaces:` (*Opzionale*) Namespace nel campo `metadata.name` di Kubernetes della risorsa da filtrare. * `resourceMatchers[].labelSelectors:` (*Opzionale*) Stringa del selettore di etichette nel campo `metadata.name` di Kubernetes della risorsa come definito in "[Documentazione Kubernetes](#)". Ad esempio: `"trident.netapp.io/os=linux"`.

+ Ad esempio:

+

```
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"]
```

+ . Dopo aver inserito i valori corretti nel file `trident-protect-snapshot-ipr-cr.yaml`, applica il CR:

+

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

Usa la CLI

Passaggi

1. Ripristina lo snapshot nello spazio dei nomi originale, sostituendo i valori tra parentesi con le informazioni del tuo ambiente. Ad esempio:

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

Verificare lo stato di un'operazione di ripristino

È possibile utilizzare la riga di comando per verificare lo stato di un'operazione di ripristino in corso, completata o non riuscita.

Passaggi

1. Utilizzare il seguente comando per recuperare lo stato dell'operazione di ripristino, sostituendo i valori tra parentesi con le informazioni dal proprio ambiente:

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

Utilizza le impostazioni di ripristino avanzate di Trident Protect

È possibile personalizzare le operazioni di ripristino utilizzando impostazioni avanzate come annotazioni, impostazioni dello spazio dei nomi e opzioni di storage per soddisfare i requisiti specifici.

Annotazioni ed etichette dello spazio dei nomi durante le operazioni di ripristino e failover

Durante le operazioni di ripristino e failover, le etichette e le annotazioni nello spazio dei nomi di destinazione vengono rese corrispondenti alle etichette e alle annotazioni nello spazio dei nomi di origine. Le etichette o le annotazioni dello spazio dei nomi di origine che non esistono nello spazio dei nomi di destinazione vengono aggiunte e tutte le etichette o annotazioni già esistenti vengono sovrascritte per corrispondere al valore dello spazio dei nomi di origine. Le etichette o le annotazioni che esistono solo nello spazio dei nomi di destinazione rimangono invariate.



Se si utilizza Red Hat OpenShift, è importante tenere presente il ruolo fondamentale delle annotazioni dello spazio dei nomi negli ambienti OpenShift. Le annotazioni dello spazio dei nomi garantiscono che i pod ripristinati aderiscano alle autorizzazioni e alle configurazioni di sicurezza appropriate definite dai vincoli del contesto di sicurezza (SCC) di OpenShift e possano accedere ai volumi senza problemi di autorizzazione. Per ulteriori informazioni, consulta la ["Documentazione sui vincoli del contesto di sicurezza di OpenShift"](#).

È possibile impedire che specifiche annotazioni nello spazio dei nomi di destinazione vengano sovrascritte impostando la variabile di ambiente Kubernetes `RESTORE_SKIP_NAMESPACE_ANNOTATIONS` prima di eseguire l'operazione di ripristino o failover. Ad esempio:

```
helm upgrade trident-protect -n trident-protect netapp-trident-protect/trident-protect \
  --set-string
  restoreSkipNamespaceAnnotations="{<annotation_key_to_skip_1>,<annotation_key_to_skip_2>}" \
  --reuse-values
```



Quando si esegue un'operazione di ripristino o di failover, tutte le annotazioni e le etichette dello spazio dei nomi specificate in `restoreSkipNamespaceAnnotations` e `restoreSkipNamespaceLabels` sono escluse dall'operazione di ripristino o di failover. Assicurarsi che queste impostazioni siano configurate durante l'installazione iniziale di Helm. Per saperne di più, consulta ["Configura impostazioni aggiuntive dell'helm chart Trident Protect"](#).

Se hai installato l'applicazione sorgente utilizzando Helm con il `--create-namespace` flag, viene riservato un trattamento speciale alla chiave dell'etichetta `name`. Durante il processo di ripristino o failover, Trident Protect copia questa etichetta nello spazio dei nomi di destinazione, ma aggiorna il valore a quello dello spazio dei nomi di destinazione se il valore della sorgente corrisponde allo spazio dei nomi di origine. Se questo valore non corrisponde allo spazio dei nomi di origine, viene copiato nello spazio dei nomi di destinazione senza modifiche.

Esempio

Il seguente esempio presenta uno spazio dei nomi di origine e uno di destinazione, ciascuno con annotazioni ed etichette diverse. Puoi vedere lo stato dello spazio dei nomi di destinazione prima e dopo l'operazione e come le annotazioni e le etichette vengono combinate o sovrascritte nello spazio dei nomi di destinazione.

Prima dell'operazione di ripristino o failover

La tabella seguente illustra lo stato degli spazi dei nomi di origine e di destinazione di esempio prima dell'operazione di ripristino o failover:

Spazio dei nomi	Annotazioni	Etichette
Namespace ns-1 (origine)	<ul style="list-style-type: none">• annotation.one/key: "updatedvalue"• annotation.two/key: "true"	<ul style="list-style-type: none">• ambiente=produzione• compliance=hipaa• name=ns-1
Namespace ns-2 (destinazione)	<ul style="list-style-type: none">• annotation.one/key: "true"• annotazione.tre/chiave: "false"	<ul style="list-style-type: none">• role=database

Dopo l'operazione di ripristino

La tabella seguente illustra lo stato dello spazio dei nomi di destinazione di esempio dopo l'operazione di ripristino o failover. Alcune chiavi sono state aggiunte, alcune sono state sovrascritte e l'`name` etichetta è stata aggiornata per corrispondere allo spazio dei nomi di destinazione:

Spazio dei nomi	Annotazioni	Etichette
Namespace ns-2 (destinazione)	<ul style="list-style-type: none">• annotation.one/key: "updatedvalue"• annotation.two/key: "true"• annotazione.tre/chiave: "false"	<ul style="list-style-type: none">• name=ns-2• compliance=hipaa• ambiente=produzione• role=database

Campi supportati

Questa sezione descrive i campi aggiuntivi disponibili per le operazioni di ripristino.

Mappatura delle storage class

L'`spec.storageClassMapping` attributo definisce una mappatura da una classe di storage presente nell'applicazione di origine a una nuova classe di storage nel cluster di destinazione. Puoi utilizzare questa opzione quando migri applicazioni tra cluster con classi di storage diverse o quando cambi il backend di storage per le operazioni di BackupRestore.

Esempio:

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

```
source: "sourceStorageClass2"
```

Annotazioni supportate

Questa sezione elenca le annotazioni supportate per configurare vari comportamenti nel sistema. Se un'annotazione non viene impostata esplicitamente dall'utente, il sistema utilizzerà il valore predefinito.

Annotazione	Tipo	Descrizione	Valore predefinito
protect.trident.netapp.io/data-mover-timeout-sec	stringa	Il tempo massimo (in secondi) consentito per l'operazione di spostamento dei dati che può essere bloccata.	"300"
protect.trident.netapp.io/kopia-content-cache-size-limit-mb	stringa	Limite massimo di dimensione (in megabyte) per la cache dei contenuti Kopia.	"1000"
protect.trident.netapp.io/pvc-bind-timeout-sec	stringa	Tempo massimo (in secondi) di attesa affinché eventuali nuove PersistentVolumeClaims (PVC) raggiungano la fase <code>Bound</code> prima che l'operazione fallisca. Si applica a tutti i tipi di restore CR (BackupRestore, BackupInplaceRestore, SnapshotRestore, SnapshotInplaceRestore). Usa un valore più alto se il backend di storage o il cluster richiedono spesso più tempo.	"1200" (20 minuti)

Modifica le risorse durante il ripristino utilizzando risorse personalizzate

Le trasformazioni delle risorse ti permettono di modificare una risorsa mentre viene ripristinata. Questo è utile quando la versione ripristinata deve essere diversa dall'originale, ad esempio aggiornando un hostname in una ConfigMap quando ripristini un'applicazione in un cluster diverso.

Come funziona la modifica delle risorse

Il `transformations` campo in `SnapshotRestore`, `BackupRestore`, `AppMirrorRelationship` e altre risorse di ripristino ti permette di modificare le risorse Kubernetes durante il processo di ripristino. Questo è utile quando l'applicazione ripristinata deve essere configurata in modo diverso dall'originale, per esempio cambiando i nomi host, gli URL del registro, i limiti delle risorse o le variabili d'ambiente.

Le trasformazioni delle risorse utilizzano ["RFC 6902 JSON Patch"](#) operazioni e ["RFC 6901 JSON Pointer"](#) percorsi per individuare e modificare campi specifici all'interno delle risorse Kubernetes.

Di seguito puoi vedere la struttura di base di un oggetto di ripristino che contiene trasformazioni di risorse:

```
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)

```

Risorse supportate

È possibile utilizzare le trasformazioni delle risorse con risorse che corrispondono ai seguenti criteri:

- **tipo** (obbligatorio): Il tipo di risorsa Kubernetes (ad esempio, ConfigMap, Deployment, Pod)
- **gruppo** (facoltativo): Il gruppo API (ad esempio, apps, route.openshift.io) — ometti per le risorse core
- **versione** (facoltativo): La versione dell'API (ad esempio, v1, v1beta1)
- **nome** (facoltativo): Applica solo a una risorsa specifica per nome



La modifica di PersistentVolumeClaims e degli spazi dei nomi non è attualmente supportata.

Operazioni supportate

Puoi usare le seguenti operazioni per modificare le risorse: add, copy, move, remove, replace e test.

Aggiungere un valore a una risorsa

Utilizza l'operazione add per aggiungere un nuovo campo o valore al percorso specificato. Puoi aggiungere dati a oggetti o array.

Utilizzare un CR

L'esempio seguente aggiunge un selettore di nodi a una Deployment risorsa:

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

Usa la CLI

Utilizzare il seguente comando per eseguire questa trasformazione tramite riga di comando:

```
tridentctl-protect create snapshotrestore <restore_name> \
--snapshot <namespace/snapshot-name> \
--namespace-mapping <source-ns>:<dest-ns> \
--transformation
'apps,v1,Deployment:add{"path":"/spec/template/spec/nodeSelector","value":{
"topology.kubernetes.io/zone":"us-east-1a","disktype":"ssd"}}'
```

Copia un valore all'interno di una risorsa

Utilizzare l'operazione `copy` per copiare un valore da un percorso a un altro all'interno della stessa risorsa. L'origine non viene modificata.

Utilizzare un CR

Il seguente esempio duplica una chiave dati per un oggetto ConfigMap:

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

Usa la CLI

Utilizzare il seguente comando per eseguire questa trasformazione tramite riga di comando:

```
tridentctl-protect create snapshotrestore <restore_name> \
--snapshot <namespace/snapshot-name> \
```

```
--namespace-mapping <source-ns>:<dest-ns> \
--transformation ',v1,ConfigMap:copy{"from":"/data/source-
key","path":"/data/backup-key"}'
```

Spostare un valore all'interno di una risorsa

Utilizzare l'operazione `move` per spostare un valore da un percorso a un altro all'interno della stessa risorsa. L'origine viene rimossa e il valore viene inserito nella destinazione.

Utilizzare un CR

Il seguente esempio rinomina una chiave dati per un oggetto `ConfigMap`:

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

Usa la CLI

Utilizzare il seguente comando per eseguire questa trasformazione tramite riga di comando:

```
tridentctl-protect create snapshotrestore <restore_name> \
--snapshot <namespace/snapshot-name> \
--namespace-mapping <source-ns>:<dest-ns> \
--transformation
',v1,ConfigMap:move{"from":"/data/OLD_KEY","path":"/data/NEW_KEY"}'
```

Rimuovere un valore da una risorsa

Utilizzare l'operazione `remove` per rimuovere un campo o un valore nel percorso specificato.

Utilizzare un CR

Il seguente esempio rimuove un'annotazione da una risorsa `ConfigMap`:

```
transformations:
- resource:
  kind: ConfigMap
  operations:
  - op: remove
```

```
path: "/metadata/annotations/kubectl.kubernetes.io~1last-  
applied-configuration"
```



Nel percorso per l'esempio precedente, ~1 è la sequenza di escape del puntatore JSON per /.

Usa la CLI

Utilizzare il seguente comando per eseguire questa trasformazione tramite riga di comando:

```
tridentctl-protect create snapshotrestore <restore_name> \  
--snapshot <namespace/snapshot-name> \  
--namespace-mapping <source-ns>:<dest-ns> \  
--transformation  
'v1,ConfigMap:remove{"path":"/metadata/annotations/kubectl.kubernetes.  
io~1last-applied-configuration"}'
```

Sostituire un valore all'interno di una risorsa

Usa l'`replace`operazione per sostituire un valore esistente all'interno di una risorsa al percorso specificato. Il percorso JSON deve già esistere.

Utilizzare un CR

Il seguente esempio cambia il nome host di un Route oggetto:

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

Usa la CLI

Utilizzare il seguente comando per eseguire questa trasformazione tramite riga di comando:

```
tridentctl-protect create snapshotrestore <restore_name> \  
--snapshot <namespace/snapshot-name> \  
--namespace-mapping <source-ns>:<dest-ns> \  
--transformation  
'route.openshift.io,v1,Route:replace{"path":"/spec/host","value":"prod.'
```

```
example.com"}'
```

Verifica la modifica della risorsa

Usa l'operazione `test` per verificare che il valore in un percorso corrisponda a quello previsto prima che vengano applicate le altre operazioni. Se il test fallisce, l'intera modifica viene annullata.

Utilizzare un CR

Il seguente esempio aggiorna `database-host` solo se `environment` è impostato su `staging`:

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

Usa la CLI

Utilizzare il seguente comando per eseguire questa trasformazione tramite riga di comando:

```
tridentctl-protect create snapshotrestore <restore_name> \
--snapshot <namespace/snapshot-name> \
--namespace-mapping <source-ns>:<dest-ns> \
--transformation
',v1,ConfigMap:test{"path":"/data/environment","value":"staging"},repla
ce{"path":"/data/database-host","value":"prod-db.example.com"}'
```

Informazioni sul copyright

Copyright © 2026 NetApp, Inc. Tutti i diritti riservati. Stampato negli Stati Uniti d'America. Nessuna porzione di questo documento soggetta a copyright può essere riprodotta in qualsiasi formato o mezzo (grafico, elettronico o meccanico, inclusi fotocopie, registrazione, nastri o storage in un sistema elettronico) senza previo consenso scritto da parte del detentore del copyright.

Il software derivato dal materiale sottoposto a copyright di NetApp è soggetto alla seguente licenza e dichiarazione di non responsabilità:

IL PRESENTE SOFTWARE VIENE FORNITO DA NETAPP "COSÌ COM'È" E SENZA QUALSIVOGLIA TIPO DI GARANZIA IMPLICITA O ESPRESSA FRA CUI, A TITOLO ESEMPLIFICATIVO E NON ESAUSTIVO, GARANZIE IMPLICITE DI COMMERCIALIZZABILITÀ E IDONEITÀ PER UNO SCOPO SPECIFICO, CHE VENGONO DECLINATE DAL PRESENTE DOCUMENTO. NETAPP NON VERRÀ CONSIDERATA RESPONSABILE IN ALCUN CASO PER QUALSIVOGLIA DANNO DIRETTO, INDIRETTO, ACCIDENTALE, SPECIALE, ESEMPLARE E CONSEGUENZIALE (COMPRESI, A TITOLO ESEMPLIFICATIVO E NON ESAUSTIVO, PROCUREMENT O SOSTITUZIONE DI MERCI O SERVIZI, IMPOSSIBILITÀ DI UTILIZZO O PERDITA DI DATI O PROFITTI OPPURE INTERRUZIONE DELL'ATTIVITÀ AZIENDALE) CAUSATO IN QUALSIVOGLIA MODO O IN RELAZIONE A QUALUNQUE TEORIA DI RESPONSABILITÀ, SIA ESSA CONTRATTUALE, RIGOROSA O DOVUTA A INSOLVENZA (COMPRESA LA NEGLIGENZA O ALTRO) INSORTA IN QUALSIASI MODO ATTRAVERSO L'UTILIZZO DEL PRESENTE SOFTWARE ANCHE IN PRESENZA DI UN PREAVVISO CIRCA L'EVENTUALITÀ DI QUESTO TIPO DI DANNI.

NetApp si riserva il diritto di modificare in qualsiasi momento qualunque prodotto descritto nel presente documento senza fornire alcun preavviso. NetApp non si assume alcuna responsabilità circa l'utilizzo dei prodotti o materiali descritti nel presente documento, con l'eccezione di quanto concordato espressamente e per iscritto da NetApp. L'utilizzo o l'acquisto del presente prodotto non comporta il rilascio di una licenza nell'ambito di un qualche diritto di brevetto, marchio commerciale o altro diritto di proprietà intellettuale di NetApp.

Il prodotto descritto in questa guida può essere protetto da uno o più brevetti degli Stati Uniti, esteri o in attesa di approvazione.

LEGENDA PER I DIRITTI SOTTOPOSTI A LIMITAZIONE: l'utilizzo, la duplicazione o la divulgazione da parte degli enti governativi sono soggetti alle limitazioni indicate nel sottoparagrafo (b)(3) della clausola Rights in Technical Data and Computer Software del DFARS 252.227-7013 (FEB 2014) e FAR 52.227-19 (DIC 2007).

I dati contenuti nel presente documento riguardano un articolo commerciale (secondo la definizione data in FAR 2.101) e sono di proprietà di NetApp, Inc. Tutti i dati tecnici e il software NetApp forniti secondo i termini del presente Contratto sono articoli aventi natura commerciale, sviluppati con finanziamenti esclusivamente privati. Il governo statunitense ha una licenza irrevocabile limitata, non esclusiva, non trasferibile, non cedibile, mondiale, per l'utilizzo dei Dati esclusivamente in connessione con e a supporto di un contratto governativo statunitense in base al quale i Dati sono distribuiti. Con la sola esclusione di quanto indicato nel presente documento, i Dati non possono essere utilizzati, divulgati, riprodotti, modificati, visualizzati o mostrati senza la previa approvazione scritta di NetApp, Inc. I diritti di licenza del governo degli Stati Uniti per il Dipartimento della Difesa sono limitati ai diritti identificati nella clausola DFARS 252.227-7015(b) (FEB 2014).

Informazioni sul marchio commerciale

NETAPP, il logo NETAPP e i marchi elencati alla pagina <http://www.netapp.com/TM> sono marchi di NetApp, Inc. Gli altri nomi di aziende e prodotti potrebbero essere marchi dei rispettivi proprietari.