Cloud ibrido con componenti gestiti dal provider

NetApp public and hybrid cloud solutions

NetApp
August 18, 2025

This PDF was generated from https://docs.netapp.com/it-it/netapp-solutions-cloud/openshift/os-pm-solution.html on August 18, 2025. Always check docs.netapp.com for the latest.

Sommario

Cloud ibrido con componenti gestiti dal provider . . . 1
- Soluzione NetApp con carichi di lavoro della piattaforma container Red Hat OpenShift gestiti . . . 1
- Distribuisci e configura la piattaforma Managed Red Hat OpenShift Container su AWS . . . 1
- Distribuisci e configura OpenShift Dedicated su Google Cloud con Google Cloud NetApp Volumes . . . 4
- Protezione dei dati . . . 6
  - Backup/Ripristino da backup . . . 7
  - Snapshot/Ripristina da snapshot . . . 7
  - Blog . . . 7
  - Dettagli passo passo per creare uno snapshot e ripristinarlo . . . 7
- Migrazione dei dati . . . 22
  - Migrazione dei dati . . . 23
- Ulteriori soluzioni NetApp Hybrid Multicloud per carichi di lavoro Red Hat OpenShift . . . 24
  - Soluzioni aggiuntive . . . 24

Cloud ibrido con componenti gestiti dal provider

Soluzione NetApp con carichi di lavoro della piattaforma container Red Hat OpenShift gestiti

I clienti potrebbero essere "nati nel cloud" oppure trovarsi in una fase del loro percorso di modernizzazione in cui sono pronti a spostare alcuni carichi di lavoro selezionati o tutti i carichi di lavoro dai loro data center al cloud. Possono scegliere di utilizzare container OpenShift gestiti dal provider e storage NetApp gestito dal provider nel cloud per eseguire i propri carichi di lavoro. Dovrebbero pianificare e distribuire i cluster di container Managed Red Hat OpenShift nel cloud per creare un ambiente di produzione di successo per i loro carichi di lavoro di container. NetApp fornisce soluzioni di storage completamente gestite per le soluzioni Managed Red Hat in tutti e tre i principali cloud pubblici.

- Amazon FSx for NetApp ONTAP (FSx ONTAP)*

FSx ONTAP garantisce protezione dei dati, affidabilità e flessibilità per le distribuzioni di container in AWS. Trident funge da fornitore di storage dinamico per utilizzare lo storage FSx ONTAP persistente per le applicazioni con stato dei clienti.

Poiché ROSA può essere distribuito in modalità HA con nodi del piano di controllo distribuiti su più zone di disponibilità, FSx ONTAP può anche essere fornito con l'opzione Multi-AZ che fornisce elevata disponibilità e protezione contro i guasti AZ.

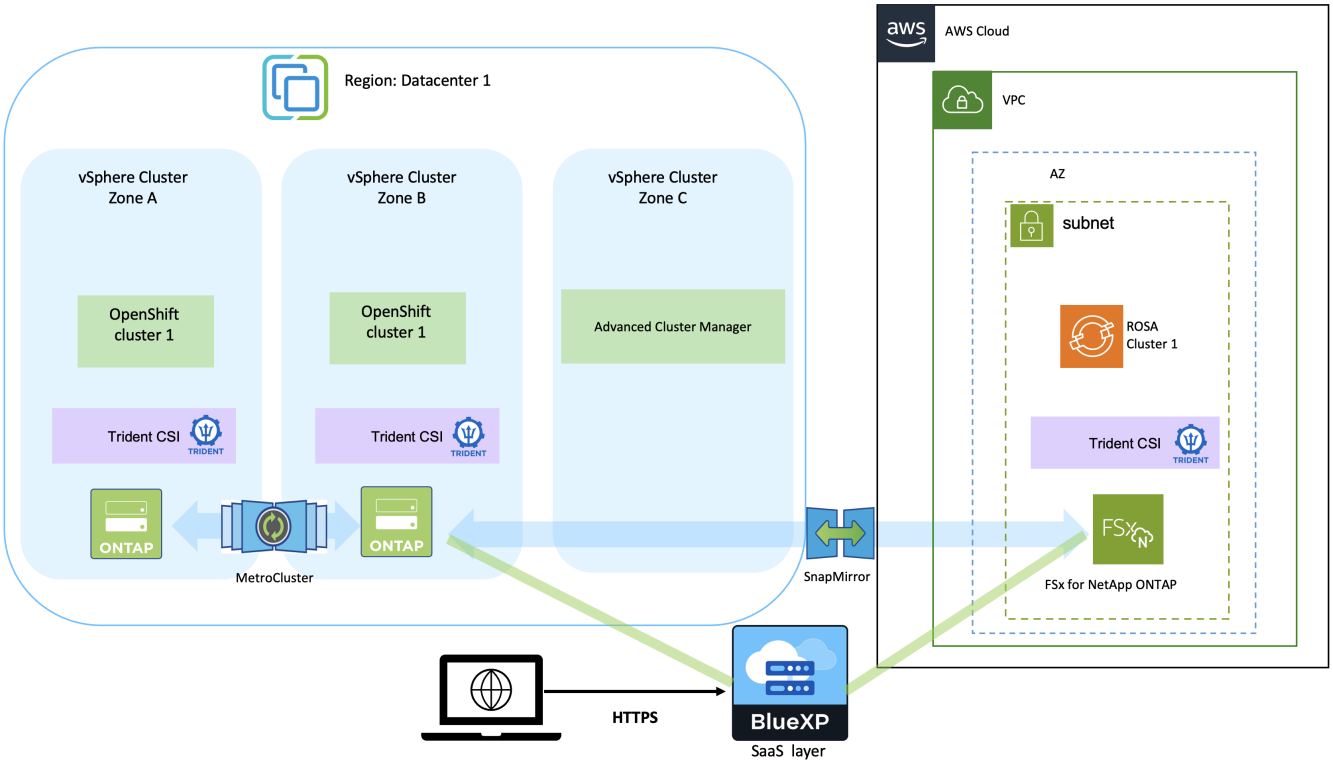
- Google Cloud NetApp Volumes*

Red Hat OpenShift Dedicated è una piattaforma applicativa completamente gestita che consente di creare, distribuire e scalare rapidamente le applicazioni nel cloud ibrido. Google Cloud NetApp Volumes fornisce volumi persistenti, portando la suite completa delle funzionalità di gestione dei dati aziendali di ONTAP alle distribuzioni OpenShift in Google Cloud.

Distribuisci e configura la piattaforma Managed Red Hat OpenShift Container su AWS

Questa sezione descrive un flusso di lavoro di alto livello per la configurazione dei cluster Managed Red Hat OpenShift su AWS (ROSA). Mostra l'utilizzo di Amazon FSx for NetApp ONTAP (FSx ONTAP) gestito come backend di storage da parte di Trident per fornire volumi persistenti. Vengono forniti dettagli sulla distribuzione di FSx ONTAP su AWS utilizzando BlueXP. Vengono inoltre forniti dettagli sull'utilizzo di BlueXP e OpenShift GitOps (Argo CD) per eseguire attività di protezione e migrazione dei dati per le applicazioni con stato sui cluster ROSA.

Ecco un diagramma che illustra i cluster ROSA distribuiti su AWS e che utilizzano FSx ONTAP come storage back-end.



Questa soluzione è stata verificata utilizzando due cluster ROSA in due VPC in AWS. Ogni cluster ROSA è stato integrato con FSx ONTAP utilizzando Trident. Esistono diversi modi per distribuire cluster ROSA e FSx ONTAP in AWS. Questa descrizione di alto livello della configurazione fornisce link alla documentazione per il metodo specifico utilizzato. È possibile fare riferimento agli altri metodi nei link pertinenti forniti nel"sezione risorse" .

Il processo di configurazione può essere suddiviso nei seguenti passaggi:

Installare i cluster ROSA

- Crea due VPC e configura la connettività peering VPC tra le VPC.
- Fare riferimento"Qui" per istruzioni su come installare i cluster ROSA.

Installa FSx ONTAP

- Installare FSx ONTAP sulle VPC da BlueXP. Fare riferimento"Qui" per creare un account BlueXP e iniziare. Fare riferimento"Qui" per installare FSx ONTAP. Fare riferimento"Qui" per creare un connettore in AWS per gestire FSx ONTAP.
- Distribuisci FSx ONTAP utilizzando AWS. Fare riferimento"Qui" per la distribuzione tramite la console AWS.

Installa Trident sui cluster ROSA (utilizzando il grafico Helm)

- Utilizzare il grafico Helm per installare Trident sui cluster ROSA. Fare riferimento al collegamento alla documentazione: https://docs.netapp.com/us-en/trident/trident-get-started/kubernetes-deploy-helm.html [qui].

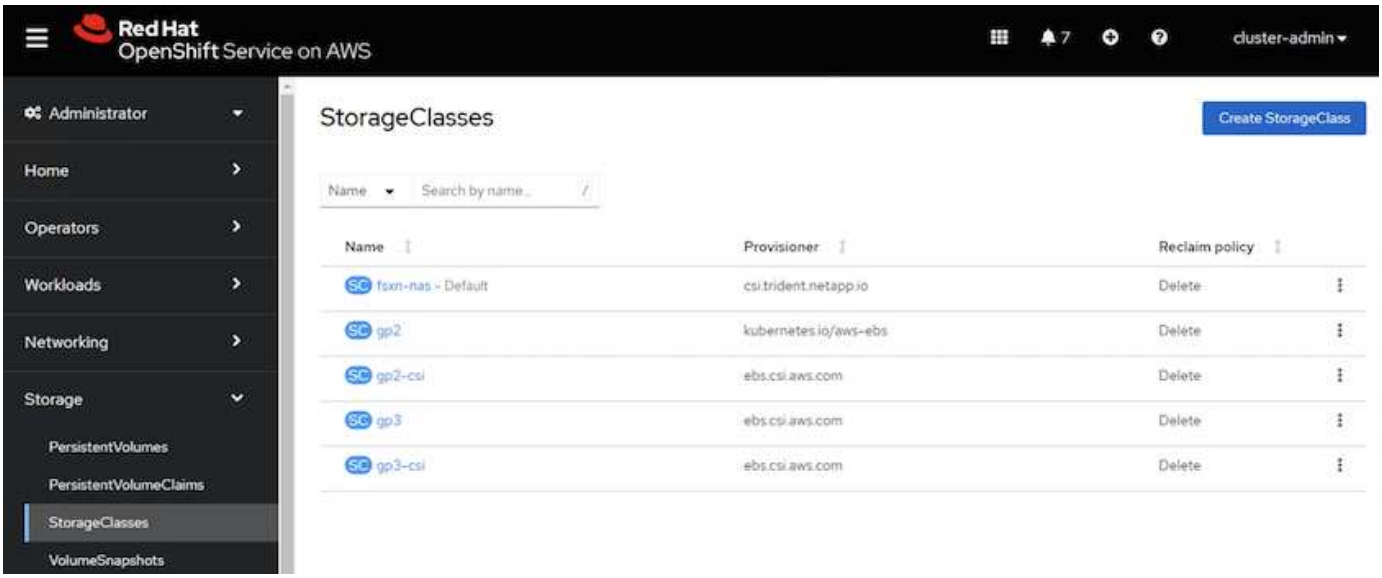
Integrazione di FSx ONTAP con Trident per cluster ROSA

OpenShift GitOps può essere utilizzato per distribuire Trident CSI su tutti i cluster gestiti non appena vengono registrati su ArgoCD tramite ApplicationSet.

```yaml
apiVersion: argoproj.io/v1alpha1
kind: ApplicationSet
metadata:
  name: trident-operator
spec:
  generators:
  - clusters: {}
    # selector:
    #   matchLabels:
    #     tridentversion: '23.04.0'
  template:
    metadata:
      name: '{{nameNormalized}}-trident'
    spec:
      destination:
        namespace: trident
        server: '{{server}}'
      source:
        repoURL: 'https://netapp.github.io/trident-helm-chart'
        targetRevision: 23.04.0
        chart: trident-operator
      project: default
      syncPolicy:
        syncOptions:
          - CreateNamespace=true
```

Creare classi di backend e storage utilizzando Trident (per FSx ONTAP)

- Fare riferimento"Qui" per i dettagli sulla creazione di classi di backend e storage.
- Imposta come predefinita la classe di archiviazione creata per FsxN con Trident CSI dalla console OpenShift. Vedi screenshot qui sotto:



Distribuisci un'applicazione utilizzando OpenShift GitOps (Argo CD)

- Installare l'operatore OpenShift GitOps sul cluster. Fare riferimento alle istruzioni"Qui" .
- Configurare una nuova istanza Argo CD per il cluster. Fare riferimento alle istruzioni"Qui" .

