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"

En función de su entorno, es posible que se necesite una configuración adicional después de instalar Astra Control Center:

- "Configurar Astra Control Center después de la instalación"

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. Puede utilizar este procedimiento para instalar Astra Control Center en entornos conectados a Internet o con conexión por aire.

Otros procedimientos de instalación

- **Instalar con RedHat OpenShift OperatorHub**: Utilice esto "procedimiento alternativo" Para instalar Astra Control Center en OpenShift con OperatorHub.
- **Instalar en la nube pública con Cloud Volumes ONTAP backend**: Uso "estos procedimientos" Para instalar Astra Control Center en Amazon Web Services (AWS), Google Cloud Platform (GCP) o Microsoft Azure con un back-end de almacenamiento de Cloud Volumes ONTAP.

Para ver una demostración del proceso de instalación de Astra Control Center, consulte "este vídeo".

Lo que necesitará

- "Antes de comenzar la instalación, prepare su entorno para la implementación de Astra Control Center".
- Si ha configurado o desea configurar directivas de seguridad de POD en su entorno, familiarícese con las directivas de seguridad de POD y cómo afectan a la instalación de Astra Control Center. Consulte "Comprender las restricciones de directivas de seguridad de POD".
- Asegúrese de que todos los servicios de API se encuentren en buen estado y estén disponibles:

  ```
  kubectl get apiservices
  ```

- Asegúrese de que el FQDN de Astra que tiene previsto utilizar se puede 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.
- Si ya existe un administrador de certificados en el clúster, tendrá que realizar algunos "requisitos previos" Por lo tanto, Astra Control Center no intenta instalar su propio administrador de certificados. De forma predeterminada, Astra Control Center instala su propio administrador de certificados durante la instalación.

Acerca de esta tarea

El proceso de instalación de Astra Control Center le ayuda a hacer lo siguiente:

- Instale los componentes de Astra en la `netapp-acc` (o nombre personalizado).
- Cree una cuenta predeterminada de administrador de propietario de Astra Control.
- Establecer una dirección de correo electrónico de usuario administrativo y una contraseña de configuración inicial predeterminada. A este usuario se le asigna el rol de propietario que se necesita para iniciar sesión por primera vez en la interfaz de usuario.
- Determine que se están ejecutando todas las pods de Astra Control Center.
- Instale la interfaz de usuario de Astra Control Center.

> No elimine el operador Astra Control Center (por ejemplo, `kubectl delete -f astra_control_center_operator_deploy.yaml`) En cualquier momento durante la instalación o el funcionamiento de Astra Control Center para evitar la eliminación de las dosis.

Pasos

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

- Descargue y extraiga 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 extraiga Astra Control Center

1. Vaya a la "Página de descargas de Astra Control Center Evaluation" En el sitio de soporte de NetApp.
2. Descargue el paquete que contiene Astra Control Center (`astra-control-center-[version].tar.gz`).
3. (Recomendado pero opcional) Descargue el paquete de certificados y firmas para Astra Control Center (`astra-control-center-certs-[version].tar.gz`) para verificar la firma del paquete:

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

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

Se mostrará la salida `Verified OK` después de una verificación correcta.

4. Extraiga las imágenes del paquete Astra Control Center:

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

Instale el complemento Astra kubectl de NetApp

El complemento de la línea de comandos Astra bectl de NetApp ahorra tiempo en la realización de tareas comunes asociadas a la puesta en marcha y la actualización de Astra Control Center.

Lo que necesitará

NetApp proporciona binarios de complementos para diferentes arquitecturas de CPU y sistemas operativos. Debe saber qué CPU y sistema operativo tiene antes de realizar esta tarea.

Pasos

1. Enumere los binarios disponibles del complemento Astra kubectl de NetApp, y anote el nombre del archivo que necesita para el sistema operativo y la arquitectura de CPU:

   La biblioteca de complementos kubectl forma parte del paquete tar y se extrae en la carpeta `kubectl-astra`.

```
ls kubectl-astra/
```

2. Mueva el binario correcto a la ruta actual y cambie el nombre a. `kubectl-astra`:

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

Agregue las imágenes al registro local

1. Complete la secuencia de pasos apropiada para el motor del contenedor:

Docker

1. Cambie al directorio raíz del tarball. Debería ver este archivo y directorio:

   ```
   acc.manifest.bundle.yaml
   acc/
   ```

2. Inserte las imágenes del paquete en el directorio de imágenes de Astra Control Center en su registro local. Realice las siguientes sustituciones antes de ejecutar el `push-images` comando:

   - Sustituya <BUNDLE_FILE> por el nombre del archivo Astra Control Bundle (`acc.manifest.bundle.yaml`).
   - Sustituya <MY_FULL_REGISTRY_PATH> por la URL del repositorio de Docker; por ejemplo, "<a href="https://<docker-registry>"" class="bare">https://<docker-registry>".
   - Reemplace <MY_REGISTRY_USER> por el nombre de usuario.
   - Sustituya <MY_REGISTRY_TOKEN> por un token autorizado para el registro.

   ```
   kubectl astra packages push-images -m <BUNDLE_FILE> -r
   <MY_FULL_REGISTRY_PATH> -u <MY_REGISTRY_USER> -p
   <MY_REGISTRY_TOKEN>
   ```

Podman

1. Cambie al directorio raíz del tarball. Debería ver este archivo y directorio:

   ```
   acc.manifest.bundle.yaml
   acc/
   ```

2. Inicie sesión en su registro:

   ```
   podman login <YOUR_REGISTRY>
   ```

3. Prepare y ejecute una de las siguientes secuencias de comandos personalizadas para la versión de Podman que utilice. Sustituya <MY_FULL_REGISTRY_PATH> por la URL del repositorio que incluye cualquier subdirectorio.

   ```
   <strong>Podman 4</strong>
   ```

```
export REGISTRY=<MY_FULL_REGISTRY_PATH>
export PACKAGENAME=acc
export PACKAGEVERSION=22.11.0-82
export DIRECTORYNAME=acc
for astraImageFile in $(ls ${DIRECTORYNAME}/images/*.tar) ; do
astraImage=$(podman load --input ${astraImageFile} | sed 's/Loaded
image: //')
astraImageNoPath=$(echo ${astraImage} | sed 's:.*/::')
podman tag ${astraImageNoPath} ${REGISTRY}/netapp/astra/
${PACKAGENAME}/${PACKAGEVERSION}/${astraImageNoPath}
podman push ${REGISTRY}/netapp/astra/${PACKAGENAME}/${
PACKAGEVERSION}/${astraImageNoPath}
done
```

```
<strong>Podman 3</strong>
```

```
export REGISTRY=<MY_FULL_REGISTRY_PATH>
export PACKAGENAME=acc
export PACKAGEVERSION=22.11.0-82
export DIRECTORYNAME=acc
for astraImageFile in $(ls ${DIRECTORYNAME}/images/*.tar) ; do
astraImage=$(podman load --input ${astraImageFile} | sed 's/Loaded
image: //')
astraImageNoPath=$(echo ${astraImage} | sed 's:.*/::')
podman tag ${astraImageNoPath} ${REGISTRY}/netapp/astra/
${PACKAGENAME}/${PACKAGEVERSION}/${astraImageNoPath}
podman push ${REGISTRY}/netapp/astra/${PACKAGENAME}/${
PACKAGEVERSION}/${astraImageNoPath}
done
```

La ruta de acceso de imagen que crea el script debe parecerse a la siguiente, dependiendo de la configuración del Registro: https://netappdownloads.jfrog.io/docker-astra-control-prod/netapp/astra/acc/22.11.0-82/image:version

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

1. Exporte el KUBECONFIG para el clúster de host de Astra Control Center:

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

> Antes de completar la instalación, asegúrese de que KUBECONFIG apunta al clúster en el que desea instalar Astra Control Center. El KUBECONFIG sólo puede contener un contexto.

2. 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:

   > El marcador de posición `your_registry_path` debe coincidir con la ubicación de las imágenes que ha cargado anteriormente (por ejemplo, `[Registry_URL]/netapp/astra/astracc/22.11.0-82`).

   ```
   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
   ```

   > Si elimina el espacio de nombres después de que se genere el secreto, vuelva a crear el espacio de nombres y, a continuación, vuelva a generar el secreto para el espacio de nombres.

   c. Cree el `netapp-acc` (o nombre personalizado).

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

   Respuesta de ejemplo:

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

d. Cree un secreto para `netapp-acc` (o nombre 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
```

Instale el operador de Astra Control Center

1. Cambie el directorio:

```
cd manifests
```

2. 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
```

> Una muestra anotada de AYLMA sigue estos pasos.

