使用Trident Protect 保护数据

NetApp public and hybrid cloud solutions

NetApp
February 04, 2026

目录

使用Trident Protect 保护数据

使用Trident Protect 为 OpenShift 容器平台中的容器应用提供数据保护

参考文档的此部分提供了使用Trident保护创建容器应用程序的快照和备份的详细信息。NetApp Trident Protect 提供高级应用程序数据管理功能，可增强由NetApp ONTAP存储系统和NetApp Trident CSI 存储配置器支持的有状态 Kubernetes 应用程序的功能和可用性。Trident Protect 创建应用程序快照和备份，这意味着不仅创建持久卷中的应用程序数据的快照和备份，还创建应用程序元数据的快照和备份。 Trident Protect 创建的快照和备份可以存储在以下任何对象存储中，并在稍后从中恢复。

- AWS S3
- Azure Blob 存储
- Google Cloud Storage
- Ontap S3
- 存储网格
- 任何其他 S3 兼容存储

Trident Protect 使用基于角色的访问控制 (RBAC) 的 Kubernetes 模型。默认情况下， Trident Protect 提供一个名为 trident-protect 的单一系统命名空间及其关联的默认服务帐户。如果您的组织拥有许多用户或特定的安全需求，您可以使用Trident Protect 的 RBAC 功能来更精细地控制对资源和命名空间的访问。

有关Trident Protect 中的 RBAC 的更多信息，请参阅"Trident保护文档"

> 集群管理员可以访问默认的 trident-protect 命名空间中的资源，也可以访问所有其他命名空间中的资源。用户无法在 trident-protect 命名空间中创建应用程序数据管理自定义资源 (CR)，如快照和备份 CR。作为最佳实践，用户需要在应用程序命名空间中创建这些 CR。

可以按照文档中提供的说明安装Trident Protect"此处"本节将展示使用Trident Protect 对容器应用程序进行数据保护和恢复应用程序的工作流程。1.快照创建（按需、按计划）2.从快照恢复（恢复到相同和不同的命名空间）3.备份创建4.从备份还原

先决条件

在为应用程序创建快照和备份之前，必须在Trident Protect 中配置对象存储来存储快照和备份。这是使用存储桶 CR 完成的。只有管理员可以创建存储桶 CR 并对其进行配置。在Trident Protect 中，存储桶 CR 被称为 AppVault。 AppVault 对象是存储桶的声明性 Kubernetes 工作流表示。 AppVault CR 包含存储桶在保护操作（例如备份、快照、恢复操作和SnapMirror复制）中使用的必要配置。

在此示例中，我们将展示如何使用ONTAP S3 作为对象存储。以下是为ONTAP S3 创建 AppVault CR 的工作流程： 1.在ONTAP集群中的 SVM 中创建 S3 对象存储服务器。2.在对象存储服务器中创建一个存储桶。3.在 SVM 中创建 S3 用户。将访问密钥和密钥保存在安全的地方。4.在 OpenShift 中，创建一个密钥来存储ONTAP S3 凭证。5.为ONTAP S3 创建 AppVault 对象

为**ONTAP S3** 配置**Trident**保护 **AppVault**

使用**ONTAP S3** 作为 **AppVault** 配置**Trident Protect** 的示例 **yaml** 文件

```
# 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]#
```

用于安装 **postgresql** 应用程序的示例 **yaml** 文件

```
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]#
```

创建快照

创建按需快照

```
# 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
Warning: apps.openshift.io/v1 DeploymentConfig is deprecated in v4.14+, unavailable in v4.10000+
Warning: 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   STORAG
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                  SOURCESNAPSHOTCONTENT   RESTORESIZE   SNAPSHOTCLASS           SNAPSHOTCONTENT
            CREATIONTIME   AGE
volumesnapshot.snapshot.storage.k8s.io/snapshot-2e94d04c-c8ea-446a-8d47-64b0bee67107-pvc-951a9910-9edb-48ae-898a-1aed9aa25dc3   true
      postgres-pvc                                          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                            368Ki         trident-snapshotclass   snapcontent-86a464d4-ffd8-4279-9cf7-88
88a097c001   13m            13m
```

创建计划 使用以下命令，将每天 15:33 创建快照，并保留两个快照和备份。

```
# 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]#
```

使用 **yaml** 创建计划

```
# 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]#
```

您可以看到按照此计划创建的快照。

```
[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]#
```