Aprire la console di Argo CD e distribuire un'app. Ad esempio, è possibile distribuire un'app Jenkins utilizzando Argo CD con un grafico Helm. Durante la creazione dell'applicazione sono stati forniti i seguenti dettagli:

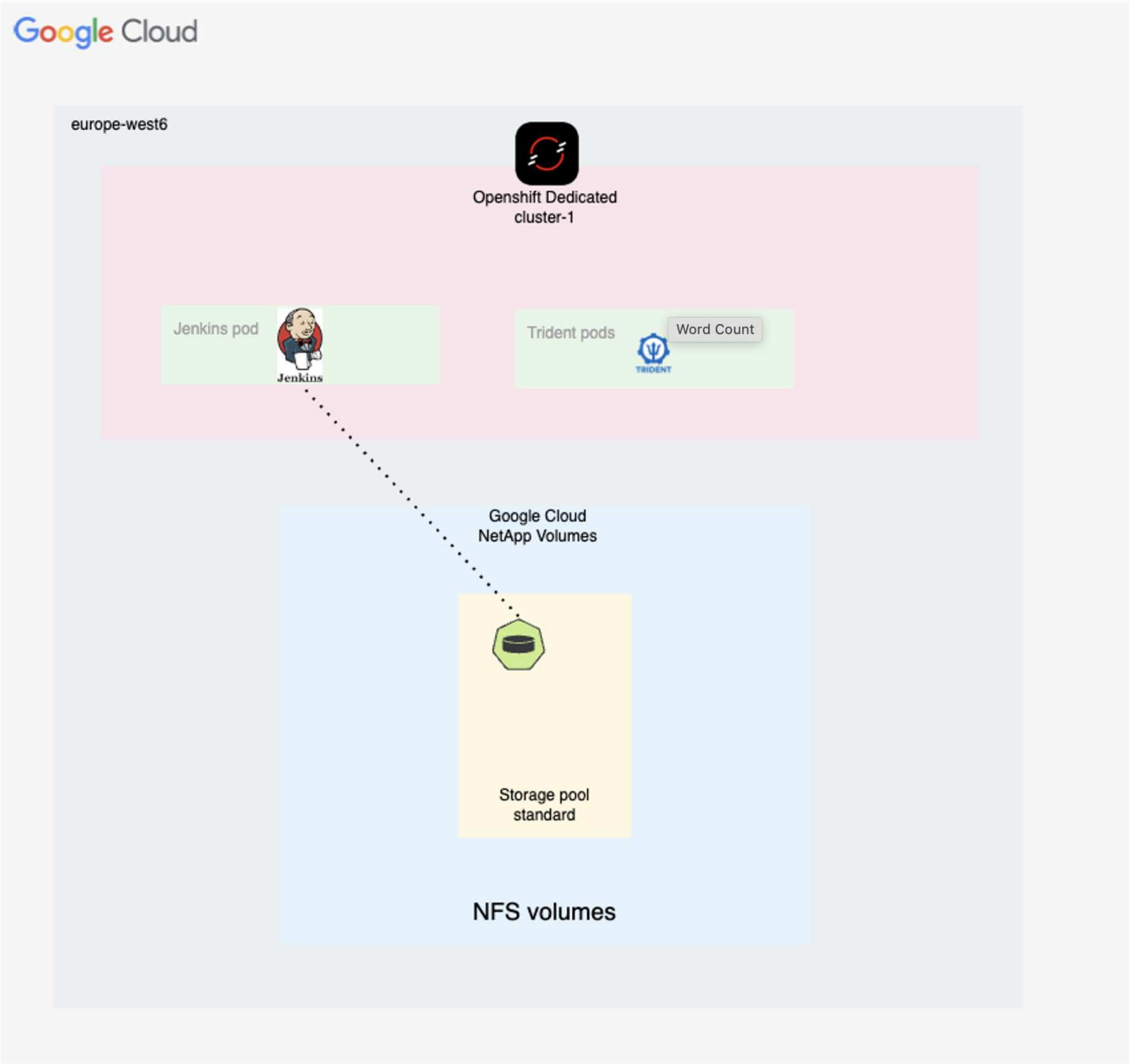
Progetto: cluster predefinito:'https://kubernetes.default.svc' (senza virgolette) Namespace: Jenkins L'URL per il grafico Helm:'https://charts.bitnami.com/bitnami' (senza virgolette)

Parametri Helm: global.storageClass: fsxn-nas

Distribuisci e configura OpenShift Dedicated su Google Cloud con Google Cloud NetApp Volumes

Questa sezione descrive un flusso di lavoro di alto livello per la configurazione di cluster OpenShift Dedicated (OSD) sulla piattaforma Google Cloud. Mostra NetApp Trident che utilizza Google Cloud NetApp Volumes come backend di archiviazione per fornire volumi persistenti per applicazioni con stato eseguite con Kubernetes.

Ecco un diagramma che illustra un cluster OSD distribuito su Google Cloud e che utilizza NetApp Volumes come storage back-end.



Il processo di configurazione può essere suddiviso nei seguenti passaggi:

Installa i cluster OSD in Google Cloud

- Se si desidera utilizzare una VPC esistente per il cluster, è necessario creare la VPC, due subnet, un router cloud e due NAT cloud GCP per il cluster OSD. Fare riferimento"Qui" per istruzioni.
- Fare riferimento"Qui" per istruzioni su come installare i cluster OSD su GCP utilizzando il modello di fatturazione Customer Cloud Subscription (CCS). OSD è incluso anche su Google Cloud Marketplace. È disponibile un video che mostra come installare OSD utilizzando la soluzione Google Cloud Marketplace"Qui" .

Abilita Google Cloud NetApp Volumes

- Fare riferimento"Qui" per informazioni sulla configurazione dell'accesso a Google Cloud NetApp Volumes. Segui tutti i passaggi fino a e incluso
- Creare un pool di archiviazione. Fare riferimento"Qui" per informazioni su come configurare un pool di archiviazione su Google Cloud NetApp Volumes. I volumi per le applicazioni Kubernetes con stato in

esecuzione su OSD verranno creati all'interno del pool di archiviazione.

Installa Trident sui cluster OSD (utilizzando la tabella Helm)

- Utilizzare una tabella Helm per installare Trident sui cluster OSD. Fare riferimento"Qui" per istruzioni su come installare Helm Chart. La tabella del timone può essere trovata"Qui" .

Integrazione di NetApp Volumes con NetApp Trident per cluster OSD

Crea classi di backend e storage utilizzando Trident (per Google Cloud NetApp Volumes)

- Per i dettagli sulla creazione del backend, fare riferimento qui.
- Se una qualsiasi delle classi di archiviazione correnti in Kubernetes è contrassegnata come predefinita, rimuovere l'annotazione modificando la classe di archiviazione.
- Creare almeno una classe di archiviazione per i volumi NetApp con il provisioner Trident CSI. Rendi predefinita solo una delle classi di archiviazione utilizzando un'annotazione. Ciò consentirà a un PVC di utilizzare questa classe di archiviazione quando non è esplicitamente indicata nel manifesto del PVC. Di seguito è riportato un esempio con l'annotazione.

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: gcnv-standard-k8s
  annotations:
    storageclass.kubernetes.io/is-default-class: "true"
provisioner: csi.trident.netapp.io
parameters:
  backendType: "google-cloud-netapp-volumes"
  trident.netapp.io/nasType: "nfs"
allowVolumeExpansion: true
```

Distribuisci un'applicazione utilizzando OpenShift GitOps (Argo CD)

- Installare l'operatore OpenShift GitOps sul cluster. Fare riferimento alle istruzioni"Qui" .
- Configurare una nuova istanza Argo CD per il cluster. Fare riferimento alle istruzioni"Qui" .

Aprire la console di Argo CD e distribuire un'app. Ad esempio, è possibile distribuire un'app Jenkins utilizzando Argo CD con un grafico Helm. Durante la creazione dell'applicazione sono stati forniti i seguenti dettagli: Progetto: cluster predefinito:'https://kubernetes.default.svc' (senza virgolette) Namespace: Jenkins L'URL per il grafico Helm:'https://charts.bitnami.com/bitnami' (senza virgolette)

Protezione dei dati

Questa pagina mostra le opzioni di protezione dei dati per i cluster Managed Red Hat OpenShift on AWS (ROSA) che utilizzano Astra Control Service. Astra Control Service (ACS) fornisce un'interfaccia utente grafica di facile utilizzo con cui è possibile aggiungere cluster, definire le applicazioni in esecuzione su di essi ed eseguire attività di gestione dei dati basate sulle applicazioni. È possibile accedere alle funzioni ACS anche tramite un'API che consente l'automazione dei flussi di lavoro.

Astra Control (ACS o ACC) è alimentato da NetApp Trident. Trident integra diversi tipi di cluster Kubernetes, come Red Hat OpenShift, EKS, AKS, SUSE Rancher, Anthos ecc., con vari tipi di storage NetApp ONTAP , come FAS/ AFF, ONTAP Select, CVO, Google Google Cloud NetApp Volumes, Azure NetApp Files e Amazon FSx ONTAP.

Questa sezione fornisce dettagli sulle seguenti opzioni di protezione dei dati tramite ACS:

- Un video che mostra il backup e il ripristino di un'applicazione ROSA in esecuzione in una regione e il ripristino in un'altra regione.
- Un video che mostra lo snapshot e il ripristino di un'applicazione ROSA.
- Dettagli passo passo sull'installazione di un cluster ROSA, Amazon FSx ONTAP, utilizzando NetApp Trident per l'integrazione con il backend di storage, installando un'applicazione PostgreSQL sul cluster ROSA, utilizzando ACS per creare uno snapshot dell'applicazione e ripristinandola da esso.
- Un blog che illustra dettagliatamente la creazione e il ripristino da uno snapshot per un'applicazione MySQL su un cluster ROSA con FSx ONTAP utilizzando ACS.

Backup/Ripristino da backup

Il video seguente mostra il backup di un'applicazione ROSA in esecuzione in una regione e il ripristino in un'altra regione.

FSx NetApp ONTAP per Red Hat OpenShift Service su AWS

Snapshot/Ripristina da snapshot

Il video seguente mostra come acquisire uno snapshot di un'applicazione ROSA e come ripristinarlo successivamente.

Snapshot/ripristino per applicazioni su cluster Red Hat OpenShift Service on AWS (ROSA) con storage Amazon FSx ONTAP

Blog

- "Utilizzo di Astra Control Service per la gestione dei dati delle app sui cluster ROSA con storage Amazon FSx"

Dettagli passo passo per creare uno snapshot e ripristinarlo

Configurazione prerequisito

- "Account AWS"
- "Account Red Hat OpenShift"
- Utente IAM con"permessi appropriati" per creare e accedere al cluster ROSA
- "Interfaccia a riga di comando AWS"
- "ROSA CLI"
- "Interfaccia a riga di comando di OpenShift"(oc)
- VPC con subnet e gateway e percorsi appropriati
- "ROSA Cluster installato"nel VPC

- "Amazon FSx ONTAP"creato nella stessa VPC
- Accesso al cluster ROSA da"Console cloud ibrida OpenShift"

Prossimi passi

1. Crea un utente amministratore ed effettua l'accesso al cluster.
2. Creare un file kubeconfig per il cluster.
3. Installare Trident sul cluster.
4. Creare una configurazione di backend, classe di archiviazione e classe di snapshot utilizzando il provisioner Trident CSI.
5. Distribuire un'applicazione PostgreSQL sul cluster.
6. Crea un database e aggiungi un record.
7. Aggiungere il cluster in ACS.
8. Definire l'applicazione in ACS.
9. Crea uno snapshot utilizzando ACS.
10. Eliminare il database nell'applicazione PostgreSQL.
11. Ripristina da uno snapshot tramite ACS.
12. Verifica che l'app sia stata ripristinata dallo snapshot.

1. Crea un utente amministratore e accedi al cluster

Accedi al cluster ROSA creando un utente amministratore con il seguente comando: (È necessario creare un utente amministratore solo se non ne hai creato uno al momento dell'installazione)