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

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

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.

```
<strong>astra_control_center_operator_deploy.yaml</strong>
```

```
---------------------------------------------------------------------
```

```
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
  strategy:
    type: Recreate
  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
        env:
        - name: ACCOP_LOG_LEVEL
          value: "2"
        - name: ACCOP_HELM_INSTALLTIMEOUT
          value: 5m
        image: [your_registry_path]/acc-operator:[version x.y.z]
        imagePullPolicy: IfNotPresent
        livenessProbe:
          httpGet:
            path: /healthz
            port: 8081
          initialDelaySeconds: 15
```

```
          periodSeconds: 20
        name: manager
        readinessProbe:
          httpGet:
            path: /readyz
            port: 8081
          initialDelaySeconds: 5
          periodSeconds: 10
        resources:
          limits:
            cpu: 300m
            memory: 750Mi
          requests:
            cpu: 100m
            memory: 75Mi
        securityContext:
          allowPrivilegeEscalation: false
      imagePullSecrets: []
      securityContext:
        runAsUser: 65532
      terminationGracePeriodSeconds: 10
```

3. 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
```

4. Verifique que los pods se estén ejecutando:

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

Configurar Astra Control Center

1. Edite el archivo de recursos personalizados (CR) del Centro de control de Astra (`astra_control_center.yaml`) para realizar las configuraciones de cuenta, soporte, registro y otras necesarias:

```
vim astra_control_center.yaml
```

> Una muestra anotada de AYLMA sigue estos pasos.

2. Modifique o confirme los siguientes ajustes:

<code>accountName</code>

Ajuste	Orientación	Tipo	Ejemplo
`accountName`	Cambie el `accountName` Cadena con el nombre que desea asociar a la cuenta Astra Control Center. Sólo puede haber un nombre de cuenta.	cadena	`Example`

<code>astraVersion</code>

Ajuste	Orientación	Tipo	Ejemplo
`astraVersion`	La versión de Astra Control Center para implementar. No se necesita ninguna acción para este ajuste, ya que el valor se rellenará previamente.	cadena	`22.11.0-82`

<code>astraAddress</code>

Ajuste	Orientación	Tipo	Ejemplo
`astraAddress`	Cambie el `astraAddress` Cadena al FQDN (recomendado) o dirección IP que desea utilizar en su navegador para acceder a Astra Control Center. Esta dirección define cómo se encontrará Astra Control Center en su centro de datos y es el mismo FQDN o la dirección IP que ha aprovisionado desde su equilibrador de carga cuando ha finalizado "Requisitos del Centro de Control de Astra". NOTA: No utilizar `http://` o. `https://` en la dirección. Copie este FQDN para utilizarlo en un paso posterior.	cadena	`astra.example.com`

<code>autoSupport</code>

Las selecciones de esta sección determinan si participará en la aplicación de soporte proactivo de NetApp, Active IQ de NetApp y dónde se envían los datos. Se requiere una conexión a Internet (puerto 442) y todos los datos de soporte se anóniman.

Ajuste	Uso	Orientación	Tipo	Ejemplo
`autoSupport.enrolled`	Uno de los dos `enrolled` o. `url` los campos deben seleccionarse	Cambiar `enrolled` Para AutoSupport a. `false` para sitios sin conexión a internet o retención `true` para sitios conectados. Un valor de `true` Permite enviar datos anónimos a NetApp con fines de soporte. La elección predeterminada es `false` E indica que no se enviará ningún dato de soporte a NetApp.	Booleano	`false` (este valor es el predeterminado)
`autoSupport.url`	Uno de los dos `enrolled` o. `url` los campos deben seleccionarse	Esta URL determina dónde se enviarán los datos anónimos.	cadena	`https://support.netapp.com/asupprod/post/1.0/postAsup`

<code>email</code>

Ajuste	Orientación	Tipo	Ejemplo
`email`	Cambie el `email` cadena en la dirección inicial predeterminada del administrador. Copie esta dirección de correo electrónico para su uso en un paso posterior. Esta dirección de correo electrónico se utilizará como nombre de usuario de la cuenta inicial para iniciar sesión en la interfaz de usuario y se le notificarán los eventos de Astra Control.	cadena	`admin@example.com`

<code>firstName</code>

Ajuste	Orientación	Tipo	Ejemplo
`firstName`	El nombre del administrador inicial predeterminado asociado con la cuenta Astra. El nombre utilizado aquí aparecerá en un encabezado de la interfaz de usuario después del primer inicio de sesión.	cadena	`SRE`

<code>LastName</code>

Ajuste	Orientación	Tipo	Ejemplo
`lastName`	Apellido del administrador inicial predeterminado asociado con la cuenta Astra. El nombre utilizado aquí aparecerá en un encabezado de la interfaz de usuario después del primer inicio de sesión.	cadena	`Admin`

<code>imageRegistry</code>

Las selecciones realizadas en esta sección definen el registro de imágenes del contenedor que aloja las imágenes de la aplicación Astra, el operador del centro de control Astra y el repositorio de Astra Control Center Helm.

Ajuste	Uso	Orientación	Tipo	Ejemplo
`imageRegistry.name`	Obligatorio	El nombre del registro de imágenes en el que se insertó las imágenes en el paso anterior. No utilizar `http://` o. `https://` en el nombre del registro.	cadena	`example.registry.com/astra`
`imageRegistry.secret`	Obligatorio si la cadena introducida para `imageRegistry.name' requires a secret.` `IMPORTANT: If you are using a registry that does not require authorization, you must delete this` `secret` línea dentro `imageRegistry` o se producirá un error en la instalación.	El nombre del secreto Kubernetes utilizado para autenticarse con el registro de imágenes.	cadena	`astra-registry-cred`

<code>storageClass</code>

Ajuste	Orientación	Tipo	Ejemplo
`storageClass`	Cambie el `storageClass` valor desde `ontap-gold` En otro recurso de la clase de almacenamiento de Trident, según lo requiera su instalación. Ejecute el comando `kubectl get sc` para determinar las clases de almacenamiento configuradas existentes. Se debe introducir una de las clases de almacenamiento basadas en Trident en el archivo de manifiesto (`astra-control-center-<version>.manifest`) Y se utilizará para Astra PVs. Si no está establecida, se utilizará la clase de almacenamiento predeterminada. NOTA: Si se ha configurado una clase de almacenamiento predeterminada, asegúrese de que es la única clase de almacenamiento que tiene la anotación predeterminada.	cadena	`ontap-gold`

<code>volumeReclaimPolicy</code>

Ajuste	Orientación	Tipo	Opciones
`volumeReclaimPolicy`	De esta forma se establece la política de reclamaciones para los vehículos de Astra. Configuración de esta directiva como `Retain` Conserva los volúmenes persistentes una vez que Astra se elimina. Configuración de esta directiva como `Delete` elimina los volúmenes persistentes después de eliminar astra. Si no se establece este valor, se conservan los VP.	cadena	• `Retain` (Este es el valor predeterminado) • `Delete`

<code>ingressType</code>

Ajuste	Orientación	Tipo	Opciones
`ingressType`	Utilice uno de los siguientes tipos de entrada:*`Generic`* (`ingressType: "Generic"`) (Predeterminado) Utilice esta opción cuando tenga otro controlador de entrada en uso o prefiera usar su propio controlador de entrada. Después de implementar Astra Control Center, deberá configurar el "controlador de entrada" Para exponer Astra Control Center con una URL.**`AccTraefik`** (`ingressType: "AccTraefik"`) Utilice esta opción cuando prefiera no configurar un controlador de entrada. 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 aún 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	cadena	• `Generic` (este es el valor predeterminado) • `AccTraefik`

<code>astraResourcesScaler</code>

Ajuste	Orientación	Tipo	Opciones
`astraResourcesScaler`	Opciones de escalado para los límites de recursos de AstraControlCenter. De forma predeterminada, Astra Control Center se despliega con solicitudes de recursos establecidas para la mayoría de los componentes de Astra. Esta configuración permite que la pila de software de Astra Control Center tenga un mejor rendimiento en entornos con un mayor nivel de carga y escalabilidad de las aplicaciones. Sin embargo, en situaciones que utilizan grupos de desarrollo o pruebas más pequeños, el campo CR `astraResourcesScalar` se puede establecer en `Off`. De este modo se deshabilitan las solicitudes de recursos y se puede implementar en clústeres más pequeños.	cadena	• `Default` (Este es el valor predeterminado) • `Off`

<code>crds</code>

Sus selecciones en esta sección determinan cómo Astra Control Center debe manejar los CRD.

Ajuste	Orientación	Tipo	Ejemplo
`crds.externalCertManager`	Si utiliza un administrador de certificados externo, cambie `externalCertManager` para `true`. El valor predeterminado `false` Hace que Astra Control Center instale sus propios CRD de administrador de certificados durante la instalación. Los crds son objetos de todo el clúster y su instalación podría tener un impacto en otras partes del clúster. Puede utilizar este indicador para indicar a Astra Control Center que el administrador del clúster instalará y gestionará estos CRD fuera de Astra Control Center.	Booleano	`False` (este valor es el predeterminado)
`crds.externalTraefik`	De forma predeterminada, Astra Control Center instalará los CRD de Traefik necesarios. Los crds son objetos de todo el clúster y su instalación podría tener un impacto en otras partes del clúster. Puede utilizar este indicador para indicar a Astra Control Center que el administrador del clúster instalará y gestionará estos CRD fuera de Astra Control Center.	Booleano	`False` (este valor es el predeterminado)

