# Información general de la instalación

Astra Control Center

NetApp
November 21, 2023

# Tabla de contenidos

# Información general de la instalación

Elija y complete uno de los siguientes procedimientos de instalación de Astra Control Center:

- "Instale Astra Control Center mediante el proceso estándar"
- "(Si utiliza Red Hat OpenShift) instale Astra Control Center mediante OpenShift OperatorHub"
- "Instale Astra Control Center con un back-end de almacenamiento de Cloud Volumes ONTAP"

# Instale Astra Control Center mediante el proceso estándar

Para instalar Astra Control Center, descargue el paquete de instalación desde el sitio de soporte de NetApp y realice los siguientes pasos para instalar Astra Control Center Operator y Astra Control Center en su entorno. Puede utilizar este procedimiento para instalar Astra Control Center en entornos conectados a Internet o con conexión por aire.

Para entornos Red Hat OpenShift, también puede utilizar un "procedimiento alternativo" Para instalar Astra Control Center con OpenShift OperatorHub.

**Lo que necesitará**

- "Antes de comenzar la instalación, prepare su entorno para la implementación de Astra Control Center".
- Asegurarse de que todos los operadores del clúster se encuentren en estado correcto y estén disponibles.

  Ejemplo de OpenShift:

  ```
  oc get clusteroperators
  ```

- Asegúrese de que todos los servicios de API se encuentren en buen estado y estén disponibles:

  Ejemplo de OpenShift:

  ```
  oc get apiservices
  ```

- El FQDN de Astra que va a utilizar debe poder enrutar a este clúster. Esto significa que tiene una entrada DNS en el servidor DNS interno o que está utilizando una ruta URL principal que ya está registrada.

**Acerca de esta tarea**

El proceso de instalación de Astra Control Center realiza lo siguiente:

- Instala los componentes de Astra en `netapp-acc` (o nombre personalizado).
- Crea una cuenta predeterminada.
- Establece una dirección de correo electrónico de usuario administrativo predeterminada y una contraseña única predeterminada de `ACC-<UUID_of_installation>` Por este ejemplo de Astra Control Center. A este usuario se le asigna el rol de propietario del sistema y es necesario iniciar sesión por primera vez en la interfaz de usuario.
- Le ayuda a determinar que se están ejecutando todas las pods de Astra Control Center.
- Instala la interfaz de usuario de Astra.

(Se aplica sólo a la versión Astra Data Store Early Access Program (EAP)) Si tiene intención de gestionar Astra Data Store mediante Astra Control Center y habilitar los flujos de trabajo de VMware, implemente Astra Control Center únicamente en `pcloud` espacio de nombres y no en `netapp-acc` espacio de nombres o un espacio de nombres personalizado que se describe en los pasos de este procedimiento.

No ejecute el siguiente comando durante todo el proceso de instalación para evitar eliminar todas las POD de Astra Control Center: `kubectl delete -f astra_control_center_operator_deploy.yaml`

Si utiliza Podman de Red Hat en lugar de Docker Engine, los comandos de Podman se pueden utilizar en lugar de los comandos de Docker.

**Pasos**

Para instalar Astra Control Center, lleve a cabo los siguientes pasos:

- Descargue y desembale el paquete Astra Control Center
- Instale el complemento Astra kubectl de NetApp
- Agregue las imágenes al registro local
- Configurar espacio de nombres y secreto para registros con requisitos de autenticación
- Instale el operador de Astra Control Center
- Configurar Astra Control Center
- Complete la instalación del centro de control de Astra y del operador
- Comprobar el estado del sistema
- Configure la entrada para el equilibrio de carga
- Inicie sesión en la interfaz de usuario de Astra Control Center

## Descargue y desembale el paquete Astra Control Center

1. Descargue el paquete Astra Control Center (`astra-control-center-[version].tar.gz`) del "Sitio de soporte de NetApp".
2. Descargue el archivo zip de los certificados y claves de Astra Control Center de "Sitio de soporte de NetApp".
3. (Opcional) Use el siguiente comando para verificar la firma del paquete:

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

4. Extraiga las imágenes:

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

## Instale el complemento Astra kubectl de NetApp

La Astra de NetApp `kubectl` El complemento de línea de comandos ahorra tiempo al realizar tareas comunes asociadas con la implementación y actualización de Astra Control Center.

### Lo que necesitará

NetApp proporciona binarios para el complemento para distintas arquitecturas de CPU y sistemas operativos. Debe saber qué CPU y sistema operativo tiene antes de realizar esta tarea. En los sistemas operativos Linux y Mac, puede utilizar `uname -a` comando para recopilar esta información.

### Pasos

1. Enumere la Astra de NetApp disponible `kubectl` Haga un complemento para binarios y anote el nombre del archivo que necesita para su sistema operativo y arquitectura de CPU:

```
ls kubectl-astra/
```

2. Copie el archivo en la misma ubicación que el estándar `kubectl` utilidad. En este ejemplo, la `kubectl` la utilidad se encuentra en `/usr/local/bin` directorio. Sustituya `<binary-name>` con el nombre del archivo que necesita:

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

## Agregue las imágenes al registro local

1. Cambie al directorio Astra:

```
cd acc
```

2. Agregue los archivos del directorio imagen de Astra Control Center al registro local.

> Consulte secuencias de comandos de ejemplo para la carga automática de imágenes a continuación.

   a. Inicie sesión en su registro:

      Docker:

```
docker login [your_registry_path]
```

      Podman:

```
podman login [your_registry_path]
```

   b. Utilice la secuencia de comandos adecuada para cargar las imágenes, etiquetar las imágenes y

empuje las imágenes en el registro local:

Docker:

```
export REGISTRY=[Docker_registry_path]
for astraImageFile in $(ls images/*.tar) ; do
  # Load to local cache. And store the name of the loaded image
trimming the 'Loaded images: '
  astraImage=$(docker load --input ${astraImageFile} | sed 's/Loaded
image: //')
  astraImage=$(echo ${astraImage} | sed 's!localhost/!!')
  # Tag with local image repo.
  docker tag ${astraImage} ${REGISTRY}/${astraImage}
  # Push to the local repo.
  docker push ${REGISTRY}/${astraImage}
done
```

Podman:

```
export REGISTRY=[Registry_path]
for astraImageFile in $(ls images/*.tar) ; do
  # Load to local cache. And store the name of the loaded image trimming
the 'Loaded images: '
  astraImage=$(podman load --input ${astraImageFile} | sed 's/Loaded
image(s): //')
  astraImage=$(echo ${astraImage} | sed 's!localhost/!!')
  # Tag with local image repo.
  podman tag ${astraImage} ${REGISTRY}/${astraImage}
  # Push to the local repo.
  podman push ${REGISTRY}/${astraImage}
done
```

## Configurar espacio de nombres y secreto para registros con requisitos de autenticación

1. Si utiliza un registro que requiere autenticación, debe hacer lo siguiente:

   a. Cree el `netapp-acc-operator` espacio de nombres:

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

   Respuesta:

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

b. Cree un secreto para `netapp-acc-operator` espacio de nombres. Añada información sobre Docker y ejecute el siguiente comando:

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

Respuesta de ejemplo:

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

c. Cree el `netapp-acc` (o espacio de nombres personalizado).

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

Respuesta de ejemplo:

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

d. Cree un secreto para `netapp-acc` (o espacio de nombres personalizado). Añada información sobre Docker y ejecute el siguiente comando:

```
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]
```

Respuesta

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

a. [[substep_kubeconfig_secret](opcional) Si desea que el clúster sea gestionado automáticamente por Astra Control Center después de la instalación, asegúrese de proporcionar el kubeconfig como secreto dentro del espacio de nombres Astra Control Center que tiene intención de implementar utilizando este comando:

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

## Instale el operador de Astra Control Center

1. Edite la implementación del operador de Astra Control Center YAML (`astra_control_center_operator_deploy.yaml`) para referirse a su registro local y secreto.

```
vim astra_control_center_operator_deploy.yaml
```

   a. Si utiliza un registro que requiere autenticación, reemplace la línea predeterminada de `imagePullSecrets: []` con lo siguiente:

```
imagePullSecrets:
- name: <name_of_secret_with_creds_to_local_registry>
```

   b. Cambiar `[your_registry_path]` para la `kube-rbac-proxy` imagen a la ruta del registro en la que se insertó la imagen en un paso anterior.

   c. Cambiar `[your_registry_path]` para la `acc-operator-controller-manager` imagen a la ruta del registro en la que se insertó la imagen en un paso anterior.

   d. (Para instalaciones que utilizan la vista previa de Astra Data Store) Consulte este problema conocido con respecto a "Los aprovisionadores de clases de almacenamiento y los cambios adicionales que deberá realizar en la 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: []
```

2. Instale el operador de Astra Control Center:

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

Respuesta de ejemplo:

```
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
```

## Configurar Astra Control Center

1. Edite el archivo de recursos personalizados (CR) del Centro de control de Astra (`astra_control_center_min.yaml`) Para realizar las configuraciones de cuenta, AutoSupport, Registro y otras necesarias:

> Si se requieren personalizaciones adicionales para su entorno, puede utilizar `astra_control_center.yaml` Como CR alternativo. `astra_control_center_min.yaml` Es la CR predeterminada y es adecuada para la mayoría de las instalaciones.