```
rosa create admin --cluster=<cluster-name>
```

Il comando fornirà un output simile al seguente. Accedi al cluster utilizzando `oc login` comando fornito nell'output.

```
W: It is recommended to add an identity provider to login to this cluster.
See 'rosa create idp --help' for more information.
I: Admin account has been added to cluster 'my-rosa-cluster'. It may take up
to a minute for the account to become active.
I: To login, run the following command:
oc login https://api.my-rosa-cluster.abcd.p1.openshiftapps.com:6443 \
--username cluster-admin \
--password FWGYL-2mkJI-00000-00000
```

> È anche possibile accedere al cluster tramite un token. Se al momento della creazione del cluster è già stato creato un utente amministratore, è possibile accedere al cluster dalla console Red Hat OpenShift Hybrid Cloud con le credenziali dell'utente amministratore. Quindi cliccando nell'angolo in alto a destra dove è visualizzato il nome dell'utente registrato, è possibile ottenere il `oc login` comando (accesso token) per la riga di comando.

2. Crea un file kubeconfig per il cluster

Seguire le procedure"Qui" per creare un file kubeconfig per il cluster ROSA. Questo file kubeconfig verrà utilizzato in seguito quando aggiungerai il cluster ad ACS.

3. Installa Trident sul cluster

Installare Trident (ultima versione) sul cluster ROSA. Per fare ciò, puoi seguire una qualsiasi delle procedure fornite"Qui" . Per installare Trident tramite helm dalla console del cluster, creare prima un progetto denominato Trident.

Quindi, dalla vista Sviluppatore, crea un repository di grafici Helm. Per il campo URL utilizzare `'https://netapp.github.io/trident-helm-chart'` . Quindi creare una versione del timone per l'operatore Trident .

Project: trident

Create Helm Chart Repository

Add helm chart repository.

Configure via: (●) Form view ( ) YAML view

Scope type

( ) Namespaced scoped (ProjectHelmChartRepository)

Add Helm Chart Repository in the selected namespace.

(●) Cluster scoped (HelmChartRepository)

Add Helm Chart Repository at the cluster level and in all namespaces.

Name *

trident

A unique name for the Helm Chart repository.

Display name

Astra Trident

A display name for the Helm Chart repository.

Description

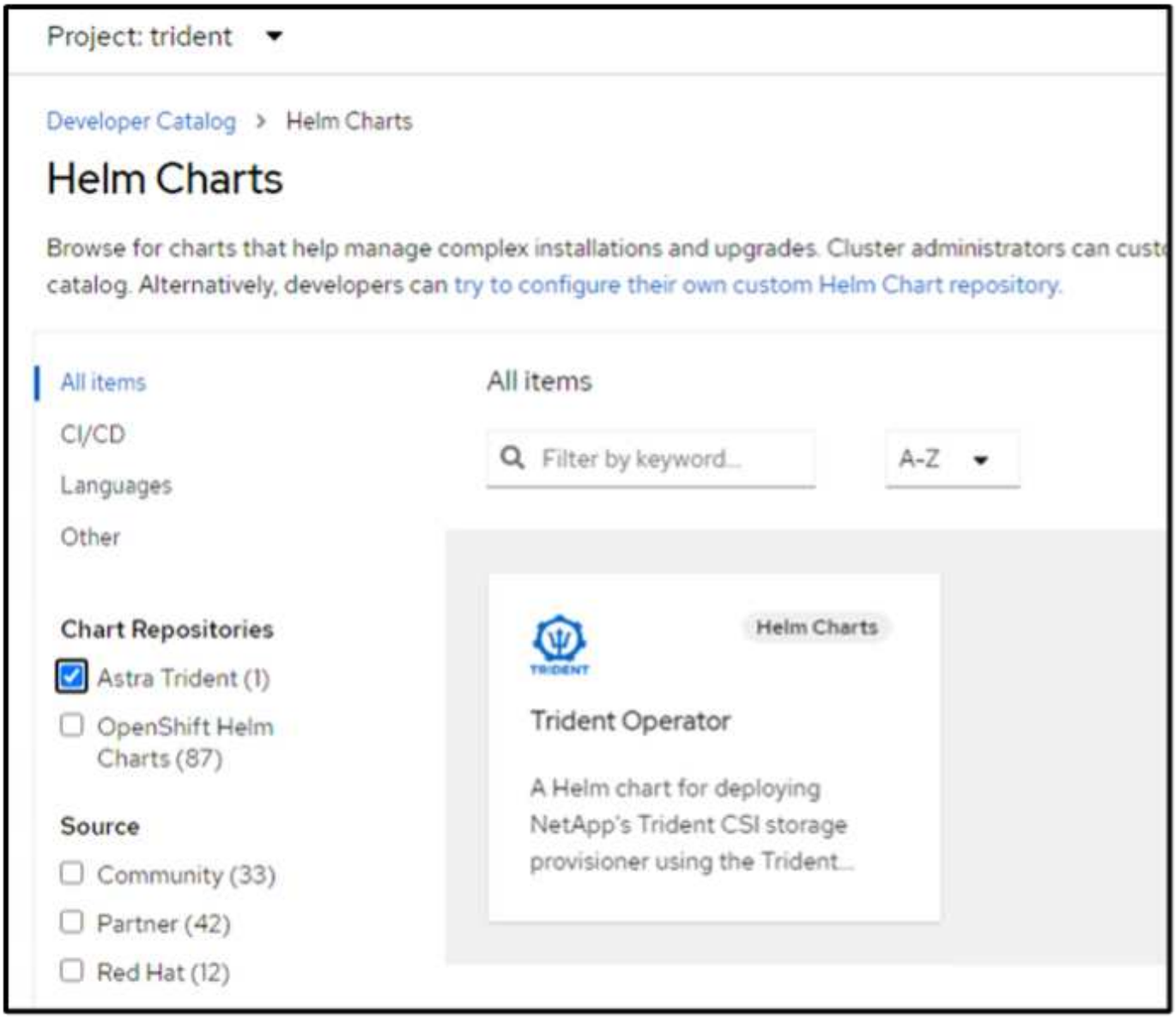
NetApp Astra Trident

A description for the Helm Chart repository.

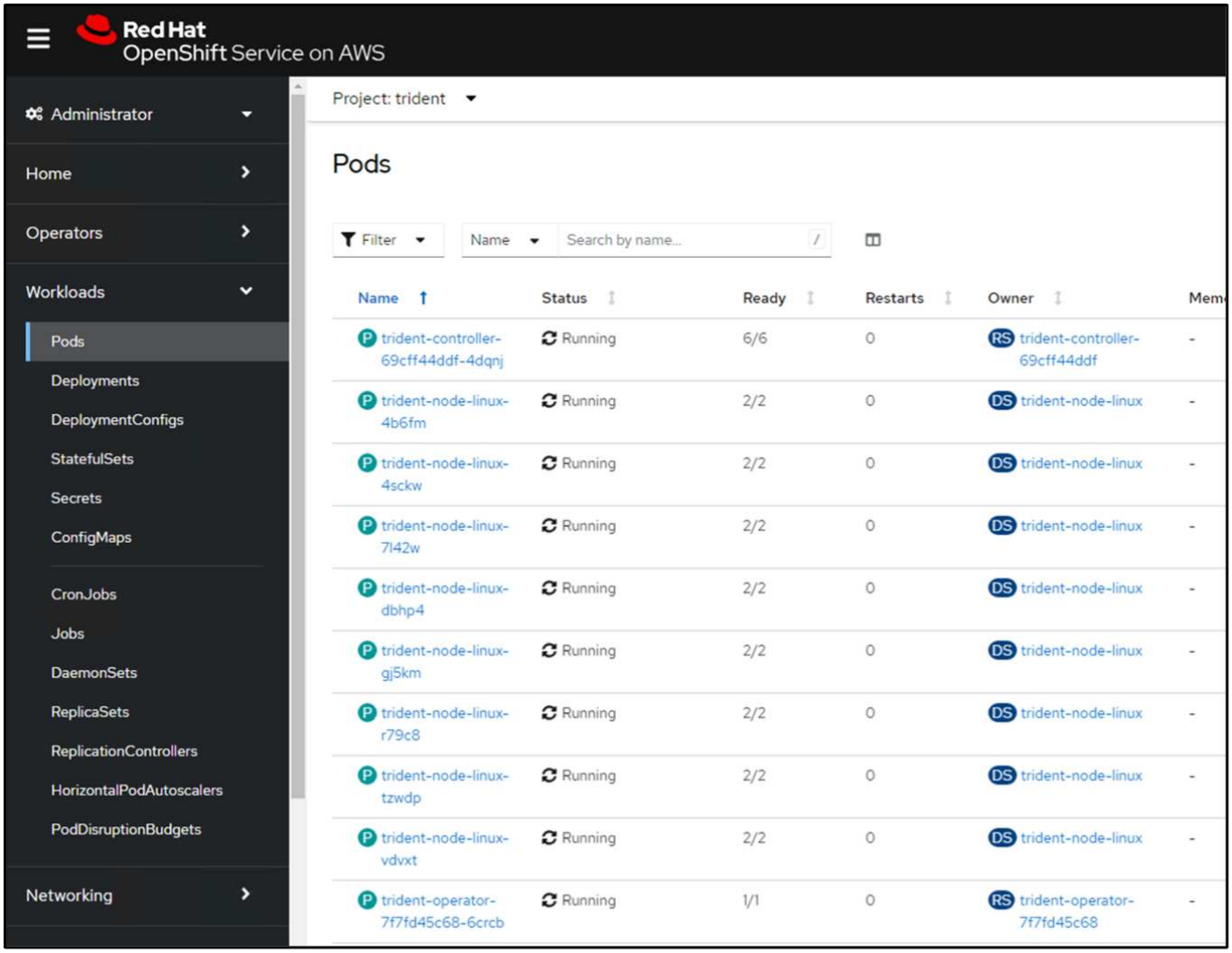
[ ] Disable usage of the repo in the developer catalog.

URL *

https://netapp.github.io/trident-helm-chart



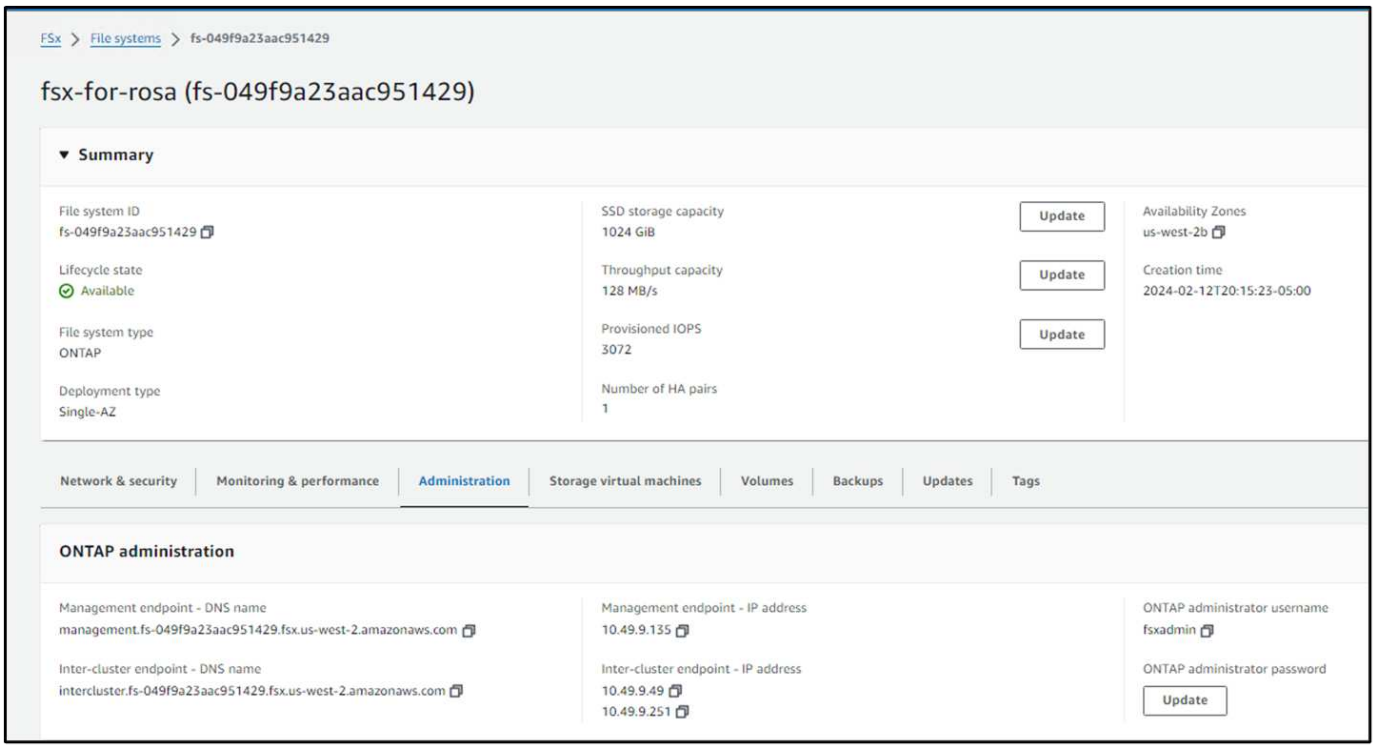
Verificare che tutti i pod Trident siano in esecuzione tornando alla vista Amministratore sulla console e selezionando i pod nel progetto Trident.



4. Crea una configurazione di backend, classe di archiviazione e classe di snapshot utilizzando il provisioner Trident CSI

Utilizzare i file yaml mostrati di seguito per creare un oggetto backend trident, un oggetto classe di archiviazione e l'oggetto Volumesnapshot. Assicurati di fornire le credenziali del file system Amazon FSx ONTAP che hai creato, il LIF di gestione e il nome del vserver del tuo file system nel file yaml di configurazione per il backend. Per ottenere questi dettagli, accedi alla console AWS per Amazon FSx e seleziona il file system, quindi vai alla scheda Amministrazione. Inoltre, fare clic su Aggiorna per impostare la password per `fsxadmin` utente.

> È possibile utilizzare la riga di comando per creare gli oggetti oppure crearli con i file yaml dalla console del cloud ibrido.



Configurazione backend Trident