```
<strong>astra_control_center.yaml</strong>
```

```
apiVersion: astra.netapp.io/v1
kind: AstraControlCenter
metadata:
  name: astra
spec:
  accountName: "Example"
  astraVersion: "ASTRA_VERSION"
  astraAddress: "astra.example.com"
  autoSupport:
    enrolled: true
  email: "[admin@example.com]"
  firstName: "SRE"
  lastName: "Admin"
  imageRegistry:
    name: "[your_registry_path]"
    secret: "astra-registry-cred"
  storageClass: "ontap-gold"
  volumeReclaimPolicy: "Retain"
  ingressType: "Generic"
  astraResourcesScaler: "Default"
  additionalValues: {}
  crds:
    externalTraefik: false
    externalCertManager: false
```

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.yaml -n [netapp-acc or custom
namespace]
```

Respuesta de ejemplo:

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

Comprobar el estado del sistema

Puede verificar el estado del sistema con comandos kubectl. Si prefiere utilizar OpenShift, puede utilizar comandos de OC comparables para realizar los pasos de verificación.

Pasos

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.

Ejemplo de respuesta

```
NAME                                          READY   STATUS
RESTARTS        AGE
acc-helm-repo-76d8d845c9-ggds2                1/1     Running   0
14m
activity-6cc67ff9f4-z48mr                     1/1     Running   2
(8m32s ago)   9m
api-token-authentication-7s67v                1/1     Running   0
8m56s
api-token-authentication-bplb4                1/1     Running   0
8m56s
api-token-authentication-p2c9z                1/1     Running   0
8m56s
asup-6cdfbc6795-md8vn                         1/1     Running   0
9m14s
authentication-9477567db-8hnc9                1/1     Running   0
7m4s
bucketservice-f4dbdfcd6-wqzkw                 1/1     Running   0
8m48s
cert-manager-bb756c7c4-wm2cv                  1/1     Running   0
14m
cert-manager-cainjector-c9bb86786-8wrf5       1/1     Running   0
14m
cert-manager-webhook-dd465db99-j2w4x          1/1     Running   0
14m
certificates-68dff9cdd6-kcvml                 1/1     Running   2
(8m43s ago)   9m2s
certificates-68dff9cdd6-rsnsb                 1/1     Running   0
9m2s
cloud-extension-69d48c956c-2s8dt              1/1     Running   3
(8m43s ago)   9m24s
cloud-insights-service-7c4f48b978-7gvlh       1/1     Running   3
(8m50s ago)   9m28s
composite-compute-7d9ff5f68-nxbhl             1/1     Running   0
8m51s
composite-volume-57b4756d64-nl66d             1/1     Running   0
9m13s
credentials-6dbc55f89f-qpzff                  1/1     Running   0
11m
entitlement-67bfb6d7-gl6kp                    1/1     Running   4
(8m33s ago)   9m38s
features-856cc4dccc-mxbdb                     1/1     Running   0
9m20s
fluent-bit-ds-4rtsp                           1/1     Running   0
6m54s
```

```
fluent-bit-ds-9rqll                        1/1     Running   0
6m54s
fluent-bit-ds-w5mp7                        1/1     Running   0
6m54s
graphql-server-7c7cc49776-jz2kn            1/1     Running   0
2m29s
identity-87c59c975-9jpnf                   1/1     Running   0
9m6s
influxdb2-0                                1/1     Running   0
13m
keycloak-operator-84ff6d59d4-qcnmc         1/1     Running   0
7m1s
krakend-cbf6c7df9-mdtzv                    1/1     Running   0
2m30s
license-5b888b78bf-plj6j                   1/1     Running   0
9m32s
login-ui-846b4664dd-fz8hv                  1/1     Running   0
2m24s
loki-0                                     1/1     Running   0
13m
metrics-facade-779cc9774-n26rw             1/1     Running   0
9m18s
monitoring-operator-974db78f-pkspq         2/2     Running   0
6m58s
nats-0                                     1/1     Running   0
13m
nats-1                                     1/1     Running   0
13m
nats-2                                     1/1     Running   0
13m
nautilus-7bdc7ddc54-49tfn                  1/1     Running   0
7m50s
nautilus-7bdc7ddc54-cwc79                  1/1     Running   0
9m36s
openapi-5584ff9f46-gbrdj                   1/1     Running   0
9m17s
openapi-5584ff9f46-z9mzk                   1/1     Running   0
9m17s
packages-bfc58cc98-lpxq9                   1/1     Running   0
8m58s
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   3
(6m15s ago)   6m56s
polaris-keycloak-1                          1/1     Running   0
4m22s
polaris-keycloak-2                          1/1     Running   0
3m41s
polaris-keycloak-db-0                       1/1     Running   0
6m56s
polaris-keycloak-db-1                       1/1     Running   0
4m23s
polaris-keycloak-db-2                       1/1     Running   0
3m36s
polaris-mongodb-0                           2/2     Running   0
13m
polaris-mongodb-1                           2/2     Running   0
13m
polaris-mongodb-2                           2/2     Running   0
12m
polaris-ui-5ccff47897-8rzgh                 1/1     Running   0
2m33s
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-6cb7bfc49b-p54xm             1/1     Running   1
(8m29s ago)   9m31s
storage-backend-metrics-5c77994586-kjn48    1/1     Running   0
8m52s
storage-provider-769fdc858c-62w54           1/1     Running   0
8m54s
task-service-9ffc484c5-kx9f4                1/1     Running   3
(8m44s ago)   9m34s
telegraf-ds-bphb9                           1/1     Running   0
6m54s
telegraf-ds-rtsm2                           1/1     Running   0
6m54s
telegraf-ds-s9h5h                           1/1     Running   0
6m54s
telegraf-rs-lbpv7                           1/1     Running   0
6m54s
telemetry-service-57cfb998db-zjx78          1/1     Running   1
(8m40s ago)   9m26s
tenancy-5d5dfbcf9f-vmbxh                    1/1     Running   0
9m5s
```

```
traefik-7b87c4c474-jmgp2                    1/1     Running   0
2m24s
traefik-7b87c4c474-t9k8x                    1/1     Running   0
2m24s
trident-svc-c78f5b6bd-nwdsq                 1/1     Running   0
9m22s
vault-controller-55bbc96668-c6425           1/1     Running   0
11m
vault-controller-55bbc96668-lq9n9           1/1     Running   0
11m
vault-controller-55bbc96668-rfkgg           1/1     Running   0
11m
```

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 que se produzca un error de registro del clúster que se indica en los registros, puede volver a intentar realizar el registro a través de la "Añada el flujo de trabajo del clúster en la interfaz de usuario de" O API.

3. Cuando todos los pods estén en ejecución, verifique que la instalación se haya realizado correctamente (`READY` es `True`) Y obtenga la contraseña de configuración inicial que utilizará cuando inicie sesión en Astra Control Center:

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

Respuesta:

```
NAME    UUID                                   VERSION     ADDRESS
READY
astra   9aa5fdae-4214-4cb7-9976-5d8b4c0ce27f  22.11.0-82  10.111.111.111
True
```

> Copie el valor de UUID. La contraseña es `ACC-` Seguido del valor UUID (`ACC-[UUID]` o, en este ejemplo, `ACC-9aa5fdae-4214-4cb7-9976-5d8b4c0ce27f`).