还创建了卷快照。

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

删除应用程序以模拟应用程序丢失

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

从快照恢复到同一命名空间

```
# tp create sir postgres-sir --snapshot postgres/hourly-3f1ee-
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 |       |
+--------------+--------------------+-----------+-------+-------+
```

应用程序及其 PVC 恢复到同一个命名空间。

```
[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]#
```

从快照还原到不同的命名空间

```
# tp create snapshotrestore postgres-restore --snapshot
postgres/hourly-3f1ee-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 |       |
+------------------+--------------------+-----------+-------+-------+
```

您可以看到应用程序已恢复到新的命名空间。

```
[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]#
```

创建备份

创建按需备份

```
# 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  |       |
+-----------------------------+-------------+-----------+--------+-------+
```

创建备份计划

上面列表中的每日和每小时备份是根据之前设置的计划创建的。

```
# 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 |       |
+-----------+-------------+----------------------+---------+-------+------+-------+
```

删除应用程序和 **PVC** 以模拟数据丢失。

```
[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]#
```

恢复到相同的命名空间 #tp create bir postgres-bir --backup postgres/hourly-3f1ee-20250224023300 -n postgres BackupInplaceRestore "postgres-bir" created。

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

应用程序和 PVC 在同一个命名空间中恢复。

```
[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   VOLUMEATTRIBUT
ESCLASS   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
```

恢复到不同的命名空间 创建一个新的命名空间。从备份恢复到新的命名空间。

```
[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]#
```

要将应用程序克隆或迁移到不同的集群（执行跨集群克隆），请在源集群上创建备份，然后将备份还原到不同的集群。确保目标集群上安装了Trident保护。

在源集群上，执行下图所示的步骤：

```
[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  |       |
+------------------------------+--------------+-----------+--------+-------+
```


从源集群，将上下文切换到目标集群。然后，确保可以从目标集群上下文访问 AppVault，并从目标集群获取 AppVault 内容。

```
[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      |  TYPE  |            NAME            |         TIMESTAMP         |
                      PATH                                                                       |
+--------------+--------------+--------+----------------------------+---------------------------+-------------------------------------
-------------------------------------------------------------------------------------------------+
| ocp-cluster11 | bbox         | backup | bboxbackup1                | 2025-01-17 15:57:49 (UTC) | bbox_68b4f05f-d5f1-4304-8c69-ad67514
fe393/backups/bboxbackup1_3960c945-8ee0-42fe-945c-c57bb0a3af6f                                   |
| ocp-cluster11 | postgres-app | backup | backup1                    | 2025-02-18 13:31:50 (UTC) | postgres-app_4d798ed5-cfa8-49ff-a5b6
-c5e2d89aeb89/backups/backup1_28e1bd9a-9b04-4412-8b96-811f9b62e2e3                               |
| ocp-cluster11 | postgres-app | backup | daily-feac1-20250222153300 | 2025-02-22 15:34:44 (UTC) | postgres-app_4d798ed5-cfa8-49ff-a5b6
-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) | postgres-app_4d798ed5-cfa8-49ff-a5b6
-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) | postgres-app_4d798ed5-cfa8-49ff-a5b6
-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) | postgres-app_4d798ed5-cfa8-49ff-a5b6
-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) | postgres-app_4d798ed5-cfa8-49ff-a5b6
-c5e2d89aeb89/backups/postgres-backup-cluster1_ec0ed3f3-5500-4e72-afa8-117a04a0b1c3    |
```

使用列表中的备份路径并创建一个备份恢复 CR 对象，如下面的命令所示。

```
# 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 |       |
+--------------------------+--------------------+-----------+--------+-------+
```

现在您可以看到应用程序 pod 和 PVC 已在目标集群中创建。

```
[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
ESCLASS   AGE
data-postgres-postgresql-0   Bound    pvc-872a5182-601b-4848-b410-fef368337d07   8Gi        RWO            sc-zoneb-san   <unset>
          13m
postgres-pvc                 Bound    pvc-caf9fa71-76a8-4645-9bb5-2ed72e72948b   5Gi        RWO            sc-zoneb-san   <unset>
          13m
[root@localhost DataProtection]#
```


使用Trident Protect 保护 Red Hat OpenShift 虚拟化中的虚拟机

使用快照和备份保护 OpenShift 虚拟化中的虚拟机。此过程包括使用ONTAP S3 对象存储创建 AppVault、配置Trident Protect 以捕获 VM 数据（包括 Kubernetes 资源对象、持久卷和内部图像），并在必要时恢复数据。

