Présentation de l'installation

Astra Control Center

NetApp
November 21, 2023

This PDF was generated from https://docs.netapp.com/fr-fr/astra-control-center-2208/get-started/acc_cluster_cr_options.html on November 21, 2023. Always check docs.netapp.com for the latest.

Sommaire

Présentation de l'installation

Choisissez l'une des procédures d'installation suivantes du centre de contrôle Astra :

- "Installer le centre de contrôle Astra en suivant la procédure standard"
- "(Si vous utilisez Red Hat OpenShift) installez Astra Control Center à l'aide d'OpenShift OperatorHub"
- "Installer le centre de contrôle Astra avec un système de stockage back-end Cloud Volumes ONTAP"

Installer le centre de contrôle Astra en suivant la procédure standard

Pour installer le centre de contrôle Astra, téléchargez le bundle d'installation sur le site de support NetApp et effectuez les opérations suivantes pour installer l'opérateur du centre de contrôle Astra et le centre de contrôle Astra dans votre environnement. Vous pouvez utiliser cette procédure pour installer Astra Control Center dans des environnements connectés à Internet ou équipés d'un filtre à air.

Pour les environnements Red Hat OpenShift, vous pouvez utiliser un "autre procédure" Pour installer Astra Control Center à l'aide d'OpenShift OperatorHub.

Ce dont vous avez besoin

- "Avant de commencer l'installation, préparez votre environnement pour le déploiement d'Astra Control Center".
- Si vous avez configuré ou que vous souhaitez configurer des stratégies de sécurité de pod dans votre environnement, familiarisez-vous avec les stratégies de sécurité de pod et leur incidence sur l'installation d'Astra Control Center. Voir "Comprendre les restrictions de la stratégie de sécurité du pod".
- S'assurer que tous les opérateurs du groupe d'instruments sont en état de fonctionnement et disponibles.

```
kubectl get clusteroperators
```

- Assurez-vous que tous les services API sont en état de santé et disponibles :

```
kubectl get apiservices
```

- Assurez-vous que le FQDN Astra que vous prévoyez d'utiliser est routable vers ce cluster. Cela signifie que vous avez une entrée DNS dans votre serveur DNS interne ou que vous utilisez une route URL de base déjà enregistrée.
- Si un cert-Manager existe déjà dans le cluster, vous devez en effectuer certaines "étapes préalables" Pour qu'Astra Control Center n'installe pas son propre cert-Manager.

Description de la tâche

La procédure d'installation d'Astra Control Center est la suivante :

- Installe les composants Astra dans le `netapp-acc` (ou espace de nom personnalisé).
- Crée un compte par défaut.
- Définit une adresse e-mail d'utilisateur administratif par défaut et un mot de passe unique par défaut. Ce

rôle propriétaire est attribué à cet utilisateur dans le système qui est nécessaire pour la première connexion à l'interface utilisateur.

- Vous aide à déterminer que toutes les POD Astra Control Center sont en cours d'exécution.
- Installe l'interface utilisateur Astra.

> (Applicable uniquement à la version EAP (Data Store Early Access Program) d'Astra) si vous prévoyez de gérer le magasin de données Astra à l'aide d'Astra Control Center et d'activer les flux de travail VMware, déployez Astra Control Center uniquement sur le `pcloud` et pas sur le `netapp-acc` espace de noms ou espace de noms personnalisé décrits dans les étapes de cette procédure.

> N'exécutez pas la commande suivante pendant l'intégralité du processus d'installation pour éviter de supprimer toutes les pods Astra Control Center : `kubectl delete -f astra_control_center_operator_deploy.yaml`

> Si vous utilisez le Podman de Red Hat au lieu de Docker Engine, vous pouvez utiliser les commandes Podman à la place des commandes Docker.

Étapes

Pour installer le centre de contrôle Astra, procédez comme suit :

- Téléchargez et déballez le pack Astra Control Center
- Installez le plug-in NetApp Astra kubectl
- Ajoutez les images à votre registre local
- Configurez l'espace de noms et le secret pour les registres avec les exigences d'authentification
- Poser le conducteur du centre de commande Astra
- Configurer le centre de contrôle Astra
- Installation complète du centre de contrôle Astra et du conducteur
- Vérifiez l'état du système
- Configurer l'entrée pour l'équilibrage de charge
- Connectez-vous à l'interface utilisateur du centre de contrôle Astra

Téléchargez et déballez le pack Astra Control Center

1. Téléchargez le pack Astra Control Center (`astra-control-center-[version].tar.gz`) du "Site de support NetApp".
2. Téléchargez le code postal des certificats et clés Astra Control Center sur le "Site de support NetApp".
3. (Facultatif) utilisez la commande suivante pour vérifier la signature du pack :

```
openssl dgst -sha256 -verify AstraControlCenter-public.pub -signature
astra-control-center-[version].tar.gz.sig astra-control-center-
[version].tar.gz
```

4. Extraire les images :

```
tar -vxzf astra-control-center-[version].tar.gz
```

Installez le plug-in NetApp Astra kubectl

NetApp Astra `kubectl` Le plug-in de ligne de commande permet de gagner du temps lors de l'exécution des tâches courantes associées au déploiement et à la mise à niveau d'Astra Control Center.

Ce dont vous avez besoin

NetApp fournit des binaires pour différents systèmes d'exploitation et architectures CPU. Avant d'effectuer cette tâche, vous devez savoir quelle unité centrale et quel système d'exploitation vous possédez. Sur les systèmes d'exploitation Linux et Mac, vous pouvez utiliser `uname -a` commande permettant de collecter ces informations.

Étapes

1. Répertoriez l'Astra de NetApp disponible `kubectl` Les binaires du plug-in, et notez le nom du fichier dont vous avez besoin pour votre système d'exploitation et l'architecture de l'UC :

```
ls kubectl-astra/
```

2. Copiez le fichier au même emplacement que la norme `kubectl` informatique. Dans cet exemple, le `kubectl` l'utilitaire se trouve dans le `/usr/local/bin` répertoire. Remplacement `<binary-name>` avec le nom du fichier dont vous avez besoin :

```
cp kubectl-astra/<binary-name> /usr/local/bin/kubectl-astra
```

Ajoutez les images à votre registre local

1. Suivez la séquence d'étapes appropriée pour votre moteur de mise en conteneurs :

Docker

1. Passez au répertoire Astra :

```
cd acc
```

2. placez les images du paquet dans le répertoire d'images Astra Control Center dans votre registre local. Exécutez les substitutions suivantes avant d'exécuter la commande :
   - Remplacez BUNDLE_FILE par le nom du fichier bundle Astra Control (par exemple, `acc.manifest.yaml`).
   - Remplacez MON_REGISTRE par l'URL du référentiel Docker.
   - Remplacez MON_REGISTRE_UTILISATEUR par le nom d'utilisateur.
   - Remplacez MON_REGISTRY_TOKEN par un jeton autorisé pour le Registre.

```
kubectl astra packages push-images -m BUNDLE_FILE -r MY_REGISTRY
-u MY_REGISTRY_USER -p MY_REGISTRY_TOKEN
```

Podman

1. Connectez-vous à votre registre :

```
podman login [your_registry_path]
```

2. Exécutez le script suivant, en procédant à la substitution <YOUR_REGISTRY> comme indiqué dans les commentaires :

```
# You need to be at the root of the tarball.
# You should see these files to confirm correct location:
#   acc.manifest.yaml
#   acc/

# Replace <YOUR_REGISTRY> with your own registry (e.g
registry.customer.com or registry.customer.com/testing, etc..)
export REGISTRY=<YOUR_REGISTRY>
export PACKAGENAME=acc
export PACKAGEVERSION=22.08.1-26
export DIRECTORYNAME=acc
for astraImageFile in $(ls ${DIRECTORYNAME}/images/*.tar) ; do
  # Load to local cache
  astraImage=$(podman load --input ${astraImageFile} | sed 's/Loaded
image(s): //')

  # Remove path and keep imageName.
  astraImageNoPath=$(echo ${astraImage} | sed 's:.*/::')

  # Tag with local image repo.
  podman tag ${astraImage} ${REGISTRY}/netapp/astra/${PACKAGENAME}
/${PACKAGEVERSION}/${astraImageNoPath}

  # Push to the local repo.
  podman push ${REGISTRY}/netapp/astra/${PACKAGENAME}/
${PACKAGEVERSION}/${astraImageNoPath}
done
```

Configurez l'espace de noms et le secret pour les registres avec les exigences d'authentification

1. Exporter le KUBECONFIG pour le groupe hôte du centre de contrôle Astra :

```
export KUBECONFIG=[file path]
```

2. Si vous utilisez un registre qui nécessite une authentification, vous devez procéder comme suit :

   a. Créer le `netapp-acc-operator` espace de noms :

```
kubectl create ns netapp-acc-operator
```

   Réponse :

```
namespace/netapp-acc-operator created
```

b. Créez un secret pour le `netapp-acc-operator` espace de noms. Ajoutez des informations sur Docker et exécutez la commande suivante :

> Le paramètre fictif `your_registry_path` doit correspondre à l'emplacement des images que vous avez téléchargées précédemment (par exemple, `[Registry_URL]/netapp/astra/astracc/22.08.1-26`).

```
kubectl create secret docker-registry astra-registry-cred -n netapp-
acc-operator --docker-server=[your_registry_path] --docker-username
=[username] --docker-password=[token]
```

Exemple de réponse :

```
secret/astra-registry-cred created
```

> Si vous supprimez l'espace de noms après la génération du secret, vous devez régénérer le secret pour l'espace de noms après la recréation de l'espace de noms.

c. Créer le `netapp-acc` (ou espace de nom personnalisé).

```
kubectl create ns [netapp-acc or custom namespace]
```

Exemple de réponse :

```
namespace/netapp-acc created
```

d. Créez un secret pour le `netapp-acc` (ou espace de nom personnalisé). Ajoutez des informations sur Docker et exécutez la commande suivante :

```
kubectl create secret docker-registry astra-registry-cred -n [netapp-
acc or custom namespace] --docker-server=[your_registry_path]
--docker-username=[username] --docker-password=[token]
```

Réponse

```
secret/astra-registry-cred created
```

a. (Facultatif) si vous souhaitez que le cluster soit automatiquement géré par Astra Control Center après

l'installation, assurez-vous de fournir le kubeconfig comme secret dans l'espace de noms de l'Astra Control Center que vous souhaitez déployer à l'aide de cette commande :

```
kubectl create secret generic [acc-kubeconfig-cred or custom secret name] --from-file=<path-to-your-kubeconfig> -n [netapp-acc or custom namespace]
```

Poser le conducteur du centre de commande Astra

1. Modifier le répertoire :

```
cd manifests
```

2. Modifiez le YAML de déploiement de l'opérateur Astra Control Center (`astra_control_center_operator_deploy.yaml`) pour faire référence à votre registre local et à votre secret.

```
vim astra_control_center_operator_deploy.yaml
```

> Un échantillon annoté YAML suit ces étapes.

a. Si vous utilisez un registre qui nécessite une authentification, remplacez la ligne par défaut de `imagePullSecrets: []` avec les éléments suivants :

```
imagePullSecrets:
- name: <astra-registry-cred>
```

b. Changer `[your_registry_path]` pour le `kube-rbac-proxy` image dans le chemin du registre où vous avez poussé les images dans un étape précédente.

c. Changer `[your_registry_path]` pour le `acc-operator-controller-manager` image dans le chemin du registre où vous avez poussé les images dans un étape précédente.

d. (Pour les installations utilisant l'aperçu d'Astra Data Store) Découvrez ce problème connu concernant "Les spécialistes en provisionnement de classe de stockage et les changements supplémentaires que vous devrez apporter au YAML".

```
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    control-plane: controller-manager
  name: acc-operator-controller-manager
  namespace: netapp-acc-operator
spec:
  replicas: 1
  selector:
    matchLabels:
      control-plane: controller-manager
  template:
    metadata:
      labels:
        control-plane: controller-manager
    spec:
      containers:
      - args:
        - --secure-listen-address=0.0.0.0:8443
        - --upstream=http://127.0.0.1:8080/
        - --logtostderr=true
        - --v=10
        image: [your_registry_path]/kube-rbac-proxy:v4.8.0
        name: kube-rbac-proxy
        ports:
        - containerPort: 8443
          name: https
      - args:
        - --health-probe-bind-address=:8081
        - --metrics-bind-address=127.0.0.1:8080
        - --leader-elect
        command:
        - /manager
        env:
        - name: ACCOP_LOG_LEVEL
          value: "2"
        image: [your_registry_path]/acc-operator:[version x.y.z]
        imagePullPolicy: IfNotPresent
      imagePullSecrets: []
```

3. Poser le conducteur du centre de commande Astra :

```
kubectl apply -f astra_control_center_operator_deploy.yaml
```

Exemple de réponse :

```
namespace/netapp-acc-operator created
customresourcedefinition.apiextensions.k8s.io/astracontrolcenters.astra.
netapp.io created
role.rbac.authorization.k8s.io/acc-operator-leader-election-role created
clusterrole.rbac.authorization.k8s.io/acc-operator-manager-role created
clusterrole.rbac.authorization.k8s.io/acc-operator-metrics-reader
created
clusterrole.rbac.authorization.k8s.io/acc-operator-proxy-role created
rolebinding.rbac.authorization.k8s.io/acc-operator-leader-election-
rolebinding created
clusterrolebinding.rbac.authorization.k8s.io/acc-operator-manager-
rolebinding created
clusterrolebinding.rbac.authorization.k8s.io/acc-operator-proxy-
rolebinding created
configmap/acc-operator-manager-config created
service/acc-operator-controller-manager-metrics-service created
deployment.apps/acc-operator-controller-manager created
```

4. Vérifiez que les pods sont en cours d'exécution :

```
kubectl get pods -n netapp-acc-operator
```

Configurer le centre de contrôle Astra

1. Modifiez le fichier de ressources personnalisées (CR) Astra Control Center (`astra_control_center_min.yaml`) Pour créer des comptes, AutoSupport, registre et autres configurations nécessaires :

> `astra_control_center_min.yaml` Est le CR par défaut et convient à la plupart des installations. Familiarisez-vous avec tous "Les options CR et leurs valeurs potentielles" Pour vous assurer de déployer le centre de contrôle Astra correctement pour votre environnement. Si d'autres personnalisations sont nécessaires pour votre environnement, vous pouvez l'utiliser `astra_control_center.yaml` En tant que CR alternatif.

```
vim astra_control_center_min.yaml
```

> Si vous utilisez un registre qui ne requiert pas d'autorisation, vous devez supprimer le `secret` ligne comprise entre `imageRegistry` sinon, l'installation échouera.

a. Changer `[your_registry_path]` vers le chemin du registre où vous avez poussé les images à l'étape précédente.

b. Modifiez le `accountName` chaîne du nom que vous souhaitez associer au compte.

c. Modifiez le `astraAddress` Chaîne du FQDN que vous souhaitez utiliser dans votre navigateur pour accéder à Astra. Ne pas utiliser `http://` ou `https://` dans l'adresse. Copier ce FQDN pour l'utiliser dans un plus tard.

d. Modifiez le `email` chaîne à l'adresse d'administrateur initiale par défaut. Copiez cette adresse e-mail pour l'utiliser dans un plus tard.

e. Changer `enrolled` Pour AutoSupport à `false` pour les sites sans connexion internet ou sans conservation `true` pour les sites connectés.

f. Si vous utilisez un cert-Manager externe, ajoutez les lignes suivantes à `spec`:

```
spec:
  crds:
    externalCertManager: true
```

g. (Facultatif) Ajouter un prénom `firstName` et nom `lastName` de l'utilisateur associé au compte. Vous pouvez effectuer cette étape maintenant ou plus tard dans l'interface utilisateur.

h. (Facultatif) modifiez le `storageClass` Valeur ajoutée pour une autre ressource de stockage Trident si votre installation l'exige.

i. (Facultatif) si vous souhaitez que le cluster soit géré automatiquement par Astra Control Center après l'installation et que vous l'ayez déjà fait créé le secret contenant le kubecconfig pour ce cluster, Indiquez le nom du secret en ajoutant un nouveau champ à ce fichier YAML appelé `astraKubeConfigSecret: "acc-kubeconfig-cred or custom secret name"`

j. Effectuez l'une des opérations suivantes :

- **Autre contrôleur d'entrée (ingressType:Generic)**: Il s'agit de l'action par défaut avec Astra Control Center. Après le déploiement du centre de contrôle Astra, vous devrez configurer le contrôleur d'entrée pour exposer le centre de contrôle Astra à une URL.

  L'installation par défaut d'Astra Control Center configure sa passerelle (`service/traefik`) pour être du type `ClusterIP`. Avec cette installation par défaut, vous devez également configurer une entrée/un contrôleur Kubernetes IngressController pour y acheminer le trafic. Si vous souhaitez utiliser une entrée, reportez-vous à la section "Configurer l'entrée pour l'équilibrage de charge".

- **Équilibreur de charge de service (ingressType:AccTraefik)**: Si vous ne souhaitez pas installer un IngressController ou créer une ressource d'entrée, définissez `ingressType` à `AccTraefik`.

  Ceci déploie le centre de contrôle Astra `traefik` Passerelle en tant que service de type Kubernetes LoadBalancer.

  Le centre de contrôle Astra utilise un service de type « équilibreur de charge » (`svc/traefik` Dans l'espace de noms du centre de contrôle Astra), et exige qu'il se voit attribuer une adresse IP externe accessible. Si des équilibreurs de charge sont autorisés dans votre environnement et que vous n'en avez pas encore configuré, vous pouvez utiliser MetalLB ou un autre équilibreur de charge de service externe pour attribuer une adresse IP externe au service. Dans la configuration du serveur DNS interne, pointez le nom DNS choisi pour Astra Control Center vers l'adresse IP à équilibrage de charge.

Pour plus de détails sur le type de service « LoadBalancer » et l'entrée, voir "De formation".

```
apiVersion: astra.netapp.io/v1
kind: AstraControlCenter
metadata:
  name: astra
spec:
  accountName: "Example"
  astraVersion: "ASTRA_VERSION"
  astraAddress: "astra.example.com"
  astraKubeConfigSecret: "acc-kubeconfig-cred or custom secret name"
  ingressType: "Generic"
  autoSupport:
    enrolled: true
  email: "[admin@example.com]"
  firstName: "SRE"
  lastName: "Admin"
  imageRegistry:
    name: "[your_registry_path]"
    secret: "astra-registry-cred"
  storageClass: "ontap-gold"
```

Installation complète du centre de contrôle Astra et du conducteur

1. Si vous ne l'avez pas déjà fait dans une étape précédente, créez le `netapp-acc` (ou personnalisée) espace de noms :

   ```
   kubectl create ns [netapp-acc or custom namespace]
   ```

   Exemple de réponse :

   ```
   namespace/netapp-acc created
   ```

2. Poser le centre de contrôle Astra dans le `netapp-acc` (ou votre espace de noms personnalisé) :

   ```
   kubectl apply -f astra_control_center_min.yaml -n [netapp-acc or custom
   namespace]
   ```

   Exemple de réponse :

```
astracontrolcenter.astra.netapp.io/astra created
```

Vérifiez l'état du système

> Si vous préférez utiliser OpenShift, vous pouvez utiliser des commandes oc comparables pour les étapes de vérification.

1. Vérifiez que tous les composants du système sont correctement installés.

   ```
   kubectl get pods -n [netapp-acc or custom namespace]
   ```

   Chaque pod doit avoir un statut de `Running`. Le déploiement des modules du système peut prendre plusieurs minutes.

Exemple de réponse

```
NAME                                        READY   STATUS    RESTARTS
AGE
acc-helm-repo-6b44d68d94-d8m55              1/1     Running   0
13m
activity-78f99ddf8-hltct                    1/1     Running   0
10m
api-token-authentication-457nl              1/1     Running   0
9m28s
api-token-authentication-dgwsz              1/1     Running   0
9m28s
api-token-authentication-hmqqc              1/1     Running   0
9m28s
asup-75fd554dc6-m6qzh                       1/1     Running   0
9m38s
authentication-6779b4c85d-92gds             1/1     Running   0
8m11s
bucketservice-7cc767f8f8-lqwr8              1/1     Running   0
9m31s
certificates-549fd5d6cb-5kmd6               1/1     Running   0
9m56s
certificates-549fd5d6cb-bkjh9               1/1     Running   0
9m56s
cloud-extension-7bcb7948b-hn8h2             1/1     Running   0
10m
cloud-insights-service-56ccf86647-fgg69     1/1     Running   0
9m46s
composite-compute-677685b9bb-7vgsf          1/1     Running   0
10m
composite-volume-657d6c5585-dnq79           1/1     Running   0
9m49s
credentials-755fd867c8-vrlmt                1/1     Running   0
11m
entitlement-86495cdf5b-nwhh2                1/1     Running   2
10m
features-5684fb8b56-8d6s8                   1/1     Running   0
10m
fluent-bit-ds-rhx7v                         1/1     Running   0
7m48s
fluent-bit-ds-rjms4                         1/1     Running   0
7m48s
fluent-bit-ds-zf5ph                         1/1     Running   0
7m48s
graphql-server-66d895f544-w6hjd             1/1     Running   0
3m29s
```

```
identity-744df448d5-rlcmm                1/1     Running   0
10m
influxdb2-0                              1/1     Running   0
13m
keycloak-operator-75c965cc54-z7csw       1/1     Running   0
8m16s
krakend-798d6df96f-9z2sk                 1/1     Running   0
3m26s
license-5fb7d75765-f8mjg                 1/1     Running   0
9m50s
login-ui-7d5b7df85d-l2s7s                1/1     Running   0
3m20s
loki-0                                   1/1     Running   0
13m
metrics-facade-599b9d7fcc-gtmgl          1/1     Running   0
9m40s
monitoring-operator-67cc74f844-cdplp     2/2     Running   0
8m11s
nats-0                                   1/1     Running   0
13m
nats-1                                   1/1     Running   0
13m
nats-2                                   1/1     Running   0
12m
nautilus-769f5b74cd-k5jxm                1/1     Running   0
9m42s
nautilus-769f5b74cd-kd9gd                1/1     Running   0
8m59s
openapi-84f6ccd8ff-76kvp                 1/1     Running   0
9m34s
packages-6f59fc67dc-4g2f5                1/1     Running   0
9m52s
polaris-consul-consul-server-0           1/1     Running   0
13m
polaris-consul-consul-server-1           1/1     Running   0
13m
polaris-consul-consul-server-2           1/1     Running   0
13m
polaris-keycloak-0                       1/1     Running   0
8m7s
polaris-keycloak-1                       1/1     Running   0
5m49s
polaris-keycloak-2                       1/1     Running   0
5m15s
polaris-keycloak-db-0                    1/1     Running   0
8m6s
```

```
polaris-keycloak-db-1                     1/1     Running   0
5m49s
polaris-keycloak-db-2                     1/1     Running   0
4m57s
polaris-mongodb-0                         2/2     Running   0
13m
polaris-mongodb-1                         2/2     Running   0
12m
polaris-mongodb-2                         2/2     Running   0
12m
polaris-ui-565f56bf7b-zwr8b               1/1     Running   0
3m19s
polaris-vault-0                           1/1     Running   0
13m
polaris-vault-1                           1/1     Running   0
13m
polaris-vault-2                           1/1     Running   0
13m
public-metrics-6d86d66444-2wbzl           1/1     Running   0
9m30s
storage-backend-metrics-77c5d98dcd-dbhg5 1/1     Running   0
9m44s
storage-provider-78c885f57c-6zcv4         1/1     Running   0
9m36s
telegraf-ds-2l2m9                         1/1     Running   0
7m48s
telegraf-ds-qfzgh                         1/1     Running   0
7m48s
telegraf-ds-shrms                         1/1     Running   0
7m48s
telegraf-rs-bjpkt                         1/1     Running   0
7m48s
telemetry-service-6684696c64-qzfdf        1/1     Running   0
10m
tenancy-6596b6c54d-vmpsm                  1/1     Running   0
10m
traefik-7489dc59f9-6mnst                  1/1     Running   0
3m19s
traefik-7489dc59f9-xrkgg                  1/1     Running   0
3m4s
trident-svc-6c8dc458f5-jswcl              1/1     Running   0
10m
vault-controller-6b954f9b76-gz9nm         1/1     Running   0
11m
```

2. (Facultatif) pour vous assurer que l'installation est terminée, vous pouvez regarder le `acc-operator` journaux utilisant la commande suivante.

```
kubectl logs deploy/acc-operator-controller-manager -n netapp-acc-
operator -c manager -f
```

> `accHost` l'enregistrement du cluster est l'une des dernières opérations. en cas de défaillance, le déploiement ne pourra pas échouer. Dans l'éventualité où un échec d'enregistrement de cluster était indiqué dans les journaux, vous pouvez réessayer d'enregistrer via le flux de production Add cluster "Dans l'interface utilisateur" Ou API.