Configure la entrada para el equilibrio de carga

Puede configurar un controlador de entrada de Kubernetes que gestione el acceso externo a los servicios. Estos procedimientos proporcionan ejemplos de configuración para un controlador de entrada si utilizó el valor predeterminado de `ingressType: "Generic"` En el recurso personalizado Astra Control Center (`astra_control_center.yaml`). No es necesario utilizar este procedimiento si se ha especificado `ingressType: "AccTraefik"` En el recurso personalizado Astra Control Center (`astra_control_center.yaml`).

Después de implementar Astra Control Center, deberá configurar el controlador Ingress para exponer Astra Control Center con una dirección URL.

Los pasos de configuración varían en función del tipo de controlador de entrada que utilice. Astra Control Center admite muchos tipos de controladores Ingress. Estos procedimientos de configuración ofrecen ejemplos de los siguientes tipos de controladores de entrada:

- Entrada Istio
- 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.

Pasos para la entrada de Istio

1. Configurar la entrada de Istio.

   > En este procedimiento se asume que Istio se implementa utilizando el perfil de configuración "predeterminado".

2. Recopile o cree el certificado y el archivo de claves privadas deseados para la puerta de enlace de entrada.

   Es posible usar un certificado firmado por CA o autofirmado. El nombre común debe ser la dirección Astra (FQDN).

   Comando de ejemplo:

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

3. Cree un secreto `tls secret name` de tipo `kubernetes.io/tls` Para una clave privada TLS y un certificado en `istio-system namespace` Tal como se describe en los secretos TLS.

   Comando de ejemplo:

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

El nombre del secreto debe coincidir con el `spec.tls.secretName` proporcionado en `istio-ingress.yaml` archivo.

4. Implemente un recurso de entrada en `netapp-acc` espacio de nombres (o con nombre personalizado) mediante el tipo de recurso v1 para un esquema (`istio-Ingress.yaml` se utiliza en este ejemplo):

```
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: [netapp-acc or custom namespace]
spec:
  ingressClassName: istio
  tls:
  - hosts:
    - <ACC address>
    secretName: [tls secret name]
  rules:
  - host: [ACC address]
    http:
      paths:
      - path: /
        pathType: Prefix
        backend:
          service:
            name: traefik
            port:
              number: 80
```

5. Aplicar los cambios:

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

6. Compruebe el estado de la entrada:

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

Respuesta:

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

7. Finalice la instalación de Astra Control Center.

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` espacio de nombres (o con nombre personalizado) mediante el tipo de recurso v1 para un esquema (`nginx-Ingress.yaml` se utiliza en este ejemplo):

```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 address>
    http:
      paths:
        - path:
          backend:
            service:
              name: traefik
              port:
                number: 80
          pathType: ImplementationSpecific
```

3. Aplicar los cambios:

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

> NetApp recomienda la instalación de la controladora nginx como una puesta en marcha en lugar de como una `daemonSet`.

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 navegador, introduzca el FQDN (incluido el `https://` prefijo) que utilizó en el `astraAddress` en la `astra_control_center.yaml` CR cuando Ha instalado Astra Control Center.
2. Acepte los certificados autofirmados si se le solicita.

   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.yaml` CR cuando Ha instalado Astra Control Center, seguido de la contraseña de configuración inicial (`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 éste es su primer inicio de sesión y olvida la contraseña y no se han creado otras cuentas de usuario administrativas, póngase en contacto con "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 deploy/acc-operator-controller-manager -n netapp-acc-
operator -c manager -f
```

El futuro

- (Opcional) en función de su entorno, post-instalación completa "pasos de configuración".
- Complete la implementación llevando a cabo "tareas de configuración".

=
:allow-uri-read:

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

- **Requisitos ambientales cumplidos**: "Antes de comenzar la instalación, prepare su entorno para la implementación de Astra Control Center".
- **Operadores de cluster sanos y servicios API**:
  - En el clúster de OpenShift, asegúrese de que todos los operadores de clúster se encuentran en buen estado:

    ```
    oc get clusteroperators
    ```

  - En el clúster OpenShift, asegúrese de que todos los servicios API se encuentran en buen estado:

    ```
    oc get apiservices
    ```

- **Dirección FQDN**: Obtenga una dirección FQDN para Astra Control Center en su centro de datos.
- **Permisos de OpenShift**: Obtenga los permisos necesarios y acceda a Red Hat OpenShift Container Platform para realizar los pasos de instalación descritos.
- **Administrador de certificados configurado**: Si ya existe un administrador de certificados en el clúster, deberá realizar algunas "requisitos previos" Por lo tanto, Astra Control Center no instala su propio administrador de certificados. De forma predeterminada, Astra Control Center instala su propio administrador de certificados durante la instalación.
- **Controlador de entrada de Kubernetes**: Si tiene un controlador de entrada de Kubernetes que gestiona el acceso externo a servicios, como el equilibrio de carga en un clúster, debe configurarlo para su uso con

Astra Control Center:

a. Crear el espacio de nombres del operador:

```
oc create namespace netapp-acc-operator
```

b. "Completar la configuración" para el tipo de controlador de entrada.

Pasos

- Descargue y extraiga 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 extraiga Astra Control Center

1. Vaya a la "Página de descargas de Astra Control Center Evaluation" En el sitio de soporte de NetApp.
2. Descargue el paquete que contiene Astra Control Center (`astra-control-center-[version].tar.gz`).
3. (Recomendado pero opcional) Descargue el paquete de certificados y firmas para Astra Control Center (`astra-control-center-certs-[version].tar.gz`) para verificar la firma del paquete:

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

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

Se mostrará la salida `Verified OK` después de una verificación correcta.

4. Extraiga las imágenes del paquete Astra Control Center:

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

Instale el complemento Astra kubectl de NetApp

El complemento de la línea de comandos Astra bectl de NetApp ahorra tiempo en la realización de tareas comunes asociadas a la puesta en marcha y la actualización de Astra Control Center.

Lo que necesitará

NetApp proporciona binarios de complementos para diferentes arquitecturas de CPU y sistemas operativos.

Debe saber qué CPU y sistema operativo tiene antes de realizar esta tarea.

Pasos

1. Enumere los binarios disponibles del complemento Astra kubectl de NetApp, y anote el nombre del archivo que necesita para el sistema operativo y la arquitectura de CPU:

   > La biblioteca de complementos kubectl forma parte del paquete tar y se extrae en la carpeta `kubectl-astra`.

   ```
   ls kubectl-astra/
   ```

2. Mueva el binario correcto a la ruta actual y cambie el nombre a. `kubectl-astra`:

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

Agregue las imágenes al registro local

1. Complete la secuencia de pasos apropiada para el motor del contenedor:

Docker

1. Cambie al directorio raíz del tarball. Debería ver este archivo y directorio:

   ```
   acc.manifest.bundle.yaml
   acc/
   ```

2. Inserte las imágenes del paquete en el directorio de imágenes de Astra Control Center en su registro local. Realice las siguientes sustituciones antes de ejecutar el `push-images` comando:
   - Sustituya <BUNDLE_FILE> por el nombre del archivo Astra Control Bundle (`acc.manifest.bundle.yaml`).
   - Sustituya <MY_FULL_REGISTRY_PATH> por la URL del repositorio de Docker; por ejemplo, "<a href="https://<docker-registry>"" class="bare">https://<docker-registry>".
   - Reemplace <MY_REGISTRY_USER> por el nombre de usuario.
   - Sustituya <MY_REGISTRY_TOKEN> por un token autorizado para el registro.

   ```
   kubectl astra packages push-images -m <BUNDLE_FILE> -r
   <MY_FULL_REGISTRY_PATH> -u <MY_REGISTRY_USER> -p
   <MY_REGISTRY_TOKEN>
   ```

Podman

1. Cambie al directorio raíz del tarball. Debería ver este archivo y directorio:

   ```
   acc.manifest.bundle.yaml
   acc/
   ```

2. Inicie sesión en su registro:

   ```
   podman login <YOUR_REGISTRY>
   ```

3. Prepare y ejecute una de las siguientes secuencias de comandos personalizadas para la versión de Podman que utilice. Sustituya <MY_FULL_REGISTRY_PATH> por la URL del repositorio que incluye cualquier subdirectorio.

   ```
   <strong>Podman 4</strong>
   ```

```
export REGISTRY=<MY_FULL_REGISTRY_PATH>
export PACKAGENAME=acc
export PACKAGEVERSION=22.11.0-82
export DIRECTORYNAME=acc
for astraImageFile in $(ls ${DIRECTORYNAME}/images/*.tar) ; do
astraImage=$(podman load --input ${astraImageFile} | sed 's/Loaded
image: //')
astraImageNoPath=$(echo ${astraImage} | sed 's:.*/::')
podman tag ${astraImageNoPath} ${REGISTRY}/netapp/astra/
${PACKAGENAME}/${PACKAGEVERSION}/${astraImageNoPath}
podman push ${REGISTRY}/netapp/astra/${PACKAGENAME}/${
PACKAGEVERSION}/${astraImageNoPath}
done
```