```
vim astra_control_center_min.yaml
```

> Las propiedades configuradas por la CR no se pueden cambiar tras la implementación inicial de Astra Control Center.

> Si está utilizando un registro que no requiere autorización, debe eliminar `secret` línea dentro `imageRegistry` o se producirá un error en la instalación.

a. Cambiar `[your_registry_path]` a la ruta de acceso del registro en la que ha insertado las imágenes en el paso anterior.

b. Cambie el `accountName` cadena al nombre que desea asociar a la cuenta.

c. Cambie el `astraAddress` Cadena al FQDN que desea utilizar en su navegador para acceder a Astra. No utilizar `http://` o. `https://` en la dirección. Copie este FQDN para utilizarlo en un paso

posterior.

d. Cambie el `email` cadena en la dirección inicial predeterminada del administrador. Copie esta dirección de correo electrónico para su uso en un paso posterior.

e. Cambiar `enrolled` Para AutoSupport a. `false` para sitios sin conexión a internet o retención `true` para sitios conectados.

f. (Opcional) Añada un nombre `firstName` y apellidos `lastName` del usuario asociado con la cuenta. Este paso se puede realizar ahora o una versión posterior dentro de la interfaz de usuario.

g. (Opcional) cambie el `storageClass` Valor en otro recurso de la clase de almacenamiento de Trident, si es necesario para su instalación.

h. (Opcional) Si desea que el clúster sea gestionado automáticamente por Astra Control Center después de la instalación y ya lo tiene se ha creado el secreto que contiene el kubeconfig para este cluster, Proporcione el nombre del secreto agregando un nuevo campo a este archivo YLMA llamado `astraKubeConfigSecret: "acc-kubeconfig-cred or custom secret name"`

i. Realice uno de los siguientes pasos:

   - **Otro controlador de entrada (ingressType:Generic)**: Esta es la acción predeterminada con Astra Control Center. Después de implementar Astra Control Center, deberá configurar el controlador Ingress para exponer Astra Control Center con una dirección URL.

     La instalación predeterminada de Astra Control Center configura su puerta de enlace (`service/traefik`) ser del tipo `ClusterIP`. Esta instalación predeterminada requiere que configure además un dispositivo de entrada/controlador de Kubernetes para enrutar el tráfico hacia él. Si desea utilizar una entrada, consulte "Configure la entrada para el equilibrio de carga".

   - **Equilibrador de carga de servicio (ingressType:AccTraefik)**: Si no desea instalar un controlador IngressController o crear un recurso de entrada, establezca `ingressType` para `AccTraefik`.

     Esto despliega el Astra Control Center `traefik` Puerta de enlace como servicio de tipo Kubernetes LoadBalancer.

     Astra Control Center utiliza un servicio del tipo "LoadBalancer" (`svc/traefik` En el espacio de nombres de Astra Control Center) y requiere que se le asigne una dirección IP externa accesible. Si se permiten equilibradores de carga en su entorno y no tiene uno configurado, puede utilizar MetalLB u otro equilibrador de carga de servicio externo para asignar una dirección IP externa al servicio. En la configuración interna del servidor DNS, debe apuntar el nombre DNS elegido para Astra Control Center a la dirección IP con equilibrio de carga.

     Para obtener más información sobre el tipo de servicio de "LoadBalancer" y la entrada, consulte "Requisitos".

```
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"
```

## Complete la instalación del centro de control de Astra y del operador

1. Si todavía no lo ha hecho en un paso anterior, cree el `netapp-acc` espacio de nombres (o personalizado):

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

   Respuesta de ejemplo:

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

2. Instale Astra Control Center en `netapp-acc` (o su espacio de nombres personalizado):

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

   Respuesta de ejemplo:

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

## Comprobar el estado del sistema

> Si prefiere utilizar OpenShift, puede utilizar comandos de OC comparables para realizar los pasos de verificación.

1. Compruebe que todos los componentes del sistema se han instalado correctamente.

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

Cada pod debe tener el estado de `Running`. Pueden tardar varios minutos en implementar los pods del sistema.

Respuesta de ejemplo:

```
NAME                                        READY   STATUS    RESTARTS
AGE
acc-helm-repo-5f75c5f564-bzqmt              1/1     Running   0
11m
activity-6b8f7cccb9-mlrn4                   1/1     Running   0
9m2s
api-token-authentication-6hznt              1/1     Running   0
8m50s
api-token-authentication-qpfgb              1/1     Running   0
8m50s
api-token-authentication-sqnb7              1/1     Running   0
8m50s
asup-5578bbdd57-dxkbp                       1/1     Running   0
9m3s
authentication-56bff4f95d-mspmq             1/1     Running   0
7m31s
bucketservice-6f7968b95d-9rrrl              1/1     Running   0
8m36s
cert-manager-5f6cf4bc4b-82khn               1/1     Running   0
6m19s
cert-manager-cainjector-76cf976458-sdrbc    1/1     Running   0
6m19s
cert-manager-webhook-5b7896bfd8-2n45j       1/1     Running   0
6m19s
cloud-extension-749d9f684c-8bdhq            1/1     Running   0
9m6s
cloud-insights-service-7d58687d9-h5tzw      1/1     Running   2
8m56s
composite-compute-968c79cb5-nv7l4           1/1     Running   0
9m11s
composite-volume-7687569985-jg9gg           1/1     Running   0
8m33s
```

```
credentials-5c9b75f4d6-nx9cz                1/1     Running   0
8m42s
entitlement-6c96fd8b78-zt7f8                1/1     Running   0
8m28s
features-5f7bfc9f68-gsjnl                   1/1     Running   0
8m57s
fluent-bit-ds-h88p7                         1/1     Running   0
7m22s
fluent-bit-ds-krhnj                         1/1     Running   0
7m23s
fluent-bit-ds-l5bjj                         1/1     Running   0
7m22s
fluent-bit-ds-lrclb                         1/1     Running   0
7m23s
fluent-bit-ds-s5t4n                         1/1     Running   0
7m23s
fluent-bit-ds-zpr6v                         1/1     Running   0
7m22s
graphql-server-5f5976f4bd-vbb4z             1/1     Running   0
7m13s
identity-56f78b8f9f-8h9p9                   1/1     Running   0
8m29s
influxdb2-0                                 1/1     Running   0
11m
krakend-6f8d995b4d-5khkl                    1/1     Running   0
7m7s
license-5b5db87c97-jmxzc                    1/1     Running   0
9m
login-ui-57b57c74b8-6xtv7                   1/1     Running   0
7m10s
loki-0                                      1/1     Running   0
11m
monitoring-operator-9dbc9c76d-8znck         2/2     Running   0
7m33s
nats-0                                      1/1     Running   0
11m
nats-1                                      1/1     Running   0
10m
nats-2                                      1/1     Running   0
10m
nautilus-6b9d88bc86-h8kfb                   1/1     Running   0
8m6s
nautilus-6b9d88bc86-vn68r                   1/1     Running   0
8m35s
openapi-b87d77dd8-5dz9h                     1/1     Running   0
9m7s
```

```
polaris-consul-consul-5ljfb                1/1     Running   0
11m
polaris-consul-consul-s5d5z                1/1     Running   0
11m
polaris-consul-consul-server-0             1/1     Running   0
11m
polaris-consul-consul-server-1             1/1     Running   0
11m
polaris-consul-consul-server-2             1/1     Running   0
11m
polaris-consul-consul-twmpq                1/1     Running   0
11m
polaris-mongodb-0                          2/2     Running   0
11m
polaris-mongodb-1                          2/2     Running   0
10m
polaris-mongodb-2                          2/2     Running   0
10m
polaris-ui-84dc87847f-zrg8w                1/1     Running   0
7m12s
polaris-vault-0                            1/1     Running   0
11m
polaris-vault-1                            1/1     Running   0
11m
polaris-vault-2                            1/1     Running   0
11m
public-metrics-657698b66f-67pgt            1/1     Running   0
8m47s
storage-backend-metrics-6848b9fd87-w7x8r   1/1     Running   0
8m39s
storage-provider-5ff5868cd5-r9hj7          1/1     Running   0
8m45s
telegraf-ds-dw4hg                          1/1     Running   0
7m23s
telegraf-ds-k92gn                          1/1     Running   0
7m23s
telegraf-ds-mmxjl                          1/1     Running   0
7m23s
telegraf-ds-nhs8s                          1/1     Running   0
7m23s
telegraf-ds-rj7lw                          1/1     Running   0
7m23s
telegraf-ds-tqrkb                          1/1     Running   0
7m23s
telegraf-rs-9mwgj                          1/1     Running   0
7m23s
```

```
telemetry-service-56c49d689b-ffrzx          1/1     Running   0
8m42s
tenancy-767c77fb9d-g9ctv                    1/1     Running   0
8m52s
traefik-5857d87f85-7pmx8                    1/1     Running   0
6m49s
traefik-5857d87f85-cpxgv                    1/1     Running   0
5m34s
traefik-5857d87f85-lvmlb                    1/1     Running   0
4m33s
traefik-5857d87f85-t2xlk                    1/1     Running   0
4m33s
traefik-5857d87f85-v9wpf                    1/1     Running   0
7m3s
trident-svc-595f84dd78-zb816                1/1     Running   0
8m54s
vault-controller-86c94fbf4f-krttq           1/1     Running   0
9m24s
```

2. (Opcional) para asegurarse de que la instalación ha finalizado, puede ver el `acc-operator` registra utilizando el siguiente comando.

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

> `accHost` el registro del clúster es una de las últimas operaciones y, si no se produce un error en la implementación, no provocará un error. En el caso de un error de registro del clúster que se indica en los registros, puede volver a intentar el registro a través del flujo de trabajo de add cluster "En la interfaz de usuario de" O API.

3. Cuando todos los pods estén en ejecución, verifique que la instalación se haya realizado correctamente. Para ello, recupere el `AstraControlCenter` Instancia instalada por el operador del Centro de control Astra.

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

4. En el YAML, compruebe el `status.deploymentState` en la respuesta para `Deployed` valor. Si la implementación no se realizó correctamente, aparece en su lugar un mensaje de error.
5. Para obtener la contraseña única que utilizará cuando inicie sesión en Astra Control Center, copie la `status.uuid` valor. La contraseña es `ACC-` Seguido del valor UUID (`ACC-[UUID]` o, en este ejemplo, `ACC-9aa5fdae-4214-4cb7-9976-5d8b4c0ce27f`).

**Detalles de AYLMA de muestra**