3. Lorsque tous les modules sont en cours d'exécution, vérifiez que l'installation a réussi (`READY` est `True`) Et obtenez le mot de passe unique que vous utiliserez lorsque vous vous connectez à Astra Control Center :

```
kubectl get AstraControlCenter -n netapp-acc
```

Réponse :

```
NAME    UUID                                      VERSION     ADDRESS
READY
astra   ACC-9aa5fdae-4214-4cb7-9976-5d8b4c0ce27f  22.08.1-26
10.111.111.111  True
```

> Copiez la valeur UUID. Le mot de passe est `ACC-` Suivi de la valeur UUID (`ACC-[UUID]` ou, dans cet exemple, `ACC-9aa5fdae-4214-4cb7-9976-5d8b4c0ce27f`).

Configurer l'entrée pour l'équilibrage de charge

Vous pouvez configurer un contrôleur d'entrée Kubernetes qui gère l'accès externe aux services, comme l'équilibrage de la charge dans un cluster.

Cette procédure explique comment configurer un contrôleur d'entrée (`ingressType:Generic`). Il s'agit de l'action par défaut avec Astra Control Center. Après le déploiement du centre de contrôle Astra, vous devrez configurer le contrôleur d'entrée pour exposer le centre de contrôle Astra à une URL.

> Si vous ne souhaitez pas configurer un contrôleur d'entrée, vous pouvez le configurer `ingressType:AccTraefik`). Le centre de contrôle Astra utilise un service de type « équilibreur de charge » (`svc/traefik` Dans l'espace de noms du centre de contrôle Astra), et exige qu'il se voit attribuer une adresse IP externe accessible. Si des équilibreurs de charge sont autorisés dans votre environnement et que vous n'en avez pas encore configuré, vous pouvez utiliser MetalLB ou un autre équilibreur de charge de service externe pour attribuer une adresse IP externe au service. Dans la configuration du serveur DNS interne, pointez le nom DNS choisi pour Astra Control Center vers l'adresse IP à équilibrage de charge. Pour plus de détails sur le type de service « LoadBalancer » et l'entrée, voir "De formation".

Les étapes diffèrent en fonction du type de contrôleur d'entrée utilisé :

- Entrée Istio
- Contrôleur d'entrée Nginx
- Contrôleur d'entrée OpenShift

Ce dont vous avez besoin

- Le requis "contrôleur d'entrée" doit déjà être déployé.
- Le "classe d'entrée" correspondant au contrôleur d'entrée doit déjà être créé.
- Vous utilisez les versions de Kubernetes entre et, y compris v1.19 et v1.22.

Étapes pour l'entrée Istio

1. Configurer l'entrée Istio.

   Cette procédure suppose que Istio est déployé à l'aide du profil de configuration par défaut.

2. Rassemblez ou créez le certificat et le fichier de clé privée souhaités pour la passerelle d'entrée.

   Vous pouvez utiliser un certificat signé par une autorité de certification ou auto-signé. Le nom commun doit être l'adresse Astra (FQDN).

   Exemple de commande :

   ```
   openssl req -x509 -nodes -days 365 -newkey rsa:2048
   -keyout tls.key -out tls.crt
   ```

3. Créez un secret `tls secret name` de type `kubernetes.io/tls` Pour une clé privée TLS et un certificat dans `istio-system namespace` Comme décrit dans les secrets TLS.

   Exemple de commande :

   ```
   kubectl create secret tls [tls secret name]
   --key="tls.key"
   --cert="tls.crt" -n istio-system
   ```

   Le nom du secret doit correspondre au `spec.tls.secretName` fourni dans `istio-ingress.yaml` fichier.

4. Déployez une ressource entrée dans `netapp-acc` (Ou espace de noms personnalisé) utilisant soit l'espace de noms v1beta1 (obsolète dans la version Kubernetes inférieure à ou 1.22) soit le type de ressource v1 pour un schéma obsolète ou un nouveau schéma :

   Résultat :

```
apiVersion: networking.k8s.io/v1beta1
kind: IngressClass
metadata:
  name: istio
spec:
  controller: istio.io/ingress-controller
---
apiVersion: networking.k8s.io/v1beta1
kind: Ingress
metadata:
  name: ingress
  namespace: istio-system
spec:
  ingressClassName: istio
  tls:
  - hosts:
    - <ACC addess>
    secretName: [tls secret name]
  rules:
  - host: [ACC addess]
    http:
      paths:
      - path: /
        pathType: Prefix
        backend:
          serviceName: traefik
          servicePort: 80
```

Pour le nouveau schéma v1, suivez cet exemple :

```
kubectl apply -f istio-Ingress.yaml
```

Résultat :

```
apiVersion: networking.k8s.io/v1
kind: IngressClass
metadata:
  name: istio
spec:
  controller: istio.io/ingress-controller
---
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: ingress
  namespace: istio-system
spec:
  ingressClassName: istio
  tls:
  - hosts:
    - <ACC addess>
    secretName: [tls secret name]
  rules:
  - host: [ACC addess]
    http:
      paths:
      - path: /
        pathType: Prefix
        backend:
          service:
            name: traefik
            port:
              number: 80
```

5. Déployez Astra Control Center comme d'habitude.
6. Vérifier l'état de l'entrée :

```
kubectl get ingress -n netapp-acc
```

Réponse :

```
NAME    CLASS HOSTS             ADDRESS         PORTS   AGE
ingress istio astra.example.com 172.16.103.248  80, 443 1h
```

Étapes du contrôleur d'entrée Nginx

1. Créer un secret de type[`kubernetes.io/tls`] Pour une clé privée TLS et un certificat dans `netapp-acc` (ou espace de noms personnalisé) comme décrit dans "Secrets TLS".

2. Déployez une ressource entrée dans `netapp-acc` (ou espace de nom personnalisé) utilisant l'un ou l'autre `v1beta1` (Obsolète dans la version Kubernetes inférieure à ou 1.22) ou `v1` type de ressource pour un schéma obsolète ou nouveau :

   a. Pour un `v1beta1` schéma obsolète, suivre cet exemple :

```
apiVersion: extensions/v1beta1
Kind: IngressClass
metadata:
  name: ingress-acc
  namespace: [netapp-acc or custom namespace]
  annotations:
    kubernetes.io/ingress.class: [class name for nginx controller]
spec:
  tls:
  - hosts:
    - <ACC address>
    secretName: [tls secret name]
  rules:
  - host: [ACC address]
    http:
      paths:
      - backend:
        serviceName: traefik
        servicePort: 80
        pathType: ImplementationSpecific
```

   b. Pour le `v1` nouveau schéma, suivez cet exemple :

```yaml
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: netapp-acc-ingress
  namespace: [netapp-acc or custom namespace]
spec:
  ingressClassName: [class name for nginx controller]
  tls:
  - hosts:
    - <ACC address>
    secretName: [tls secret name]
  rules:
  - host: <ACC addess>
    http:
      paths:
        - path:
          backend:
            service:
              name: traefik
              port:
                number: 80
          pathType: ImplementationSpecific
```

Étapes du contrôleur d'entrée OpenShift

1. Procurez-vous votre certificat et obtenez les fichiers de clé, de certificat et d'autorité de certification prêts à l'emploi par la route OpenShift.
2. Création de la route OpenShift :

```
oc create route edge --service=traefik
--port=web -n [netapp-acc or custom namespace]
--insecure-policy=Redirect --hostname=<ACC address>
--cert=cert.pem --key=key.pem
```

Connectez-vous à l'interface utilisateur du centre de contrôle Astra

Après avoir installé Astra Control Center, vous modifierez le mot de passe de l'administrateur par défaut et vous connecterez au tableau de bord de l'interface utilisateur de Astra Control Center.

Étapes

1. Dans un navigateur, entrez le FQDN que vous avez utilisé dans le `astraAddress` dans le `astra_control_center_min.yaml` CR quand Vous avez installé Astra Control Center.
2. Acceptez les certificats auto-signés lorsque vous y êtes invité.

> Vous pouvez créer un certificat personnalisé après la connexion.

3. Dans la page de connexion à Astra Control Center, entrez la valeur que vous avez utilisée `email` dans `astra_control_center_min.yaml` CR quand Vous avez installé Astra Control Center, suivi du mot de passe à usage unique (`ACC-[UUID]`).

> Si vous saisissez trois fois un mot de passe incorrect, le compte admin est verrouillé pendant 15 minutes.

4. Sélectionnez **connexion**.
5. Modifiez le mot de passe lorsque vous y êtes invité.

> Si c'est votre premier login et que vous oubliez le mot de passe et qu'aucun autre compte utilisateur administratif n'a encore été créé, contactez le support NetApp pour obtenir de l'aide pour la récupération de mot de passe.

6. (Facultatif) supprimez le certificat TLS auto-signé existant et remplacez-le par un "Certificat TLS personnalisé signé par une autorité de certification".

Dépanner l'installation

Si l'un des services est dans `Error` état, vous pouvez inspecter les journaux. Rechercher les codes de réponse API dans la plage 400 à 500. Ceux-ci indiquent l'endroit où un échec s'est produit.

Étapes

1. Pour inspecter les journaux de l'opérateur de l'Astra Control Center, entrez ce qui suit :

```
kubectl logs --follow -n netapp-acc-operator $(kubectl get pods -n
netapp-acc-operator -o name) -c manager
```

Et la suite

Terminez le déploiement en effectuant le processus "tâches de configuration".

=
:allow-uri-read:

Comprendre les restrictions de la stratégie de sécurité du pod

Astra Control Center prend en charge la limitation des privilèges via les politiques de sécurité du pod (PSP). Les stratégies de sécurité des pods vous permettent de limiter les utilisateurs ou les groupes capables d'exécuter des conteneurs et les privilèges dont ils disposent.

Certaines distributions Kubernetes, telles que RKE2, ont une stratégie de sécurité de pod par défaut trop restrictive et provoquent des problèmes lors de l'installation d'Astra Control Center.

Vous pouvez utiliser les informations et exemples inclus ici pour comprendre les politiques de sécurité du pod que le Control Center d'Astra et configurer les règles de sécurité du pod qui fournissent la protection dont vous avez besoin sans interférer avec les fonctions du Control Center d'Astra.

PSP installé par Astra Control Center

Astra Control Center crée plusieurs politiques de sécurité de pod pendant l'installation. Certaines sont permanentes, certaines d'entre elles sont créées pendant certaines opérations et sont supprimées une fois l'opération terminée.

PSP créé lors de l'installation

Lors de l'installation d'Astra Control Center, l'opérateur d'Astra Control Center installe une stratégie de sécurité de pod personnalisée, un objet de rôle et un objet RoleBinding pour prendre en charge le déploiement des services Astra Control Center dans l'espace de noms Astra Control Center.

La nouvelle règle et les objets ont les attributs suivants :

```
kubectl get psp

NAME                            PRIV    CAPS          SELINUX    RUNASUSER
FSGROUP     SUPGROUP    READONLYROOTFS   VOLUMES
avp-psp                         false                 RunAsAny   RunAsAny
RunAsAny    RunAsAny    false            *
netapp-astra-deployment-psp     false                 RunAsAny   RunAsAny
RunAsAny    RunAsAny    false            *

kubectl get role

NAME                                     CREATED AT
netapp-astra-deployment-role             2022-06-27T19:34:58Z

kubectl get rolebinding

NAME                                     ROLE
AGE
netapp-astra-deployment-rb               Role/netapp-astra-deployment-role
32m
```

PSP créé pendant les opérations de sauvegarde

Pendant les opérations de sauvegarde, Astra Control Center crée une règle de sécurité dynamique de pod, un objet ClusterRole et un objet RoleBinding. Ils prennent en charge le processus de sauvegarde, qui se produit dans un espace de noms distinct.

La nouvelle règle et les objets ont les attributs suivants :

```
kubectl get psp

NAME                            PRIV    CAPS
SELINUX    RUNASUSER          FSGROUP     SUPGROUP    READONLYROOTFS
VOLUMES
netapp-astra-backup             false   DAC_READ_SEARCH
RunAsAny   RunAsAny           RunAsAny    RunAsAny    false            *

kubectl get role

NAME                   CREATED AT
netapp-astra-backup    2022-07-21T00:00:00Z

kubectl get rolebinding

NAME                   ROLE                        AGE
netapp-astra-backup    Role/netapp-astra-backup    62s
```

PSP créé lors de la gestion du cluster

Lorsque vous gérez un cluster, Astra Control Center installe l'opérateur de surveillance netapp dans le cluster géré. Cet opérateur crée une politique de sécurité pod, un objet ClusterRole et un objet RoleBinding pour déployer des services de télémétrie dans l'espace de noms Astra Control Center.

La nouvelle règle et les objets ont les attributs suivants :

```
kubectl get psp

NAME                            PRIV    CAPS
SELINUX    RUNASUSER          FSGROUP     SUPGROUP    READONLYROOTFS
VOLUMES
netapp-monitoring-psp-nkmo      true    AUDIT_WRITE,NET_ADMIN,NET_RAW
RunAsAny   RunAsAny           RunAsAny    RunAsAny    false            *

kubectl get role

NAME                                          CREATED AT
netapp-monitoring-role-privileged             2022-07-21T00:00:00Z

kubectl get rolebinding

NAME                                                     ROLE
AGE
netapp-monitoring-role-binding-privileged                Role/netapp-
monitoring-role-privileged              2m5s
```

Activer la communication réseau entre les espaces de noms

Certains environnements utilisent les constructions NetworkPolicy pour limiter le trafic entre les espaces de noms. L'opérateur d'Astra Control Center, Astra Control Center et le plug-in Astra pour VMware vSphere sont tous dans des espaces de noms différents. Les services de ces différents espaces de noms doivent être capables de communiquer les uns avec les autres. Pour activer cette communication, procédez comme suit.

Étapes

1. Supprimez toutes les ressources NetworkPolicy qui existent dans l'espace de noms Astra Control Center :

```
kubectl get networkpolicy -n netapp-acc
```

2. Pour chaque objet NetworkPolicy renvoyé par la commande précédente, utilisez la commande suivante pour le supprimer. Remplacez <NOM_OBJET> par le nom de l'objet renvoyé :

```
kubectl delete networkpolicy <OBJECT_NAME> -n netapp-acc
```

3. Appliquez le fichier de ressources suivant pour configurer l'objet de stratégie réseau-acc-avp pour permettre à Astra Plugin pour les services VMware vSphere de faire des demandes aux services Astra Control Center. Remplacez entre parenthèses <> par les informations relatives à votre environnement :

```
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
  name: acc-avp-network-policy
  namespace: <ACC_NAMESPACE_NAME> # REPLACE THIS WITH THE ASTRA CONTROL
CENTER NAMESPACE NAME
spec:
  podSelector: {}
  policyTypes:
    - Ingress
  ingress:
    - from:
        - namespaceSelector:
            matchLabels:
              kubernetes.io/metadata.name: <PLUGIN_NAMESPACE_NAME> #
REPLACE THIS WITH THE ASTRA PLUGIN FOR VMWARE VSPHERE NAMESPACE NAME
```

4. Appliquez le fichier de ressources suivant pour configurer l'objet de stratégie réseau ACC-opérateur pour permettre à l'opérateur Astra Control Center de communiquer avec les services Astra Control Center. Remplacez entre parenthèses <> par les informations relatives à votre environnement :

```
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
  name: acc-operator-network-policy
  namespace: <ACC_NAMESPACE_NAME> # REPLACE THIS WITH THE ASTRA CONTROL
CENTER NAMESPACE NAME
spec:
  podSelector: {}
  policyTypes:
    - Ingress
  ingress:
    - from:
        - namespaceSelector:
            matchLabels:
              kubernetes.io/metadata.name: <NETAPP-ACC-OPERATOR> #
REPLACE THIS WITH THE OPERATOR NAMESPACE NAME
```

Supprimer les limitations de ressources

Certains environnements utilisent les objets ResourceQuotas et LimitRanges pour empêcher les ressources d'un namespace de consommer l'ensemble des CPU et de la mémoire disponibles sur le cluster. Le centre de contrôle Astra ne fixe pas de limites maximales, il ne sera donc pas conforme à ces ressources. Vous devez les supprimer des espaces de noms où vous prévoyez d'installer Astra Control Center.

Vous pouvez suivre les étapes suivantes pour récupérer et supprimer ces quotas et ces limites. Dans ces exemples, la sortie de la commande est affichée immédiatement après la commande.

Étapes

1. Obtenez les quotas de ressources dans l'espace de noms netapp-acc :

   ```
   kubectl get quota -n netapp-acc
   ```

   Réponse :

   ```
   NAME          AGE   REQUEST                                          LIMIT
   pods-high     16s   requests.cpu: 0/20, requests.memory: 0/100Gi
   limits.cpu: 0/200, limits.memory: 0/1000Gi
   pods-low      15s   requests.cpu: 0/1, requests.memory: 0/1Gi
   limits.cpu: 0/2, limits.memory: 0/2Gi
   pods-medium   16s   requests.cpu: 0/10, requests.memory: 0/20Gi
   limits.cpu: 0/20, limits.memory: 0/200Gi
   ```

2. Supprimez tous les quotas de ressources par nom :

```
kubectl delete resourcequota  pods-high -n netapp-acc
```

```
kubectl delete resourcequota  pods-low -n netapp-acc
```

```
kubectl delete resourcequota  pods-medium -n netapp-acc
```

3. Consultez les plages de limite dans l'espace de noms netapp-acc :

```
kubectl get limits -n netapp-acc
```

Réponse :

```
NAME              CREATED AT
cpu-limit-range   2022-06-27T19:01:23Z
```

4. Supprimez les plages de limite par nom :

```
kubectl delete limitrange cpu-limit-range -n netapp-acc
```

=
:allow-uri-read:

Installez Astra Control Center à l'aide d'OpenShift OperatorHub

Si vous utilisez Red Hat OpenShift, vous pouvez installer Astra Control Center à l'aide de l'opérateur certifié Red Hat. Utiliser cette procédure pour installer le centre de contrôle Astra à partir du "Catalogue de l'écosystème Red Hat" Ou utilisez Red Hat OpenShift Container Platform.

Une fois cette procédure terminée, vous devez revenir à la procédure d'installation pour terminer le "les étapes restantes" pour vérifier que l'installation a réussi et ouvrir une session.

Ce dont vous avez besoin

- "Avant de commencer l'installation, préparez votre environnement pour le déploiement d'Astra Control Center".
- Depuis votre cluster OpenShift, assurez-vous que tous les opérateurs de clusters sont en état sain (`available` est `true`) :

```
oc get clusteroperators
```

- Depuis votre cluster OpenShift, assurez-vous que tous les services d'API sont en état sain (`available` est `true`) :

```
oc get apiservices
```

- Créez une adresse FQDN pour Astra Control Center dans votre centre de données.
- Obtenez les autorisations nécessaires et l'accès à Red Hat OpenShift Container Platform pour effectuer les étapes d'installation décrites.
- Si un cert-Manager existe déjà dans le cluster, vous devez en effectuer certaines "étapes préalables" Pour qu'Astra Control Center n'installe pas son propre cert-Manager.

Étapes

- Téléchargez et déballez le pack Astra Control Center
- Installez le plug-in NetApp Astra kubectl
- Ajoutez les images à votre registre local
- Recherchez la page d'installation de l'opérateur
- Poser l'opérateur
- Poser le centre de contrôle Astra

Téléchargez et déballez le pack Astra Control Center

1. Téléchargez le pack Astra Control Center (`astra-control-center-[version].tar.gz`) du "Site de support NetApp".
2. Téléchargez le code postal des certificats et clés Astra Control Center sur le "Site de support NetApp".
3. (Facultatif) utilisez la commande suivante pour vérifier la signature du pack :

```
openssl dgst -sha256 -verify AstraControlCenter-public.pub -signature
astra-control-center-[version].tar.gz.sig astra-control-center-
[version].tar.gz
```

4. Extraire les images :

```
tar -vxzf astra-control-center-[version].tar.gz
```

Installez le plug-in NetApp Astra kubectl

NetApp Astra `kubectl` Le plug-in de ligne de commande permet de gagner du temps lors de l'exécution des tâches courantes associées au déploiement et à la mise à niveau d'Astra Control Center.

Ce dont vous avez besoin

NetApp fournit des binaires pour différents systèmes d'exploitation et architectures CPU. Avant d'effectuer cette tâche, vous devez savoir quelle unité centrale et quel système d'exploitation vous possédez. Sur les systèmes d'exploitation Linux et Mac, vous pouvez utiliser `uname -a` commande permettant de collecter ces

informations.

Étapes

1. Répertoriez l'Astra de NetApp disponible `kubectl` Les binaires du plug-in, et notez le nom du fichier dont vous avez besoin pour votre système d'exploitation et l'architecture de l'UC :

```
ls kubectl-astra/
```

2. Copiez le fichier au même emplacement que la norme `kubectl` informatique. Dans cet exemple, le `kubectl` l'utilitaire se trouve dans le `/usr/local/bin` répertoire. Remplacement `<binary-name>` avec le nom du fichier dont vous avez besoin :

```
cp kubectl-astra/<binary-name> /usr/local/bin/kubectl-astra
```

Ajoutez les images à votre registre local

1. Suivez la séquence d'étapes appropriée pour votre moteur de mise en conteneurs :

Docker

1. Passez au répertoire Astra :

```
cd acc
```

2. placez les images du paquet dans le répertoire d'images Astra Control Center dans votre registre local. Exécutez les substitutions suivantes avant d'exécuter la commande :
   - Remplacez BUNDLE_FILE par le nom du fichier bundle Astra Control (par exemple, `acc.manifest.yaml`).
   - Remplacez MON_REGISTRE par l'URL du référentiel Docker.
   - Remplacez MON_REGISTRE_UTILISATEUR par le nom d'utilisateur.
   - Remplacez MON_REGISTRY_TOKEN par un jeton autorisé pour le Registre.

```
kubectl astra packages push-images -m BUNDLE_FILE -r MY_REGISTRY
-u MY_REGISTRY_USER -p MY_REGISTRY_TOKEN
```

Podman

1. Connectez-vous à votre registre :

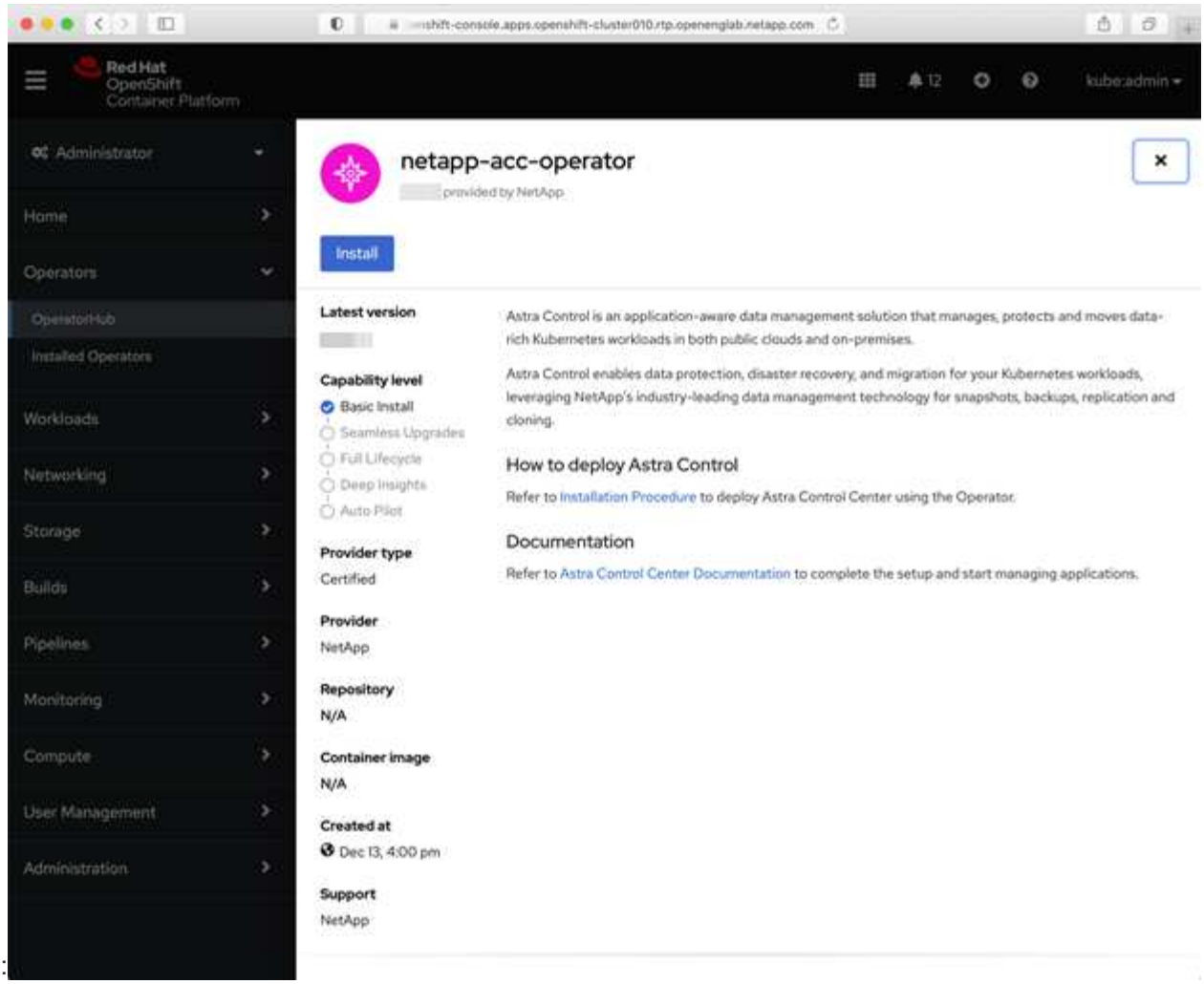
```
podman login [your_registry_path]
```

2. Exécutez le script suivant, en procédant à la substitution <YOUR_REGISTRY> comme indiqué dans les commentaires :

```
# You need to be at the root of the tarball.
# You should see these files to confirm correct location:
#   acc.manifest.yaml
#   acc/

# Replace <YOUR_REGISTRY> with your own registry (e.g
registry.customer.com or registry.customer.com/testing, etc..)
export REGISTRY=<YOUR_REGISTRY>
export PACKAGENAME=acc
export PACKAGEVERSION=22.08.1-26
export DIRECTORYNAME=acc
for astraImageFile in $(ls ${DIRECTORYNAME}/images/*.tar) ; do
  # Load to local cache
  astraImage=$(podman load --input ${astraImageFile} | sed 's/Loaded
image(s): //')

  # Remove path and keep imageName.
  astraImageNoPath=$(echo ${astraImage} | sed 's:.*/::')

  # Tag with local image repo.
  podman tag ${astraImage} ${REGISTRY}/netapp/astra/${PACKAGENAME}
/${PACKAGEVERSION}/${astraImageNoPath}

  # Push to the local repo.
  podman push ${REGISTRY}/netapp/astra/${PACKAGENAME}/
${PACKAGEVERSION}/${astraImageNoPath}
done
```

Recherchez la page d'installation de l'opérateur

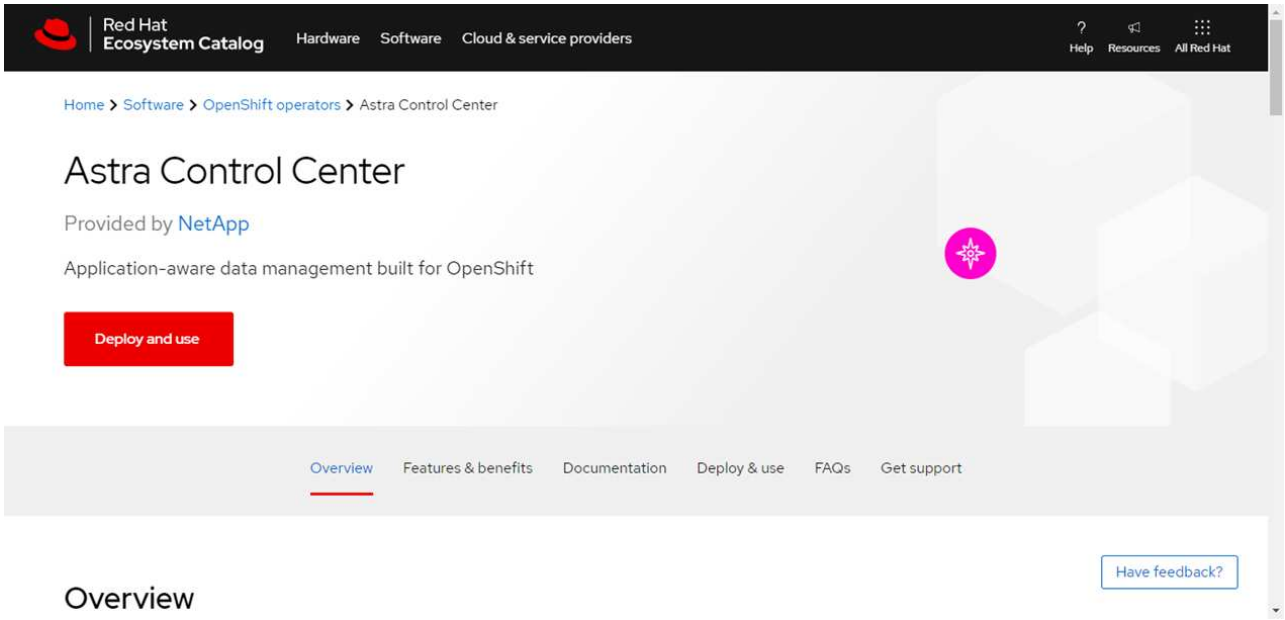
1. Effectuez l'une des procédures suivantes pour accéder à la page d'installation de l'opérateur :
    - Depuis la console Web Red Hat OpenShift



:

i. Connectez-vous à l'interface utilisateur de OpenShift Container Platform.
ii. Dans le menu latéral, sélectionnez **Operators > OperatorHub**.
iii. Sélectionnez l'opérateur du centre de contrôle Astra NetApp.
iv. Sélectionnez **installer**.

- À partir du catalogue de l'écosystème Red Hat
  :



i. Sélectionnez le centre de contrôle NetApp Astra "opérateur".

ii. Sélectionnez **déployer et utiliser**.

Poser l'opérateur

1. Complétez la page **Install Operator** et installez l'opérateur :

   L'opérateur sera disponible dans tous les namespaces du cluster.

   a. Sélectionnez l'espace de noms de l'opérateur ou `netapp-acc-operator` l'espace de noms sera créé automatiquement dans le cadre de l'installation de l'opérateur.

   b. Sélectionnez une stratégie d'approbation manuelle ou automatique.

   L'approbation manuelle est recommandée. Une seule instance d'opérateur doit s'exécuter par cluster.

   c. Sélectionnez **installer**.

   Si vous avez sélectionné une stratégie d'approbation manuelle, vous serez invité à approuver le plan d'installation manuelle pour cet opérateur.

2. Depuis la console, accéder au menu OperatorHub et vérifier que l'opérateur a bien installé.

Poser le centre de contrôle Astra

1. Depuis la console dans la vue détaillée du conducteur du centre de contrôle Astra, sélectionnez `Create instance` Dans la section API fournies.

2. Complétez le `Create AstraControlCenter` champ de formulaire :

   a. Conservez ou ajustez le nom du centre de contrôle Astra.

   b. (Facultatif) Activer ou désactiver Auto support. Il est recommandé de conserver la fonctionnalité Auto support.

c. Entrez l'adresse du centre de contrôle Astra. N'entrez pas `http://` ou `https://` dans l'adresse.
d. Entrez la version Astra Control Center, par exemple 21.12.60.
e. Entrez un nom de compte, une adresse e-mail et un nom d'administrateur.
f. Conservez la règle de récupération du volume par défaut.
g. Dans **image Registry**, entrez le chemin d'accès au registre d'images du conteneur local. N'entrez pas `http://` ou `https://` dans l'adresse.
h. Si vous utilisez un registre qui nécessite une authentification, saisissez le secret.
i. Entrez le prénom de l'administrateur.
j. Configurer l'évolutivité des ressources.
k. Conservez la classe de stockage par défaut.
l. Définissez les préférences de gestion de CRD.

3. Sélectionnez `Create`.

Et la suite

Vérifier que le centre de contrôle Astra a été correctement installé et terminer le "les étapes restantes" pour vous connecter. De plus, vous terminez le déploiement en effectuant également des opérations "tâches de configuration".

Installer le centre de contrôle Astra avec un système de stockage back-end Cloud Volumes ONTAP

Avec Astra Control Center, vous pouvez gérer les applications dans un environnement de cloud hybride avec des clusters Kubernetes et des instances Cloud Volumes ONTAP autogérés. Vous pouvez déployer Astra Control Center dans vos clusters Kubernetes sur site ou dans l'un des clusters Kubernetes autogéré dans l'environnement cloud.

Dans l'un de ces déploiements, vous pouvez effectuer des opérations de gestion des données d'application en utilisant Cloud Volumes ONTAP comme système back-end. Vous pouvez également configurer un compartiment S3 en tant que cible de sauvegarde.

Pour installer Astra Control Center dans Amazon Web Services (AWS), Google Cloud Platform (GCP) et Microsoft Azure avec un système back-end de stockage Cloud Volumes ONTAP, effectuez les opérations suivantes en fonction de votre environnement cloud.

- Déploiement d'Astra Control Center dans Amazon Web Services
- Déployez Astra Control Center dans Google Cloud Platform
- Déploiement d'Astra Control Center dans Microsoft Azure

Vous pouvez gérer vos applications dans des distributions avec des clusters Kubernetes autogérés, tels qu'OpenShift Container Platform (OCP). Seuls les clusters OCP autogérés sont validés pour le déploiement d'Astra Control Center.

Déploiement d'Astra Control Center dans Amazon Web Services

Vous pouvez déployer Astra Control Center sur un cluster Kubernetes autogéré, hébergé dans un cloud public Amazon Web Services (AWS).

Ce dont vous avez besoin pour AWS

Avant de déployer Astra Control Center dans AWS, vous aurez besoin des éléments suivants :

- Licence Astra Control Center. Voir "Exigences de licence d'Astra Control Center".
- "Découvrez les exigences d'Astra Control Center".
- Compte NetApp Cloud Central
- En cas d'utilisation des autorisations OCP, Red Hat OpenShift Container Platform (OCP) (au niveau de l'espace de noms pour créer des pods)
- Les identifiants AWS, l'ID d'accès et la clé secrète avec des autorisations qui vous permettent de créer des compartiments et des connecteurs
- Accès et connexion au registre d'instance de conteneur souple (ECR) du compte AWS
- Zone hébergée sur AWS et entrée route 53 nécessaires pour accéder à l'interface utilisateur de contrôle Astra

Exigences de l'environnement opérationnel pour AWS

Astra Control Center requiert l'environnement opérationnel suivant pour AWS :

- Red Hat OpenShift Container Platform 4.8

> Assurez-vous que l'environnement d'exploitation que vous choisissez d'héberger est conforme aux exigences de base en matière de ressources décrites dans la documentation officielle de l'environnement.

Le Centre de contrôle Astra requiert les ressources suivantes en plus des exigences de l'environnement en matière de ressources :

Composant	Conditions requises
Backend la capacité de stockage Cloud Volumes ONTAP	300 Go au moins disponibles
Nœuds workers (exigence AWS EC2)	Au moins 3 nœuds workers au total, avec 4 cœurs de vCPU et 12 Go de RAM chacun
Équilibrage de la charge	Type de service « LoadBalancer » disponible pour que le trafic d'entrée soit envoyé aux services du cluster d'environnement opérationnel
FQDN	Méthode permettant de pointer le FQDN de Astra Control Center vers l'adresse IP à charge équilibrée
Astra Trident (installé dans le cadre de la découverte du cluster Kubernetes dans NetApp Cloud Manager)	Astra Trident 21.04 ou version ultérieure installé et configuré et NetApp ONTAP 9.5 ou version ultérieure en tant que système de stockage back-end

Composant	Conditions requises
Registre d'images	Vous devez disposer d'un registre privé existant, comme AWS Elastic Container Registry, auquel vous pouvez pousser les images de création Astra Control Center. Vous devez fournir l'URL du registre d'images où vous allez télécharger les images. Le cluster hébergé par Astra Control Center et le cluster géré doivent avoir accès au même registre d'images pour pouvoir sauvegarder et restaurer des applications à l'aide de l'image Restic.
Configuration d'Astra Trident et ONTAP	Avec Astra Control Center, il est nécessaire de créer une classe de stockage et de la définir comme classe de stockage par défaut. L'Astra Control Center prend en charge les classes de stockage Kubernetes suivantes de ONTAP qui sont créées lorsque vous importez le cluster Kubernetes dans NetApp Cloud Manager. Découvrez Astra Trident : • `vsaworkingenvironment-<>-ha-nas csi.trident.netapp.io` • `vsaworkingenvironment-<>-ha-san csi.trident.netapp.io` • `vsaworkingenvironment-<>-single-nas csi.trident.netapp.io` • `vsaworkingenvironment-<>-single-san csi.trident.netapp.io`

De telles exigences supposent que Astra Control Center est la seule application qui s'exécute dans l'environnement opérationnel. Si l'environnement exécute des applications supplémentaires, ajustez ces exigences minimales en conséquence.

Le jeton de Registre AWS expire dans 12 heures. Après cela, vous devrez renouveler le code secret de Registre d'images Docker.

Présentation du déploiement pour AWS

Voici un aperçu du processus d'installation d'Astra Control Center pour AWS avec Cloud Volumes ONTAP en tant que système de stockage back-end.

Chacune de ces étapes est expliquée en détail ci-dessous.

1. Assurez-vous que vous disposez de suffisamment d'autorisations IAM.
2. Installez un cluster Red Hat OpenShift sur AWS.
3. Configurez AWS.
4. Configurez NetApp Cloud Manager.
5. Poser le centre de contrôle Astra.

Assurez-vous que vous disposez de suffisamment d'autorisations IAM

Assurez-vous de disposer de suffisamment de rôles et d'autorisations IAM pour installer un cluster RedHat OpenShift et un connecteur NetApp Cloud Manager.

Voir "Identifiants AWS initiaux".

Installez un cluster Red Hat OpenShift sur AWS

Installez un cluster Red Hat OpenShift Container Platform sur AWS.

Pour obtenir des instructions d'installation, reportez-vous à la section "Installation d'un cluster sur AWS dans OpenShift Container Platform".

Configurez AWS

Configurez ensuite AWS pour créer un réseau virtuel, configurez les instances de calcul EC2, créez un compartiment AWS S3, créez un registre d'objets élastiques (ECR) pour héberger les images d'Astra Control Center et envoyez les images dans ce registre.

Suivez la documentation AWS pour suivre la procédure ci-dessous. Voir "Documentation d'installation d'AWS".

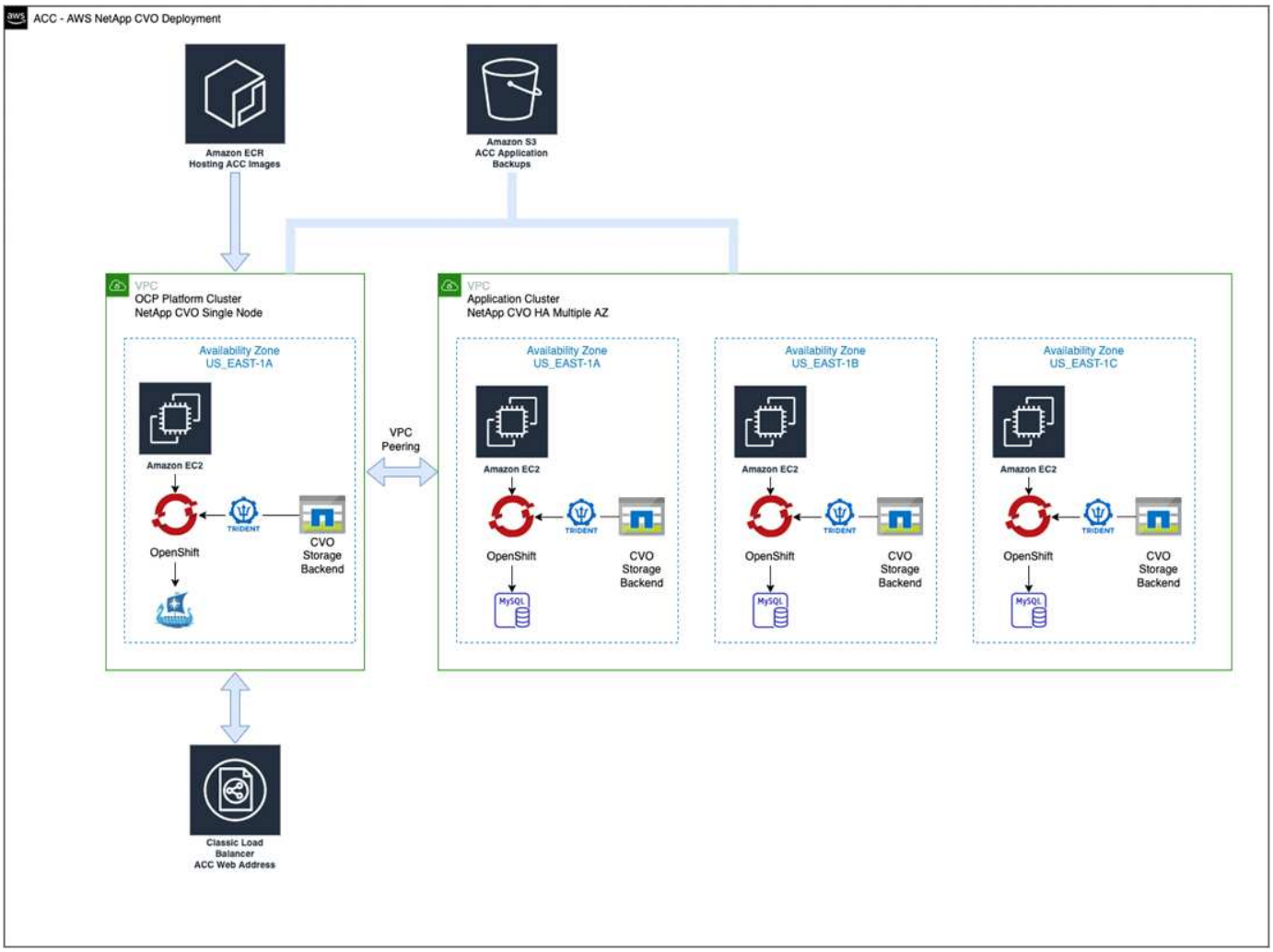
1. Créez un réseau virtuel AWS.
2. Vérifiez les instances de calcul EC2. Il peut s'agir d'un serveur bare Metal ou de machines virtuelles dans AWS.
3. Si le type d'instance ne correspond pas déjà aux exigences de ressources minimales Astra pour les nœuds maîtres et workers, modifiez le type d'instance dans AWS afin qu'il réponde aux exigences de l'Astra. Voir "Exigences du centre de contrôle Astra".
4. Créez au moins un compartiment AWS S3 pour stocker vos sauvegardes.
5. Créez un registre AWS Elastic Container (ECR) pour héberger toutes les images ACC.

> Si vous ne créez pas d'ECR, le centre de contrôle Astra ne peut pas accéder aux données de surveillance à partir d'un cluster contenant Cloud Volumes ONTAP avec un back-end AWS. Le problème survient lorsque le cluster que vous essayez de découvrir et de gérer à l'aide d'Astra Control Center n'a pas accès à AWS ECR.

6. Poussez les images ACC dans le registre défini.

> Le token AWS Elastic Container Registry (ECR) expire au bout de 12 heures et provoque l'échec des opérations de clonage inter-cluster. Ce problème survient lors de la gestion d'un système back-end à partir d'Cloud Volumes ONTAP configuré pour AWS. Pour corriger ce problème, authentifiez-vous à nouveau avec l'ECR et générez un nouveau secret pour que les opérations de clonage puissent reprendre avec succès.

Voici un exemple de déploiement AWS :



Configurez NetApp Cloud Manager

Avec Cloud Manager, créez un espace de travail, ajoutez un connecteur à AWS, créez un environnement de travail et importez le cluster.

Suivez la documentation de Cloud Manager pour effectuer les étapes suivantes. Voir les éléments suivants :

- "Mise en route de Cloud Volumes ONTAP dans AWS".
- "Créez un connecteur dans AWS à l'aide de Cloud Manager"

Étapes

1. Ajoutez vos identifiants à Cloud Manager.
2. Créez un espace de travail.
3. Ajoutez un connecteur pour AWS. Choisissez AWS en tant que fournisseur.
4. Créez un environnement de travail pour votre environnement cloud.
   a. Emplacement : « Amazon Web Services (AWS) »
   b. Type : « Cloud Volumes ONTAP HA »
5. Importer le cluster OpenShift Le cluster se connecte à l'environnement de travail que vous venez de créer.
   a. Pour en savoir plus sur le cluster NetApp, sélectionnez **K8s** > **liste des clusters** > **Détails du cluster**.

b. Notez la version Trident dans le coin supérieur droit.

c. Notez les classes de stockage du cluster Cloud Volumes ONTAP indiquant NetApp comme provisionneur.

   Cela importe votre cluster Red Hat OpenShift et lui attribue une classe de stockage par défaut. Vous sélectionnez la classe de stockage. Trident est automatiquement installé dans le cadre du processus d'importation et de détection.

6. Noter tous les volumes et volumes persistants sur ce déploiement Cloud Volumes ONTAP

> Cloud Volumes ONTAP peut fonctionner comme un seul nœud ou en mode haute disponibilité. Si la HA est activée, noter l'état de la HA et l'état du déploiement du nœud en cours dans AWS.

Poser le centre de contrôle Astra

Respectez la norme "Instructions d'installation du centre de contrôle Astra".

> AWS utilise le type de compartiment S3 générique.

Déployez Astra Control Center dans Google Cloud Platform

Vous pouvez déployer Astra Control Center sur un cluster Kubernetes autogéré, hébergé dans un cloud public Google Cloud Platform (GCP).

Éléments requis pour GCP

Avant de déployer Astra Control Center dans GCP, vous aurez besoin des éléments suivants :

- Licence Astra Control Center. Voir "Exigences de licence d'Astra Control Center".
- "Découvrez les exigences d'Astra Control Center".
- Compte NetApp Cloud Central
- Si vous utilisez OCP, Red Hat OpenShift Container Platform (OCP) 4.10
- En cas d'utilisation des autorisations OCP, Red Hat OpenShift Container Platform (OCP) (au niveau de l'espace de noms pour créer des pods)
- Compte de service GCP avec les autorisations qui vous permettent de créer des compartiments et des connecteurs

Exigences de l'environnement opérationnel pour GCP

> Assurez-vous que l'environnement d'exploitation que vous choisissez d'héberger est conforme aux exigences de base en matière de ressources décrites dans la documentation officielle de l'environnement.

Le Centre de contrôle Astra requiert les ressources suivantes en plus des exigences de l'environnement en matière de ressources :

Composant	Conditions requises
Backend la capacité de stockage Cloud Volumes ONTAP	300 Go au moins disponibles

Composant	Conditions requises
Nœuds workers (exigences de calcul GCP)	Au moins 3 nœuds workers au total, avec 4 cœurs de vCPU et 12 Go de RAM chacun
Équilibrage de la charge	Type de service « LoadBalancer » disponible pour que le trafic d'entrée soit envoyé aux services du cluster d'environnement opérationnel
FQDN (ZONE DNS GCP)	Méthode permettant de pointer le FQDN de Astra Control Center vers l'adresse IP à charge équilibrée
Astra Trident (installé dans le cadre de la découverte du cluster Kubernetes dans NetApp Cloud Manager)	Astra Trident 21.04 ou version ultérieure installé et configuré et NetApp ONTAP 9.5 ou version ultérieure en tant que système de stockage back-end
Registre d'images	Vous devez disposer d'un registre privé existant, tel que le registre de conteneurs Google, auquel vous pouvez pousser les images de création d'Astra Control Center. Vous devez fournir l'URL du registre d'images où vous allez télécharger les images. Vous devez activer l'accès anonyme pour extraire les images Restic pour les sauvegardes.
Configuration d'Astra Trident et ONTAP	Avec Astra Control Center, il est nécessaire de créer une classe de stockage et de la définir comme classe de stockage par défaut. L'Astra Control Center prend en charge les classes de stockage Kubernetes suivantes de ONTAP qui sont créées lorsque vous importez le cluster Kubernetes dans NetApp Cloud Manager. Découvrez Astra Trident : • `vsaworkingenvironment-<>-ha-nas csi.trident.netapp.io` • `vsaworkingenvironment-<>-ha-san csi.trident.netapp.io` • `vsaworkingenvironment-<>-single-nas csi.trident.netapp.io` • `vsaworkingenvironment-<>-single-san csi.trident.netapp.io`

De telles exigences supposent que Astra Control Center est la seule application qui s'exécute dans l'environnement opérationnel. Si l'environnement exécute des applications supplémentaires, ajustez ces exigences minimales en conséquence.

Présentation du déploiement pour GCP

Voici un aperçu du processus d'installation d'Astra Control Center sur un cluster OCP autogéré dans GCP avec Cloud Volumes ONTAP comme système de stockage principal.

Chacune de ces étapes est expliquée en détail ci-dessous.

1. Installez un cluster Red Hat OpenShift sur GCP.
2. Création d'un projet GCP et d'un cloud privé virtuel.
3. Assurez-vous que vous disposez de suffisamment d'autorisations IAM.
4. Configurez GCP.
5. Configurez NetApp Cloud Manager.
6. Installer et configurer le centre de contrôle Astra.

Installez un cluster Red Hat OpenShift sur GCP

La première étape consiste à installer un cluster Red Hat OpenShift sur GCP.

Pour les instructions d'installation, reportez-vous aux sections suivantes :

- "Installation d'un cluster OpenShift dans GCP"
- "Création d'un compte de service GCP"

Création d'un projet GCP et d'un cloud privé virtuel

Créez au moins un projet GCP et un cloud privé virtuel (VPC).

> OpenShift peut créer ses propres groupes de ressources. En plus de ces VPC, vous devez également définir un VPC GCP. Voir la documentation OpenShift.

Vous pouvez créer un groupe de ressources de cluster de plate-forme et un groupe de ressources de cluster OpenShift d'application cible.

Assurez-vous que vous disposez de suffisamment d'autorisations IAM

Assurez-vous de disposer de suffisamment de rôles et d'autorisations IAM pour installer un cluster RedHat OpenShift et un connecteur NetApp Cloud Manager.

Voir "Identifiants et autorisations GCP initiaux".

Configurez GCP

Configurez ensuite GCP pour créer un VPC, configurez des instances de calcul, créez un stockage objet Google Cloud, créez un registre de conteneurs Google pour héberger les images d'Astra Control Center et envoyez les images vers ce registre.

Suivez la documentation GCP pour effectuer les étapes suivantes. Voir installation du cluster OpenShift dans GCP.

1. Créez un projet GCP et un VPC dans le GCP que vous prévoyez d'utiliser pour le cluster OCP avec le backend CVO.
2. Vérifiez les instances de calcul. Il peut s'agir d'un serveur bare Metal ou de machines virtuelles dans GCP.
3. Si le type d'instance ne correspond pas déjà aux exigences de ressources minimales Astra pour les nœuds maîtres et workers, modifiez le type d'instance dans GCP afin qu'il réponde aux exigences de l'Astra. Voir "Exigences du centre de contrôle Astra".
4. Créez au moins un compartiment de stockage cloud GCP pour stocker vos sauvegardes.

5. Créez un secret, requis pour l'accès au compartiment.
6. Créez un registre de conteneurs Google pour héberger toutes les images du centre de contrôle Astra.
7. Configurez l'accès du registre de conteneurs Google pour le transfert/transfert de Docker pour toutes les images du centre de contrôle Astra.

   Exemple : les images ACC peuvent être transmises à ce registre en entrant le script suivant :

   ```
   gcloud auth activate-service-account <service account email address>
   --key-file=<GCP Service Account JSON file>
   ```

   Ce script nécessite un fichier manifeste Astra Control Center et votre emplacement dans le registre d'images Google.

   Exemple :

   ```
   manifestfile=astra-control-center-<version>.manifest
   GCP_CR_REGISTRY=<target image repository>
   ASTRA_REGISTRY=<source ACC image repository>

   while IFS= read -r image; do
       echo "image: $ASTRA_REGISTRY/$image $GCP_CR_REGISTRY/$image"
       root_image=${image%:*}
       echo $root_image
       docker pull $ASTRA_REGISTRY/$image
       docker tag $ASTRA_REGISTRY/$image $GCP_CR_REGISTRY/$image
       docker push $GCP_CR_REGISTRY/$image
   done < astra-control-center-22.04.41.manifest
   ```

8. Configurer les zones DNS.

Configurez NetApp Cloud Manager

Avec Cloud Manager, créez un espace de travail, ajoutez un connecteur à GCP, créez un environnement de travail et importez le cluster.

Suivez la documentation de Cloud Manager pour effectuer les étapes suivantes. Voir "Mise en route de Cloud Volumes ONTAP dans GCP".

Ce dont vous avez besoin

- Accès au compte de services GCP avec les autorisations IAM et les rôles requis

Étapes

1. Ajoutez vos identifiants à Cloud Manager. Voir "Ajout de comptes GCP".
2. Ajoutez un connecteur pour GCP.
   a. Choisissez GCP comme fournisseur.
   b. Entrez les identifiants GCP. Voir "Création d'un connecteur dans GCP à partir de Cloud Manager".

   c. S'assurer que le connecteur est en marche et basculer vers ce connecteur.
3. Créez un environnement de travail pour votre environnement cloud.
   a. Emplacement : « GCP »
   b. Type : « Cloud Volumes ONTAP HA »
4. Importer le cluster OpenShift Le cluster se connecte à l'environnement de travail que vous venez de créer.
   a. Pour en savoir plus sur le cluster NetApp, sélectionnez **K8s** > **liste des clusters** > **Détails du cluster**.
   b. Notez la version Trident dans le coin supérieur droit.
   c. Notez les classes de stockage du cluster Cloud Volumes ONTAP indiquant « NetApp » comme provisionneur.

      Cela importe votre cluster Red Hat OpenShift et lui attribue une classe de stockage par défaut. Vous sélectionnez la classe de stockage. Trident est automatiquement installé dans le cadre du processus d'importation et de détection.
5. Noter tous les volumes et volumes persistants sur ce déploiement Cloud Volumes ONTAP

> Cloud Volumes ONTAP peut fonctionner comme un seul nœud ou en haute disponibilité. Si la haute disponibilité est activée, notez l'état de la haute disponibilité et l'état du déploiement des nœuds exécutés dans GCP.

Poser le centre de contrôle Astra

Respectez la norme "Instructions d'installation du centre de contrôle Astra".

> GCP utilise le type de compartiment S3 générique.

1. Générez le secret Docker pour extraire des images pour l'installation du centre de contrôle Astra :

```
kubectl create secret docker-registry <secret name>
--docker-server=<Registry location>
--docker-username=_json_key
--docker-password="$(cat <GCP Service Account JSON file>)"
--namespace=pcloud
```

Déploiement d'Astra Control Center dans Microsoft Azure

Vous pouvez déployer Astra Control Center sur un cluster Kubernetes autogéré, hébergé dans un cloud public Microsoft Azure.

Ce dont vous avez besoin pour Azure

Avant de déployer Astra Control Center dans Azure, vous aurez besoin des éléments suivants :

- Licence Astra Control Center. Voir "Exigences de licence d'Astra Control Center".
- "Découvrez les exigences d'Astra Control Center".
- Compte NetApp Cloud Central

- Si vous utilisez OCP, Red Hat OpenShift Container Platform (OCP) 4.8
- En cas d'utilisation des autorisations OCP, Red Hat OpenShift Container Platform (OCP) (au niveau de l'espace de noms pour créer des pods)
- Les identifiants Azure avec autorisations qui vous permettent de créer des compartiments et des connecteurs

Exigences de l'environnement opérationnel pour Azure

Assurez-vous que l'environnement d'exploitation que vous choisissez d'héberger est conforme aux exigences de base en matière de ressources décrites dans la documentation officielle de l'environnement.

Le Centre de contrôle Astra requiert les ressources suivantes en plus des exigences de l'environnement en matière de ressources :

Voir "Exigences relatives à l'environnement opérationnel d'Astra Control Center".

Composant	Conditions requises
Backend la capacité de stockage Cloud Volumes ONTAP	300 Go au moins disponibles
Nœuds worker (exigences de calcul Azure)	Au moins 3 nœuds workers au total, avec 4 cœurs de vCPU et 12 Go de RAM chacun
Équilibrage de la charge	Type de service « LoadBalancer » disponible pour que le trafic d'entrée soit envoyé aux services du cluster d'environnement opérationnel
FQDN (zone Azure DNS)	Méthode permettant de pointer le FQDN de Astra Control Center vers l'adresse IP à charge équilibrée
Astra Trident (installé dans le cadre de la découverte du cluster Kubernetes dans NetApp Cloud Manager)	Astra Trident 21.04 ou version ultérieure installé et configuré et NetApp ONTAP version 9.5 ou ultérieure sera utilisé comme système de stockage back-end
Registre d'images	Vous devez disposer d'un registre privé existant, tel que le registre de conteneur Azure (ACR), auquel vous pouvez pousser les images de création d'Astra Control Center. Vous devez fournir l'URL du registre d'images où vous allez télécharger les images. Vous devez activer l'accès anonyme pour extraire les images Restic pour les sauvegardes.

Composant	Conditions requises
Configuration d'Astra Trident et ONTAP	Avec Astra Control Center, il est nécessaire de créer une classe de stockage et de la définir comme classe de stockage par défaut. L'Astra Control Center prend en charge les classes de stockage Kubernetes suivantes de ONTAP qui sont créées lorsque vous importez le cluster Kubernetes dans NetApp Cloud Manager. Découvrez Astra Trident : • `vsaworkingenvironment-<>-ha-nas csi.trident.netapp.io` • `vsaworkingenvironment-<>-ha-san csi.trident.netapp.io` • `vsaworkingenvironment-<>-single-nas csi.trident.netapp.io` • `vsaworkingenvironment-<>-single-san csi.trident.netapp.io`

De telles exigences supposent que Astra Control Center est la seule application qui s'exécute dans l'environnement opérationnel. Si l'environnement exécute des applications supplémentaires, ajustez ces exigences minimales en conséquence.

Présentation du déploiement pour Azure

Voici un aperçu du processus d'installation d'Astra Control Center pour Azure.

Chacune de ces étapes est expliquée en détail ci-dessous.

1. Installez un cluster Red Hat OpenShift sur Azure.
2. Créez des groupes de ressources Azure.
3. Assurez-vous que vous disposez de suffisamment d'autorisations IAM.
4. Configurez Azure.
5. Configurez NetApp Cloud Manager.
6. Installer et configurer le centre de contrôle Astra.

Installez un cluster Red Hat OpenShift sur Azure

La première étape consiste à installer un cluster Red Hat OpenShift sur Azure.

Pour obtenir des instructions d'installation, reportez-vous à la documentation RedHat sur "Installation du cluster OpenShift sur Azure" et "Installation d'un compte Azure".

Créez des groupes de ressources Azure

Créez au moins un groupe de ressources Azure.

OpenShift peut créer ses propres groupes de ressources. En plus de ces groupes, vous devez également définir des groupes de ressources Azure. Voir la documentation OpenShift.

Vous pouvez créer un groupe de ressources de cluster de plate-forme et un groupe de ressources de cluster OpenShift d'application cible.

Assurez-vous que vous disposez de suffisamment d'autorisations IAM

Assurez-vous de disposer de suffisamment de rôles et d'autorisations IAM pour installer un cluster RedHat OpenShift et un connecteur NetApp Cloud Manager.

Voir "Identifiants et autorisations Azure".

Configurez Azure

Configurez ensuite Azure pour créer un réseau virtuel, configurez des instances de calcul, créez un conteneur Azure Blob Container Register, créez un ACR (Azure Container Register) pour héberger les images d'Astra Control Center et envoyez les images dans ce registre.

Suivez la documentation Azure pour suivre les étapes ci-dessous. Voir "Installation du cluster OpenShift sur Azure".

1. Créez un réseau virtuel Azure.
2. Vérifiez les instances de calcul. Il peut s'agir d'un serveur bare Metal ou de machines virtuelles dans Azure.
3. Si le type d'instance ne correspond pas déjà aux exigences de ressources minimales Astra pour les nœuds maîtres et workers, modifiez le type d'instance dans Azure afin qu'il réponde aux exigences de l'Astra. Voir "Exigences du centre de contrôle Astra".
4. Créez au moins un conteneur Azure Blob pour stocker vos sauvegardes.
5. Créez un compte de stockage. Vous aurez besoin d'un compte de stockage pour créer un conteneur à utiliser comme compartiment dans Astra Control Center.
6. Créez un secret, requis pour l'accès au compartiment.
7. Créez un registre de conteneurs Azure (ACR) pour héberger toutes les images du centre de contrôle Astra.
8. Configurer l'accès ACR pour Docker pousser/extraire toutes les images du centre de contrôle Astra.
9. Envoyez les images ACC dans ce registre en entrant le script suivant :

   ```
   az acr login -n <AZ ACR URL/Location>
   This script requires ACC manifest file and your Azure ACR location.
   ```

   Exemple :

```
manifestfile=astra-control-center-<version>.manifest
AZ_ACR_REGISTRY=<target image repository>
ASTRA_REGISTRY=<source ACC image repository>

while IFS= read -r image; do
    echo "image: $ASTRA_REGISTRY/$image $AZ_ACR_REGISTRY/$image"
    root_image=${image%:*}
    echo $root_image
    docker pull $ASTRA_REGISTRY/$image
    docker tag $ASTRA_REGISTRY/$image $AZ_ACR_REGISTRYY/$image
    docker push $AZ_ACR_REGISTRY/$image
done < astra-control-center-22.04.41.manifest
```

10. Configurer les zones DNS.

Configurez NetApp Cloud Manager

Avec Cloud Manager, créez un espace de travail, ajoutez un connecteur à Azure, créez un environnement de travail et importez le cluster.

Suivez la documentation de Cloud Manager pour effectuer les étapes suivantes. Voir "Mise en route de Cloud Manager dans Azure".

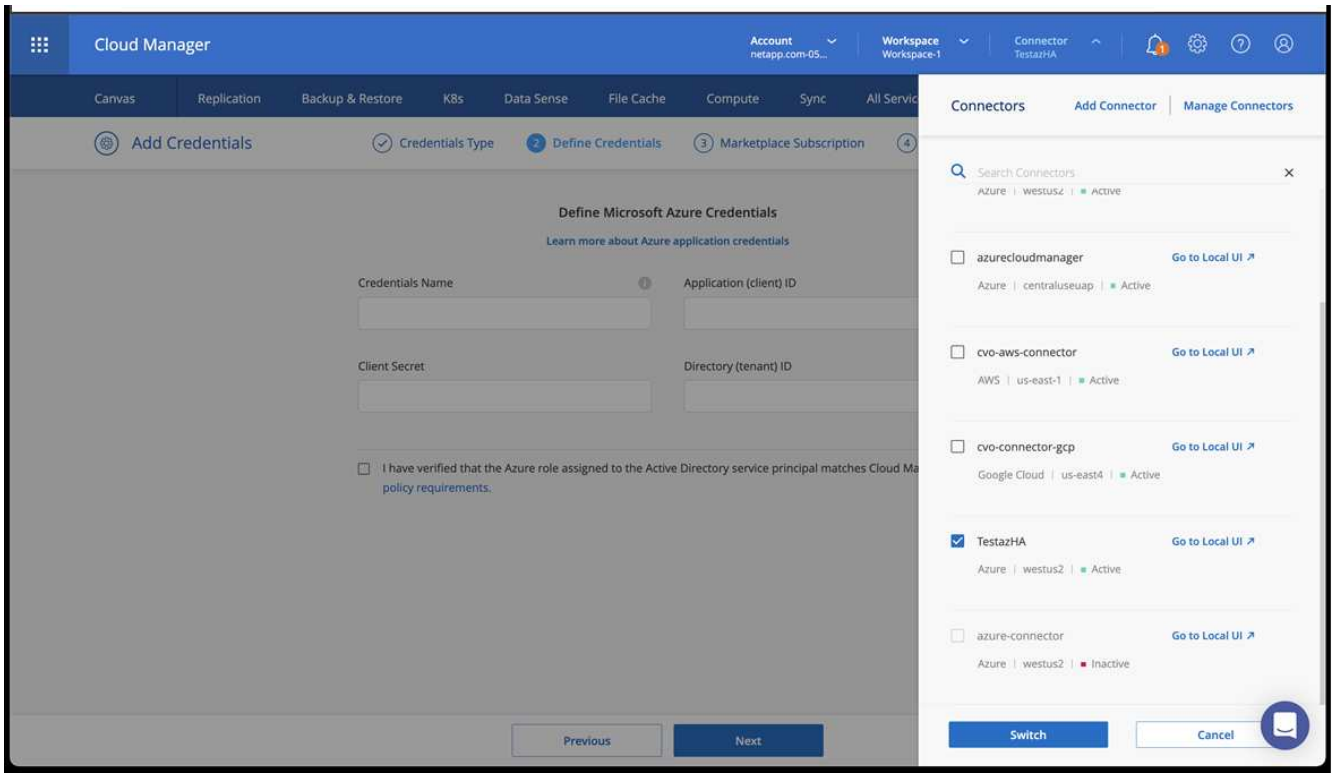
Ce dont vous avez besoin

Accès au compte Azure avec les autorisations IAM et les rôles requis

Étapes

1. Ajoutez vos identifiants à Cloud Manager.
2. Ajoutez un connecteur pour Azure. Voir "Règles de Cloud Manager".
   a. Choisissez **Azure** comme fournisseur.
   b. Vous pouvez entrer les identifiants Azure, notamment l'ID de l'application, le secret client et l'ID du répertoire (locataire).

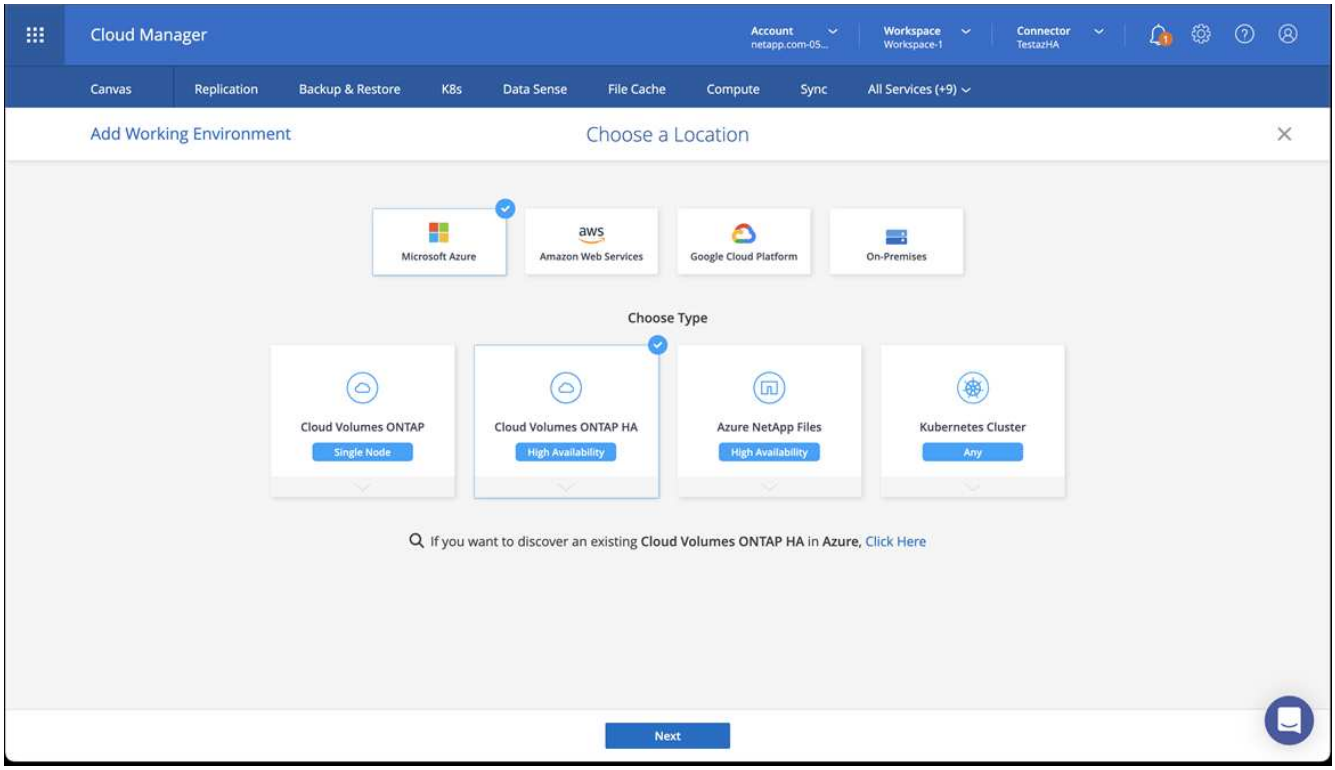
      Voir "Création d'un connecteur dans Azure à partir de Cloud Manager".
3. S'assurer que le connecteur est en marche et basculer vers ce connecteur.



4. Créez un environnement de travail pour votre environnement cloud.

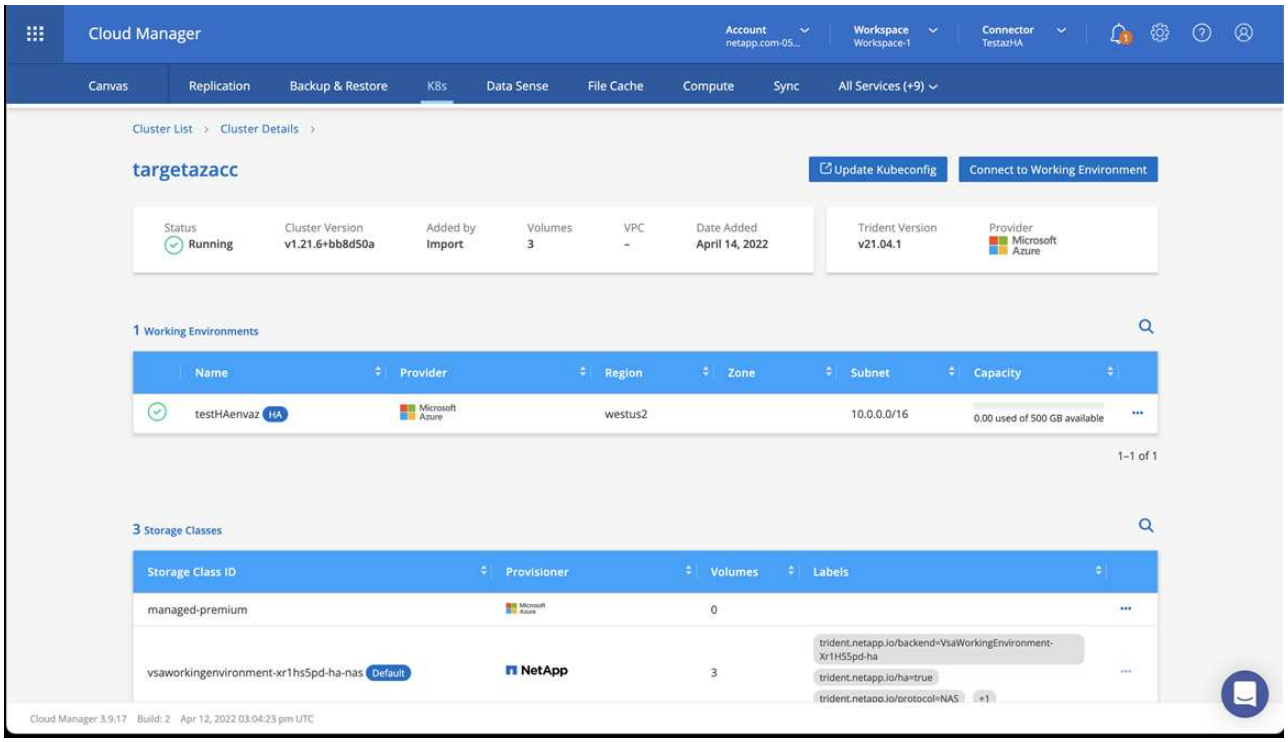
   a. Emplacement : « Microsoft Azure ».

   b. Type : « Cloud Volumes ONTAP HA ».



5. Importer le cluster OpenShift Le cluster se connecte à l'environnement de travail que vous venez de créer.

   a. Pour en savoir plus sur le cluster NetApp, sélectionnez **K8s** > **liste des clusters** > **Détails du cluster**.



b. Notez la version Trident dans le coin supérieur droit.

c. Notez les classes de stockage du cluster Cloud Volumes ONTAP indiquant NetApp comme provisionneur.

Cela importe votre cluster Red Hat OpenShift et attribue une classe de stockage par défaut. Vous sélectionnez la classe de stockage. Trident est automatiquement installé dans le cadre du processus d'importation et de détection.

6. Noter tous les volumes et volumes persistants sur ce déploiement Cloud Volumes ONTAP
7. Cloud Volumes ONTAP peut fonctionner comme un seul nœud ou en mode haute disponibilité. Si la HA est activée, noter l'état de la HA et l'état du déploiement du nœud en cours d'exécution dans Azure.

Installer et configurer le centre de contrôle Astra

Installer le centre de contrôle Astra de série "instructions d'installation".

Avec Astra Control Center, ajoutez un compartiment Azure. Voir "Configurer le centre de contrôle Astra et ajouter des seaux".

Informations sur le copyright

Copyright © 2023 NetApp, Inc. Tous droits réservés. Imprimé aux États-Unis. Aucune partie de ce document protégé par copyright ne peut être reproduite sous quelque forme que ce soit ou selon quelque méthode que ce soit (graphique, électronique ou mécanique, notamment par photocopie, enregistrement ou stockage dans un système de récupération électronique) sans l'autorisation écrite préalable du détenteur du droit de copyright.

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

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

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

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

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

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

Informations sur les marques commerciales

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