```
<strong>Podman 3</strong>
```

```
export REGISTRY=<MY_FULL_REGISTRY_PATH>
export PACKAGENAME=acc
export PACKAGEVERSION=22.11.0-82
export DIRECTORYNAME=acc
for astraImageFile in $(ls ${DIRECTORYNAME}/images/*.tar) ; do
astraImage=$(podman load --input ${astraImageFile} | sed 's/Loaded
image: //')
astraImageNoPath=$(echo ${astraImage} | sed 's:.*/::')
podman tag ${astraImageNoPath} ${REGISTRY}/netapp/astra/
${PACKAGENAME}/${PACKAGEVERSION}/${astraImageNoPath}
podman push ${REGISTRY}/netapp/astra/${PACKAGENAME}/${
PACKAGEVERSION}/${astraImageNoPath}
done
```

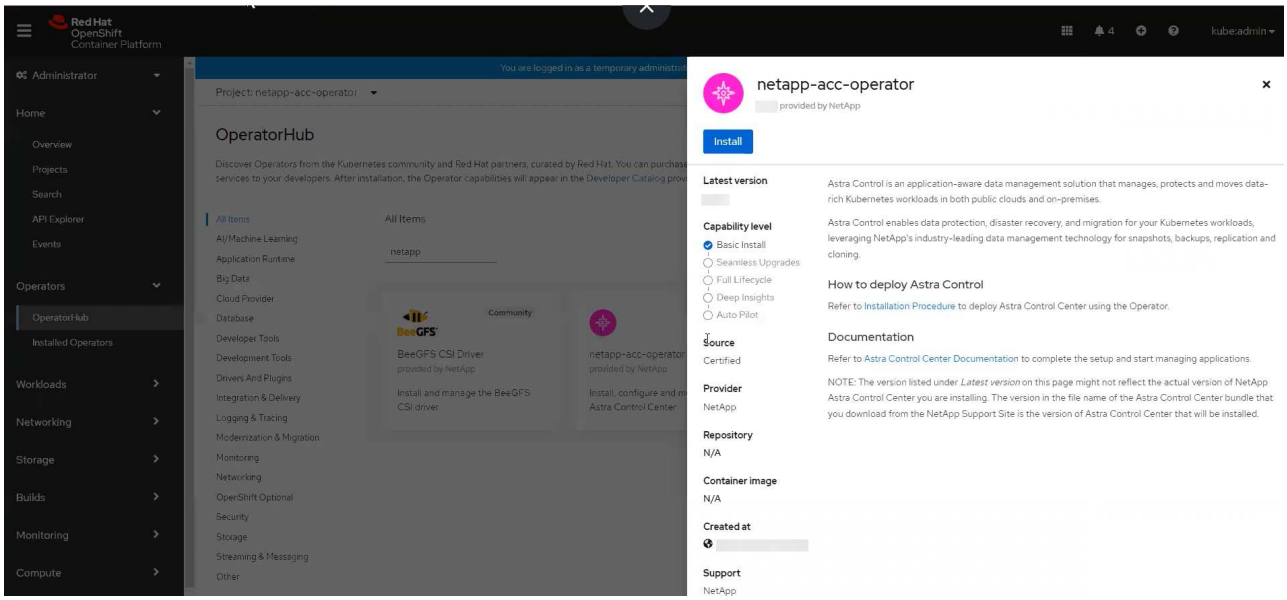
> La ruta de acceso de imagen que crea el script debe parecerse a la siguiente, dependiendo de la configuración del Registro: https://netappdownloads.jfrog.io/docker-astra-control-prod/netapp/astra/acc/22.11.0-82/image:version

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**.

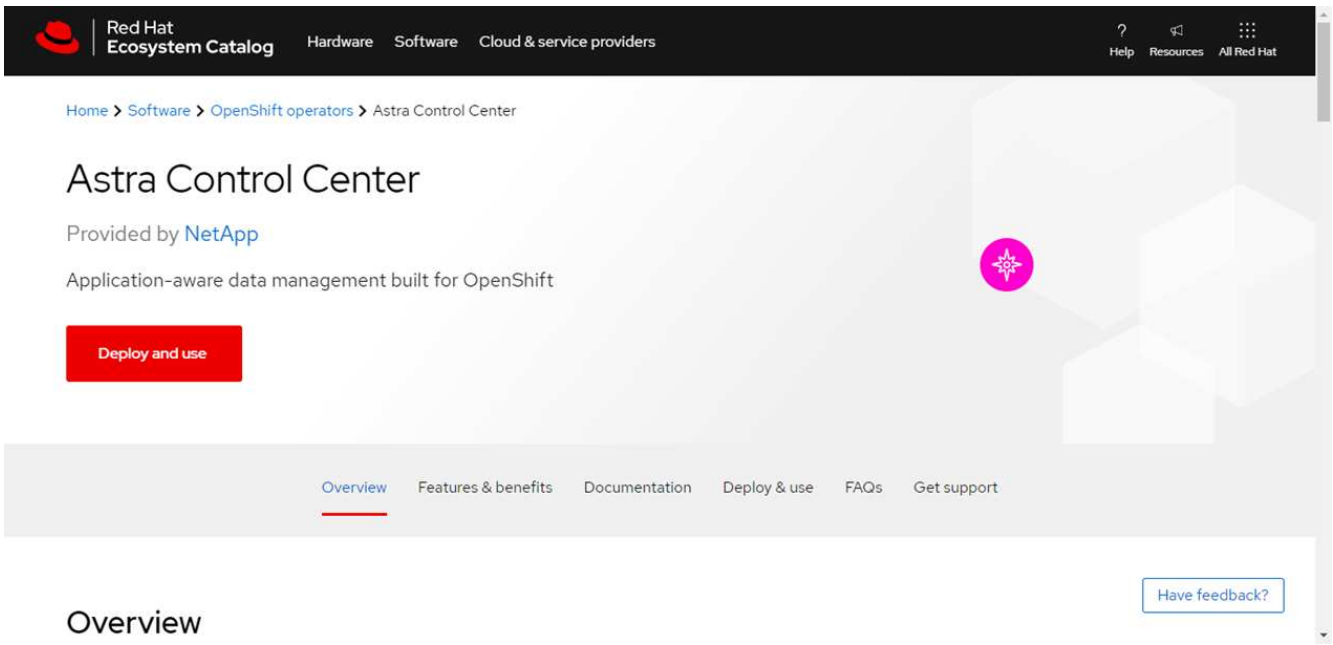
iii. Busque y seleccione el operador Centro de control Astra de NetApp.



- 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**.

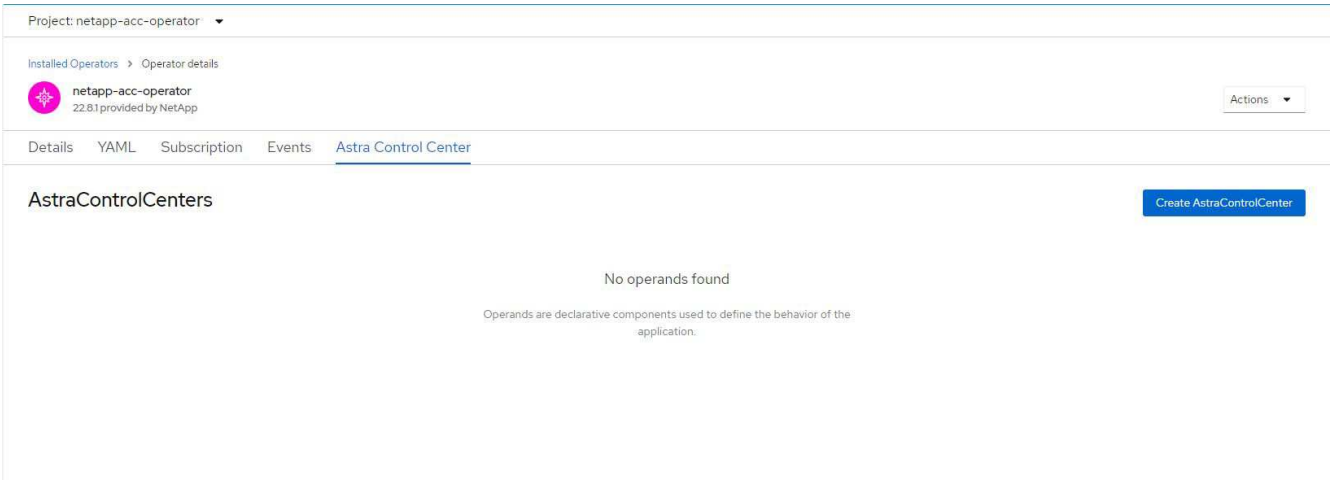
> Si ha seleccionado una estrategia de aprobación manual, se le pedirá que apruebe el plan de instalación manual para este operador.

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

Instalar Astra Control Center

1. Desde la consola de la pestaña **Astra Control Center** del operador Astra Control Center, seleccione **Crear AstraControlCenter**

.



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

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

   b. Agregue etiquetas para Astra Control Center.

   c. Habilitar o deshabilitar AutoSupport. Se recomienda conservar la funcionalidad de AutoSupport.

   d. Introduzca el FQDN o la dirección IP de Astra Control Center. No entre `http://` o. `https://` en el campo de dirección.

   e. Introduzca la versión de Astra Control Center; por ejemplo, 22.04.1.

   f. Introduzca un nombre de cuenta, una dirección de correo electrónico y un apellido de administrador.

   g. Seleccione una política de reclamaciones de volumen de `Retain`, `Recycle`, o. `Delete`. El valor predeterminado es `Retain`.

   h. Seleccione el tipo de entrada:

      - **`Generic`** (`ingressType: "Generic"`) (Predeterminado)

        Utilice esta opción cuando tenga otro controlador de entrada en uso o prefiera utilizar su propio controlador de entrada. Después de implementar Astra Control Center, deberá configurar el "controlador de entrada" Para exponer Astra Control Center con una URL.

- **AccTraefik** (`ingressType: "AccTraefik"`)

  Utilice esta opción cuando prefiera no configurar un controlador de entrada. Esto despliega el Astra Control Center `traefik` Puerta de enlace como servicio de tipo "LoadBalancer" de Kubernetes.

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 aún 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".

a. En **Registro de imágenes**, introduzca la ruta de registro de la imagen del contenedor local. No entre `http://` o. `https://` en el campo de dirección.
b. Si utiliza un registro de imágenes que requiere autenticación, introduzca el secreto de imagen.

   > Si utiliza un registro que requiere autenticación, cree un secreto en el clúster.

c. Introduzca el nombre del administrador.
d. Configure el escalado de recursos.
e. Proporcione la clase de almacenamiento predeterminada.

   > Si se configura una clase de almacenamiento predeterminada, asegúrese de que es la única clase de almacenamiento que tiene la anotación predeterminada.

f. Defina las preferencias de manejo de CRD.

3. Seleccione la vista YAML para revisar los ajustes seleccionados.
4. Seleccione `Create`.

Cree un secreto de registro

Si utiliza un registro que requiere autenticación, cree un secreto en el clúster OpenShift y escriba el nombre secreto en el `Create AstraControlCenter` campo de formulario.

1. Cree un espacio de nombres para el operador Astra Control Center:

```
oc create ns [netapp-acc-operator or custom namespace]
```

2. Cree un secreto en este espacio de nombres:

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

Astra Control sólo admite secretos de registro Docker.

3. Complete los campos restantes en El campo de formulario Create AstraControlCenter.

El futuro

Complete el "pasos restantes" Para verificar que Astra Control Center se ha instalado correctamente, configure un controlador de entrada (opcional) e inicie sesión en la interfaz de usuario. Además, tendrá que realizar "tareas de configuración" tras completar la instalación.

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), Google Cloud Platform (GCP) y Microsoft Azure con una 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 Google Cloud Platform
- Ponga en marcha Astra Control Center en Microsoft Azure

Puede gestionar sus aplicaciones en distribuciones con clústeres de Kubernetes autogestionados, como OpenShift Container Platform (OCP). Sólo se validan los clústeres OCP autogestionados para la implantación de Astra Control Center.

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).

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
- Si utiliza OCP, los permisos de Red Hat OpenShift Container Platform (OCP) (en el 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 BlueXP, anteriormente Cloud Manager)	Astra Trident 21.04 o posterior instalado y configurado y NetApp ONTAP versión 9.5 o posterior como back-end de almacenamiento
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. 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.

Componente	Requisito
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 de Kubernetes a NetApp BlueXP (anteriormente Cloud Manager). Los proporciona 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`

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. Configuración de BlueXP de NetApp para AWS.
5. Instale Astra Control Center para AWS.

Compruebe que dispone de suficientes permisos IAM

Asegúrese de tener suficientes roles y permisos de IAM que le permiten instalar un clúster RedHat OpenShift y un conector BlueXP de NetApp (anteriormente Cloud Manager).

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".

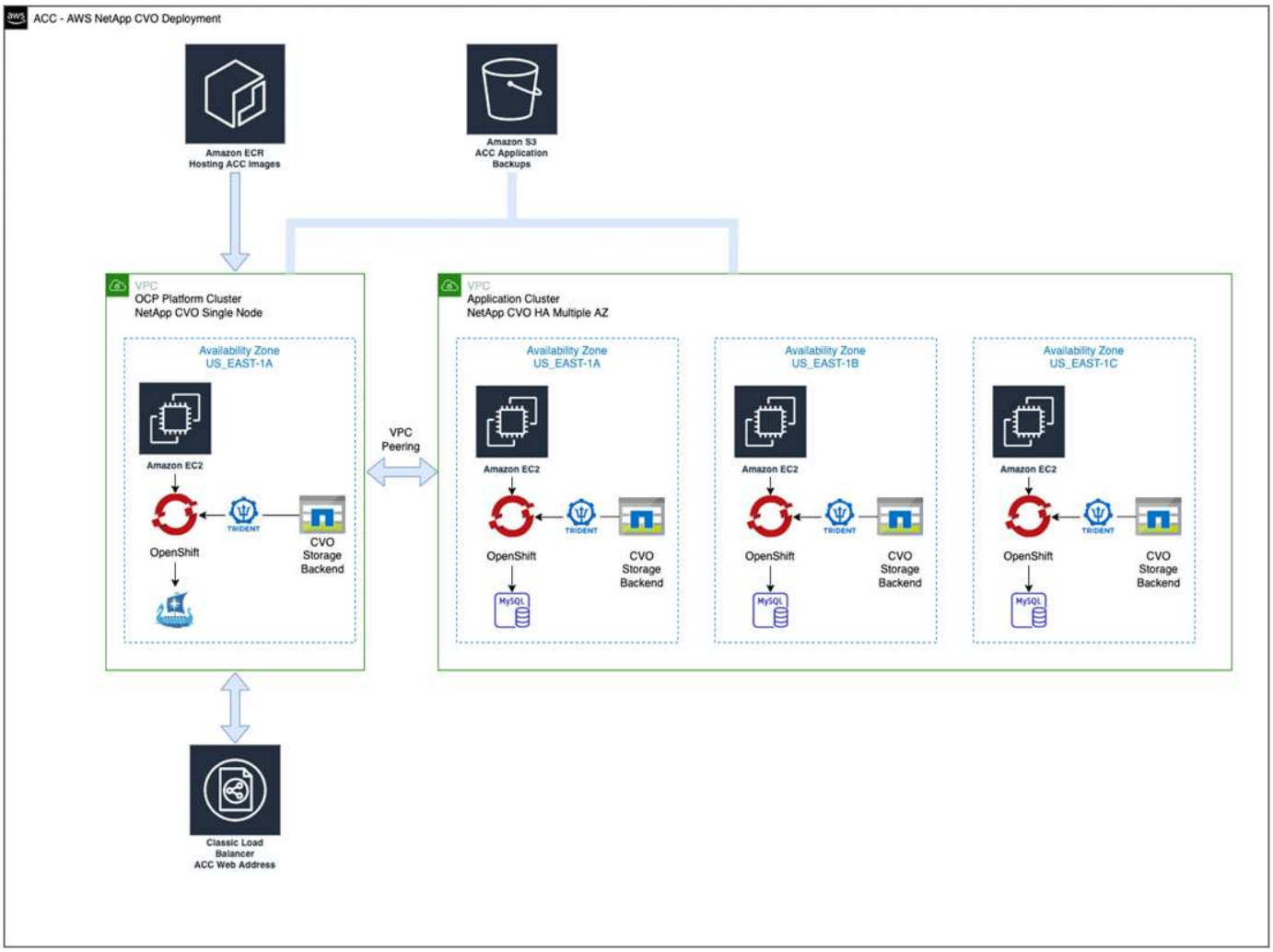
1. Cree una red virtual AWS.
2. Revise las instancias de computación EC2. Puede ser un servidor con configuración básica o máquinas virtuales en AWS.
3. Si el tipo de instancia no coincide con los requisitos mínimos de recursos de Astra para los nodos maestros y trabajadores, cambie el tipo de instancia 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:



Configuración de BlueXP de NetApp para AWS

Con NetApp BlueXP (anteriormente 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 BlueXP para completar los siguientes pasos. Consulte lo siguiente:

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

Pasos

1. Agregue sus credenciales a BlueXP.
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.

Instale Astra Control Center para AWS

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

> AWS utiliza el tipo de bloque Generic S3.

Ponga en marcha Astra Control Center en Google Cloud Platform

Puede poner en marcha Astra Control Center en un clúster de Kubernetes autogestionado alojado en un cloud público de Google Cloud Platform (GCP).

Qué necesitará para GCP

Antes de implementar Astra Control Center en GCP, 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
- Si se utiliza OCP, Red Hat OpenShift Container Platform (OCP) 4.10
- Si utiliza OCP, los permisos de Red Hat OpenShift Container Platform (OCP) (en el nivel de espacio de nombres para crear pods)
- Cuenta de servicio de GCP con permisos que le permiten crear cubos y conectores

Requisitos del entorno operativo para GCP

> 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

Componente	Requisito
Nodos de trabajo (requisitos de computación de GCP)	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 GCP)	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 BlueXP, anteriormente Cloud Manager)	Astra Trident 21.04 o posterior instalado y configurado y NetApp ONTAP versión 9.5 o posterior como back-end de almacenamiento
Registro de imágenes	Debe tener un registro privado existente, como Google 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. 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 BlueXP de NetApp. Los proporciona 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`

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 de puesta en marcha para GCP

A continuación se ofrece una descripción general del proceso de instalación de Astra Control Center en un clúster OCP autogestionado en GCP con Cloud Volumes ONTAP como back-end de almacenamiento.

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

1. Instale un clúster RedHat OpenShift en GCP.
2. Cree un proyecto de GCP y una nube privada virtual.
3. Compruebe que dispone de suficientes permisos IAM.
4. Configure GCP.
5. Configuración de NetApp BlueXP para GCP.
6. Instale Astra Control Center para GCP.

Instale un clúster RedHat OpenShift en GCP

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

Para obtener instrucciones de instalación, consulte lo siguiente:

- "Instalar un clúster OpenShift en GCP"
- "Creación de una cuenta de servicio de GCP"

Cree un proyecto de GCP y una nube privada virtual

Cree al menos un proyecto de GCP y una nube privada virtual (VPC).

> OpenShift podría crear sus propios grupos de recursos. Además de ellas, debe definir también un VPC de GCP. 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 que le permiten instalar un clúster RedHat OpenShift y un conector BlueXP de NetApp (anteriormente Cloud Manager).

Consulte "Credenciales y permisos iniciales de GCP".

Configure GCP

A continuación, configure GCP para crear un VPC, configure instancias de computación, cree un almacenamiento de objetos de Google Cloud, cree un Registro de contenedor de Google para alojar las imágenes de Astra Control Center y empuje las imágenes a este registro.

Siga la documentación de GCP para completar los siguientes pasos. Consulte instalación del clúster OpenShift en GCP.

1. Cree un proyecto de GCP y VPC en el GCP que planea utilizar para el clúster de OCP con el back-end de CVO.
2. Revise las instancias de computación. Puede tratarse de un servidor de configuración básica o máquinas virtuales en GCP.
3. Si el tipo de instancia no coincide con los requisitos mínimos de recursos de Astra para los nodos maestro y trabajador, cambie el tipo de instancia de GCP para que cumpla los requisitos de Astra. Consulte "Requisitos del Centro de Control de Astra".

4. Cree al menos un bloque de almacenamiento en cloud de GCP para almacenar sus backups.
5. Crear un secreto, que es necesario para el acceso a bloques.
6. Cree un registro de Google Container para alojar todas las imágenes de Astra Control Center.
7. Configure el acceso al registro de contenedores de Google para inserción/extracción de Docker para todas las imágenes de Astra Control Center.

   Ejemplo: Las imágenes ACC se pueden insertar en este registro introduciendo la siguiente secuencia de comandos:

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

   Este script requiere un archivo de manifiesto de Astra Control Center y su ubicación del Registro de imágenes de Google.

   Ejemplo:

   ```
   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. Configure zonas DNS.