```
apiVersion: v1
kind: Secret
metadata:
  name: backend-tbc-ontap-nas-secret
type: Opaque
stringData:
  username: fsxadmin
  password: <password>
---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: ontap-nas
spec:
  version: 1
  storageDriverName: ontap-nas
  managementLIF: <management lif>
  backendName: ontap-nas
  svm: fsx
  credentials:
    name: backend-tbc-ontap-nas-secret
```

Classe di archiviazione

```yaml
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: ontap-nas
provisioner: csi.trident.netapp.io
parameters:
  backendType: "ontap-nas"
  media: "ssd"
  provisioningType: "thin"
  snapshots: "true"
allowVolumeExpansion: true
```

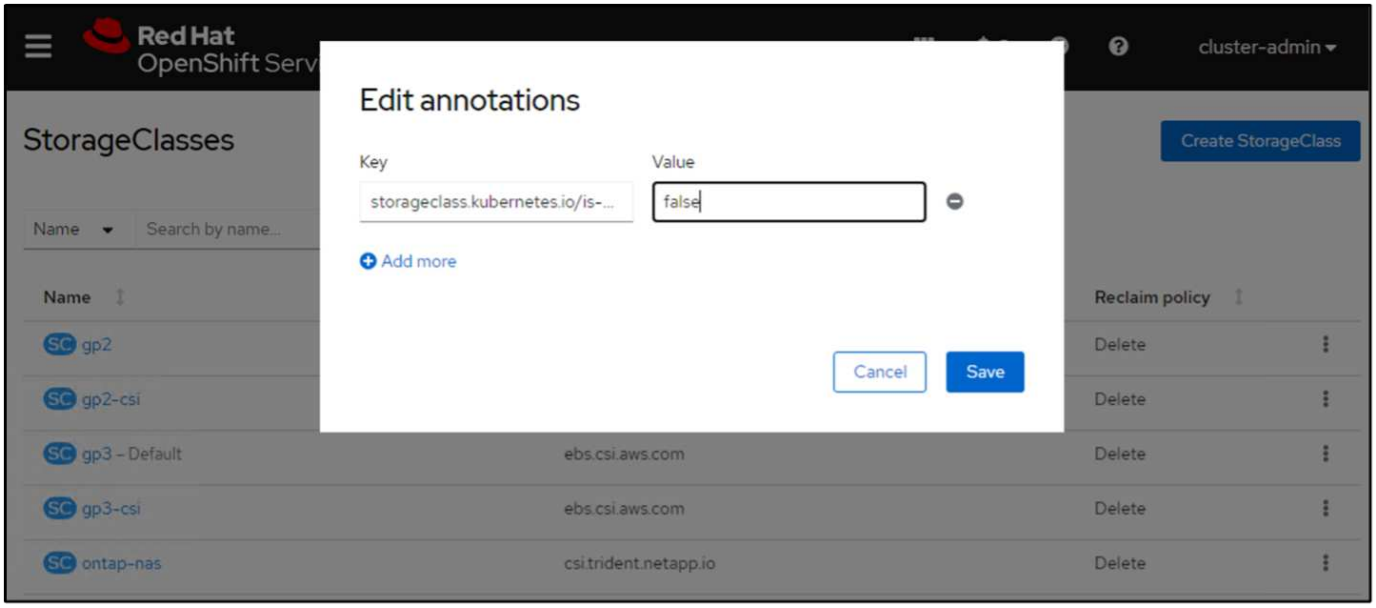
classe snapshot

```yaml
apiVersion: snapshot.storage.k8s.io/v1
kind: VolumeSnapshotClass
metadata:
  name: trident-snapshotclass
driver: csi.trident.netapp.io
deletionPolicy: Delete
```

Verificare che gli oggetti backend, storage class e trident-snapshotclass siano stati creati emettendo i comandi mostrati di seguito.

```
[ec2-user@ip-10-49-11-132 storage]$ kubectl get tbc -n trident
NAME        BACKEND NAME   BACKEND UUID                           PHASE   STATUS
ontap-nas   ontap-nas      8a5e4583-2dac-46bb-b01e-fa7c3816f121   Bound   Success
[ec2-user@ip-10-49-11-132 storage]$ kubectl get sc
NAME            PROVISIONER             RECLAIMPOLICY   VOLUMEBINDINGMODE      ALLOWVOLUMEEXPANSION   AGE
gp2             kubernetes.io/aws-ebs   Delete          WaitForFirstConsumer   true                   3h23m
gp2-csi         ebs.csi.aws.com         Delete          WaitForFirstConsumer   true                   3h19m
gp3 (default)   ebs.csi.aws.com         Delete          WaitForFirstConsumer   true                   3h23m
gp3-csi         ebs.csi.aws.com         Delete          WaitForFirstConsumer   true                   3h19m
ontap-nas       csi.trident.netapp.io   Delete          Immediate              true                   141m
[ec2-user@ip-10-49-11-132 storage]$ kubectl get Volumesnapshotclass
NAME                    DRIVER                  DELETIONPOLICY   AGE
csi-aws-vsc             ebs.csi.aws.com         Delete           3h19m
trident-snapshotclass   csi.trident.netapp.io   Delete           6m56s
[ec2-user@ip-10-49-11-132 storage]$
```

A questo punto, una modifica importante da apportare è impostare ontap-nas come classe di archiviazione predefinita anziché gp3, in modo che l'app PostgreSQL che distribuirai in seguito possa utilizzare la classe di archiviazione predefinita. Nella console Openshift del tuo cluster, in Storage seleziona StorageClasses. Modificare l'annotazione della classe predefinita corrente su false e aggiungere l'annotazione storageclass.kubernetes.io/is-default-class impostata su true per la classe di archiviazione ontap-nas.



5. Distribuisci un'applicazione PostgreSQL sul cluster

È possibile distribuire l'applicazione dalla riga di comando come segue:

```
helm install postgresql bitnami/postgresql -n postgresql --create-namespace
```

```
[ec2-user@ip-10-49-11-132 astra]$ helm install postgresql bitnami/postgresql -n postgresql --create-namespace
NAME: postgresql
LAST DEPLOYED: Tue Feb 13 14:46:16 2024
NAMESPACE: postgresql
STATUS: deployed
REVISION: 1
TEST SUITE: None
NOTES:
CHART NAME: postgresql
CHART VERSION: 14.0.4
APP VERSION: 16.2.0

** Please be patient while the chart is being deployed **

PostgreSQL can be accessed via port 5432 on the following DNS names from within your cluster:

    postgresql.postgresql.svc.cluster.local - Read/Write connection

To get the password for "postgres" run:

    export POSTGRES_PASSWORD=$(kubectl get secret --namespace postgresql postgresql -o jsonpath="{.data.postgres-password}" | base64 -d)

To connect to your database run the following command:

    kubectl run postgresql-client --rm --tty -i --restart='Never' --namespace postgresql --image docker.io/bitnami/postgresql:16.2.0-debian-11-r1 --env="PG
PASSWORD=$POSTGRES_PASSWORD" \
      --command -- psql --host postgresql -U postgres -d postgres -p 5432

    > NOTE: If you access the container using bash, make sure that you execute "/opt/bitnami/scripts/postgresql/entrypoint.sh /bin/bash" in order to avoid
the error "psql: local user with ID 1001} does not exist"

To connect to your database from outside the cluster execute the following commands:

    kubectl port-forward --namespace postgresql svc/postgresql 5432:5432 &
    PGPASSWORD="$POSTGRES_PASSWORD" psql --host 127.0.0.1 -U postgres -d postgres -p 5432

WARNING: The configured password will be ignored on new installation in case when previous PostgreSQL release was deleted through the helm command. In that
 case, old PVC will have an old password, and setting it through helm won't take effect. Deleting persistent volumes (PVs) will solve the issue.
[ec2-user@ip-10-49-11-132 astra]$
```

Se non vedi i pod dell'applicazione in esecuzione, potrebbe essersi verificato un errore dovuto a vincoli del contesto di sicurezza.

```
[ec2-user@ip-10-49-11-132 astra]$ kubectl get all -n postgresql
NAME                    TYPE        CLUSTER-IP     EXTERNAL-IP   PORT(S)    AGE
service/postgresql      ClusterIP   172.30.245.50  <none>        5432/TCP   12m
service/postgresql-hl   ClusterIP   None           <none>        5432/TCP   12m

NAME                          READY   AGE
statefulset.apps/postgresql   0/1     12m
[ec2-user@ip-10-49-11-132 astra]$ kubectl get events -n postgresql
LAST SEEN   TYPE      REASON                  OBJECT                                  MESSAGE
2m39s       Normal    WaitForFirstConsumer    persistentvolumeclaim/data-postgresql-0 waiting for first consumer to be created before binding
12m         Normal    SuccessfulCreate        statefulset/postgresql                  create Claim data-postgresql-0 Pod postgresql-0 in StatefulSet postg
resql success
107s        Warning   FailedCreate            statefulset/postgresql                  create Pod postgresql-0 in StatefulSet postgresql failed error: pods
 "postgresql-0" is forbidden: unable to validate against any security context constraint: [provider "trident-controller": Forbidden: not usable by user or
serviceaccount, provider "anyuid": Forbidden: not usable by user or serviceaccount, provider restricted-v2: .spec.securityContext.fsGroup: Invalid value: [
]int64{1001}: 1001 is not an allowed group, provider restricted-v2: .containers[0].runAsUser: Invalid value: 1001: must be in the ranges: [1001010000, 1001
019999], provider "restricted": Forbidden: not usable by user or serviceaccount, provider "nonroot-v2": Forbidden: not usable by user or serviceaccount, pr
ovider "nonroot": Forbidden: not usable by user or serviceaccount, provider "pcap-dedicated-admins": Forbidden: not usable by user or serviceaccount, provi
der "hostmount-anyuid": Forbidden: not usable by user or serviceaccount, provider "machine-api-termination-handler": Forbidden: not usable by user or servi
ceaccount, provider "hostnetwork-v2": Forbidden: not usable by user or serviceaccount, provider "hostnetwork": Forbidden: not usable by user or serviceacco
unt, provider "hostaccess": Forbidden: not usable by user or serviceaccount, provider "splunkforwarder": Forbidden: not usable by user or serviceaccount, p
rovider "trident-node-linux": Forbidden: not usable by user or serviceaccount, provider "node-exporter": Forbidden: not usable by user or serviceaccount, p
rovider "privileged": Forbidden: not usable by user or serviceaccount]
[ec2-user@ip-10-49-11-132 astra]$
```

Correggi l'errore modificando il `runAsUser` E `fsGroup` campi in `statefuleset.apps/postgresql` oggetto con l'uid che si trova nell'output del `oc get project` comando come mostrato di seguito.

```
[ec2-user@ip-10-49-11-132 astra]$ oc get project postgresql -o yaml | grep uid-range
    openshift.io/sa.scc.uid-range: 1001010000/10000
[ec2-user@ip-10-49-11-132 astra]$ oc edit -n postgresql statefulset.apps/postgresql
statefulset.apps/postgresql edited
[ec2-user@ip-10-49-11-132 astra]$
```

L'app PostgreSQL deve essere in esecuzione e utilizzare volumi persistenti supportati dallo storage Amazon FSx ONTAP .

```
[ec2-user@ip-10-49-11-132 astra]$ oc get pods -n postgresql
NAME            READY   STATUS    RESTARTS   AGE
postgresql-0    1/1     Running   0          2m46s
[ec2-user@ip-10-49-11-132 astra]$
```

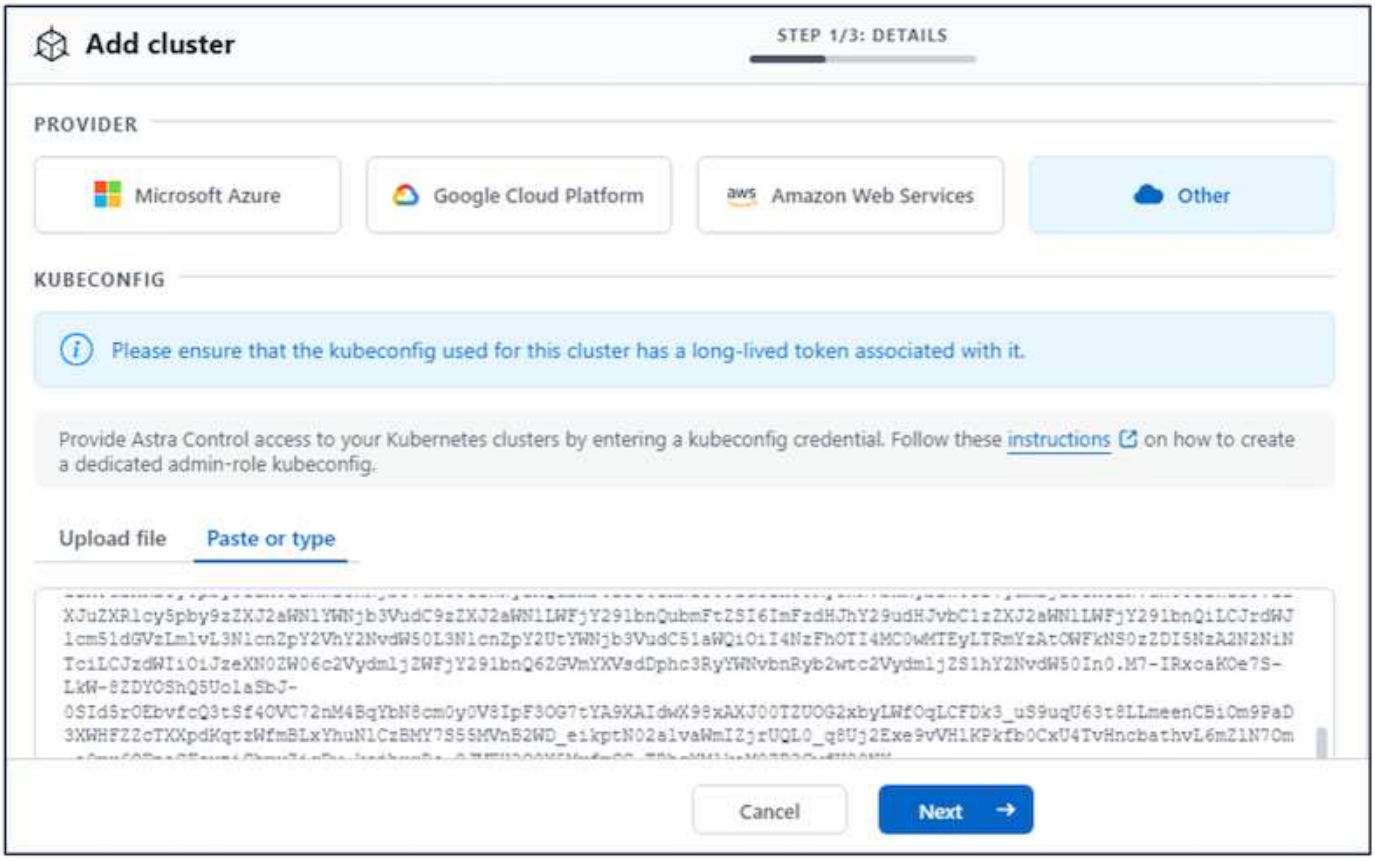
```
[ec2-user@ip-10-49-11-132 storage]$ kubectl get pvc -n postgresql
NAME                STATUS   VOLUME                                     CAPACITY   ACCESS MODES   STORAGECLASS   AGE
data-postgresql-0   Bound    pvc-dd09524a-de75-4825-9424-03a9b91195ca   8Gi        RWO            ontap-nas      4m2s
[ec2-user@ip-10-49-11-132 storage]$
```

6. Crea un database e aggiungi un record

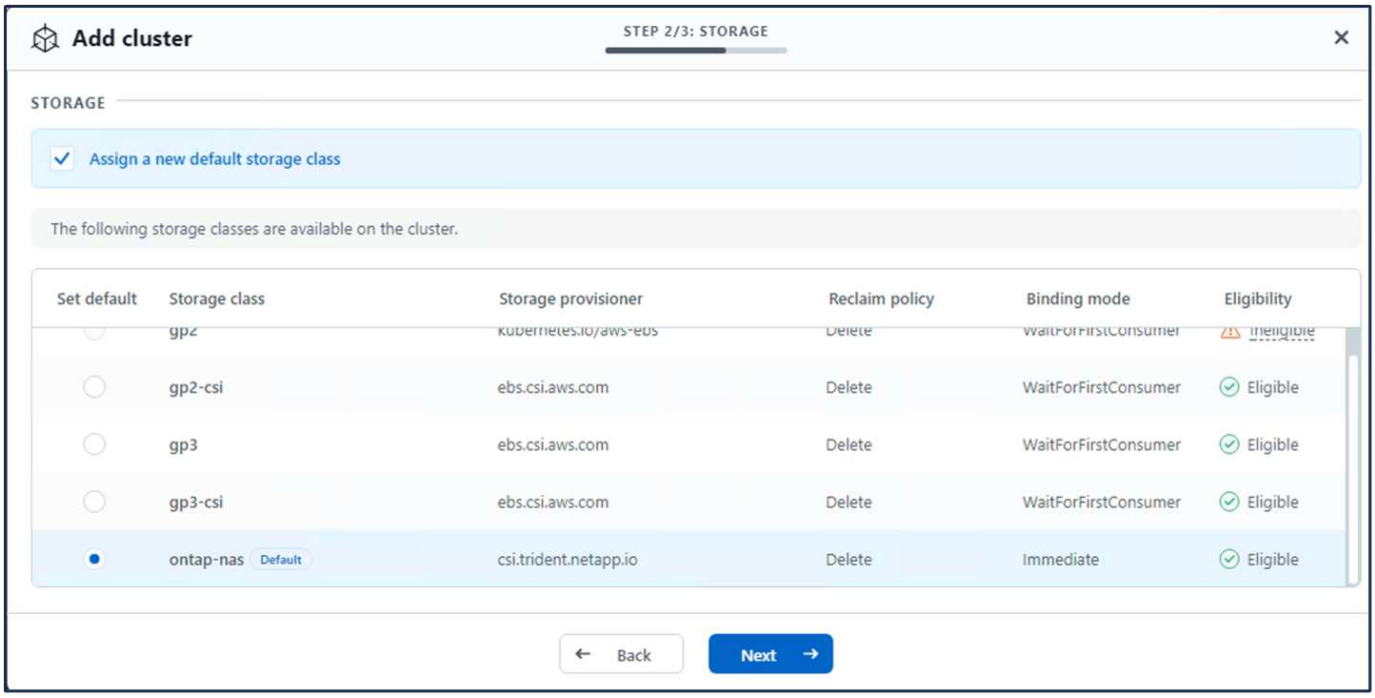
```
[ec2-user@ip-10-49-11-132 astra]$ export POSTGRES_PASSWORD=$(kubectl get secret --namespace postgresql postgresql -o jsonpath="
{.data.postgres-password}" | base64 -d)
[ec2-user@ip-10-49-11-132 astra]$ kubectl run postgresql-client --rm --tty -i --restart='Never' --namespace postgresql --image
docker.io/bitnami/postgresql:16.2.0-debian-11-r1 --env="PGPASSWORD=$POSTGRES_PASSWORD" \
> --command -- psql --host postgresql -U postgres -d postgres -p 5432
Warning: would violate PodSecurity "restricted:v1.24": allowPrivilegeEscalation != false (container "postgresql-client" must se
t securityContext.allowPrivilegeEscalation=false), unrestricted capabilities (container "postgresql-client" must set securityCo
ntext.capabilities.drop=["ALL"]), runAsNonRoot != true (pod or container "postgresql-client" must set securityContext.runAsNonR
oot=true), seccompProfile (pod or container "postgresql-client" must set securityContext.seccompProfile.type to "RuntimeDefault
" or "Localhost")
If you don't see a command prompt, try pressing enter.

postgres=# CREATE DATABASE erp;
CREATE DATABASE
postgres=# \c erp
You are now connected to database "erp" as user "postgres".
erp=# CREATE TABLE PERSONS(ID INT PRIMARY KEY NOT NULL, FIRSTNAME TEXT NOT NULL, LASTNAME TEXT NOT NULL);
CREATE TABLE
erp=# INSERT INTO PERSONS VALUES(1,'John','Doe');
INSERT 0 1
erp=# \dt
         List of relations
 Schema |  Name   | Type  |  Owner
--------+---------+-------+----------
 public | persons | table | postgres
(1 row)

erp=# SELECT * FROM persons;
 id | firstname | lastname
----+-----------+----------
  1 | John      | Doe
(1 row)
```

7. Aggiungi il cluster in ACS

Accedi ad ACS. Selezionare il cluster e fare clic su Aggiungi. Seleziona altro e carica o incolla il file kubeconfig.



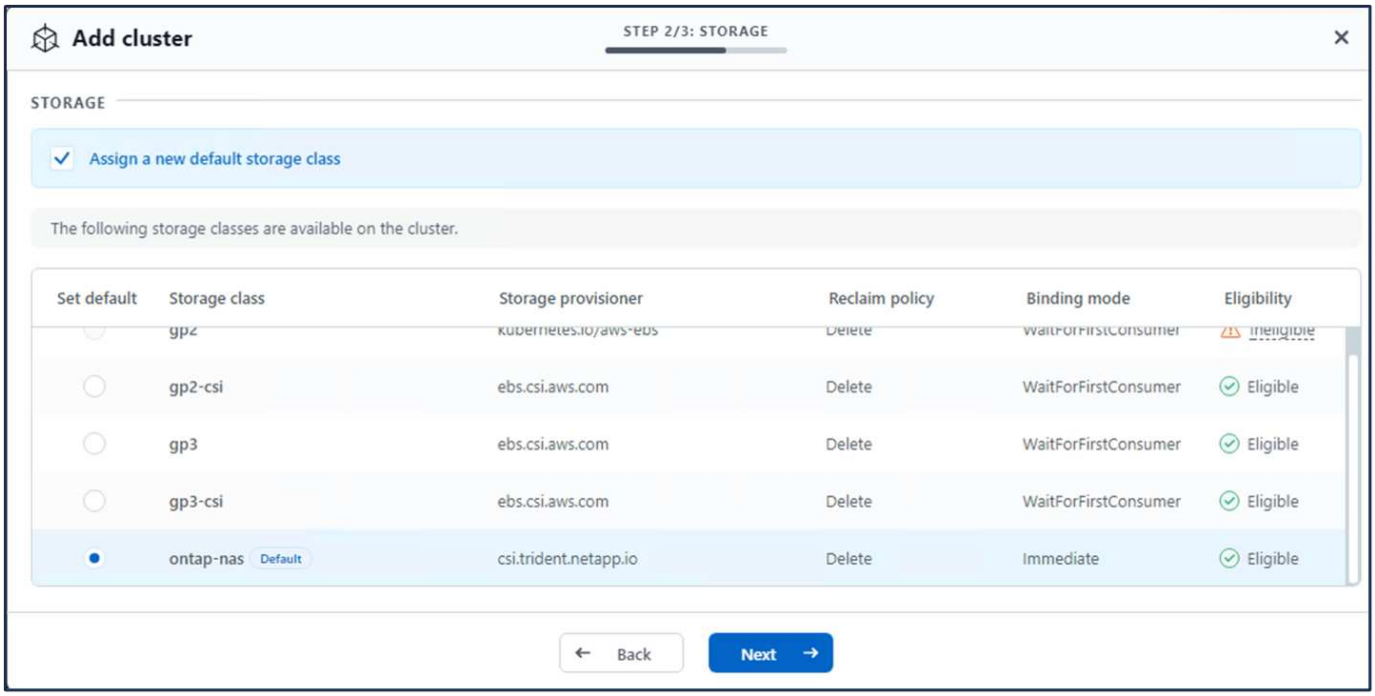
Fare clic su **Avanti** e selezionare ontap-nas come classe di archiviazione predefinita per ACS. Fare clic su **Avanti**, rivedere i dettagli e **Aggiungere** il cluster.



8. Definire l'applicazione in ACS

Definire l'applicazione PostgreSQL in ACS. Dalla landing page, seleziona **Applicazioni**, **Definisci** e compila i dettagli appropriati. Fare clic su **Avanti** un paio di volte, rivedere i dettagli e fare clic su **Definisci**.

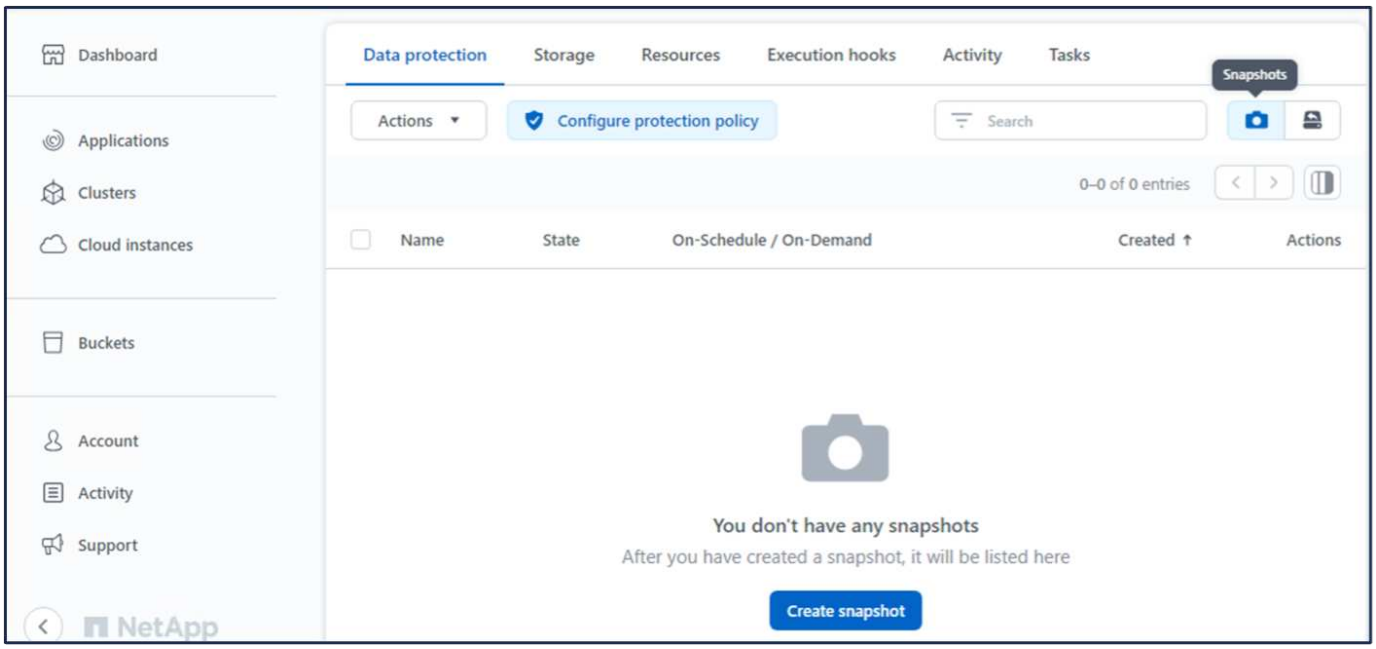
L'applicazione viene aggiunta ad ACS.

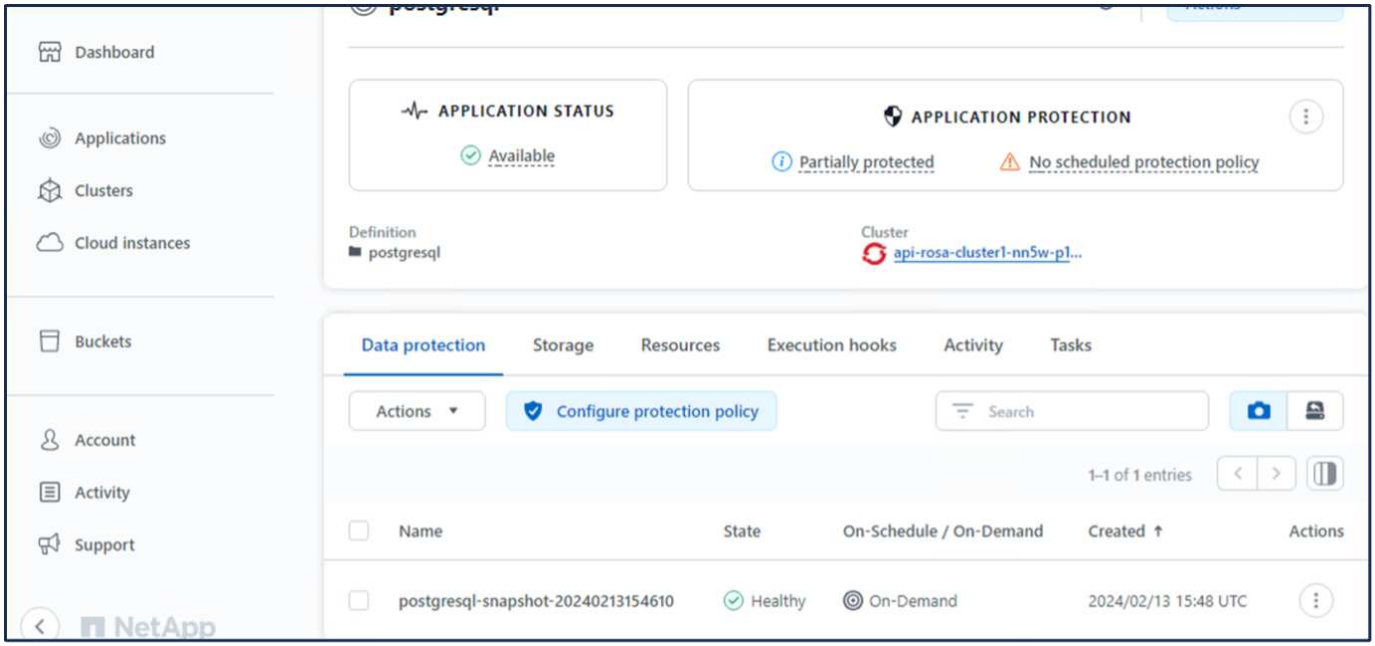


9. Crea uno snapshot utilizzando ACS

Esistono molti modi per creare uno snapshot in ACS. È possibile selezionare l'applicazione e creare uno snapshot dalla pagina che mostra i dettagli dell'applicazione. È possibile fare clic su Crea snapshot per creare uno snapshot su richiesta o configurare un criterio di protezione.

Per creare uno snapshot on-demand, basta cliccare su **Crea snapshot**, specificare un nome, esaminare i dettagli e cliccare su **Snapshot**. Una volta completata l'operazione, lo stato dello snapshot cambia in Integro.





10. Elimina il database nell'applicazione PostgreSQL

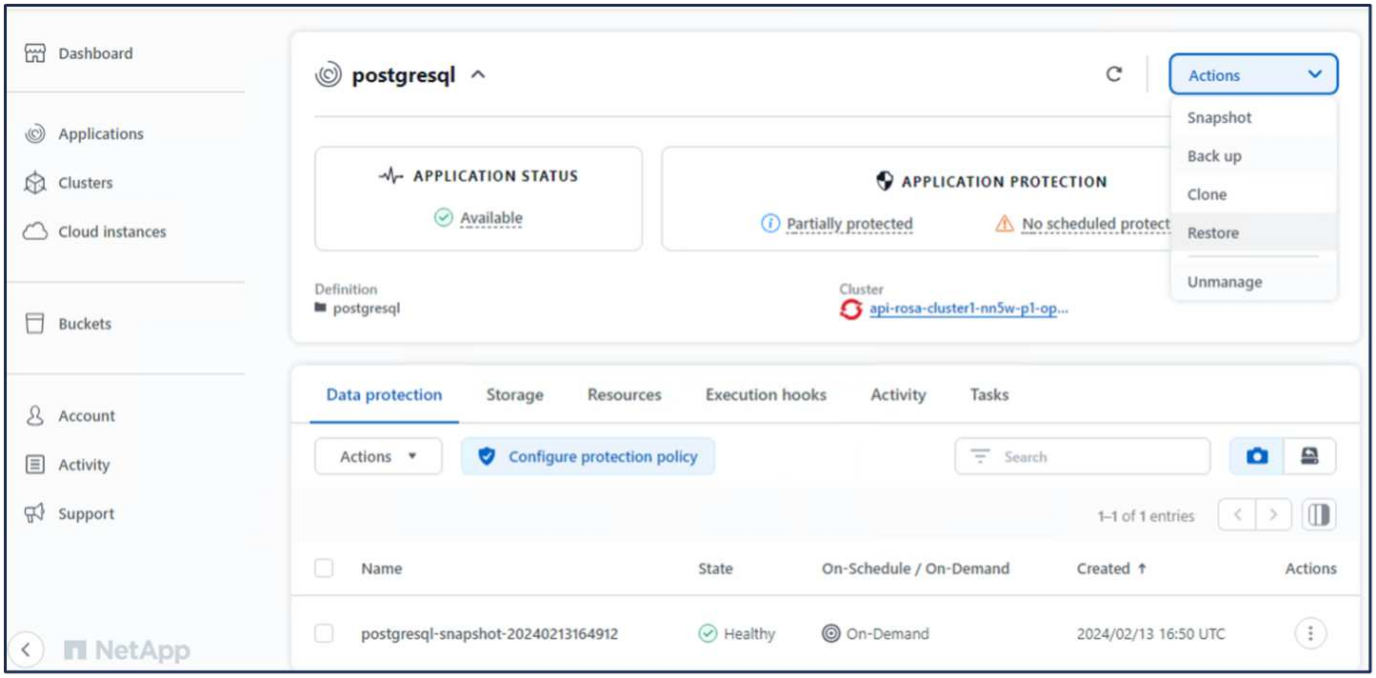
Accedi nuovamente a PostgreSQL, elenca i database disponibili, elimina quello creato in precedenza ed esegui nuovamente l'elenco per assicurarti che il database sia stato eliminato.

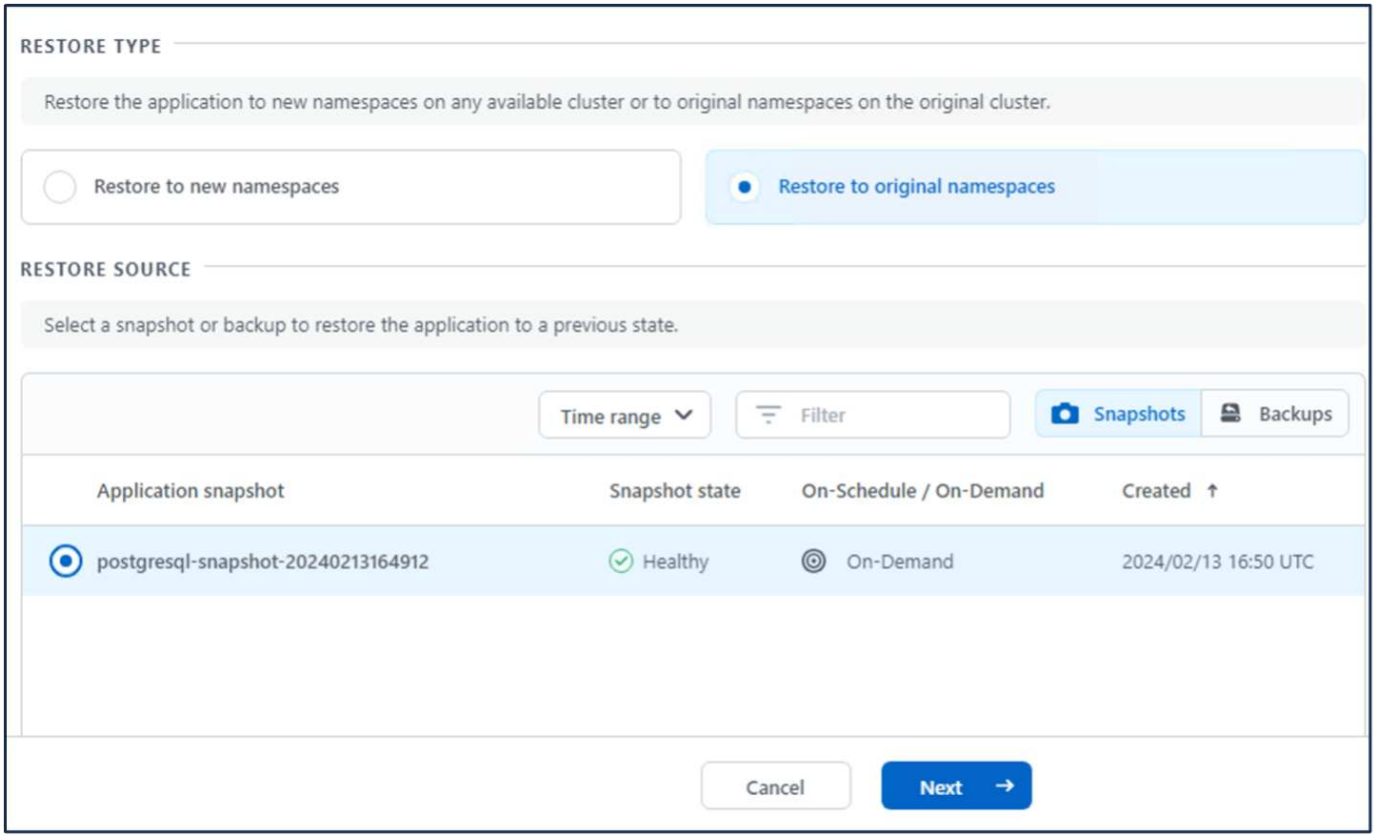
```
postgres=# \l
                                                    List of databases
   Name    |  Owner   | Encoding | Locale Provider |   Collate   |    Ctype    | ICU Locale | ICU Rules |   Access priv
-----------+----------+----------+-----------------+-------------+-------------+------------+-----------+---------------
 erp       | postgres | UTF8     | libc            | en_US.UTF-8 | en_US.UTF-8 |            |           |
 postgres  | postgres | UTF8     | libc            | en_US.UTF-8 | en_US.UTF-8 |            |           |
 template0 | postgres | UTF8     | libc            | en_US.UTF-8 | en_US.UTF-8 |            |           | =c/postgres
+          |          |          |                 |             |             |            |           | postgres=CTc/
 template1 | postgres | UTF8     | libc            | en_US.UTF-8 | en_US.UTF-8 |            |           | =c/postgres
+          |          |          |                 |             |             |            |           | postgres=CTc/
(4 rows)

postgres=# DROP DATABASE erp;
DROP DATABASE
postgres=# \l
                                                    List of databases
   Name    |  Owner   | Encoding | Locale Provider |   Collate   |    Ctype    | ICU Locale | ICU Rules |   Access priv
-----------+----------+----------+-----------------+-------------+-------------+------------+-----------+---------------
 postgres  | postgres | UTF8     | libc            | en_US.UTF-8 | en_US.UTF-8 |            |           |
 template0 | postgres | UTF8     | libc            | en_US.UTF-8 | en_US.UTF-8 |            |           | =c/postgres
+          |          |          |                 |             |             |            |           | postgres=CTc/
 template1 | postgres | UTF8     | libc            | en_US.UTF-8 | en_US.UTF-8 |            |           | =c/postgres
+          |          |          |                 |             |             |            |           | postgres=CTc/
(3 rows)
```

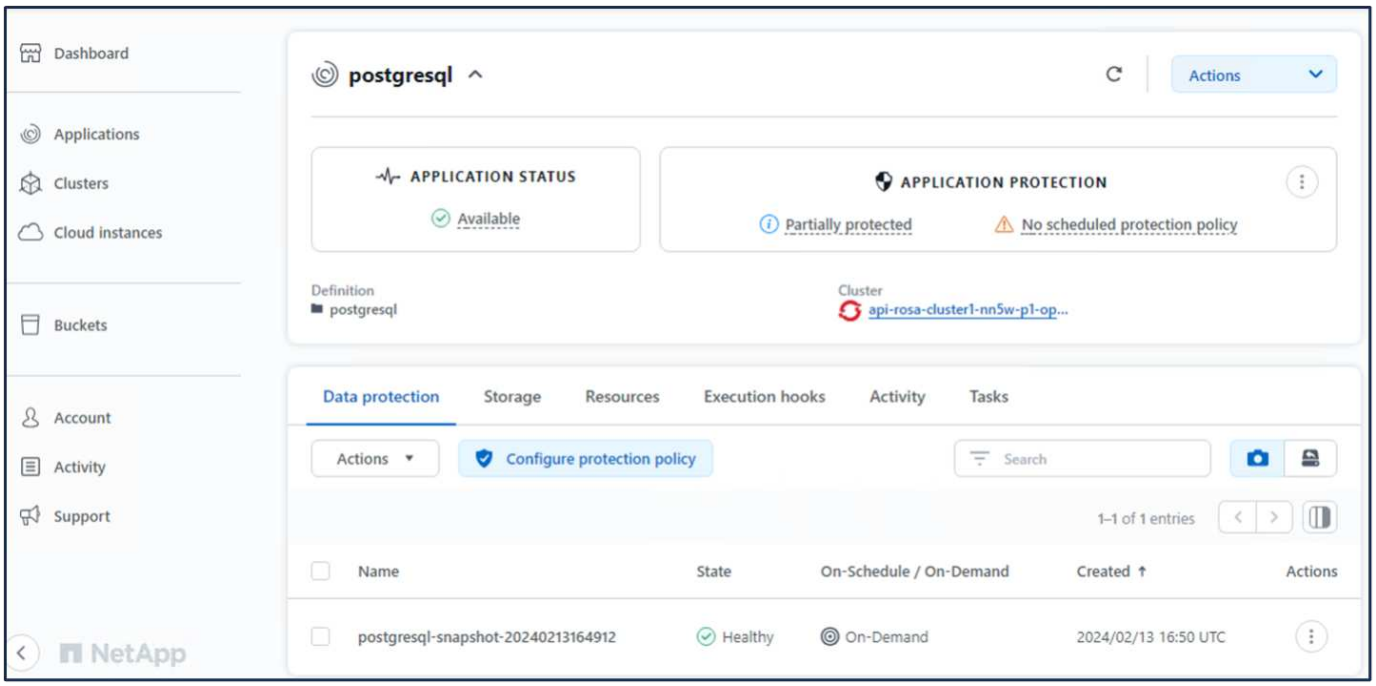
11. Ripristina da uno snapshot utilizzando ACS

Per ripristinare l'applicazione da uno snapshot, vai alla landing page dell'interfaccia utente di ACS, seleziona l'applicazione e seleziona Ripristina. È necessario scegliere uno snapshot o un backup da cui effettuare il

ripristino. (In genere, ne verrebbero creati più di uno in base a una policy configurata). Effettuare le scelte appropriate nelle schermate successive e quindi fare clic su **Ripristina**. Dopo il ripristino dall'istantanea, lo stato dell'applicazione passa da Ripristino in corso a Disponibile.







12. Verifica che la tua app sia stata ripristinata dallo snapshot

Accedi al client PostgreSQL e dovresti ora vedere la tabella e il record nella tabella che avevi in precedenza. Questo è tutto. Basta cliccare su un pulsante e la tua applicazione verrà ripristinata a uno stato precedente. Ecco quanto è semplice per i nostri clienti il nostro lavoro con Astra Control.

```
[ec2-user@ip-10-49-11-132 ~]$ kubectl run postgresql-client --rm --tty -i --restart='Never' --namespace postgresql --image docker.io/bitnami/postgresql:16.
2.0-debian-11-r1 --env="PGPASSWORD=$POSTGRES_PASSWORD" --command -- psql --host postgresql -U postgres -d postgres -p 5432
Warning: would violate PodSecurity "restricted:v1.24": allowPrivilegeEscalation != false (container "postgresql-client" must set securityContext.allowPrivi
legeEscalation=false), unrestricted capabilities (container "postgresql-client" must set securityContext.capabilities.drop=["ALL"]), runAsNonRoot != true (
pod or container "postgresql-client" must set securityContext.runAsNonRoot=true), seccompProfile (pod or container "postgresql-client" must set securityCon
text.seccompProfile.type to "RuntimeDefault" or "Localhost")
If you don't see a command prompt, try pressing enter.

postgres=# \l
                                                  List of databases
   Name    |  Owner   | Encoding | Locale Provider |   Collate   |    Ctype    | ICU Locale | ICU Rules |   Access privileges
-----------+----------+----------+-----------------+-------------+-------------+------------+-----------+-----------------------
 erp       | postgres | UTF8     | libc            | en_US.UTF-8 | en_US.UTF-8 |            |           |
 postgres  | postgres | UTF8     | libc            | en_US.UTF-8 | en_US.UTF-8 |            |           |
 template0 | postgres | UTF8     | libc            | en_US.UTF-8 | en_US.UTF-8 |            |           | =c/postgres          +
           |          |          |                 |             |             |            |           | postgres=CTc/postgres
 template1 | postgres | UTF8     | libc            | en_US.UTF-8 | en_US.UTF-8 |            |           | =c/postgres          +
           |          |          |                 |             |             |            |           | postgres=CTc/postgres
(4 rows)

postgres=# \c erp
You are now connected to database "erp" as user "postgres".
erp=# \dt
          List of relations
 Schema |  Name   | Type  |  Owner
--------+---------+-------+----------
 public | persons | table | postgres
(1 row)

erp=# SELECT * from PERSONS;
 id | firstname | lastname
----+-----------+----------
  1 | John      | Doe
(1 row)
```

Migrazione dei dati

Questa pagina mostra le opzioni di migrazione dei dati per i carichi di lavoro dei container sui cluster Managed Red Hat OpenShift utilizzando FSx ONTAP per l'archiviazione persistente.

Migrazione dei dati

Il servizio Red Hat OpenShift su AWS e Amazon FSx for NetApp ONTAP (FSx ONTAP) fanno parte del portafoglio di servizi di AWS. FSx ONTAP è disponibile nelle opzioni Single AZ o Multi-AZ. L'opzione Multi-Az garantisce la protezione dei dati in caso di guasti della zona di disponibilità. FSx ONTAP può essere integrato con Trident per fornire storage persistente per le applicazioni sui cluster ROSA.

Integrazione di FSx ONTAP con Trident utilizzando il grafico Helm

Integrazione del cluster ROSA con Amazon FSx ONTAP

La migrazione delle applicazioni container comporta:

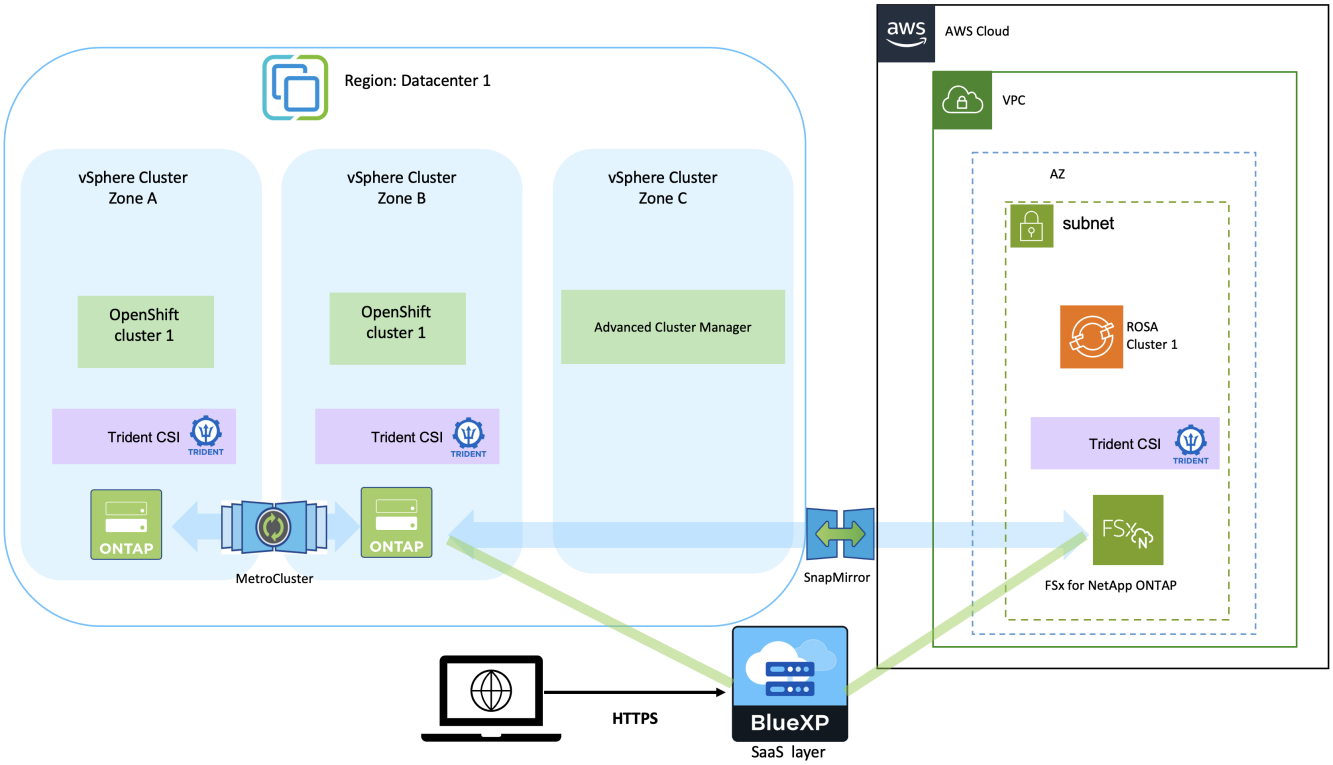
- Volumi persistenti: questo può essere ottenuto utilizzando BlueXP. Un'altra opzione è quella di utilizzare Trident Protect per gestire le migrazioni delle applicazioni container dall'ambiente locale a quello cloud. L'automazione può essere utilizzata per lo stesso scopo.
- Metadati dell'applicazione: questo può essere ottenuto utilizzando OpenShift GitOps (Argo CD).

Failover e fail-back delle applicazioni sul cluster ROSA utilizzando FSx ONTAP per l'archiviazione persistente

Il video seguente è una dimostrazione di scenari di failover e fail-back delle applicazioni utilizzando BlueXP e Argo CD.

Failover e fail-back delle applicazioni sul cluster ROSA

Soluzione di protezione e migrazione dei dati per carichi di lavoro OpenShift Container



Ulteriori soluzioni NetApp Hybrid Multicloud per carichi di lavoro Red Hat OpenShift

Soluzioni aggiuntive

Ulteriori soluzioni sono disponibili nelle seguenti sezioni:

Per le soluzioni Red Hat OpenShift Container, vedere"Qui" .

Per le soluzioni Red Hat OpenShift Virtualization in sede, vedere"Qui" .

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 COMMERCIABILITÀ 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 CONSEQUENZIALE (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.