



Proteggi i dati utilizzando Trident Protect

NetApp public and hybrid cloud solutions

NetApp
August 18, 2025

Sommario

- Proteggi i dati utilizzando Trident Protect 1
 - Protezione dei dati per le app container in OpenShift Container Platform utilizzando Trident Protect 1
- Proteggi le VM in Red Hat OpenShift Virtualization utilizzando Trident Protect 14
 - Crea App Vault per Object Storage 15
 - Creare una VM in OpenShift Virtualization 17
 - Crea un'app 21
 - Proteggi l'app creando un backup 22
 - Ripristina dal backup 25
 - Proteggi l'app utilizzando gli snapshot 29
 - Ripristina da snapshot 32
 - Ripristina una VM specifica 36
 - Dimostrazione video 41

Proteggi i dati utilizzando Trident Protect

Protezione dei dati per le app container in OpenShift Container Platform utilizzando Trident Protect

Questa sezione del documento di riferimento fornisce dettagli sulla creazione di snapshot e backup di app container utilizzando Trident Protect. NetApp Trident Protect offre funzionalità avanzate di gestione dei dati delle applicazioni che migliorano la funzionalità e la disponibilità delle applicazioni Kubernetes con stato supportate dai sistemi di storage NetApp ONTAP e dal provisioner di storage NetApp Trident CSI. Trident Protect crea snapshot e backup delle applicazioni, il che significa che non vengono creati solo snapshot e backup dei dati delle applicazioni in volumi persistenti, ma anche snapshot e backup dei metadati delle applicazioni. Gli snapshot e i backup creati da Trident Protect possono essere archiviati in uno qualsiasi dei seguenti Object Storage e ripristinati in un secondo momento.

- AWS S3
- Archiviazione BLOB di Azure
- Google Cloud Storage
- Ontap S3
- StorageGrid
- qualsiasi altro storage compatibile con S3

Trident Protect utilizza il modello Kubernetes di controllo degli accessi basato sui ruoli (RBAC). Per impostazione predefinita, Trident Protect fornisce un singolo namespace di sistema denominato `trident-protect` e il relativo account di servizio predefinito. Se la tua organizzazione ha molti utenti o esigenze di sicurezza specifiche, puoi utilizzare le funzionalità RBAC di Trident Protect per ottenere un controllo più granulare sull'accesso alle risorse e agli spazi dei nomi.

Ulteriori informazioni su RBAC in Trident Protect possono essere trovate nel ["Documentazione di protezione Trident"](#)



L'amministratore del cluster ha accesso alle risorse nello spazio dei nomi `trident-protect` predefinito e può anche accedere alle risorse in tutti gli altri spazi dei nomi. Gli utenti non possono creare risorse personalizzate (CR) per la gestione dei dati delle applicazioni, come CR Snapshot e Backup, nello spazio dei nomi `trident-protect`. Come buona pratica, gli utenti dovranno creare tali CR nello spazio dei nomi dell'applicazione.

Trident Protect può essere installato seguendo le istruzioni fornite nella documentazione ["Qui"](#). Questa sezione illustrerà il flusso di lavoro per la protezione dei dati delle applicazioni contenitore e il ripristino delle applicazioni utilizzando Trident Protect. 1. Creazione di snapshot (su richiesta e secondo programma) 2. Ripristina da snapshot (ripristina nello stesso namespace e in uno diverso) 3. Creazione del backup 4. Ripristina dal backup

Prerequisito

Prima di creare snapshot e backup per un'applicazione, è necessario configurare un Object Storage in Trident Protect per archiviare snapshot e backup. Ciò avviene utilizzando il bucket CR. Solo gli amministratori possono creare un bucket CR e configurarlo. In Trident Protect il bucket CR è noto come AppVault. Gli oggetti AppVault sono la rappresentazione dichiarativa del flusso di lavoro Kubernetes di un bucket di archiviazione. Un CR di AppVault contiene le configurazioni necessarie affinché un bucket venga utilizzato nelle operazioni di protezione, come backup, snapshot, operazioni di ripristino e replica SnapMirror .

In questo esempio mostreremo l'uso di ONTAP S3 come storage di oggetti. Ecco il flusso di lavoro per la creazione di AppVault CR per ONTAP S3: 1. Creare un server di archiviazione oggetti S3 nell'SVM nel cluster ONTAP . 2. Crea un bucket nell'Object Store Server. 3. Creare un utente S3 nell'SVM. Conservare la chiave di accesso e la chiave segreta in un luogo sicuro. 4. In OpenShift, crea un segreto per archiviare le credenziali ONTAP S3. 5. Creare un oggetto AppVault per ONTAP S3

Configurare Trident Protect AppVault per ONTAP S3

Esempio di file yaml per la configurazione Trident Protect con ONTAP S3 come AppVault

```
# alias tp='tridentctl-protect'

appvault-secret.yaml

apiVersion: v1
stringData:
  accessKeyID: "<access key id created for a user to access ONTAP S3
bucket>"
  secretAccessKey: "corresponding Secret Access Key"
#data:
# base 64 encoded values
#  accessKeyID: <base64 access key id created for a user to access
ONTAP S3 bucket>
#  secretAccessKey: <base 64  Secret Access Key>
kind: Secret
metadata:
  name: appvault-secret
  namespace: trident-protect
type: Opaque

appvault.yaml

apiVersion: protect.trident.netapp.io/v1
kind: AppVault
metadata:
  name: ontap-s3-appvault
  namespace: trident-protect
spec:
  providerConfig:
```

```

azure:
  accountName: ""
  bucketName: ""
  endpoint: ""
gcp:
  bucketName: ""
  projectID: ""
s3:
  bucketName: <bucket-name for storing the snapshots and backups>
  endpoint: <endpoint IP for S3>
  secure: "false"
  skipCertValidation: "true"
providerCredentials:
  accessKeyID:
    valueFromSecret:
      key: accessKeyID
      name: appvault-secret
  secretAccessKey:
    valueFromSecret:
      key: secretAccessKey
      name: appvault-secret
providerType: OntapS3

# oc create -f appvault-secret.yaml -n trident-protect
# oc create -f appvault.yaml -n trident-protect

```

```

[root@localhost openshift-benchmark]#
[root@localhost openshift-benchmark]# tp get appvault -n trident-protect
+-----+-----+-----+-----+-----+
| NAME          | PROVIDER | STATE   | AGE   | ERROR |
+-----+-----+-----+-----+-----+
| ontap-s3-appvault | OntapS3  | Available | 2d2h |      |
+-----+-----+-----+-----+-----+
[root@localhost openshift-benchmark]# █

```

Esempio di file yaml per l'installazione dell'app PostgreSQL

```

postgres.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: postgres
spec:
  replicas: 1
  selector:
    matchLabels:

```

```

    app: postgres
template:
  metadata:
    labels:
      app: postgres
  spec:
    containers:
      - name: postgres
        image: postgres:14
        env:
          - name: POSTGRES_USER
            #value: "myuser"
            value: "admin"
          - name: POSTGRES_PASSWORD
            #value: "mypassword"
            value: "adminpass"
          - name: POSTGRES_DB
            value: "mydb"
          - name: PGDATA
            value: "/var/lib/postgresql/data/pgdata"
        ports:
          - containerPort: 5432
        volumeMounts:
          - name: postgres-storage
            mountPath: /var/lib/postgresql/data
    volumes:
      - name: postgres-storage
        persistentVolumeClaim:
          claimName: postgres-pvc
---
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: postgres-pvc
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 5Gi
---
apiVersion: v1
kind: Service
metadata:
  name: postgres
spec:

```

```
selector:
  app: postgres
ports:
- protocol: TCP
  port: 5432
  targetPort: 5432
type: ClusterIP
```

Now create the Trident protect application CR for the postgres app. Include the objects in the namespace postgres and create it in the postgres namespace.

```
# tp create app postgres-app --namespaces postgres -n postgres
```

```
[root@localhost RedHat]# tp get app -n postgres
```

NAME	NAMESPACES	STATE	AGE
postgres-app	postgres	Ready	24s

```
[root@localhost RedHat]#
```

```
# tp create snapshot postgres-snap1 --app postgres-app --appvault
ontap-s3-appvault -n postgres
Snapshot "postgres-snap1" created.
```

```
[root@localhost RedHat]# tp get snapshot -n postgres
+-----+-----+-----+-----+-----+
| NAME | APP REF | STATE | AGE | ERROR |
+-----+-----+-----+-----+
| postgres-snap1 | postgres-app | Completed | 19s | |
+-----+-----+-----+-----+
[root@localhost RedHat]#
```

```
[root@localhost DataProtection]# oc get all,pvc,volumesnapshot -n postgres
apps.openshift.io/v1 DeploymentConfig is deprecated in v4.14+, unavailable in v4.10000+
kubevirt.io/v1 VirtualMachineInstancePresets is now deprecated and will be removed in v2.
NAME READY STATUS RESTARTS AGE
pod/postgres-cd9d6ccb-jfx49 1/1 Running 0 3h47m

NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE
service/postgres ClusterIP 172.30.132.112 <none> 5432/TCP 3h47m

NAME READY UP-TO-DATE AVAILABLE AGE
deployment.apps/postgres 1/1 1 1 3h47m

NAME DESIRED CURRENT READY AGE
replicaset.apps/postgres-cd9d6ccb 1 1 1 3h47m

NAME STATUS VOLUME CAPACITY ACCESS MODES STORAGECLASS
ECLASS VOLUMEATTRIBUTESCLASS AGE
persistentvolumeclaim/data-postgres-postgresql-0 Bound pvc-9f89514e-3f2c-41ad-b7a3-792cea503f00 8Gi RWO sc-zon
ea-nas <unset> 4h40m
persistentvolumeclaim/postgres-pvc Bound pvc-951a9910-9edb-48ae-898a-1aed9aa25dc3 5Gi RWO sc-zon
ea-nas <unset> 3h47m

NAME READYTO
USE SOURCEPVC CREATIONTIME AGE SOURCESNAPSHOTCONTENT RESTORESIZE SNAPSHOTCLASS SNAPSHOTCONTENT
volumesnapshot.snapshot.storage.k8s.io/snapshot-2e94d04c-c8ea-446a-8d47-64b0bee67107-pvc-951a9910-9edb-48ae-898a-1aed9aa25dc3 true
postgres-pvc 13m 13m 53676Ki trident-snapshotclass snapcontent-796ea7f8-59a0-493e-bbd8-3a
e76fe9036c 13m 13m
volumesnapshot.snapshot.storage.k8s.io/snapshot-2e94d04c-c8ea-446a-8d47-64b0bee67107-pvc-9f89514e-3f2c-41ad-b7a3-792cea503f00 true
data-postgres-postgresql-0 13m 13m 368Ki trident-snapshotclass snapcontent-86a464d4-ffd8-4279-9cf7-88
88a097c001 13m 13m
```

Creazione di una pianificazione Utilizzando il seguente comando, gli snapshot verranno creati ogni giorno alle 15:33 e verranno conservati due snapshot e due backup.

```
# tp create schedule schedule1 --app postgres-app --appvault ontap-s3-
appvault --backup-retention 2 --snapshot-retention 2 --granularity
Daily --hour 15 --minute 33 --data-mover Restic -n postgres
Schedule "schedule1" created.
```



```
[root@localhost DataProtection]# tp get schedule -n postgres
```

NAME	APP	SCHEDULE	ENABLED	STATE	AGE	ERROR
schedule1	postgres-app	Daily:hour=15,min=33	true		17s	

```
[root@localhost DataProtection]#
```

Creazione di una pianificazione tramite yaml

```
# tp create schedule schedule2 --app postgres-app --appvault ontap-s3-  
appvault --backup-retention 2 --snapshot-retention 2 --granularity  
Daily --hour 15 --minute 33 --data-mover Restic -n postgres --dry-run >  
hourly-snapshotschedule.yaml
```

```
cat hourly-snapshotschedule.yaml
```

```
apiVersion: protect.trident.netapp.io/v1  
kind: Schedule  
metadata:  
  creationTimestamp: null  
  name: schedule2  
  namespace: postgres  
spec:  
  appVaultRef: ontap-s3-appvault  
  applicationRef: postgres-app  
  backupRetention: "2"  
  dataMover: Restic  
  dayOfMonth: ""  
  dayOfWeek: ""  
  enabled: true  
  granularity: Hourly  
  #hour: "15"  
  minute: "33"  
  recurrenceRule: ""  
  snapshotRetention: "2"  
status: {}
```

```
[root@localhost DataProtection]# tp get schedule -n postgres
```

NAME	APP	SCHEDULE	ENABLED	STATE	AGE	ERROR
schedule1	postgres-app	Daily:hour=15,min=33	true		8d7h	
schedule2	postgres-app	Hourly:min=33	true		8d7h	

```
[root@localhost DataProtection]#
```

È possibile visualizzare gli snapshot creati in base a questa pianificazione.

```
[root@localhost DataProtection]# tp get snap -n postgres
```

NAME	APP REF	STATE	AGE	ERROR
hourly-3f1ee-20250214183300	postgres-app	Completed	19s	
postgres-snap1	postgres-app	Completed	1h25m	

```
[root@localhost DataProtection]#
```

Vengono creati anche snapshot del volume.

```
[root@localhost DataProtection]# oc get volumesnapshots -n postgres
```

NAME	CREATIONTIME	AGE	READYTOUSE	SOURCEPVC
snapshot-2e94d04c-c8ea-446a-8d47-64b0bee67107-pvc-951a9910-9edb-48ae-898a-1aed9aa25dc3-59a0-493e-bbd8-3ae76fe9036c	114m	114m	true	postgres-pvc
snapshot-2e94d04c-c8ea-446a-8d47-64b0bee67107-pvc-9f89514e-3f2c-41ad-b7a3-792cea503f00-ffd8-4279-9cf7-8888a097c001	114m	114m	true	data-postgres-postgresql-0
snapshot-ce75a274-ecb2-48c9-a0a5-94c10f8e6cb1-pvc-951a9910-9edb-48ae-898a-1aed9aa25dc3-7adc-4042-a8c9-7606d1103ead	30m	30m	true	postgres-pvc
snapshot-ce75a274-ecb2-48c9-a0a5-94c10f8e6cb1-pvc-9f89514e-3f2c-41ad-b7a3-792cea503f00-3a82-43f6-9868-dcadd2cc04e2	30m	30m	true	data-postgres-postgresql-0

Elimina l'applicazione per simulare la perdita dell'applicazione

```
# oc delete deployment/postgres -n postgres
# oc get pod,pvc -n postgres
No resources found in postgres namespace.
```

Ripristina da Snapshot allo stesso namespace

```
# tp create sir postgres-sir --snapshot postgres/hourly-3flee-20250214183300 -n postgres
SnapshotInplaceRestore "postgres-sir" created.
```

```
[root@localhost DataProtection]# tp get sir -n postgres
```

NAME	APPVAULT	STATE	AGE	ERROR
postgres-sir	ontap-s3-appvault	Completed	2m39s	

L'applicazione e il suo PVC vengono ripristinati nello stesso namespace.

```
[root@localhost DataProtection]# oc get pods,pvc -n postgres
```

NAME	READY	STATUS	RESTARTS	AGE
pod/postgres-cd9d6ccb-x85tg	1/1	Running	0	98s

NAME	STATUS	VOLUME	CAPACITY	ACCESS MODES	STORAGECLASS	VOLUMEATTRIBUTESCLASS	AGE
persistentvolumeclaim/data-postgres-postgresql-0	Bound	pvc-b2cd67fd-fe4d-49b1-9e06-a53bf7be575e	8Gi	RWO	sc-zonea-nas	<unset>	105s
persistentvolumeclaim/postgres-pvc	Bound	pvc-2d549395-0cc6-4529-b2b9-7361bfb14fa8	5Gi	RWO	sc-zonea-nas	<unset>	105s

```
[root@localhost DataProtection]#
```

Ripristina da Snapshot a uno spazio dei nomi diverso

```
# tp create snapshotrestore postgres-restore --snapshot postgres/hourly-3flee-20250214183300 --namespace-mapping postgres:postgres-restore -n postgres-restore
SnapshotRestore "postgres-restore" created.
```

```
[root@localhost DataProtection]# tp get snapshotrestore -n postgres-restore
```

NAME	APPVAULT	STATE	AGE	ERROR
postgres-restore	ontap-s3-appvault	Completed	1m15s	

Puoi vedere che l'applicazione è stata ripristinata in un nuovo namespace.

```
[root@localhost DataProtection]#
[root@localhost DataProtection]# oc get pods,pvc -n postgres-restore
```

NAME	READY	STATUS	RESTARTS	AGE
pod/postgres-cd9d6ccb-pfxw9	1/1	Running	0	8d

NAME	STATUS	VOLUME	CAPACITY	ACCESS MODES	STORAGECLASS
persistentvolumeclaim/data-postgres-postgresql-0	Bound	pvc-3dff4e42-828d-49f9-84e9-4daf75479292	8Gi	RWO	sc-zonea-nas
persistentvolumeclaim/postgres-pvc	Bound	pvc-f57321b2-f21f-4eb7-8f86-4a17f915318b	5Gi	RWO	sc-zonea-nas

```
[root@localhost DataProtection]#
```

Crea backup

Creazione di un backup su richiesta

```
# tp create backup postgres-backup1 --app postgres-app --appvault
ontap-s3-appvault -n postgres
Backup "postgres-backup1" created.
```

```
[root@localhost DataProtection]# tp get backup -n postgres
```

NAME	APP REF	STATE	AGE	ERROR
backup1	postgres-app	Completed	5d12h	
daily-feac1-20250222153300	postgres-app	Completed	1d10h	
daily-feac1-20250223153300	postgres-app	Completed	10h36m	
hourly-3f1ee-20250224003300	postgres-app	Completed	1h36m	
hourly-3f1ee-20250224013300	postgres-app	Completed	36m27s	
postgres-backup1	postgres-app	Completed	6m19s	

Creazione di una pianificazione per il backup

I backup giornalieri e orari nell'elenco sopra vengono creati in base alla pianificazione impostata in precedenza.

```
# tp create schedule schedule1 --app postgres-app --appvault ontap-s3-
appvault --backup-retention 2 --snapshot-retention 2 --granularity
Daily --hour 15 --minute 33 --data-mover Restic -n postgres
Schedule "schedule1" created.
```

```
[root@localhost DataProtection]# tp get schedule -n postgres
```

NAME	APP	SCHEDULE	ENABLED	STATE	AGE	ERROR
schedule1	postgres-app	Daily:hour=15,min=33	true		9d8h	
schedule2	postgres-app	Hourly:min=33	true		9d8h	

Ripristina dal backup

Eliminare l'applicazione e i PVC per simulare una perdita di dati.

```
[root@localhost DataProtection]# oc get pods -n postgres
NAME                                READY   STATUS    RESTARTS   AGE
postgres-cd9d6ccb-dftkt            1/1     Running   0           11s
[root@localhost DataProtection]# oc get deployment -n postgres
NAME    READY   UP-TO-DATE   AVAILABLE   AGE
postgres 1/1      1             1           20s
[root@localhost DataProtection]# oc delete deployment/postgres -n postgres
deployment.apps "postgres" deleted
[root@localhost DataProtection]# oc get pods -n postgres
No resources found in postgres namespace.
[root@localhost DataProtection]# oc get pvc -n postgres
NAME                                STATUS   VOLUME                                     CAPACITY   ACCESS MODES   STORAGECLASS   VOLUMEATTRIBUTE
SCLASS   AGE
data-postgres-postgresql-0          Bound    pvc-b2cd67fd-fe4d-49b1-9e06-a53bf7be575e  8Gi        RWO             sc-zonea-nas   <unset>
5d13h
postgres-pvc                        Bound    pvc-2d549395-0cc6-4529-b2b9-7361bfb14fa8  5Gi        RWO             sc-zonea-nas   <unset>
5d13h
[root@localhost DataProtection]# oc delete pvc/data-postgres-postgresql-0 -n postgres
persistentvolumeclaim "data-postgres-postgresql-0" deleted
[root@localhost DataProtection]# oc delete pvc/postgres-pvc -n postgres
persistentvolumeclaim "postgres-pvc" deleted
[root@localhost DataProtection]# oc get pvc -n postgres
No resources found in postgres namespace.
[root@localhost DataProtection]#
```

Activate Windows
Go to Settings to activate Windows.

Ripristina nello stesso namespace #tp create bir postgres-bir --backup postgres/hourly-3f1ee-20250224023300 -n postgres BackupInplaceRestore "postgres-bir" creato.

```
[root@localhost DataProtection]# tp get bir -n postgres
+-----+-----+-----+-----+-----+
| NAME | APPVAULT | STATE | AGE | ERROR |
+-----+-----+-----+-----+-----+
| postgres-bir | ontap-s3-appvault | Completed | 2m19s | |
+-----+-----+-----+-----+-----+
```

L'applicazione e i PVC vengono ripristinati nello stesso namespace.

```
[root@localhost DataProtection]# oc get pods -n postgres
NAME                                READY   STATUS    RESTARTS   AGE
postgres-cd9d6ccb-t857w            1/1     Running   0           10m
[root@localhost DataProtection]# oc get pvc -n postgres
NAME                                STATUS   VOLUME                                     CAPACITY   ACCESS MODES   STORAGECLASS   VOLUMEATTRIBUTE
SCLASS   AGE
data-postgres-postgresql-0          Bound    pvc-0a849c19-16fe-466f-9733-85e82a8b1677  8Gi        RWO             sc-zonea-nas   <unset>
10m
postgres-pvc                        Bound    pvc-ded304ea-02d4-4225-b606-63007666ad66  5Gi        RWO             sc-zonea-nas   <unset>
10m
```

Ripristina in uno spazio dei nomi diverso Crea un nuovo spazio dei nomi. Ripristina da un backup al nuovo namespace.

```

[root@localhost DataProtection]# oc create ns postgres-restore-from-backup
namespace/postgres-restore-from-backup created
[root@localhost DataProtection]# tp create backuprestore postgres-restore-from-backup --backup postgres/postgres-backup1 --namespace-map
ping postgres:postgres-restore-from-backup -n postgres-restore-from-backup
BackupRestore "postgres-restore-from-backup" created.
[root@localhost DataProtection]# oc get backuprestore -n postgres-restore-from-backup
NAME                                STATE      ERROR      AGE
postgres-restore-from-backup        Running    37s
[root@localhost DataProtection]# oc get backuprestore -n postgres-restore-from-backup
NAME                                STATE      ERROR      AGE
postgres-restore-from-backup        Running    56s
[root@localhost DataProtection]# oc get backuprestore -n postgres-restore-from-backup
NAME                                STATE      ERROR      AGE
postgres-restore-from-backup        Completed  2m52s
[root@localhost DataProtection]# oc get pods -n postgres-restore-from-backup
NAME                                READY      STATUS      RESTARTS      AGE
postgres-cd9d6ccb-p659p             1/1        Running    0              2m9s
[root@localhost DataProtection]# oc get pvc -n postgres-restore-from-backup
NAME                                STATUS      VOLUME                                     CAPACITY   ACCESS MODES   STORAGECLASS   VOLUMEATTRIBUT
ESCLASS   AGE
data-postgres-postgresql-0          Bound      pvc-36df7399-95da-4c67-a621-af9434015bdb   8Gi        RWO              sc-zonea-nas   <unset>
2m18s
postgres-pvc                        Bound      pvc-633de3aa-a4f9-4f3b-93cc-e91afbd4fe02   5Gi        RWO              sc-zonea-nas   <unset>
2m18s
[root@localhost DataProtection]#

```

Activate Windows
 Go to Settings to activate Windows.

Migrazione delle applicazioni

Per clonare o migrare un'applicazione su un cluster diverso (eseguire un clone tra cluster), creare un backup sul cluster di origine, quindi ripristinare il backup su un cluster diverso. Assicurarsi che Trident Protect sia installato sul cluster di destinazione.

Sul cluster di origine, eseguire i passaggi mostrati nell'immagine seguente:

```
[root@localhost DataProtection]# tp create backup postgres-backup-cluster1 --app postgres-app --appvault ontap-s3-appvault -n postgres
Backup "postgres-backup-cluster1" created.
[root@localhost DataProtection]# tp get backup -n postgres
```

NAME	APP REF	STATE	AGE	ERROR
backup1	postgres-app	Completed	5d14h	
daily-feac1-20250222153300	postgres-app	Completed	1d12h	
daily-feac1-20250223153300	postgres-app	Completed	12h18m	
hourly-3f1ee-20250224023300	postgres-app	Completed	1h18m	
hourly-3f1ee-20250224033300	postgres-app	Completed	18m26s	
postgres-backup-cluster1	postgres-app	Running	35s	
postgres-backup1	postgres-app	Completed	1h48m	

```
[root@localhost DataProtection]# tp get backup -n postgres
```

NAME	APP REF	STATE	AGE	ERROR
backup1	postgres-app	Completed	5d14h	
daily-feac1-20250222153300	postgres-app	Completed	1d12h	
daily-feac1-20250223153300	postgres-app	Completed	12h19m	
hourly-3f1ee-20250224023300	postgres-app	Completed	1h19m	
hourly-3f1ee-20250224033300	postgres-app	Completed	19m41s	
postgres-backup-cluster1	postgres-app	Completed	1m50s	
postgres-backup1	postgres-app	Completed	1h49m	

Activate Windows
Go to Settings to activate Windows.

Dal cluster di origine, cambia contesto al cluster di destinazione. Quindi, assicurati che AppVault sia accessibile dal contesto del cluster di destinazione e ottieni i contenuti di AppVault dal cluster di destinazione.

```
[root@localhost DataProtection]# kubectl config use-context default/api-bm-cluster5-min-ocpv-sddc-netapp-com:6443/kube:admin
Switched to context "default/api-bm-cluster5-min-ocpv-sddc-netapp-com:6443/kube:admin".
[root@localhost DataProtection]# tp get appvault -n trident-protect
```

NAME	PROVIDER	STATE	AGE	ERROR
ontap-s3-appvault	OntapS3	Available	3d6h	

```
[root@localhost DataProtection]# tp get appvaultcontent ontap-s3-appvault --show-resources backup --show-paths
```

CLUSTER	APP PATH	TYPE	NAME	TIMESTAMP
ocp-cluster11	bbox	backup	bboxbackup1	2025-01-17 15:57:49 (UTC)
fe393/backups/bboxbackup1_3960c945-8ee0-42fe-945c-c57bb0a3af6f				
ocp-cluster11	postgres-app	backup	backup1	2025-02-18 13:31:50 (UTC)
-c5e2d89aeb89/backups/backup1_28e1bd9a-9b04-4412-8b96-811f9b62e2e3				
ocp-cluster11	postgres-app	backup	daily-feac1-20250222153300	2025-02-22 15:34:44 (UTC)
-c5e2d89aeb89/backups/daily-feac1-20250222153300_23d1386b-09f0-456f-aa49-a5865fd48abd				
ocp-cluster11	postgres-app	backup	daily-feac1-20250223153300	2025-02-23 15:34:42 (UTC)
-c5e2d89aeb89/backups/daily-feac1-20250223153300_c492a4d1-38a9-4472-9684-4705c12a206d				
ocp-cluster11	postgres-app	backup	hourly-3f1ee-20250224033300	2025-02-24 03:34:44 (UTC)
-c5e2d89aeb89/backups/hourly-3f1ee-20250224033300_3d09ab81-f6fe-47fa-a699-28006160cdbc				
ocp-cluster11	postgres-app	backup	hourly-3f1ee-20250224043300	2025-02-24 04:34:47 (UTC)
-c5e2d89aeb89/backups/hourly-3f1ee-20250224043300_66805e4f-7631-48a6-98f7-d34bb8626031				
ocp-cluster11	postgres-app	backup	postgres-backup-cluster1	2025-02-24 03:52:36 (UTC)
-c5e2d89aeb89/backups/postgres-backup-cluster1_ec0ed3f3-5500-4e72-afa8-117a04a0b1c3				

Utilizzare il percorso di backup dall'elenco e creare un oggetto CR backuprestore come mostrato nel comando seguente.

```
# tp create backuprestore backup-restore-cluster2 --namespace-mapping
postgres:postgres --appvault ontap-s3-appvault --path postgres-
app_4d798ed5-cfa8-49ff-a5b6-c5e2d89aeb89/backups/postgres-backup-
cluster1_ec0ed3f3-5500-4e72-afa8-117a04a0b1c3 -n postgres
BackupRestore "backup-restore-cluster2" created.
```

```
[root@localhost DataProtection]# tp get backuprestore -n postgres
```

NAME	APPVAULT	STATE	AGE	ERROR
backup-restore-cluster2	ontap-s3-appvault	Completed	12m41s	

Ora puoi vedere che i pod dell'applicazione e i pvc sono stati creati nel cluster di destinazione.

```
[root@localhost DataProtection]# oc get pods -n postgres
```

NAME	READY	STATUS	RESTARTS	AGE
postgres-cd9d6ccb-2lvcq	1/1	Running	0	13m

```
[root@localhost DataProtection]# oc get pvc -n postgres
```

NAME	STATUS	VOLUME	CAPACITY	ACCESS MODES	STORAGECLASS	VOLUMEATTRIBUT
data-postgres-postgresql-0	Bound	pvc-872a5182-601b-4848-b410-fef368337d07	8Gi	RWO	sc-zoneb-san	<unset>
postgres-pvc	Bound	pvc-caf9fa71-76a8-4645-9bb5-2ed72e72948b	5Gi	RWO	sc-zoneb-san	<unset>

Proteggi le VM in Red Hat OpenShift Virtualization utilizzando Trident Protect

Proteggi le VM in OpenShift Virtualization utilizzando snapshot e backup. Questa procedura include la creazione di un AppVault utilizzando l'archiviazione di oggetti ONTAP S3, la configurazione Trident Protect per acquisire i dati della VM, inclusi gli oggetti delle risorse Kubernetes, i volumi persistenti e le immagini interne, e il ripristino dei dati quando necessario.

Le macchine virtuali nell'ambiente di virtualizzazione OpenShift sono applicazioni containerizzate che vengono eseguite nei nodi worker della piattaforma OpenShift Container. È importante proteggere i metadati delle VM e i dischi persistenti delle VM, in modo che sia possibile recuperarli quando vengono persi o danneggiati.

I dischi persistenti delle VM di virtualizzazione OpenShift possono essere supportati dallo storage ONTAP integrato nel cluster OpenShift utilizzando ["Trident CSI"](#). In questa sezione utilizziamo ["Trident protetto"](#) per creare snapshot e backup di VM, inclusi i relativi volumi di dati su ONTAP Object Storage.

Quando necessario, eseguiamo il ripristino da uno snapshot o da un backup.

Trident Protect consente snapshot, backup, ripristino e disaster recovery di applicazioni e VM su un cluster OpenShift. Per le VM di OpenShift Virtualization, i dati che possono essere protetti con Trident Protect includono oggetti di risorse Kubernetes associati alle VM, volumi persistenti e immagini interne.

Di seguito sono riportate le versioni dei vari componenti utilizzati per gli esempi in questa sezione

- "Cluster OpenShift 4.17"
- "OpenShift Virtualization installato tramite OpenShift Virtualization Operator fornito da Red Hat"
- "Trident 25.02"
- "Trident protect 25.02"
- "ONTAP 9.16"

Crea App Vault per Object Storage

Crea AppVault

Prima di creare snapshot e backup per un'applicazione o una macchina virtuale, è necessario configurare un Object Storage in Trident Protect per archiviare snapshot e backup. Ciò avviene utilizzando il bucket CR. Solo gli amministratori possono creare un bucket CR e configurarlo. In Trident Protect il bucket CR è noto come AppVault. Gli oggetti AppVault sono la rappresentazione dichiarativa del flusso di lavoro Kubernetes di un bucket di archiviazione. Un CR di AppVault contiene le configurazioni necessarie affinché un bucket venga utilizzato nelle operazioni di protezione, come backup, snapshot, operazioni di ripristino e replica SnapMirror.

In questo esempio mostreremo l'uso di ONTAP S3 come storage di oggetti. Ecco il flusso di lavoro per la creazione di AppVault CR per ONTAP S3: 1. Creare un server di archiviazione oggetti S3 nell'SVM nel cluster ONTAP. 2. Crea un bucket nell'Object Store Server. 3. Creare un utente S3 nell'SVM. Conservare la chiave di accesso e la chiave segreta in un luogo sicuro. 4. In OpenShift, crea un segreto per archiviare le credenziali ONTAP S3. 5. Creare un oggetto AppVault per ONTAP S3

Configurare Trident Protect AppVault per ONTAP S3

```
# alias tp='tridentctl-protect'

# cat appvault-secret.yaml
apiVersion: v1
stringData:
  accessKeyID: "<access key of S3>"
  secretAccessKey: "<secret access key of S3>"
# you can also provide base 64 encoded values instead of string values
#data:
# base 64 encoded values
#  accessKeyID: < base 64 encoded access key>
#  secretAccessKey: <base 64 encoded secretAccess key>
kind: Secret
metadata:
  name: appvault-secret
  namespace: trident-protect
type: Opaque

# cat appvault.yaml
apiVersion: protect.trident.netapp.io/v1
kind: AppVault
metadata:
  name: ontap-s3-appvault
  namespace: trident-protect
spec:
  providerConfig:
    azure:
      accountName: ""
      bucketName: ""
      endpoint: ""
```

```

gcp:
  bucketName: ""
  projectID: ""
s3:
  bucketName: trident-protect
  endpoint: <lif for S3 access>
  secure: "false"
  skipCertValidation: "true"
providerCredentials:
  accessKeyID:
    valueFromSecret:
      key: accessKeyID
      name: appvault-secret
  secretAccessKey:
    valueFromSecret:
      key: secretAccessKey
      name: appvault-secret
providerType: OntapS3

# oc create -f appvault-secret.yaml -n trident-protect
# oc create -f appvault.yaml -n trident-protect

```

```

[root@localhost VM-DataProtection]# tp get appvault
+-----+-----+-----+-----+-----+-----+
| NAME          | PROVIDER | STATE   | ERROR | MESSAGE | AGE   |
+-----+-----+-----+-----+-----+-----+
| ontap-s3-appvault | OntapS3  | Available |      |          | 8d17h |
+-----+-----+-----+-----+-----+-----+
[root@localhost VM-DataProtection]# █

```

Creare una VM in OpenShift Virtualization

Creare una VM in OpenShift Virtualization

Gli screenshot seguenti mostrano la creazione della VM (demo-fedora nello spazio dei nomi demo) dalla console utilizzando il modello. Il disco radice sceglie automaticamente la classe di archiviazione predefinita, quindi verificare che la classe di archiviazione predefinita sia impostata correttamente. In questa configurazione, la classe di archiviazione predefinita è **sc-zonea-san**. Quando si crea il disco aggiuntivo, assicurarsi di scegliere la classe di archiviazione **sc-zonea-san** e di selezionare la casella di controllo **"Applica impostazioni di archiviazione ottimizzate"**. In questo modo le modalità di accesso verranno impostate su RWX e la modalità volume su Blocco.



Trident supporta la modalità di accesso RWX in modalità Block Volume per SAN (iSCSI, NVMe/TCP e FC). (È la modalità di accesso predefinita per NAS). La modalità di accesso RWX è necessaria se in un secondo momento è necessario eseguire la migrazione live delle VM.

StorageClasses

Name ▾ Search by name... /

Name ▴	Provisioner ▴
 sc-zonea-san - Default	csi.trident.netapp.io
 thin-csi	csi.vsphere.vmware.com

Catalog

VirtualMachines

Templates

InstanceTypes

Preferences

Bootable volumes

MigrationPolicies

Checkups

Migration >

Networking >

Storage ▾

PersistentVolumes

Project: demo ▾

VirtualMachines



No VirtualMachines found

Click Create VirtualMachine to create your first VirtualMachine or view the [catalog](#) tab to c

Create VirtualMachine ▾

From InstanceType

From template

With YAML



Fedora VM

fedora-server-small

×

▼ Template info

Operating system

Fedora VM

Workload type

Server (default)

Description

Template for Fedora Linux 39 VM or newer. A PVC with the Fedora disk image must be

▼ Storage ?

☐ Boot from CD ?

Disk source * ?

Template default ▼

Disk size *

-

30

+

GiB ▼

Quick create VirtualMachine

VirtualMachine name *

demo-fedora

Project

demo

Public SSH key

Not configured 

☒ Start this VirtualMachine after creation

Quick create VirtualMachine

Customize VirtualMachine

Activate Windows

Go to Settings to activate Windows.

Cancel

Catalog

Customize and create VirtualMachine

YAML

Template: Fedora VM

Overview

YAML

Scheduling

Environment

Network interfaces

Disks

Scripts

Metadata

Add disk ▼

Empty disk (blank)

Create a disk with no contents.

Use existing

Ephemeral disk (Container image)

Any changes are lost upon reboot

Volume

Add a volume already available on the cluster.

Volume snapshot

Add a snapshot available on the cluster to the VirtualMachine.

☒ Clone volume

Clone a volume available on the cluster and add it to the VirtualMachine.

Create VirtualMachine

Cancel

Mount Windows drivers disk

Drive	Interface	Storage class	
Disk	virtio	-	⋮
Disk	virtio	-	⋮

Activate Windows

Go to Settings to activate Window

Add disk



disk1

Disk size *



30



GiB



Type

Disk



Hot plug is enabled only for "Disk" and "Lun" types

Interface *

VirtIO



Hot plug is enabled only for "SCSI" interface

StorageClass

 sc-zonea-san



☒ Apply optimized StorageProfile settings

Optimized values Access mode: ReadWriteMany, Volume mode: Block.

Save

Cancel

Project: demo

[Overview](#)
[YAML](#)
[Scheduling](#)
[Environment](#)
[Network interfaces](#)
[Disks](#)
[Scripts](#)
[Metadata](#)

Add disk

Filter

Search by name...

Mount Windows drivers disk

Name	Source	Size	Drive	Interface	Storage class
cloudinitdisk	Other	-	Disk	virtio	-
disk1	Other	30 GiB	Disk	virtio	sc-zonea-san
rootdisk	PVC (auto import)	30 GiB	Disk	virtio	-

bootable

☒ Start this VirtualMachine after creation

Create VirtualMachine

Cancel

Activate Windows

Go to Settings to activate V

```
[root@localhost VM-DataProtection]# oc get vm,pods,pvc -n demo
NAME                                     AGE      STATUS   READY
virtualmachine.kubevirt.io/demo-fedora  2m54s   Running  True

NAME                                     READY   STATUS   RESTARTS   AGE
pod/virt-launcher-demo-fedora-69cts     1/1     Running  0          110s

NAME                                     STATUS   VOLUME                                     CAPACITY   ACCESS MODES   STORAGECLASS
persistentvolumeclaim/demo-fedora        Bound    pvc-406d9d57-4a5e-4123-93c1-13070b7b0140  30Gi       RWX             sc-zonea-san
persistentvolumeclaim/dv-demo-fedora-fuchsia-shrew-87 Bound    pvc-311f2f81-1d25-4a9a-b0cb-836b4e702f04  30Gi       RWX             sc-zonea-san
[root@localhost VM-DataProtection]#
```

Crea un'app

Crea app

Creare un'app Trident Protect per la VM

Nell'esempio, lo spazio dei nomi demo ha una VM e tutte le risorse dello spazio dei nomi vengono incluse durante la creazione dell'app.

```
# alias tp='tridentctl-protect'
# tp create app demo-vm --namespaces demo -n demo --dry-run > app.yaml

# cat app.yaml
apiVersion: protect.trident.netapp.io/v1
kind: Application
metadata:
  creationTimestamp: null
  name: demo-vm
  namespace: demo
spec:
  includedNamespaces:
  - namespace: demo
# oc create -f app.yaml -n demo
```

```
[root@localhost VM-DataProtection]# tp get app -n demo
+-----+-----+-----+-----+
| NAME   | NAMESPACES | STATE | AGE |
+-----+-----+-----+-----+
| demo-vm | demo       | Ready | 45s |
+-----+-----+-----+-----+
[root@localhost VM-DataProtection]#
```

Proteggi l'app creando un backup

Crea backup

Crea un backup su richiesta

Creare un backup per l'app (demo-vm) creata in precedenza che includa tutte le risorse nello spazio dei nomi demo. Fornire il nome dell'appvault in cui verranno archiviati i backup.

```
# tp create backup demo-vm-backup-on-demand --app demo-vm --appvault  
ontap-s3-appvault -n demo  
Backup "demo-vm-backup-on-demand" created.
```

```
[root@localhost VM-DataProtection]# tp get backup -n demo
```

NAME	APP	RECLAIM POLICY	STATE	ERROR	AGE
demo-vm-backup-on-demand	demo-vm	Retain	Completed		12m53s

```
[root@localhost VM-DataProtection]#
```

Crea backup pianificati

Creare una pianificazione per i backup specificando la granularità e il numero di backup da conservare.

```
# tp create schedule backup-schedule1 --app demo-vm --appvault ontap-
s3-appvault --granularity Hourly --minute 45 --backup-retention 1 -n
demo --dry-run>backup-schedule-demo-vm.yaml
schedule.protect.trident.netapp.io/backup-schedule1 created

#cat backup-schedule-demo-vm.yaml
apiVersion: protect.trident.netapp.io/v1
kind: Schedule
metadata:
  creationTimestamp: null
  name: backup-schedule1
  namespace: demo
spec:
  appVaultRef: ontap-s3-appvault
  applicationRef: demo-vm
  backupRetention: "1"
  dayOfMonth: ""
  dayOfWeek: ""
  enabled: true
  granularity: Hourly
  hour: ""
  minute: "45"
  recurrenceRule: ""
  snapshotRetention: "0"
status: {}
# oc create -f backup-schedule-demo-vm.yaml -n demo
```

```
[root@localhost VM-DataProtection]# tp get schedule -n demo
```

NAME	APP	SCHEDULE	ENABLED	STATE	ERROR	AGE
backup-schedule1	demo-vm	Hourly:min=45	true			9s

```
[root@localhost VM-DataProtection]# tp get backups -n demo
```

NAME	APP	RECLAIM POLICY	STATE	ERROR	AGE
demo-vm-backup-on-demand	demo-vm	Retain	Completed		44m4s
hourly-4c094-20250312144500	demo-vm	Retain	Completed		20m34s

```
[root@localhost VM-DataProtection]#
```

Ripristina dal backup

Ripristina dai backup

Ripristina la VM nello stesso namespace

Nell'esempio, il backup demo-vm-backup-on-demand contiene il backup con demo-app per la VM Fedora.

Per prima cosa, elimina la VM e assicurati che i PVC, il pod e gli oggetti VM vengano eliminati dallo spazio dei nomi "demo"

```
[root@localhost VM-DataProtection]# oc get vm,pods,pvc -n demo
NAME                                     AGE   STATUS   READY
virtualmachine.kubevirt.io/demo-fedora  59m   Running  True

NAME                                     READY   STATUS   RESTARTS   AGE
pod/virt-launcher-demo-fedora-69cts     1/1     Running  0           58m

NAME                                     STATUS   VOLUME                                     CAPACITY
CLASS   VOLUMEATTRIBUTESCLASS   AGE
persistentvolumeclaim/demo-fedora          Bound    pvc-406d9d57-4a5e-4123-93c1-13070b7b0140  30Gi
a-san   <unset>                  59m
persistentvolumeclaim/dv-demo-fedora-fuchsia-shrew-87 Bound    pvc-311f2f81-1d25-4a9a-b0cb-836b4e702f04  30Gi
a-san   <unset>                  59m
[root@localhost VM-DataProtection]#
[root@localhost VM-DataProtection]# oc delete vm demo-fedora -n demo
virtualmachine.kubevirt.io "demo-fedora" deleted
[root@localhost VM-DataProtection]# oc get vm,pods,pvc -n demo
No resources found in demo namespace.
[root@localhost VM-DataProtection]#
```

Activate
Go to Settings

Ora, crea un oggetto di ripristino tramite backup sul posto.

```
# tp create bir demo-fedora-restore --backup demo/demo-vm-backup-on-
demand -n demo --dry-run>vm-demo-bir.yaml

# cat vm-demo-bir.yaml
apiVersion: protect.trident.netapp.io/v1
kind: BackupInplaceRestore
metadata:
  annotations:
    protect.trident.netapp.io/max-parallel-restore-jobs: "25"
  creationTimestamp: null
  name: demo-fedora-restore
  namespace: demo
spec:
  appArchivePath: demo-vm_cc8adc7a-0c28-460b-a32f-
0a7b3d353e13/backups/demo-vm-backup-on-demand_f6af3513-9739-480e-88c7-
4cca45808a80
  appVaultRef: ontap-s3-appvault
  resourceFilter: {}
status:
  postRestoreExecHooksRunResults: null
  state: ""

# oc create -f vm-demo-bir.yaml -n demo
backupinplacerestore.protect.trident.netapp.io/demo-fedora-restore
created
```

```
[root@localhost VM-DataProtection]# tp get bir -n demo
```

NAME	APPVAULT	STATE	ERROR	AGE
demo-fedora-restore	ontap-s3-appvault	Completed		28m17s

```
[root@localhost VM-DataProtection]#
```

Verificare che la VM, i pod e i PVC siano ripristinati

```
[root@localhost VM-DataProtection]# oc get vm,pods,pvc -n demo
```

NAME	AGE	STATUS	READY
virtualmachine.kubevirt.io/demo-fedora	116s	Running	True

NAME	READY	STATUS	RESTARTS	AGE
pod/virt-launcher-demo-fedora-9kfxh	1/1	Running	0	116s

NAME	STATUS	VOLUME	CAPACITY	ACCESS MODES	STORAGECLASS
persistentvolumeclaim/demo-fedora	Bound	pvc-6f69a62c-285c-4980-b0dd-6c85baccf346	30Gi	RWX	sc-zonea-san
persistentvolumeclaim/dv-demo-fedora-fuchsia-shrew-87	Bound	pvc-81d82d82-7aca-40fc-8f8f-6e99246e63f8	30Gi	RWX	sc-zonea-san

```
[root@localhost VM-DataProtection]#
```

Ripristina la VM in uno spazio dei nomi diverso

Per prima cosa crea un nuovo namespace in cui vuoi ripristinare l'app, in questo esempio demo2. Quindi creare un oggetto di backup e ripristino

```
# tp create br demo2-fedora-restore --backup demo/hourly-4c094-20250312154500 --namespace-mapping demo:demo2 -n demo2 --dry-run>vm-demo2-br.yaml

# cat vm-demo2-br.yaml
apiVersion: protect.trident.netapp.io/v1
kind: BackupRestore
metadata:
  annotations:
    protect.trident.netapp.io/max-parallel-restore-jobs: "25"
  creationTimestamp: null
  name: demo2-fedora-restore
  namespace: demo2
spec:
  appArchivePath: demo-vm_cc8adc7a-0c28-460b-a32f-0a7b3d353e13/backups/hourly-4c094-20250312154500_aaa14543-a3fa-41f1-a04c-44b1664d0f81
  appVaultRef: ontap-s3-appvault
  namespaceMapping:
    - destination: demo2
      source: demo
  resourceFilter: {}
status:
  conditions: null
  postRestoreExecHooksRunResults: null
  state: ""
# oc create -f vm-demo2-br.yaml -n demo2
```

```
[root@localhost VM-DataProtection]# tp get br -n demo2
```

NAME	APPVAULT	STATE	ERROR	AGE
demo2-fedora-restore	ontap-s3-appvault	Completed		38m52s

Verificare che la VM, i pod e i pvc siano creati nel nuovo namespace demo2.

```
[root@localhost VM-DataProtection]#
[root@localhost VM-DataProtection]# oc get vm,pods,pvc -n demo2
```

NAME	AGE	STATUS	READY
virtualmachine.kubevirt.io/demo-fedora	5m8s	Running	True

NAME	READY	STATUS	RESTARTS	AGE
pod/virt-launcher-demo-fedora-c7xc6	1/1	Running	0	5m7s

NAME	STATUS	VOLUME	CAPACITY	ACCESS MODES	STORAGECLASS
persistentvolumeclaim/demo-fedora	Bound	pvc-4d278ae2-76cc-46f1-bbf8-071ae75e4a82	30Gi	RWX	sc-zonea-san
persistentvolumeclaim/dv-demo-fedora-fuchsia-shrew-87	Bound	pvc-9b96d09c-7226-4ffc-829b-2cee88e7a117	30Gi	RWX	sc-zonea-san

```
[root@localhost VM-DataProtection]#
```

Proteggi l'app utilizzando gli snapshot

Crea snapshot

Crea uno snapshot on-demand Crea uno snapshot per l'app e specifica l'appvault in cui deve essere archiviato.

```
# tp create snapshot demo-vm-snapshot-ondemand --app demo-vm --appvault
ontap-s3-appvault -n demo --dry-run
# cat demo-vm-snapshot-on-demand.yaml
apiVersion: protect.trident.netapp.io/v1
kind: Snapshot
metadata:
  creationTimestamp: null
  name: demo-vm-snapshot-ondemand
  namespace: demo
spec:
  appVaultRef: ontap-s3-appvault
  applicationRef: demo-vm
  completionTimeout: 0s
  volumeSnapshotsCreatedTimeout: 0s
  volumeSnapshotsReadyToUseTimeout: 0s
status:
  conditions: null
  postSnapshotExecHooksRunResults: null
  preSnapshotExecHooksRunResults: null
  state: ""

# oc create -f demo-vm-snapshot-on-demand.yaml
snapshot.protect.trident.netapp.io/demo-vm-snapshot-ondemand created
```

```
[root@localhost VM-DataProtection]#
[root@localhost VM-DataProtection]# oc get vm,pods,pvc -n demo2
NAME                                AGE    STATUS    READY
virtualmachine.kubevirt.io/demo-fedora  5m8s   Running   True

NAME                                READY    STATUS    RESTARTS   AGE
pod/virt-launcher-demo-fedora-c7xc6  1/1      Running   0           5m7s

NAME                                STATUS    VOLUME                                     CAPACITY   ACCESS MODES   STORAGECLASS
persistentvolumeclaim/demo-fedora    Bound     pvc-4d278ae2-76cc-46f1-bbf8-071ae75e4a82  30Gi       RNX             sc-zonea-san
persistentvolumeclaim/dv-demo-fedora-fuchsia-shrew-87  Bound     pvc-9b96d09c-7226-4ffc-829b-2cee88e7a117  30Gi       RWX             sc-zonea-san
[root@localhost VM-DataProtection]#
```

Crea una pianificazione per gli snapshot Crea una pianificazione per gli snapshot. Specificare la granularità e il numero di snapshot da conservare.

```
# tp create Schedule snapshot-schedule1 --app demo-vm --appvault ontap-
s3-appvault --granularity Hourly --minute 50 --snapshot-retention 1 -n
demo --dry-run>snapshot-schedule-demo-vm.yaml

# cat snapshot-schedule-demo-vm.yaml
apiVersion: protect.trident.netapp.io/v1
kind: Schedule
metadata:
  creationTimestamp: null
  name: snapshot-schedule1
  namespace: demo
spec:
  appVaultRef: ontap-s3-appvault
  applicationRef: demo-vm
  backupRetention: "0"
  dayOfMonth: ""
  dayOfWeek: ""
  enabled: true
  granularity: Hourly
  hour: ""
  minute: "50"
  recurrenceRule: ""
  snapshotRetention: "1"
status: {}

# oc create -f snapshot-schedule-demo-vm.yaml
schedule.protect.trident.netapp.io/snapshot-schedule1 created
```

```
[root@localhost VM-DataProtection]# tp get schedule -n demo
```

NAME	APP	SCHEDULE	ENABLED	STATE	ERROR	AGE
backup-schedule1	demo-vm	Hourly:min=45	true			5d23h
snapshot-schedule1	demo-vm	Hourly:min=50	true			12s

```
[root@localhost VM-DataProtection]#
```

```
[root@localhost VM-DataProtection]# tp get snapshots -n demo
```

NAME	APP	RECLAIM POLICY	STATE	ERROR	AGE
backup-39b67e1c-f875-4045-93df-78634bae9dfb	demo-vm	Delete	Completed		6m29s
demo-vm-snapshot-ondemand	demo-vm	Delete	Completed		21m30s
hourly-51839-20250318135000	demo-vm	Delete	Completed		1m29s

```
[root@localhost VM-DataProtection]#
```

Ripristina da snapshot

Ripristina da snapshot

Ripristinare la VM dallo snapshot nello stesso namespace Eliminare la VM demo-fedora dal namespace demo2.

```
[root@localhost RedHat]# oc get vm,pvc -n demo
NAME                                     AGE   STATUS   READY
virtualmachine.kubevirt.io/demo-fedora  28h   Running  True

NAME                                     STATUS   VOLUME
ORAGECLASS  VOLUMEATTRIBUTESCLASS  AGE
persistentvolumeclaim/demo-fedora          Bound    pvc-e0d5f79d-dff9-450d-be0e-90ab6880b7af
-zonea-san  <unset>          28h
persistentvolumeclaim/dv-demo-fedora-fuchsia-shrew-87  Bound    pvc-e6f7890a-70c7-4538-9035-5e2e9379511d
-zonea-san  <unset>          28h
[root@localhost RedHat]# oc delete virtualmachine.kubevirt.io/demo-fedora -n demo
virtualmachine.kubevirt.io "demo-fedora" deleted
[root@localhost RedHat]#
```

Act
Got

Creare un oggetto snapshot-in-place-restore dallo snapshot della VM.

```
# tp create sir demo-fedora-restore-from-snapshot --snapshot demo/demo-vm-snapshot-ondemand -n demo --dry-run>vm-demo-sir.yaml

# cat vm-demo-sir.yaml
apiVersion: protect.trident.netapp.io/v1
kind: SnapshotInplaceRestore
metadata:
  creationTimestamp: null
  name: demo-fedora-restore-from-snapshot
  namespace: demo
spec:
  appArchivePath: demo-vm_cc8adc7a-0c28-460b-a32f-0a7b3d353e13/snapshots/20250318132959_demo-vm-snapshot-ondemand_e3025972-30c0-4940-828a-47c276d7b034
  appVaultRef: ontap-s3-appvault
  resourceFilter: {}
status:
  conditions: null
  postRestoreExecHooksRunResults: null
  state: ""

# oc create -f vm-demo-sir.yaml
snapshotinplacerestore.protect.trident.netapp.io/demo-fedora-restore-from-snapshot created
```

```
[root@localhost VM-DataProtection]# tp get sir -n demo
```

NAME	APPVAULT	STATE	ERROR	AGE
demo-fedora-restore-from-snapshot	ontap-s3-appvault	Completed		58m17s

```
[root@localhost VM-DataProtection]#
```

Verificare che la VM e i suoi PVC siano creati nello spazio dei nomi demo.

```
[root@localhost RedHat]# oc get vm,pvc -n demo
```

NAME	AGE	STATUS	READY
virtualmachine.kubevirt.io/demo-fedora	5m17s	Running	True

NAME	STATUS	VOLUME
persistentvolumeclaim/demo-fedora	Bound	pvc-e2f418b0-1b97-40fc-9cb8-943b370d85bc
persistentvolumeclaim/dv-demo-fedora-fuchsia-shrew-87	Bound	pvc-db085154-079f-45ad-9e62-9656e913d01c

```
[root@localhost RedHat]#
```

Ripristina la VM dallo snapshot in uno spazio dei nomi diverso

Eliminare la VM nello spazio dei nomi demo2 precedentemente ripristinata dal backup.

```
[root@localhost VM-DataProtection]# oc get vm,pods,pvc -n demo2
```

NAME	AGE	STATUS	READY
virtualmachine.kubevirt.io/demo-centos	3m12s	Running	True
virtualmachine.kubevirt.io/demo-fedora	3m11s	Running	True

NAME	READY	STATUS	RESTARTS	AGE
pod/virt-launcher-demo-centos-w77rr	1/1	Running	0	3m11s
pod/virt-launcher-demo-fedora-wwdtc	1/1	Running	0	3m11s

NAME	STORAGECLASS	VOLUMEATTRIBUTESCLASS	AGE	STATUS	VOLUME	CAPACITY	ACCESS	MODES
persistentvolumeclaim/demo-centos	sc-zonea-san	<unset>	3m26s	Bound	pvc-7aafd5e2-d4cf-4af6-a259-c68e016ec6cd	30Gi	RWX	
persistentvolumeclaim/demo-fedora	sc-zonea-san	<unset>	3m25s	Bound	pvc-c1f8145a-56ea-42c5-abb4-6457b6853e1c	30Gi	RWX	
persistentvolumeclaim/dv-demo-centos-lavender-tortoise-34	sc-zonea-san	<unset>	3m20s	Bound	pvc-e82b6d98-6762-4473-9bb8-7e98efcb7987	30Gi	RWX	
persistentvolumeclaim/dv-demo-fedora-fuchsia-shrew-87	sc-zonea-san	<unset>	3m16s	Bound	pvc-ee13cfbc-2921-4129-a214-a0f6b10335cd	30Gi	RWX	

```
[root@localhost VM-DataProtection]# oc delete virtualmachine.kubevirt.io/demo-fedora -n demo2
```

```
virtualmachine.kubevirt.io "demo-fedora" deleted
```

Creare l'oggetto di ripristino dello snapshot dallo snapshot e fornire la mappatura dello spazio dei nomi.


```
# tp create sr demo2-fedora-restore-from-snapshot --snapshot demo/demo-vm-snapshot-ondemand --namespace-mapping demo:demo2 -n demo2 --dry-run>vm-demo2-sr.yaml

# cat vm-demo2-sr.yaml
apiVersion: protect.trident.netapp.io/v1
kind: SnapshotRestore
metadata:
  creationTimestamp: null
  name: demo2-fedora-restore-from-snapshot
  namespace: demo2
spec:
  appArchivePath: demo-vm_cc8adc7a-0c28-460b-a32f-0a7b3d353e13/snapshots/20250318132959_demo-vm-snapshot-ondemand_e3025972-30c0-4940-828a-47c276d7b034
  appVaultRef: ontap-s3-appvault
  namespaceMapping:
  - destination: demo2
    source: demo
  resourceFilter: {}
status:
  postRestoreExecHooksRunResults: null
  state: ""

# oc create -f vm-demo2-sr.yaml
snapshotrestore.protect.trident.netapp.io/demo2-fedora-restore-from-snapshot created
```

```
[root@localhost VM-DataProtection]# tp get sr -n demo2
```

NAME	APPVAULT	STATE	ERROR	AGE
demo2-fedora-restore-from-snapshot	ontap-s3-appvault	Completed		15m22s

Verificare che la VM e i suoi PVC siano ripristinati nel nuovo namespace demo2.

```
[root@localhost RedHat]# oc get vm,pvc -n demo2
```

NAME	AGE	STATUS	READY
virtualmachine.kubevirt.io/demo-fedora	29h	Running	True

NAME	ORAGECLASS	VOLUMEATTRIBUTESCLASS	AGE	STATUS	VOLUME
persistentvolumeclaim/demo-fedora				Bound	pvc-35dcd9b2-4fca-486c-af9e-596bc5b4dc15
-zonea-san	<unset>		29h		
persistentvolumeclaim/dv-demo-fedora-fuchsia-shrew-87				Bound	pvc-575a3111-382f-4933-a778-0089fa1ea2af
-zonea-san	<unset>		29h		

```
[root@localhost RedHat]#
```

Ripristina una VM specifica

Selezione di VM specifiche in uno spazio dei nomi per creare snapshot/backup e ripristinare

Nell'esempio precedente, avevamo una singola VM all'interno di uno spazio dei nomi. Includendo l'intero namespace nel backup, sono state acquisite tutte le risorse associate a quella VM. Nell'esempio seguente, aggiungiamo un'altra VM allo stesso namespace e creiamo un'app solo per questa nuova VM utilizzando un selettore di etichette.

Crea una nuova VM (demo-centos vm) nello spazio dei nomi demo

```
[root@localhost VM-DataProtection]# oc get vm,pod,pvc -n demo
NAME                                     AGE      STATUS    READY
virtualmachine.kubevirt.io/demo-centos  2m47s   Running   True
virtualmachine.kubevirt.io/demo-fedora   81m     Running   True

NAME                                     READY    STATUS    RESTARTS   AGE
pod/virt-launcher-demo-centos-2nq6g     1/1      Running   0           113s
pod/virt-launcher-demo-fedora-9kfxh     1/1      Running   0           81m

NAME                                     STATUS    VOLUME                                     CAPACITY   ACCESS MODES   STORAGECLASS
persistentvolumeclaim/demo-centos        Bound     pvc-ed0f492b-0109-471d-b395-9077ae5f1fa7  30Gi       RWX             sc-zonea-san
persistentvolumeclaim/demo-fedora        Bound     pvc-6f69a62c-285c-4980-b0dd-6c85baccf346  30Gi       RWX             sc-zonea-san
persistentvolumeclaim/dv-demo-centos-lavender-tortoise-34  Bound     pvc-3c01142a-4344-4293-ae67-7d3925c56211  30Gi       RWX             sc-zonea-san
persistentvolumeclaim/dv-demo-fedora-fuchsia-shrew-87     Bound     pvc-81d82d82-7aca-40fc-8f8f-6e99246e63f8  30Gi       RWX             sc-zonea-san
[root@localhost VM-DataProtection]#
```

Etichettare la VM demo-centos e le risorse associate

```
[root@localhost VM-DataProtection]# oc label vm demo-centos category=protect-demo-centos -n demo
virtualmachine.kubevirt.io/demo-centos labeled
[root@localhost VM-DataProtection]# oc label pvc demo-centos category=protect-demo-centos -n demo
persistentvolumeclaim/demo-centos labeled
[root@localhost VM-DataProtection]# oc label pvc dv-demo-centos-lavender-tortoise-34 category=protect-demo-centos -n demo
persistentvolumeclaim/dv-demo-centos-lavender-tortoise-34 labeled
[root@localhost VM-DataProtection]#
```

Verificare che la VM demo-centos e i pvc abbiano le etichette

```
[root@localhost VM-DataProtection]# oc get vm --show-labels -n demo
NAME      AGE      STATUS    READY   LABELS
demo-centos 6m31s   Running   True    app=demo-centos,category=protect-demo-centos,kubevirt.io/dynamic-credentials-support=true,vm.kubevirt.io/template.namespace=openshift,vm.kubevirt.io/template.revision=1,vm.kubevirt.io/template.version=v0.31.1,vm.kubevirt.io/template=centos-stream9-server-small
demo-fedora 85m     Running   True    app=demo-fedora,vm.kubevirt.io/template.namespace=openshift,vm.kubevirt.io/template.revision=1,vm.kubevirt.io/template.version=v0.31.1,vm.kubevirt.io/template=fedora-server-small
[root@localhost VM-DataProtection]#
```

```
[root@localhost VM-DataProtection]# oc get pvc --show-labels -n demo
NAME      STATUS    VOLUME                                     CAPACITY   ACCESS MODES   STORAGECLASS   VOLUMEATTRIBUTESCLASS   AGE      LABELS
demo-centos Bound     pvc-ed0f492b-0109-471d-b395-9077ae5f1fa7  30Gi       RWX             sc-zonea-san   <unset>                  7m41s   app.kubernetes.io/component=storage,app.kubernetes.io/managed-by=cdi-controller,app.kubernetes.io/part-of=hyperconverged-cluster,app.kubernetes.io/version=4.17.5,app=containerized-data-importer,category=protect-demo-centos,instancetype.kubevirt.io/default-instancetype=ul.medium,instancetype.kubevirt.io/default-preference=centos-stream9,kubevirt.io/created-by=a6a7b49c-669a-4e21-aa78-20743671b284
demo-fedora Bound     pvc-6f69a62c-285c-4980-b0dd-6c85baccf346  30Gi       RWX             sc-zonea-san   <unset>                  86m     app.kubernetes.io/component=storage,app.kubernetes.io/managed-by=cdi-controller,app.kubernetes.io/part-of=hyperconverged-cluster,app.kubernetes.io/version=4.17.5,app=containerized-data-importer,instancetype.kubevirt.io/default-instancetype=ul.medium,instancetype.kubevirt.io/default-preference=fedora,kubevirt.io/created-by=7d5184e9-22f0-4456-9afe-3d1904c430f9
dv-demo-centos-lavender-tortoise-34 Bound     pvc-3c01142a-4344-4293-ae67-7d3925c56211  30Gi       RWX             sc-zonea-san   <unset>                  7m41s   app.kubernetes.io/component=storage,app.kubernetes.io/managed-by=cdi-controller,app.kubernetes.io/part-of=hyperconverged-cluster,app.kubernetes.io/version=4.17.5,app=containerized-data-importer,category=protect-demo-centos,kubevirt.io/created-by=a6a7b49c-669a-4e21-aa78-20743671b284
dv-demo-fedora-fuchsia-shrew-87 Bound     pvc-81d82d82-7aca-40fc-8f8f-6e99246e63f8  30Gi       RWX             sc-zonea-san   <unset>                  86m     app.kubernetes.io/component=storage,app.kubernetes.io/managed-by=cdi-controller,app.kubernetes.io/part-of=hyperconverged-cluster,app.kubernetes.io/version=4.17.5,app=containerized-data-importer,kubevirt.io/created-by=7d5184e9-22f0-4456-9afe-3d1904c430f9
[root@localhost VM-DataProtection]#
```

Crea un'app solo per una VM specifica (demo-centos) utilizzando il selettore di etichette

```
# tp create app demo-centos-app --namespaces 'demo(category=protect-
demo-centos)' -n demo --dry-run>demo-centos-app.yaml

# cat demo-centos-app.yaml

apiVersion: protect.trident.netapp.io/v1
kind: Application
metadata:
  creationTimestamp: null
  name: demo-centos-app
  namespace: demo
spec:
  includedNamespaces:
  - labelSelector:
      matchLabels:
        category: protect-demo-centos
        namespace: demo
status:
  conditions: null

# oc create -f demo-centos-app.yaml -n demo
application.protect.trident.netapp.io/demo-centos-app created
```

```
[root@localhost VM-DataProtection]# tp get app -n demo
```

NAME	NAMESPACES	STATE	AGE
demo-centos-app	demo	Ready	56s
demo-vm	demo	Ready	4h6m

Il metodo per creare backup e snapshot su richiesta e in base a una pianificazione è lo stesso mostrato in precedenza. Poiché l'app trident-protect utilizzata per creare snapshot o backup contiene solo la VM specifica dello spazio dei nomi, il ripristino da essi ripristina solo una VM specifica. Di seguito è riportato un esempio di operazione di backup/ripristino.

Crea un backup di una VM specifica in uno spazio dei nomi utilizzando la relativa app

Nei passaggi precedenti è stata creata un'app utilizzando selettori di etichette per includere solo la macchina virtuale CentOS nello spazio dei nomi demo. Crea un backup (backup su richiesta, in questo esempio) per questa app.

```
# tp create backup demo-centos-backup-on-demand --app demo-centos-app
--appvault ontap-s3-appvault -n demo
Backup "demo-centos-backup-on-demand" created.
```

NAME	APP	RECLAIM POLICY	STATE	ERROR	AGE
demo-centos-backup-on-demand	demo-centos-app	Retain	Completed		13m22s
demo-vm-backup-on-demand	demo-vm	Retain	Completed		4h19m
hourly-4c094-20250312174500	demo-vm	Retain	Completed		56m17s

Ripristina una VM specifica nello stesso namespace Il backup di una VM specifica (centos) è stato creato utilizzando l'app corrispondente. Se da questo viene creato un backup-in-place-restore o un backup-restore, verrà ripristinata solo questa specifica VM. Eliminare la VM Centos.

```
[root@localhost RedHat]# oc get vm,pvc -n demo
NAME                                     AGE      STATUS    READY
virtualmachine.kubevirt.io/demo-centos  4m27s    Running   True
virtualmachine.kubevirt.io/demo-fedora    4m27s    Running   True

NAME                                     STATUS    VOLUME
persistentvolumeclaim/demo-centos        Bound     pvc-e8faeaf8-fc0c-4d92-96de-c83a335a7a17
  sc-zonea-san <unset>                     4m33s
persistentvolumeclaim/demo-fedora        Bound     pvc-e2f418b0-1b97-40fc-9cb8-943b370d85bc
  sc-zonea-san <unset>                     4m33s
persistentvolumeclaim/dv-demo-centos-lavender-tortoise-34 Bound     pvc-66eb7996-1420-4513-a67c-2824f08534da
  sc-zonea-san <unset>                     4m33s
persistentvolumeclaim/dv-demo-fedora-fuchsia-shrew-87    Bound     pvc-db085154-079f-45ad-9e62-9656e913d01c
  sc-zonea-san <unset>                     4m32s

[root@localhost RedHat]# oc delete virtualmachine.kubevirt.io/demo-centos -n demo
virtualmachine.kubevirt.io "demo-centos" deleted
[root@localhost RedHat]# oc get vm,pvc -n demo
NAME                                     AGE      STATUS    READY
virtualmachine.kubevirt.io/demo-fedora    5m17s    Running   True

NAME                                     STATUS    VOLUME
persistentvolumeclaim/demo-fedora        Bound     pvc-e2f418b0-1b97-40fc-9cb8-943b370d85bc
persistentvolumeclaim/dv-demo-fedora-fuchsia-shrew-87    Bound     pvc-db085154-079f-45ad-9e62-9656e913d01c
[root@localhost RedHat]#
```

Creare un ripristino sul posto del backup da demo-centos-backup-on-demand e verificare che la VM centos sia stata ricreata.

```
#tp create bir demo-centos-restore --backup demo/demo-centos-backup-on-demand -n demo
BackupInplaceRestore "demo-centos-restore" created.
```



```
[root@localhost RedHat]# tp get br -n demo
```

NAME	APPVAULT	STATE	ERROR	AGE
demo-centos-restore	ontap-s3-appvault	Completed		57m9s
demo-fedora-restore	ontap-s3-appvault	Completed		7d5h

```
[root@localhost RedHat]# oc get vm,pvc -n demo
```

NAME	AGE	STATUS	READY
virtualmachine.kubevirt.io/demo-centos	29m	Running	True
virtualmachine.kubevirt.io/demo-fedora	85m	Running	True

NAME	STORAGECLASS	VOLUMEATTRIBUTESCLASS	AGE	STATUS	VOLUME
persistentvolumeclaim/demo-centos	sc-zonea-san	<unset>	29m	Bound	pvc-82954bf7-4a7e-4e0c-9a04-4fa152e1b0ef
persistentvolumeclaim/demo-fedora	sc-zonea-san	<unset>	85m	Bound	pvc-e2f418b0-1b97-40fc-9cb8-943b370d85bc
persistentvolumeclaim/dv-demo-centos-lavender-tortoise-34	sc-zonea-san	<unset>	29m	Bound	pvc-2a8d4eb5-ed6d-4408-b85d-e218e9a5d4b0
persistentvolumeclaim/dv-demo-fedora-fuchsia-shrew-87	sc-zonea-san	<unset>	85m	Bound	pvc-db085154-079f-45ad-9e62-9656e913d01c

```
[root@localhost RedHat]#
```

Ripristinare una VM specifica in uno spazio dei nomi diverso Creare un ripristino di backup in uno spazio dei nomi diverso (demo3) da demo-centos-backup-on-demand e verificare che la VM centos sia stata ricreata.

```
# tp create br demo2-centos-restore --backup demo/demo-centos-backup-on-demand --namespace-mapping demo:demo3 -n demo3
BackupRestore "demo2-centos-restore" created.
```

```
[root@localhost RedHat]#
```

```
[root@localhost RedHat]# tp get br -n demo3
```

NAME	APPVAULT	STATE	ERROR	AGE
demo2-centos-restore	ontap-s3-appvault	Completed		52m57s

```
[root@localhost RedHat]#
```

```
[root@localhost RedHat]#
```

```
[root@localhost RedHat]# oc get vm,pvc -n demo3
```

NAME	AGE	STATUS	READY
virtualmachine.kubevirt.io/demo-centos	19m	Running	True

NAME	STORAGECLASS	VOLUMEATTRIBUTESCLASS	AGE	STATUS	VOLUME
persistentvolumeclaim/demo-centos	sc-zonea-san	<unset>	19m	Bound	pvc-0a14e38f-07de-4e09-8f88-14a9a8bb45c2
persistentvolumeclaim/dv-demo-centos-lavender-tortoise-34	sc-zonea-san	<unset>	19m	Bound	pvc-d4f9cf2f-264c-4d02-94bf-0db28b427acc

```
[root@localhost RedHat]#
```

Dimostrazione video

Il seguente video mostra una dimostrazione della protezione di una VM tramite snapshot

[Protezione di una VM](#)

Informazioni sul copyright

Copyright © 2025 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.