Configuración de NetApp BlueXP para GCP

Con NetApp BlueXP (anteriormente Cloud Manager), cree un espacio de trabajo, añada un conector a GCP, cree un entorno de trabajo e importe el clúster.

Siga la documentación de BlueXP para completar los siguientes pasos. Consulte "Introducción a Cloud Volumes ONTAP en GCP".

Lo que necesitará

- Acceso a la cuenta de servicio de GCP con los permisos y roles de IAM necesarios

Pasos

1. Agregue sus credenciales a BlueXP. Consulte "Adición de cuentas de GCP".
2. Agregue un conector para GCP.

a. Elija "GCP" como el proveedor.
b. Introduzca las credenciales de GCP. Consulte "Creación de un conector en GCP desde BlueXP".
c. Asegúrese de que el conector está en marcha y cambie a dicho conector.

3. Cree un entorno de trabajo para su entorno de cloud.
   a. Ubicación: "GCP"
   b. Tipo: "Cloud Volumes ONTAP ha"
4. 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. Tenga en cuenta las clases de almacenamiento del clúster de 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.
5. 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 puesta en marcha del nodo que se ejecutan en GCP.

Instale Astra Control Center para GCP

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

> GCP utiliza el tipo de bloque Generic S3.

1. Genere el secreto Docker para obtener imágenes de la instalación de Astra Control Center:

```
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
```

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
- Si se utiliza OCP, Red Hat OpenShift Container Platform (OCP) 4.8
- Si utiliza OCP, los permisos de Red Hat OpenShift Container Platform (OCP) (en el 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
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 BlueXP)	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. Es necesario habilitar el acceso anónimo para extraer imágenes RTIC para realizar copias de seguridad.

Componente	Requisito
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 BlueXP de NetApp. Los proporciona 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`

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. Configuración de NetApp BlueXP (anteriormente Cloud Manager) para Azure.
6. Instalar y configurar Astra Control Center para Azure.

Instale un clúster RedHat OpenShift en Azure

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

Para obtener instrucciones de instalación, consulte lo siguiente:

- "Instalando el clúster de OpenShift en Azure".
- "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 IAM para poder instalar un clúster RedHat OpenShift y un conector BlueXP 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.

Configuración de NetApp BlueXP (anteriormente Cloud Manager) para Azure

Con BlueXP (anteriormente 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 BlueXP para completar los siguientes pasos. Consulte "Introducción a BlueXP en Azure".