```
name: astra
   namespace: netapp-acc
   resourceVersion: "104424560"
   selfLink: /apis/astra.netapp.io/v1/namespaces/netapp-
acc/astracontrolcenters/astra
   uid: 9aa5fdae-4214-4cb7-9976-5d8b4c0ce27f
 spec:
   accountName: Example
   astraAddress: astra.example.com
   astraVersion: 21.12.60
   autoSupport:
     enrolled: true
     url: https://support.netapp.com/asupprod/post/1.0/postAsup
   crds: {}
   email: admin@example.com
   firstName: SRE
   imageRegistry:
     name: registry_name/astra
     secret: astra-registry-cred
   lastName: Admin
 status:
   accConditionHistory:
     items:
     - astraVersion: 21.12.60
       condition:
         lastTransitionTime: "2021-11-23T02:23:59Z"
         message: Deploying is currently in progress.
         reason: InProgress
         status: "False"
         type: Ready
       generation: 2
       observedSpec:
         accountName: Example
         astraAddress: astra.example.com
         astraVersion: 21.12.60
         autoSupport:
           enrolled: true
           url: https://support.netapp.com/asupprod/post/1.0/postAsup
         crds: {}
         email: admin@example.com
         firstName: SRE
         imageRegistry:
           name: registry_name/astra
           secret: astra-registry-cred
```

```
        lastName: Admin
      timestamp: "2021-11-23T02:23:59Z"
    - astraVersion: 21.12.60
      condition:
        lastTransitionTime: "2021-11-23T02:23:59Z"
        message: Deploying is currently in progress.
        reason: InProgress
        status: "True"
        type: Deploying
      generation: 2
      observedSpec:
        accountName: Example
        astraAddress: astra.example.com
        astraVersion: 21.12.60
        autoSupport:
          enrolled: true
          url: https://support.netapp.com/asupprod/post/1.0/postAsup
        crds: {}
        email: admin@example.com
        firstName: SRE
        imageRegistry:
          name: registry_name/astra
          secret: astra-registry-cred
        lastName: Admin
      timestamp: "2021-11-23T02:23:59Z"
    - astraVersion: 21.12.60
      condition:
        lastTransitionTime: "2021-11-23T02:29:41Z"
        message: Post Install was successful
        observedGeneration: 2
        reason: Complete
        status: "True"
        type: PostInstallComplete
      generation: 2
      observedSpec:
        accountName: Example
        astraAddress: astra.example.com
        astraVersion: 21.12.60
        autoSupport:
          enrolled: true
          url: https://support.netapp.com/asupprod/post/1.0/postAsup
        crds: {}
        email: admin@example.com
        firstName: SRE
        imageRegistry:
          name: registry_name/astra
```

```
          secret: astra-registry-cred
        lastName: Admin
      timestamp: "2021-11-23T02:29:41Z"
    - astraVersion: 21.12.60
      condition:
        lastTransitionTime: "2021-11-23T02:29:41Z"
        message: Deploying succeeded.
        reason: Complete
        status: "False"
        type: Deploying
      generation: 2
      observedGeneration: 2
      observedSpec:
        accountName: Example
        astraAddress: astra.example.com
        astraVersion: 21.12.60
        autoSupport:
          enrolled: true
          url: https://support.netapp.com/asupprod/post/1.0/postAsup
        crds: {}
        email: admin@example.com
        firstName: SRE
        imageRegistry:
          name: registry_name/astra
          secret: astra-registry-cred
        lastName: Admin
      observedVersion: 21.12.60
      timestamp: "2021-11-23T02:29:41Z"
    - astraVersion: 21.12.60
      condition:
        lastTransitionTime: "2021-11-23T02:29:41Z"
        message: Astra is deployed
        reason: Complete
        status: "True"
        type: Deployed
      generation: 2
      observedGeneration: 2
      observedSpec:
        accountName: Example
        astraAddress: astra.example.com
        astraVersion: 21.12.60
        autoSupport:
          enrolled: true
          url: https://support.netapp.com/asupprod/post/1.0/postAsup
        crds: {}
        email: admin@example.com
```

```
        firstName: SRE
        imageRegistry:
          name: registry_name/astra
          secret: astra-registry-cred
        lastName: Admin
      observedVersion: 21.12.60
      timestamp: "2021-11-23T02:29:41Z"
    - astraVersion: 21.12.60
      condition:
        lastTransitionTime: "2021-11-23T02:29:41Z"
        message: Astra is deployed
        reason: Complete
        status: "True"
        type: Ready
      generation: 2
      observedGeneration: 2
      observedSpec:
        accountName: Example
        astraAddress: astra.example.com
        astraVersion: 21.12.60
        autoSupport:
          enrolled: true
          url: https://support.netapp.com/asupprod/post/1.0/postAsup
        crds: {}
        email: admin@example.com
        firstName: SRE
        imageRegistry:
          name: registry_name/astra
          secret: astra-registry-cred
        lastName: Admin
      observedVersion: 21.12.60
      timestamp: "2021-11-23T02:29:41Z"
  certManager: deploy
  cluster:
    type: OCP
    vendorVersion: 4.7.5
    version: v1.20.0+bafe72f
  conditions:
  - lastTransitionTime: "2021-12-08T16:19:55Z"
    message: Astra is deployed
    reason: Complete
    status: "True"
    type: Ready
  - lastTransitionTime: "2021-12-08T16:19:55Z"
    message: Deploying succeeded.
    reason: Complete
```

```
      status: "False"
      type: Deploying
    - lastTransitionTime: "2021-12-08T16:19:53Z"
      message: Post Install was successful
      observedGeneration: 2
      reason: Complete
      status: "True"
      type: PostInstallComplete
    - lastTransitionTime: "2021-12-08T16:19:55Z"
      message: Astra is deployed
      reason: Complete
      status: "True"
      type: Deployed
    deploymentState: Deployed
    observedGeneration: 2
    observedSpec:
      accountName: Example
      astraAddress: astra.example.com
      astraVersion: 21.12.60
      autoSupport:
        enrolled: true
        url: https://support.netapp.com/asupprod/post/1.0/postAsup
      crds: {}
      email: admin@example.com
      firstName: SRE
      imageRegistry:
        name: registry_name/astra
        secret: astra-registry-cred
      lastName: Admin
    observedVersion: 21.12.60
    postInstall: Complete
    uuid: 9aa5fdae-4214-4cb7-9976-5d8b4c0ce27f
kind: List
metadata:
 resourceVersion: ""
 selfLink: ""
```

## Configure la entrada para el equilibrio de carga

Puede configurar una controladora de entrada de Kubernetes que gestione el acceso externo a los servicios, como el equilibrio de carga en un clúster.

Este procedimiento explica cómo configurar un controlador de entrada (`ingressType:Generic`). Esta es la acción predeterminada con Astra Control Center. Después de implementar Astra Control Center, deberá configurar el controlador Ingress para exponer Astra Control Center con una dirección URL.

> Si no desea configurar un controlador de entrada, puede configurarlo `ingressType:AccTraefik)`. Astra Control Center utiliza un servicio del tipo "LoadBalancer" (`svc/traefik` En el espacio de nombres de Astra Control Center) y requiere que se le asigne una dirección IP externa accesible. Si se permiten equilibradores de carga en su entorno y no tiene uno configurado, puede utilizar MetalLB u otro equilibrador de carga de servicio externo para asignar una dirección IP externa al servicio. En la configuración interna del servidor DNS, debe apuntar el nombre DNS elegido para Astra Control Center a la dirección IP con equilibrio de carga. Para obtener más información sobre el tipo de servicio de "LoadBalancer" y la entrada, consulte "Requisitos".

Los pasos varían en función del tipo de controlador de entrada que utilice:

- Controlador de entrada nginx
- Controlador OpenShift Ingress

**Lo que necesitará**

- El requerido "controlador de entrada" ya debe ponerse en marcha.
- La "clase de entrada" ya se debe crear la correspondiente al controlador de entrada.
- Se utilizan versiones de Kubernetes entre e incluidas v1.19 y v1.22.

**Pasos para el controlador de entrada Nginx**

1. Cree un secreto de tipo[`kubernetes.io/tls`] Para una clave privada TLS y un certificado en `netapp-acc` (o nombre personalizado) como se describe en "Secretos TLS".
2. Implemente un recurso de entrada en `netapp-acc` (o nombre personalizado) mediante el `v1beta1` (Obsoleto en la versión de Kubernetes inferior a o 1.22) o. `v1` tipo de recurso para un esquema obsoleto o nuevo:
   a. Para un `v1beta1` esquema obsoleto, siga este ejemplo:

```
apiVersion: extensions/v1beta1
kind: Ingress
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. Para la `v1` nuevo esquema, siga este ejemplo:

```
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
```

**Pasos para el controlador de entrada de OpenShift**

1. Obtenga su certificado y consiga los archivos de clave, certificado y CA listos para su uso por la ruta OpenShift.
2. Cree la ruta 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
```

## Inicie sesión en la interfaz de usuario de Astra Control Center

Después de instalar Astra Control Center, cambiará la contraseña del administrador predeterminado e inicie sesión en el panel de interfaz de usuario de Astra Control Center.

**Pasos**

1. En un explorador, introduzca el FQDN que utilizó en `astraAddress` en la `astra_control_center_min.yaml` CR cuando Ha instalado Astra Control Center.
2. Acepte los certificados autofirmados cuando se le solicite.

> Se puede crear un certificado personalizado después de iniciar sesión.

3. En la página de inicio de sesión de Astra Control Center, introduzca el valor utilizado `email` pulg `astra_control_center_min.yaml` CR cuando Ha instalado Astra Control Center, seguido de la contraseña única (`ACC-[UUID]`).

> Si introduce una contraseña incorrecta tres veces, la cuenta de administrador se bloqueará durante 15 minutos.

4. Seleccione **Iniciar sesión**.
5. Cambie la contraseña cuando se le solicite.

> Si este es su primer inicio de sesión y olvida la contraseña y aún no se han creado otras cuentas de usuario administrativas, comuníquese con el servicio de soporte de NetApp para obtener ayuda para la recuperación de contraseñas.

6. (Opcional) quite el certificado TLS autofirmado existente y sustitúyalo por un "Certificado TLS personalizado firmado por una entidad de certificación (CA)".

## Solucione los problemas de instalación

Si alguno de los servicios está en `Error` puede inspeccionar los registros. Busque códigos de respuesta API en la gama 400 a 500. Esos indican el lugar donde ocurrió un fracaso.

**Pasos**

1. Para inspeccionar los registros del operador de Astra Control Center, introduzca lo siguiente:

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

## El futuro

Complete la implementación llevando a cabo "tareas de configuración".

# Instale Astra Control Center utilizando OpenShift OperatorHub

Si utiliza Red Hat OpenShift, puede instalar Astra Control Center mediante el operador certificado de Red Hat. Utilice este procedimiento para instalar Astra Control Center desde "Catálogo de Red Hat Ecosystem" O con Red Hat OpenShift Container Platform.

Después de completar este procedimiento, debe volver al procedimiento de instalación para completar el "pasos restantes" para verificar que la instalación se ha realizado correctamente e iniciar sesión.

**Lo que necesitará**

- "Antes de comenzar la instalación, prepare su entorno para la implementación de Astra Control Center".
- En el clúster OpenShift, asegúrese de que todos los operadores de clúster se encuentran en buen estado

(`available` es `true`):

```
oc get clusteroperators
```

- Desde su clúster OpenShift, asegúrese de que todos los servicios API se encuentran en buen estado (`available` es `true`):

```
oc get apiservices
```

- Ha creado una dirección FQDN para Astra Control Center en su centro de datos.
- Dispone de los permisos necesarios y de acceso a Red Hat OpenShift Container Platform para realizar los pasos de instalación descritos.

**Pasos**

- Descargue y desembale el paquete Astra Control Center
- Instale el complemento Astra kubectl de NetApp
- Agregue las imágenes al registro local
- Busque la página de instalación del operador
- Instale el operador
- Instalar Astra Control Center

## Descargue y desembale el paquete Astra Control Center

1. Descargue el paquete Astra Control Center (`astra-control-center-[version].tar.gz`) del "Sitio de soporte de NetApp".
2. Descargue el archivo zip de los certificados y claves de Astra Control Center de "Sitio de soporte de NetApp".
3. (Opcional) Use el siguiente comando para verificar la firma del paquete:

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

4. Extraiga las imágenes:

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

## Instale el complemento Astra kubectl de NetApp

La Astra de NetApp `kubectl` El complemento de línea de comandos ahorra tiempo al realizar tareas comunes asociadas con la implementación y actualización de Astra Control Center.

**Lo que necesitará**

NetApp proporciona binarios para el complemento para distintas arquitecturas de CPU y sistemas operativos. Debe saber qué CPU y sistema operativo tiene antes de realizar esta tarea. En los sistemas operativos Linux y Mac, puede utilizar `uname -a` comando para recopilar esta información.

**Pasos**

1. Enumere la Astra de NetApp disponible `kubectl` Haga un complemento para binarios y anote el nombre del archivo que necesita para su sistema operativo y arquitectura de CPU:

```
ls kubectl-astra/
```

2. Copie el archivo en la misma ubicación que el estándar `kubectl` utilidad. En este ejemplo, la `kubectl` la utilidad se encuentra en `/usr/local/bin` directorio. Sustituya `<binary-name>` con el nombre del archivo que necesita:

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

## Agregue las imágenes al registro local

1. Cambie al directorio Astra:

```
cd acc
```

2. Agregue los archivos del directorio imagen de Astra Control Center al registro local.

> Consulte secuencias de comandos de ejemplo para la carga automática de imágenes a continuación.

   a. Inicie sesión en su registro:

      Docker:

```
docker login [your_registry_path]
```

      Podman:

```
podman login [your_registry_path]
```

   b. Utilice la secuencia de comandos adecuada para cargar las imágenes, etiquetar las imágenes y empuje las imágenes en el registro local:

      Docker:

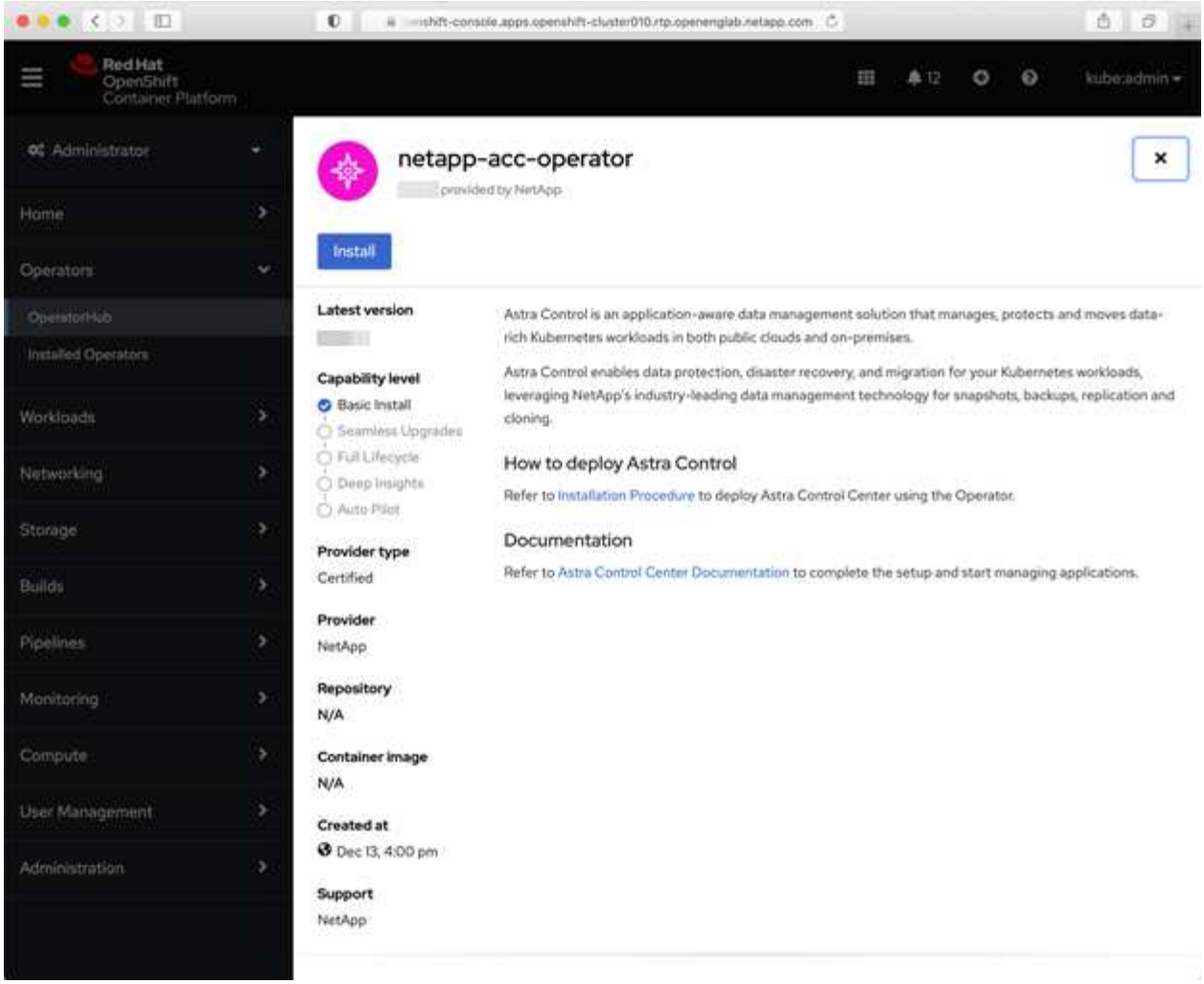
```
export REGISTRY=[Docker_registry_path]
for astraImageFile in $(ls images/*.tar) ; do
  # Load to local cache. And store the name of the loaded image
trimming the 'Loaded images: '
  astraImage=$(docker load --input ${astraImageFile} | sed 's/Loaded
image: //')
  astraImage=$(echo ${astraImage} | sed 's!localhost/!!')
  # Tag with local image repo.
  docker tag ${astraImage} ${REGISTRY}/${astraImage}
  # Push to the local repo.
  docker push ${REGISTRY}/${astraImage}
done
```

Podman:

```
export REGISTRY=[Registry_path]
for astraImageFile in $(ls images/*.tar) ; do
  # Load to local cache. And store the name of the loaded image trimming
the 'Loaded images: '
  astraImage=$(podman load --input ${astraImageFile} | sed 's/Loaded
image(s): //')
  astraImage=$(echo ${astraImage} | sed 's!localhost/!!')
  # Tag with local image repo.
  podman tag ${astraImage} ${REGISTRY}/${astraImage}
  # Push to the local repo.
  podman push ${REGISTRY}/${astraImage}
done
```

## Busque la página de instalación del operador

1. Realice uno de los siguientes procedimientos para acceder a la página de instalación del operador:
   - Desde la consola web de Red Hat OpenShift:



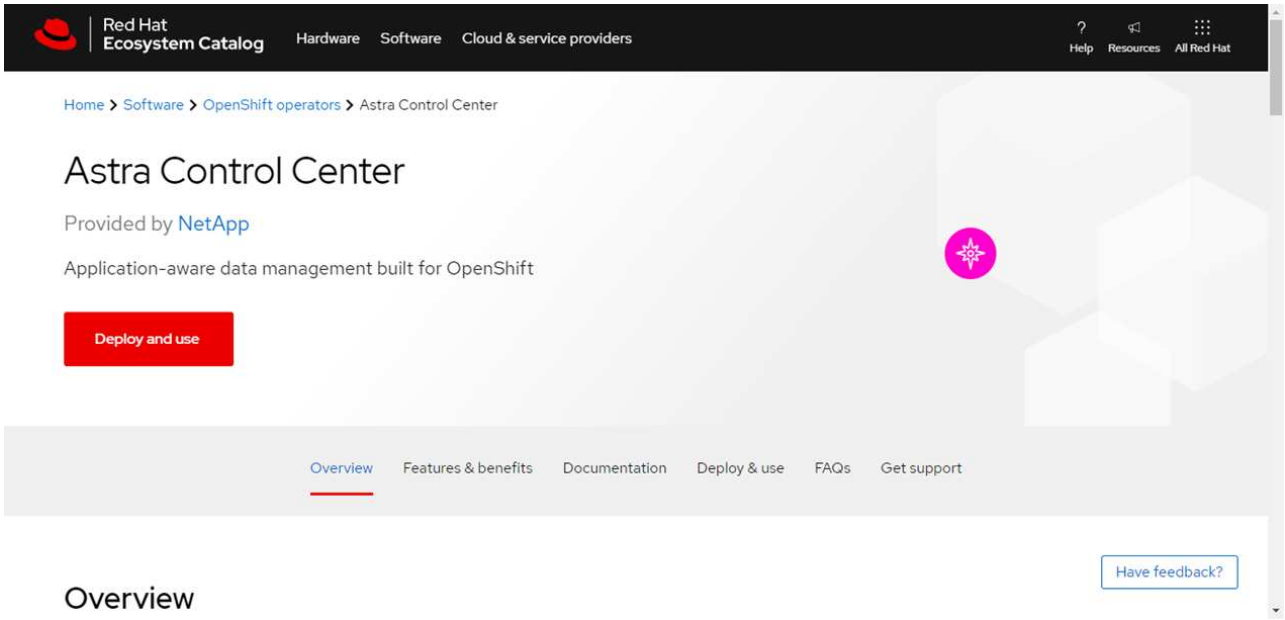

i. Inicie sesión en la IU de OpenShift Container Platform.

ii. En el menú lateral, seleccione **operadores > OperatorHub**.

iii. Seleccione el operador NetApp Astra Control Center.

iv. Seleccione **instalar**.

- En el catálogo de ecosistemas de Red Hat:




i. Seleccione Astra Control Center de NetApp "operador".

ii. Seleccione **desplegar y utilizar**.

## Instale el operador

1. Complete la página **Install Operator** e instale el operador:

> El operador estará disponible en todos los espacios de nombres del clúster.

   a. Seleccione el espacio de nombres del operador o. `netapp-acc-operator` el espacio de nombres se creará automáticamente como parte de la instalación del operador.

   b. Seleccione una estrategia de aprobación manual o automática.

> Se recomienda la aprobación manual. Solo debe tener una instancia de operador en ejecución por clúster.

   c. Seleccione **instalar**.

> Si ha seleccionado una estrategia de aprobación manual, se le pedirá que apruebe el plan de instalación manual para este operador.

2. Desde la consola, vaya al menú OperatorHub y confirme que el operador se ha instalado correctamente.

## Instalar Astra Control Center

1. En la consola de la vista de detalles del operador del Centro de control de Astra, seleccione `Create instance` En la sección proporcionada API.

2. Complete el `Create AstraControlCenter` campo de formulario:

   a. Mantenga o ajuste el nombre del Centro de control de Astra.

   b. (Opcional) Habilitar o deshabilitar AutoSupport. Se recomienda conservar la funcionalidad de AutoSupport.

c. Introduzca la dirección de Astra Control Center. No entre `http://` o. `https://` en la dirección.
d. Introduzca la versión de Astra Control Center; por ejemplo, 21.12.60.
e. Introduzca un nombre de cuenta, una dirección de correo electrónico y un apellido de administrador.
f. Conserve la política de reclamaciones de volumen predeterminada.
g. En **Registro de imágenes**, introduzca la ruta de registro de la imagen del contenedor local. No entre `http://` o. `https://` en la dirección.
h. Si utiliza un registro que requiere autenticación, introduzca el secreto.
i. Introduzca el nombre del administrador.
j. Configure el escalado de recursos.
k. Conserve la clase de almacenamiento predeterminada.
l. Defina las preferencias de manejo de CRD.

3. Seleccione `Create`.

### El futuro

Compruebe que la instalación de Astra Control Center se ha realizado correctamente y complete el "pasos restantes" para iniciar sesión. Además, completará la implementación siguiendo este proceso "tareas de configuración".

# Instale Astra Control Center con un back-end de almacenamiento de Cloud Volumes ONTAP

Con Astra Control Center, puede gestionar sus aplicaciones en un entorno de cloud híbrido con clústeres de Kubernetes e instancias de Cloud Volumes ONTAP autogestionados. Puede poner en marcha Astra Control Center en sus clústeres de Kubernetes en las instalaciones o en uno de los clústeres de Kubernetes autogestionados en el entorno de cloud.

Con una de estas puestas en marcha, puede realizar operaciones de gestión de datos de aplicaciones utilizando Cloud Volumes ONTAP como back-end de almacenamiento. También es posible configurar un bloque de S3 como destino de backup.

Para instalar Astra Control Center en Amazon Web Services (AWS) y Microsoft Azure con un back-end de almacenamiento de Cloud Volumes ONTAP, realice los siguientes pasos en función de su entorno de cloud.

- Ponga en marcha Astra Control Center en Amazon Web Services
- Ponga en marcha Astra Control Center en Microsoft Azure

## Ponga en marcha Astra Control Center en Amazon Web Services

Puede poner en marcha Astra Control Center en un clúster de Kubernetes autogestionado alojado en un cloud público de Amazon Web Services (AWS).

Sólo se admiten clústeres autogestionados de OpenShift Container Platform (OCP) para implementar Astra Control Center.

### Lo que necesitará para AWS

Antes de poner en marcha Astra Control Center en AWS, necesitará los siguientes elementos:

- Licencia Astra Control Center. Consulte "Requisitos de licencia de Astra Control Center".
- "Cumpla los requisitos de Astra Control Center".
- Cuenta de Cloud Central de NetApp
- Permisos de Red Hat OpenShift Container Platform (OCP) (a nivel de espacio de nombres para crear pods)
- Credenciales de AWS, Access ID y Secret Key con permisos que permiten crear cubos y conectores
- Acceso e inicio de sesión del Elastic Container Registry (ECR) de la cuenta de AWS
- Se requieren entradas de zona alojada de AWS y ruta 53 para acceder a la interfaz de usuario de Astra Control

### Requisitos de los entornos operativos para AWS

Astra Control Center requiere los siguientes entornos operativos para AWS:

- OpenShift Container Platform de Red Hat 4.8

> Asegúrese de que el entorno operativo que elija para alojar Astra Control Center cumple los requisitos de recursos básicos que se describen en la documentación oficial del entorno.

Astra Control Center requiere los siguientes recursos además de los requisitos de recursos del entorno:

| Componente | Requisito |
|---|---|
| **Capacidad de almacenamiento Cloud Volumes ONTAP de back-end de NetApp** | 300 GB como mínimo disponible |
| **Nodos de trabajo (requisitos de AWS EC2)** | Al menos 3 nodos de trabajo en total, con 4 núcleos vCPU y 12 GB de RAM en cada uno |
| **Equilibrador de carga** | Tipo de servicio "LoadBalancer" disponible para que el tráfico de entrada se envíe a los servicios en el clúster de entorno operativo |
| **FQDN** | Método para señalar el FQDN de Astra Control Center a la dirección IP de carga equilibrada |
| **Astra Trident (instalado como parte de la detección de clústeres de Kubernetes en NetApp Cloud Manager)** | Astra Trident 21.04 o posterior instalado y configurado y NetApp ONTAP versión 9.5 o posterior como back-end de almacenamiento |

| Componente | Requisito |
| --- | --- |
| **Registro de imágenes** | Debe tener un registro privado existente, como AWS Elastic Container Registry, al que puede insertar imágenes de creación de Astra Control Center. Debe proporcionar la dirección URL del registro de imágenes donde cargará las imágenes.<br><br>El clúster alojado de Astra Control Center y el clúster gestionado deben tener acceso al mismo registro de imágenes para poder realizar copias de seguridad y restaurar aplicaciones mediante la imagen basada en Restic. |
| **Configuración de Astra Trident/ONTAP** | Astra Control Center requiere que se cree una clase de almacenamiento y se establezca como la clase de almacenamiento predeterminada. Astra Control Center es compatible con las siguientes clases de almacenamiento ONTAP Kubernetes que se crean al importar su clúster Kubernetes a Cloud Manager de NetApp. Los proporciona Astra Trident:<br><br>• `vsaworkingenvironment-<>-ha-nas csi.trident.netapp.io`<br>• `vsaworkingenvironment-<>-ha-san csi.trident.netapp.io`<br>• `vsaworkingenvironment-<>-single-nas csi.trident.netapp.io`<br>• `vsaworkingenvironment-<>-single-san csi.trident.netapp.io` |

Estos requisitos suponen que Astra Control Center es la única aplicación que se ejecuta en el entorno operativo. Si el entorno ejecuta aplicaciones adicionales, ajuste estos requisitos mínimos según corresponda.

El token del registro de AWS caduca en 12 horas, después del cual deberá renovar el secreto del registro de imagen Docker.

### Información general sobre la implementación para AWS

He aquí una descripción general del proceso de instalación de Astra Control Center para AWS con Cloud Volumes ONTAP como back-end de almacenamiento.

Cada uno de estos pasos se explica más detalladamente a continuación.

1. Compruebe que dispone de suficientes permisos IAM.
2. Instale un clúster RedHat OpenShift en AWS.
3. Configure AWS.
4. Configure Cloud Manager de NetApp.
5. Instalar Astra Control Center.

### Compruebe que dispone de suficientes permisos IAM

Asegúrese de tener suficientes roles y permisos de IAM para poder instalar un clúster RedHat OpenShift y un conector Cloud Manager de NetApp.

Consulte "Credenciales iniciales de AWS".

### Instale un clúster RedHat OpenShift en AWS

Instale un clúster RedHat OpenShift Container Platform en AWS.

Para obtener instrucciones de instalación, consulte "Instalación de un clúster en AWS en OpenShift Container Platform".

### Configure AWS

A continuación, configure AWS para crear una red virtual, configurar instancias de computación EC2, crear un bloque de AWS S3, crear un Elastic Container Register (ECR) para alojar las imágenes de Astra Control Center y empujar las imágenes a este registro.

Siga la documentación de AWS para completar los pasos siguientes. Consulte "Documentación de instalación de AWS".

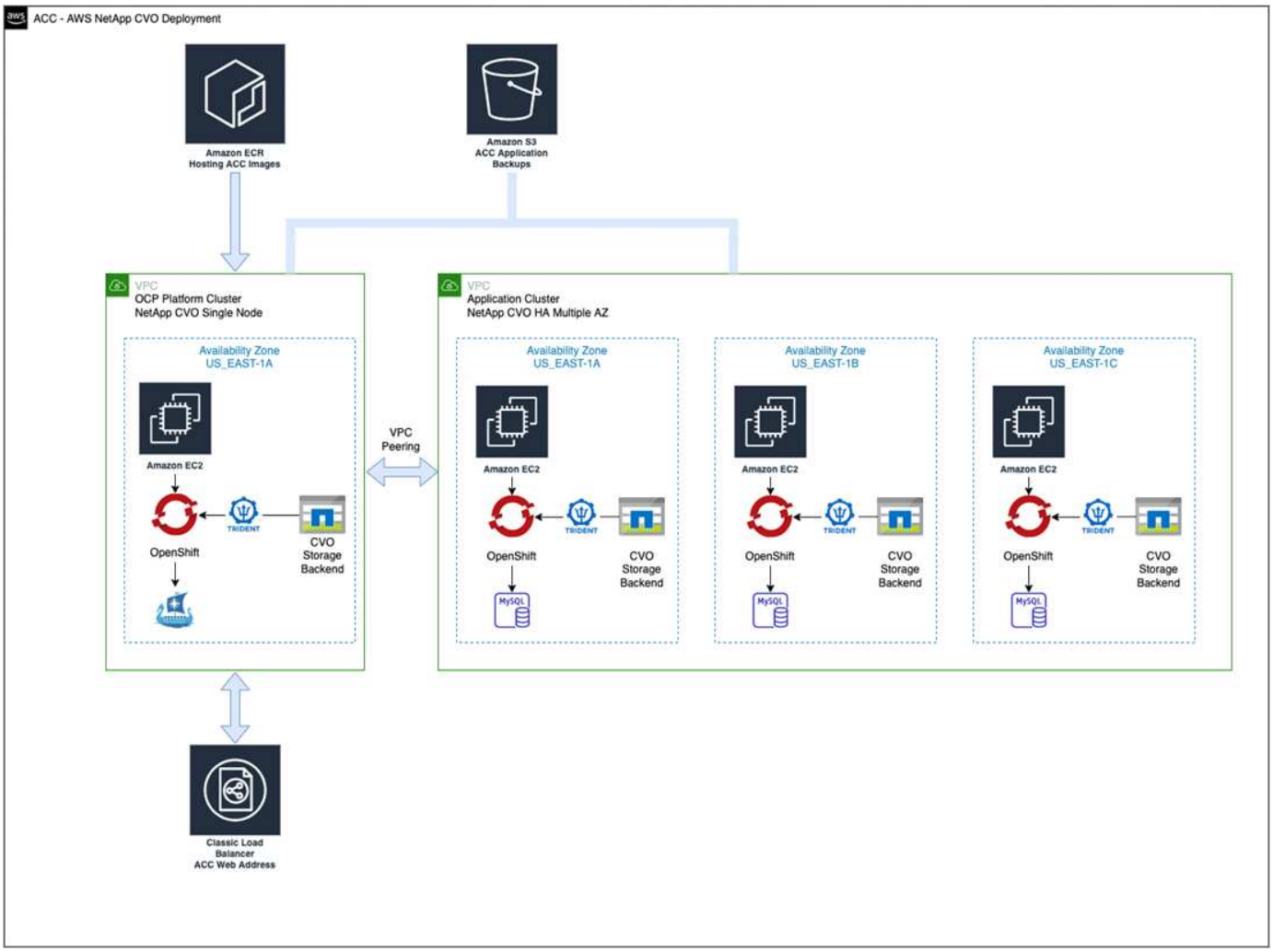
1. Cree una red virtual AWS.
2. Revise las instancias de computación EC2. Puede ser un servidor con configuración básica o máquinas virtuales en AWS.
3. Si el tipo de instancia no coincide con los requisitos mínimos de recursos de Astra para los nodos maestros y trabajadores, cambie el tipo de instancia en AWS para cumplir los requisitos de Astra. Consulte "Requisitos del Centro de Control de Astra".
4. Cree al menos un bloque de AWS S3 para almacenar los backups.
5. Cree un AWS Elastic Container Registry (ECR) para alojar todas las imágenes ACC.

> Si no crea la ECR, Astra Control Center no puede acceder a los datos de supervisión de un clúster que contenga Cloud Volumes ONTAP con un back-end de AWS. El problema se produce cuando el clúster que intenta detectar y gestionar mediante Astra Control Center no tiene acceso a AWS ECR.

6. Inserte las imágenes ACC en el registro definido.

> El token del registro de contenedor elástico de AWS (ECR) caduca al cabo de 12 horas y provoca errores en las operaciones de clonado de varios clústeres. Este problema ocurre cuando se gestiona un back-end de almacenamiento desde Cloud Volumes ONTAP configurado para AWS. Para corregir este problema, vuelva a autenticarse con la ECR y genere un nuevo secreto para que las operaciones de clonación se reanuden correctamente.

A continuación mostramos un ejemplo de una puesta en marcha de AWS:




### Configure Cloud Manager de NetApp

Con Cloud Manager, cree un espacio de trabajo, añada un conector a AWS, cree un entorno de trabajo e importe el clúster.

Siga la documentación de Cloud Manager para completar los siguientes pasos. Consulte lo siguiente:

- "Introducción a Cloud Volumes ONTAP en AWS".
- "Cree un conector en AWS mediante Cloud Manager"

**Pasos**

1. Añada sus credenciales a Cloud Manager.
2. Crear un área de trabajo.
3. Agregue un conector para AWS. Elija AWS como proveedor.
4. Cree un entorno de trabajo para su entorno de cloud.
   a. Ubicación: "Amazon Web Services (AWS)"
   b. Tipo: "Cloud Volumes ONTAP ha"
5. Importe el clúster OpenShift. El clúster se conectará al entorno de trabajo que acaba de crear.
   a. Consulte los detalles del clúster de NetApp seleccionando **K8s** > **Lista de clústeres** > **Detalles del clúster**.

b. En la esquina superior derecha, tenga en cuenta la versión de Trident.

c. Observe las clases de almacenamiento del clúster Cloud Volumes ONTAP que muestran NetApp como el aprovisionador.

   Esto importa su clúster de Red Hat OpenShift y le asigna una clase de almacenamiento predeterminada. Seleccione la clase de almacenamiento. Trident se instala automáticamente como parte del proceso de importación y detección.

6. Obsérvese todos los volúmenes y volúmenes persistentes en esta puesta en marcha de Cloud Volumes ONTAP.

> Cloud Volumes ONTAP puede funcionar como un nodo único o en alta disponibilidad. Si está habilitada, anote el estado de alta disponibilidad y el estado de implementación del nodo que se ejecutan en AWS.

#### Instalar Astra Control Center

Siga la norma "Instrucciones de instalación de Astra Control Center".

## Ponga en marcha Astra Control Center en Microsoft Azure

Puede poner en marcha Astra Control Center en un clúster de Kubernetes autogestionado que se aloja en un cloud público de Microsoft Azure.

#### Lo que necesitará para Azure

Antes de poner en marcha Astra Control Center en Azure, necesitará los siguientes elementos:

- Licencia Astra Control Center. Consulte "Requisitos de licencia de Astra Control Center".
- "Cumpla los requisitos de Astra Control Center".
- Cuenta de Cloud Central de NetApp
- Red Hat OpenShift Container Platform (OCP) 4.8
- Permisos de Red Hat OpenShift Container Platform (OCP) (a nivel de espacio de nombres para crear pods)
- Credenciales de Azure con permisos que le permiten crear cubos y conectores

#### Requisitos del entorno operativo para Azure

Asegúrese de que el entorno operativo que elija para alojar Astra Control Center cumple los requisitos de recursos básicos que se describen en la documentación oficial del entorno.

Astra Control Center requiere los siguientes recursos además de los requisitos de recursos del entorno:

Consulte "Requisitos del entorno operativo del Centro de control de Astra".

| Componente | Requisito |
|---|---|
| **Capacidad de almacenamiento Cloud Volumes ONTAP de back-end de NetApp** | 300 GB como mínimo disponible |

| Componente | Requisito |
| --- | --- |
| **Nodos de trabajo (requisitos de computación de Azure)** | Al menos 3 nodos de trabajo en total, con 4 núcleos vCPU y 12 GB de RAM en cada uno |
| **Equilibrador de carga** | Tipo de servicio "LoadBalancer" disponible para que el tráfico de entrada se envíe a los servicios en el clúster de entorno operativo |
| **FQDN (zona DNS de Azure)** | Método para señalar el FQDN de Astra Control Center a la dirección IP de carga equilibrada |
| **Astra Trident (instalado como parte de la detección de clústeres de Kubernetes en NetApp Cloud Manager)** | Como back-end de almacenamiento, se usará Astra Trident 21.04 o posterior instalado y configurado, y NetApp ONTAP versión 9.5 o posterior |
| **Registro de imágenes** | Debe disponer de un registro privado existente, como Azure Container Registry (ACR), al que puede insertar imágenes de creación de Astra Control Center. Debe proporcionar la dirección URL del registro de imágenes donde cargará las imágenes.<br>Es necesario habilitar el acceso anónimo para extraer imágenes RTIC para realizar copias de seguridad. |
| **Configuración de Astra Trident/ONTAP** | Astra Control Center requiere que se cree una clase de almacenamiento y se establezca como la clase de almacenamiento predeterminada. Astra Control Center es compatible con las siguientes clases de almacenamiento ONTAP Kubernetes que se crean al importar su clúster Kubernetes a Cloud Manager de NetApp. Los proporciona Astra Trident:<br>• `vsaworkingenvironment-<>-ha-nas csi.trident.netapp.io`<br>• `vsaworkingenvironment-<>-ha-san csi.trident.netapp.io`<br>• `vsaworkingenvironment-<>-single-nas csi.trident.netapp.io`<br>• `vsaworkingenvironment-<>-single-san csi.trident.netapp.io` |

Estos requisitos suponen que Astra Control Center es la única aplicación que se ejecuta en el entorno operativo. Si el entorno ejecuta aplicaciones adicionales, ajuste estos requisitos mínimos según corresponda.

### Información general sobre la implementación para Azure

A continuación se ofrece una descripción general del proceso de instalación de Astra Control Center para Azure.

Cada uno de estos pasos se explica más detalladamente a continuación.

1. Instale un clúster RedHat OpenShift en Azure.
2. Cree grupos de recursos de Azure.
3. Compruebe que dispone de suficientes permisos IAM.
4. Configure Azure.
5. Configure Cloud Manager de NetApp.
6. Instalar y configurar Astra Control Center.

### Instale un clúster RedHat OpenShift en Azure

El primer paso es instalar un clúster RedHat OpenShift en Azure.

Para obtener instrucciones de instalación, consulte la documentación de RedHat en "Instalación del clúster OpenShift en Azure" y.. "Instalar una cuenta de Azure".

### Cree grupos de recursos de Azure

Cree al menos un grupo de recursos de Azure.

> OpenShift podría crear sus propios grupos de recursos. Además de estos, también debe definir los grupos de recursos de Azure. Consulte la documentación de OpenShift.

Es posible que desee crear un grupo de recursos de clúster de plataforma y un grupo de recursos de clúster de aplicación OpenShift de destino.

### Compruebe que dispone de suficientes permisos IAM

Asegúrese de tener suficientes roles y permisos de IAM para poder instalar un clúster RedHat OpenShift y un conector Cloud Manager de NetApp.

Consulte "Credenciales y permisos de Azure".

### Configure Azure

A continuación, configure Azure para crear una red virtual, configurar instancias de computación, crear un contenedor de Azure Blob, crear un registro de contenedores de Azure (ACR) para alojar las imágenes de Astra Control Center y colocar las imágenes en este registro.

Siga la documentación de Azure para completar los siguientes pasos. Consulte "Instalando el clúster de OpenShift en Azure".

1. Cree una red virtual de Azure.
2. Revise las instancias de computación. Puede ser un servidor con configuración básica o máquinas virtuales en Azure.
3. Si el tipo de instancia no coincide con los requisitos mínimos de recursos de Astra para los nodos maestros y trabajadores, cambie el tipo de instancia en Azure para cumplir los requisitos de Astra. Consulte "Requisitos del Centro de Control de Astra".
4. Cree al menos un contenedor de Azure Blob para almacenar los backups.
5. Cree una cuenta de almacenamiento. Necesitará una cuenta de almacenamiento para crear un contenedor que se utilizará como bloque en Astra Control Center.

6. Crear un secreto, que es necesario para el acceso a bloques.
7. Cree un Azure Container Registry (ACR) para alojar todas las imágenes de Astra Control Center.
8. Configure el acceso ACR para pulsar/extraer todas las imágenes del Centro de control de Astra.
9. Inserte las imágenes ACC en este registro introduciendo el siguiente script:

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

**Ejemplo**:

```
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. Configure zonas DNS.

### Configure Cloud Manager de NetApp

Con Cloud Manager, cree un espacio de trabajo, añada un conector a Azure, cree un entorno de trabajo e importe el clúster.

Siga la documentación de Cloud Manager para completar los siguientes pasos. Consulte "Introducción a Cloud Manager en Azure".

#### Lo que necesitará

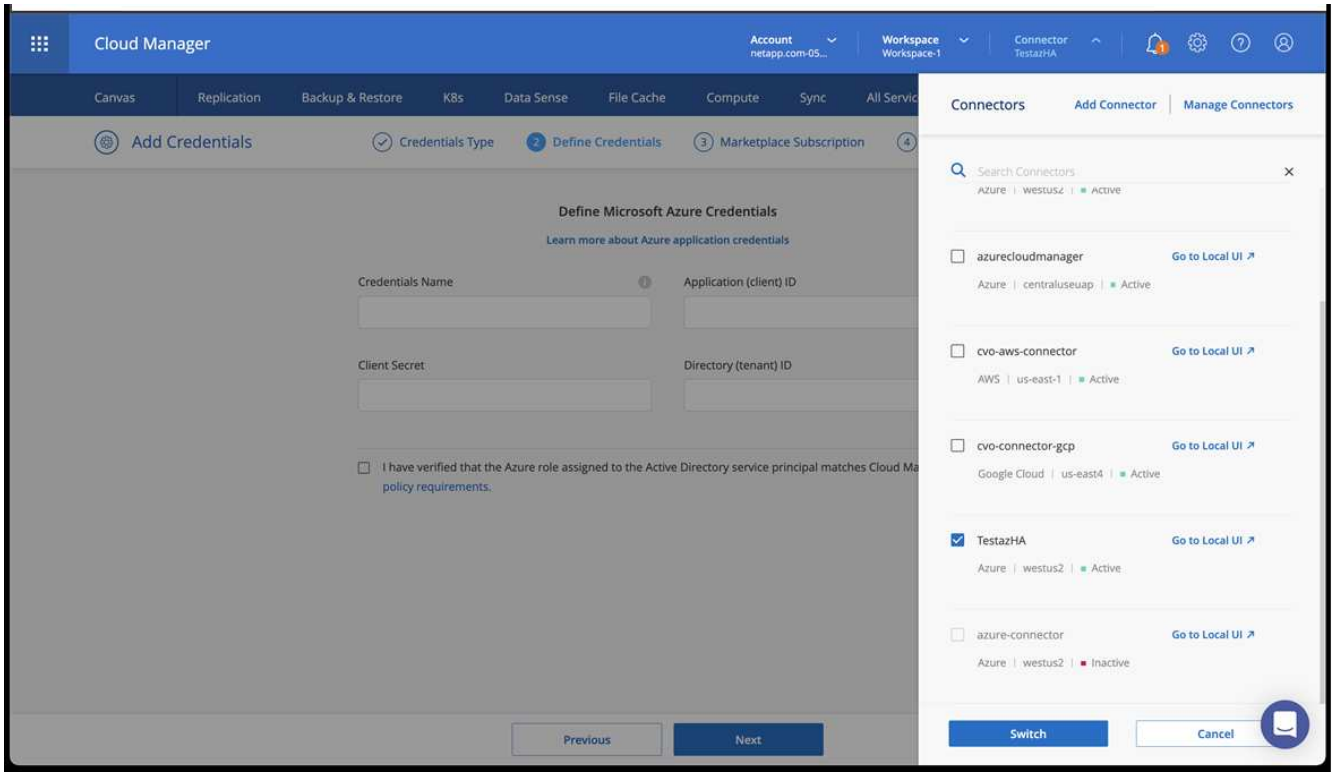
Acceso a la cuenta de Azure con los permisos y roles de IAM necesarios

#### Pasos

1. Añada sus credenciales a Cloud Manager.
2. Agregue un conector para Azure. Consulte "Políticas de Cloud Manager".
   a. Elija **Azure** como proveedor.
   b. Introduzca las credenciales de Azure, incluidos el ID de aplicación, el secreto de cliente y el ID del directorio (inquilino).

      Consulte "Crear un conector en Azure desde Cloud Manager".

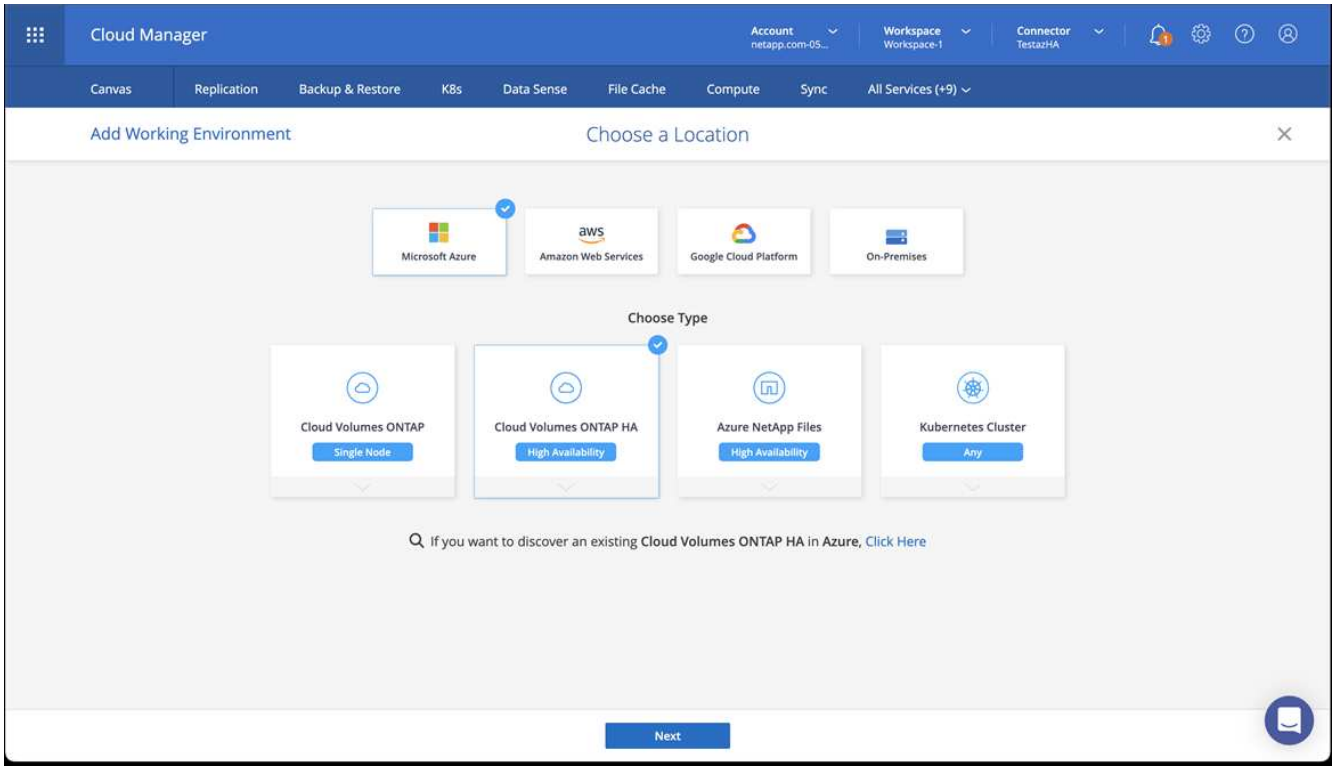
3. Asegúrese de que el conector está en marcha y cambie a dicho conector.




4. Cree un entorno de trabajo para su entorno de cloud.

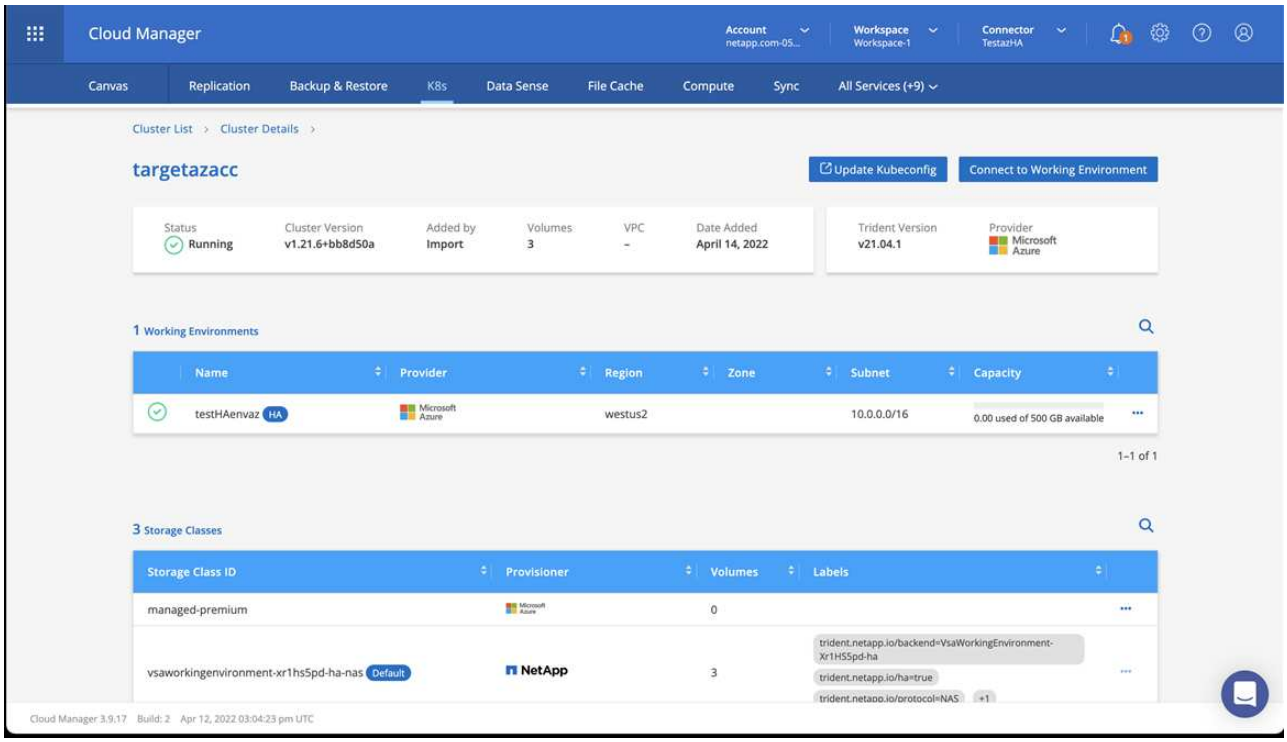
   a. Ubicación: "Microsoft Azure".

   b. Tipo: "Cloud Volumes ONTAP ha".




5. Importe el clúster OpenShift. El clúster se conectará al entorno de trabajo que acaba de crear.

a. Consulte los detalles del clúster de NetApp seleccionando **K8s** > **Lista de clústeres** > **Detalles del clúster**.




b. En la esquina superior derecha, tenga en cuenta la versión de Trident.

c. Observe las clases de almacenamiento del clúster Cloud Volumes ONTAP que muestran NetApp como el aprovisionador.

Esto importa su clúster de Red Hat OpenShift y asigna una clase de almacenamiento predeterminada. Seleccione la clase de almacenamiento. Trident se instala automáticamente como parte del proceso de importación y detección.

6. Obsérvese todos los volúmenes y volúmenes persistentes en esta puesta en marcha de Cloud Volumes ONTAP.

7. Cloud Volumes ONTAP puede funcionar como un nodo único o en alta disponibilidad. Si ha está habilitada, anote el estado de alta disponibilidad y el estado de puesta en marcha del nodo que se ejecutan en Azure.

### Instalar y configurar Astra Control Center

Instale Astra Control Center con el estándar "instrucciones de instalación".

Con Astra Control Center, añada un bucket de Azure. Consulte "Configure Astra Control Center y añada cucharones".