OpenShift 虚拟化环境中的虚拟机是在 OpenShift 容器平台的工作节点中运行的容器化应用程序。保护虚拟机元数据以及虚拟机的持久磁盘非常重要，以便在它们丢失或损坏时可以恢复它们。

OpenShift 虚拟化虚拟机的持久磁盘可由集成到 OpenShift 集群的ONTAP存储支持，使用"Trident犯罪现场调查"。在本节中我们使用"Trident保护"创建虚拟机（包括其数据卷）的快照和备份到ONTAP对象存储。

然后，我们在需要时从快照或备份中恢复。

Trident Protect 支持 OpenShift 集群上的应用程序和虚拟机的快照、备份、恢复和灾难恢复。对于 OpenShift 虚拟化虚拟机，可以使用Trident Protect 保护的数据包括与虚拟机关联的 Kubernetes 资源对象、持久卷和内部图像。

以下是本节示例使用的各个组件的版本

- "OpenShift 集群 4.17"
- "通过 Red Hat 提供的 OpenShift Virtualization Operator 安装 OpenShift Virtualization"

- "Trident25.02"
- "Trident保护 25.02"
- "ONTAP 9.16"

为对象存储创建 **App Vault**

创建 **AppVault**

在为应用程序或虚拟机创建快照和备份之前，必须在Trident Protect 中配置对象存储来存储快照和备份。这是使用存储桶 CR 完成的。只有管理员可以创建存储桶 CR 并对其进行配置。在Trident Protect 中，存储桶 CR 被称为 AppVault。 AppVault 对象是存储桶的声明性 Kubernetes 工作流表示。 AppVault CR 包含存储桶在保护操作（例如备份、快照、恢复操作和SnapMirror复制）中使用的必要配置。

在此示例中，我们将展示如何使用ONTAP S3 作为对象存储。以下是为ONTAP S3 创建 AppVault CR 的工作流程： 1.在ONTAP集群中的 SVM 中创建 S3 对象存储服务器。2.在对象存储服务器中创建一个存储桶。3.在 SVM 中创建 S3 用户。将访问密钥和密钥保存在安全的地方。4.在 OpenShift 中，创建一个密钥来存储ONTAP S3 凭证。5.为ONTAP S3 创建 AppVault 对象

为**ONTAP S3** 配置**Trident**保护 **AppVault**

```
# 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
```

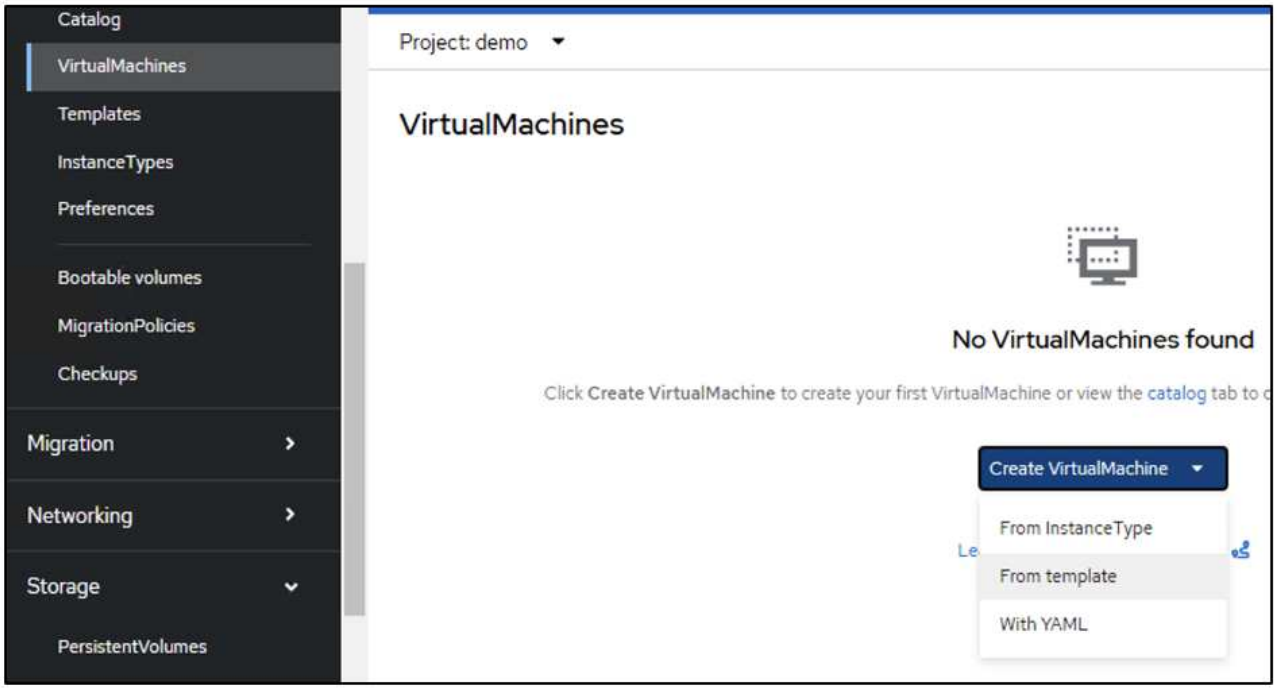
```
[root@localhost VM-DataProtection]# tp get appvault
+-------------------+----------+-----------+-------+---------+-------+
|       NAME        | PROVIDER |   STATE   | ERROR | MESSAGE |  AGE  |
+-------------------+----------+-----------+-------+---------+-------+
| ontap-s3-appvault | OntapS3  |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

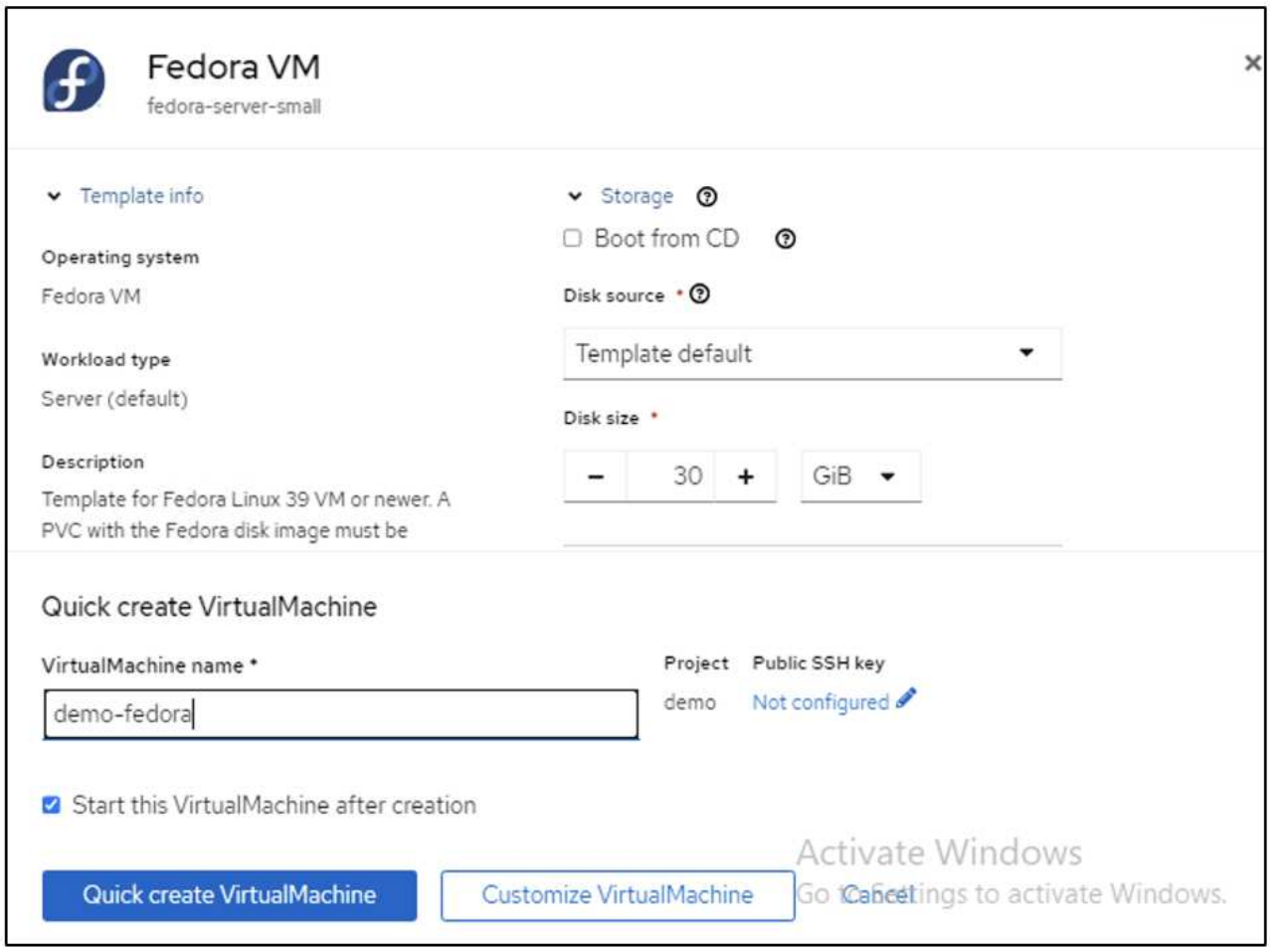
在 OpenShift Virtualization 中创建 VM

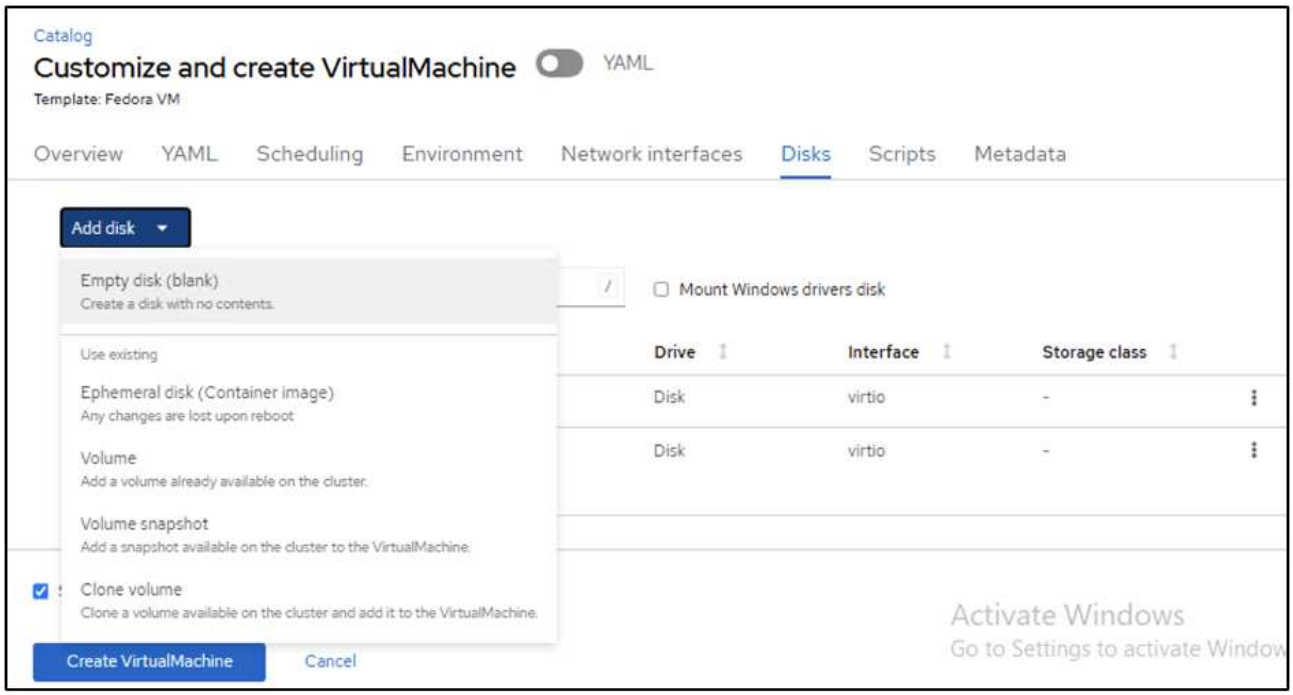
在 **OpenShift Virtualization** 中创建 **VM**

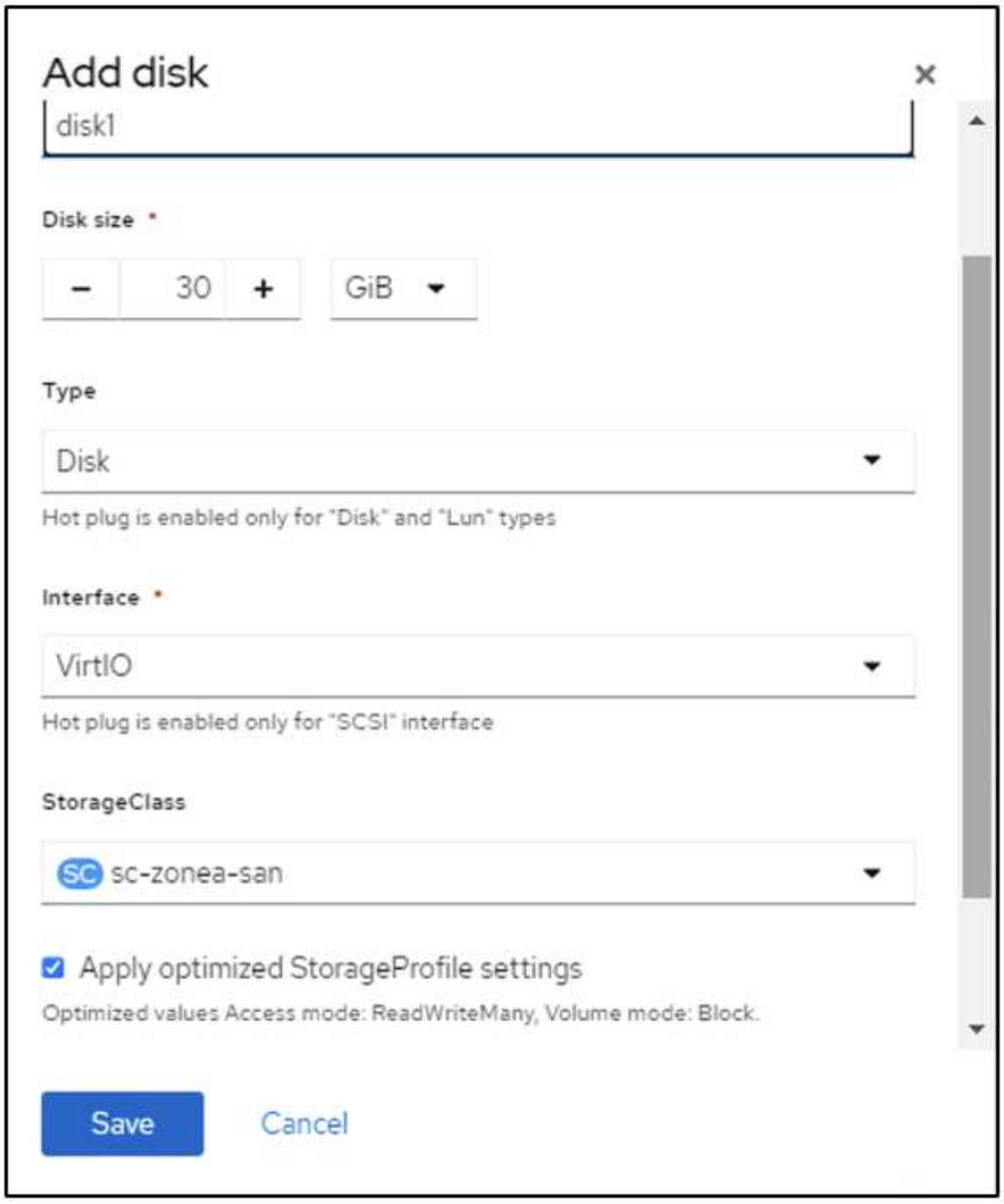
以下屏幕截图显示了使用模板从控制台创建 VM（命名空间 demo 中的 demo-fedora）。根磁盘会自动选择默认存储类别，因此，请验证默认存储类别是否设置得当。在此设置中，默认存储类是**sc-zonea-san**。确保在创建附加磁盘时选择存储类 sc-zonea-san 并选中“应用优化存储设置”复选框。这会将访问模式设置为 RWX，将卷模式设置为阻止。

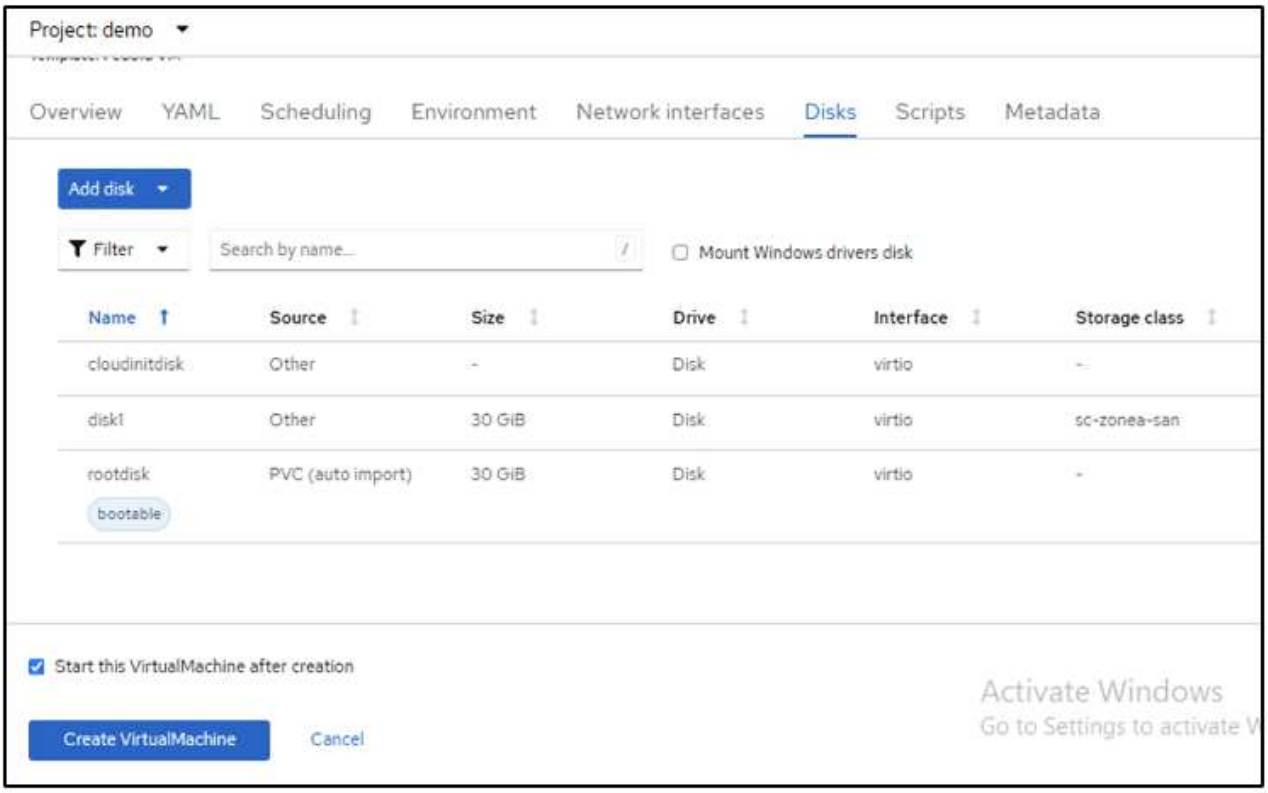
Trident支持 SAN（iSCSI、NVMe/TCP 和 FC）块卷模式下的 RWX 访问模式。 （这是NAS的默认访问模式）。如果您稍后需要对虚拟机进行实时迁移，则需要 RWX 访问模式。




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
Cancel
Activate Windows
Go to Settings to activate Windows.


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.
Mount Windows drivers disk
Drive
Interface
Storage class
Disk
virtio
Disk
virtio
Create VirtualMachine
Cancel
Activate Windows
Go to Settings to activate Windows.


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 sc-zonea-san
Apply optimized StorageProfile settings
Optimized values Access mode: ReadWriteMany, Volume mode: Block.
Save
Cancel




```
[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]#
```

创建应用程序

创建应用程序

为虚拟机创建一个 **Trident** 保护应用程序

在示例中，demo 命名空间有一个 VM，并且在创建应用程序时包含了该命名空间的所有资源。

```
# 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]#
```

通过创建备份来保护应用程序

创建按需备份

为之前创建的应用程序（demo-vm）创建一个备份，其中包括 demo 命名空间中的所有资源。提供将存储备份的 appvault 名称。

```
# 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]#
```

按计划创建备份

创建备份计划，指定要保留的备份粒度和数量。

```
# 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]#
```

从备份还原

从备份还原

将虚拟机恢复到相同的命名空间

在示例中，备份 demo-vm-backup-on-demand 包含 fedora VM 的 demo-app 备份。

首先，删除虚拟机，并确保 PVC、pod 和虚拟机对象从命名空间“demo”中删除

```
[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]#
```

现在，创建一个就地备份恢复对象。

```
# 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]#
```

验证虚拟机、Pod 和 PVC 是否已恢复

```
[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]#
```

将虚拟机还原到不同的命名空间

首先创建一个要将应用程序恢复到的新命名空间，在本例中为 demo2。然后创建备份恢复对象

```
# 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 |
+----------------------+-------------------+-----------+-------+--------+
```

验证是否在新命名空间 demo2 中创建了 VM、pod 和 pvc。

```
[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]#
```

使用快照保护应用程序

创建快照

创建按需快照 为应用程序创建快照并指定需要存储它的 appvault。

```
# 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       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]#
```

为快照创建计划 为快照创建计划。指定要保留的粒度和快照数量。

```
# 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]#
```

从快照还原

从快照还原

将虚拟机从快照恢复到同一个命名空间 从 demo2 命名空间中删除虚拟机 demo-fedora。

```
[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]#
```

从虚拟机的快照创建快照就地恢复对象。

```
# 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]#
```

验证虚拟机及其 PVC 是否在 demo 命名空间中创建。

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

将虚拟机从快照还原到不同的命名空间

删除先前从备份中恢复的 demo2 命名空间中的虚拟机。

```
[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                                                  STATUS   VOLUME                                     CAPACITY   ACCESS MODES
  STORAGECLASS   VOLUMEATTRIBUTESCLASS   AGE
persistentvolumeclaim/demo-centos                     Bound    pvc-7aafd5e2-d4cf-4af6-a259-c68e016ec6cd   30Gi       RWX
  sc-zonea-san   <unset>                 3m26s
persistentvolumeclaim/demo-fedora                     Bound    pvc-c1f8145a-56ea-42c5-abb4-6457b6853e1c   30Gi       RWX
  sc-zonea-san   <unset>                 3m25s
persistentvolumeclaim/dv-demo-centos-lavender-tortoise-34   Bound    pvc-e82b6d98-6762-4473-9bb8-7e98efcb7987   30Gi       RWX
  sc-zonea-san   <unset>                 3m20s
persistentvolumeclaim/dv-demo-fedora-fuchsia-shrew-87   Bound    pvc-ee13cfbc-2921-4129-a214-a0f6b10335cd   30Gi       RWX
  sc-zonea-san   <unset>                 3m16s
[root@localhost VM-DataProtection]# oc delete virtualmachine.kubevirt.io/demo-fedora -n demo2
virtualmachine.kubevirt.io "demo-fedora" deleted
```

从快照创建快照恢复对象并提供命名空间映射。

```
# 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 |
+-----------------------------------+-------------------+-----------+-------+--------+
```

验证虚拟机及其 PVC 是否在新的命名空间 demo2 中恢复。

```
[root@localhost RedHat]# oc get vm,pvc -n demo2
NAME                                  AGE   STATUS    READY
virtualmachine.kubevirt.io/demo-fedora   29h   Running   True

NAME                                                    STATUS   VOLUME
ORAGECLASS   VOLUMEATTRIBUTESCLASS   AGE
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]#
```

恢复特定虚拟机

选择命名空间中的特定虚拟机来创建快照/备份和恢复

在前面的例子中，我们在命名空间内有一个虚拟机。通过将整个命名空间包含在备份中，可以捕获与该虚拟机相关的所有资源。在下面的示例中，我们将另一个 VM 添加到同一个命名空间，并使用标签选择器为这个新 VM 创建一个应用程序。

在 **demo** 命名空间中创建一个新的虚拟机（**demo-centos vm**）

```
[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]#
```

标记 **demo-centos vm** 及其相关资源

```
[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]#
```

验证 **demo-centos vm** 和 **pvcs** 是否已获得标签

```
[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]#
[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=u1.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=u1.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]#
```

使用标签选择器仅为特定虚拟机（**demo-centos**）创建应用程序

```
# 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 |
+-----------------+------------+-------+------+
```

按需和按计划创建备份和快照的方法与前面所示的相同。由于用于创建快照或备份的 trident-protect 应用程序仅包含来自命名空间的特定 VM，因此从它们恢复只会恢复特定的 VM。下面显示了一个备份/恢复操作的示例。

使用其对应的应用程序创建命名空间中特定虚拟机的备份

在前面的步骤中，使用标签选择器创建了一个应用程序，以仅包含演示命名空间中的 centos vm。为此应用程序创建备份（在本例中为按需备份）。

```
# 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 |
+------------------------------+-----------------+----------------+-----------+-------+--------+
```

将特定的 **VM** 恢复到同一个命名空间 使用相应的应用程序创建了特定的 VM（centos）的备份。如果从此创建了备份就地恢复或备份恢复，则仅恢复此特定的 VM。删除 Centos VM。

```
[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
  STORAGECLASS   VOLUMEATTRIBUTESCLASS   AGE
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]#
```

从 demo-centos-backup-on-demand 创建备份就地恢复并验证 centos VM 是否已重新创建。

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

将特定的 **VM** 恢复到不同的命名空间 从 demo-centos-backup-on-demand 创建到不同命名空间 (demo3) 的备份恢复，并验证 centos VM 是否已重新创建。

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

视频演示

以下视频演示了如何使用快照保护虚拟机

保护虚拟机