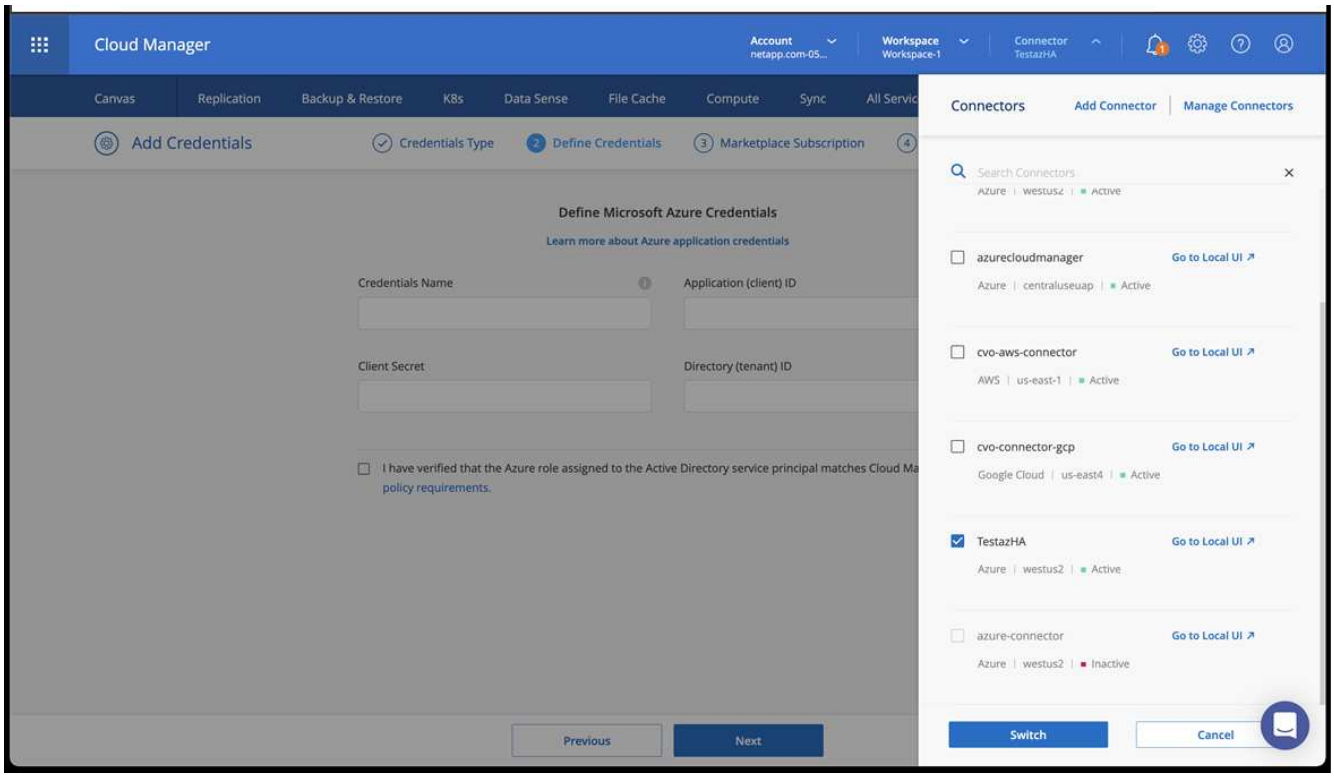
Lo que necesitará

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

Pasos

1. Agregue sus credenciales a BlueXP.
2. Agregue un conector para Azure. Consulte "Políticas de BlueXP".
   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).

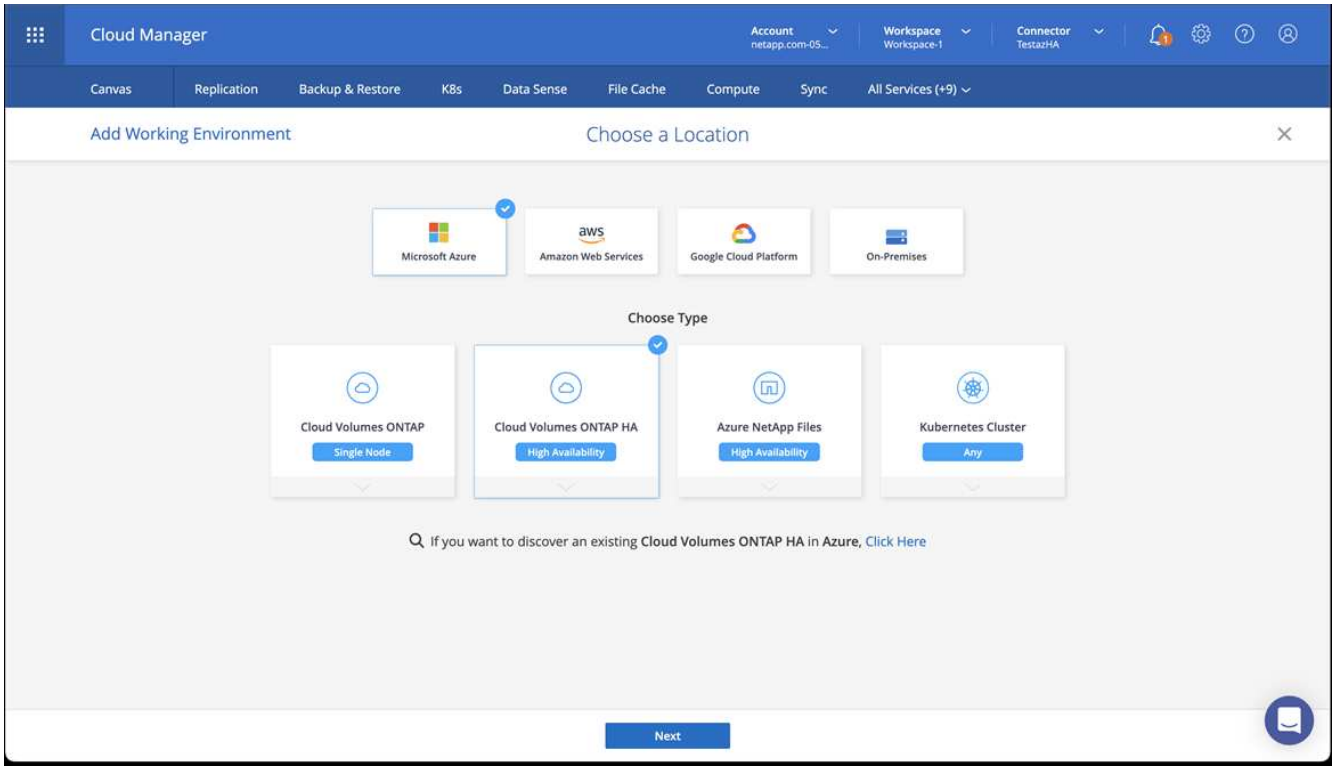
      Consulte "Creación de un conector en Azure desde BlueXPr".
3. Asegúrese de que el conector está en marcha y cambie a dicho conector.



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

   a. Ubicación: "Microsoft Azure".

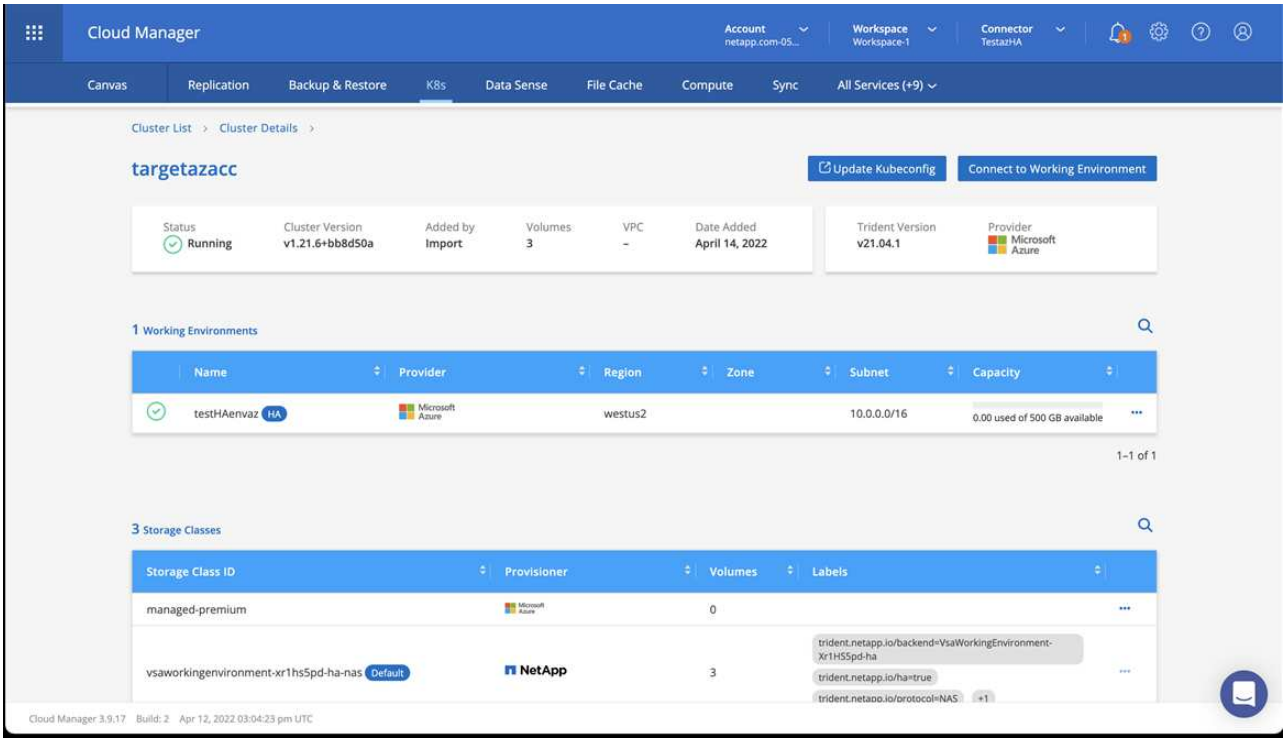
   b. Tipo: "Cloud Volumes ONTAP ha".



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 para Azure

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".

=
:allow-uri-read: