Manos a la obra

Astra Control Center

NetApp
November 21, 2023

Tabla de contenidos

Manos a la obra

Requisitos del Centro de Control de Astra

Comience verificando la preparación de su entorno operativo, clústeres de aplicaciones, aplicaciones, licencias y explorador web.

Requisitos del entorno operativo

Astra Control Center requiere uno de los siguientes tipos de entornos operativos:

- OpenShift Container Platform de Red Hat 4.6.8, 4.7 o 4.8
- Ranchero 2.5
- Kubernetes 1.19 a 1.21 (incluido 1.21.x)

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 de back-end ONTAP	300 GB como mínimo disponible
Nodos de trabajo	Al menos 3 nodos de trabajo en total, con 4 núcleos de CPU y 12 GB de RAM 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
Resolución del FQDN	Método para señalar el FQDN de Astra Control Center a la dirección IP de carga equilibrada
Astra Trident	• Astra Trident 21.04 o posterior instalado y configurado si se usará la versión 9.5 o posterior de ONTAP de NetApp como back-end de almacenamiento • Astra Trident 21.10.1 o posterior instalada y configurada si se usará la vista previa de Astra Data Store como back-end de almacenamiento

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.

- **Registro de imágenes**: Debe tener un registro de imágenes Docker privado existente en el que puede insertar imágenes de creación de Astra Control Center. Debe proporcionar la dirección URL del registro de imágenes donde cargará las imágenes.
- **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 los siguientes controladores de ONTAP proporcionados por Astra Trident:

- ontap-nas
- san ontap
- ontap-san-economía

> Durante la clonación de aplicaciones en entornos OpenShift, Astra Control Center debe permitir a OpenShift montar volúmenes y cambiar la propiedad de los archivos. Por este motivo, es necesario configurar una política de exportación de volúmenes ONTAP para permitir estas operaciones. Puede hacerlo con los siguientes comandos:
>
> 1. `export-policy rule modify -vserver <storage virtual machine name> -policyname <policy name> -ruleindex 1 -superuser sys`
> 2. `export-policy rule modify -vserver <storage virtual machine name> -policyname <policy name> -ruleindex 1 -anon 65534`

> Si tiene pensado añadir un segundo entorno operativo OpenShift como recurso informático gestionado, debe asegurarse de que la función Astra Trident Volume Snapshot esté habilitada. Para habilitar y probar copias Snapshot de volumen con Astra Trident, "Consulte las instrucciones oficiales de la Astra Trident".

Requisitos del clúster de aplicaciones

Astra Control Center tiene los siguientes requisitos para los clústeres que tiene previsto gestionar desde Astra Control Center. Estos requisitos también se aplican si el clúster que tiene previsto gestionar es el clúster de entorno operativo que aloja Astra Control Center.

- La versión más reciente de Kubernetes "componente de controladora snapshot" está instalado
- Una Astra Trident "volumesnapshotclass object" ha sido definido por un administrador
- Existe una clase de almacenamiento de Kubernetes predeterminada en el clúster
- Se configura al menos una clase de almacenamiento para que use Astra Trident

> Su clúster de aplicaciones debe tener un `kubeconfig.yaml` archivo que define sólo un elemento *context*. Consulte la documentación de Kubernetes para "información sobre la creación de archivos kubeconfig".

> Cuando administre clústeres de aplicaciones en un entorno Rancher, modifique el contexto predeterminado del clúster de aplicaciones en `kubeconfig` Archivo proporcionado por Rancher para utilizar un contexto de plano de control en lugar del contexto del servidor API Rancher. Esto reduce la carga en el servidor API de Rancher y mejora el rendimiento.

Y gestión de aplicaciones

Astra Control tiene los siguientes requisitos de gestión de aplicaciones:

- **Licencia**: Para gestionar aplicaciones mediante Astra Control Center, necesita una licencia Astra Control Center.
- **Namespaces**: Astra Control requiere que una aplicación no abarque más de un único espacio de nombres, pero un espacio de nombres puede contener más de una aplicación.

- **StorageClass**: Si instala una aplicación con StorageClass definida explícitamente y necesita clonar la aplicación, el clúster de destino para la operación de clonado debe tener el StorageClass especificado originalmente. Se producirá un error al clonar una aplicación con un tipo de almacenamiento establecido explícitamente en un clúster que no tenga el mismo tipo de almacenamiento.
- **Recursos de Kubernetes**: Las aplicaciones que usan recursos de Kubernetes no recopilados por Astra Control podrían no tener funciones completas de gestión de datos de aplicaciones. Astra Control recopila los siguientes recursos de Kubernetes:
  - Función de clúster
  - ClusterRoleBinding
  - ConfigMap
  - CustomResourceDefinition
  - Recurso personalizado
  - DemonSet
  - Puesta en marcha
  - DeploymentConfig
  - Entrada
  - MutatingWebhook
  - Claim persistente
  - Pod
  - Replicaset
  - RoleBinding
  - Función
  - Ruta
  - Secreto
  - Servicio
  - ServiceAccount
  - Statilusionados Set
  - ValidadoWebhook

Métodos de instalación de aplicaciones compatibles

Astra Control es compatible con los siguientes métodos de instalación de aplicaciones:

- **Fichero manifiesto**: Astra Control admite aplicaciones instaladas desde un archivo manifiesto mediante kubectl. Por ejemplo:

  ```
  kubectl apply -f myapp.yaml
  ```

- **Helm 3**: Si utiliza Helm para instalar aplicaciones, Astra Control requiere Helm versión 3. Es totalmente compatible con la gestión y clonación de aplicaciones instaladas con Helm 3 (o actualizadas de Helm 2 a Helm 3). No se admite la administración de aplicaciones instaladas con Helm 2.
- **Aplicaciones implementadas por el operador**: Astra Control admite aplicaciones instaladas con

operadores de ámbito de espacio de nombres. A continuación, se enumeran algunas aplicaciones que se han validado para este modelo de instalación:

- "Apache K8ssandra"
- "Jenkins CI"
- "Clúster Percona XtraDB"

> Un operador y la aplicación que instale deben usar el mismo espacio de nombres; es posible que deba modificar el archivo .yaml de despliegue para que el operador se asegure de que así sea.

Acceso a Internet

Debe determinar si tiene acceso externo a Internet. Si no lo hace, es posible que algunas funcionalidades sean limitadas, como recibir datos de supervisión y métricas de Cloud Insights de NetApp, o enviar paquetes de soporte al "Sitio de soporte de NetApp".

Licencia

Astra Control Center requiere una licencia de Astra Control Center para obtener todas las funciones. Obtenga una licencia de evaluación o una licencia completa de NetApp. Sin una licencia, no podrá:

- Defina aplicaciones personalizadas
- Cree instantáneas o clones de las aplicaciones existentes
- Configure las políticas de protección de datos

Si desea probar Astra Control Center, puede "utilice una licencia de evaluación de 90 días".

Tipo de servicio "LoadBalancer" para clústeres de Kubernetes en las instalaciones

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 los equilibradores de carga están permitidos en su entorno y no tiene uno configurado, puede usar "MetalLB" Para asignar automáticamente 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.

Requisitos de red

El entorno operativo que aloja Astra Control Center se comunica mediante los siguientes puertos TCP. Debe asegurarse de que estos puertos estén permitidos a través de cualquier firewall y configurar firewalls para permitir que cualquier tráfico de salida HTTPS que se origine en la red Astra. Algunos puertos requieren conectividad de ambos modos entre el entorno que aloja Astra Control Center y cada clúster gestionado (se indica si procede).

Origen	Destino	Puerto	Protocolo	Específico
PC cliente	Astra Control Center	443	HTTPS	Acceso de interfaz de usuario/API: Asegúrese de que este puerto está abierto de ambas formas entre el clúster que aloja a Astra Control Center y cada clúster gestionado
Consumidor de métricas	Nodo de trabajo de Astra Control Center	9090	HTTPS	Comunicación de datos de métricas: Asegúrese de que cada clúster gestionado pueda acceder a este puerto en el clúster que aloja a Astra Control Center (se requiere una comunicación bidireccional)
Astra Control Center	Servicio Cloud Insights alojado	443	HTTPS	Comunicación de Cloud Insights
Astra Control Center	Proveedor de bloques de almacenamiento Amazon S3	443	HTTPS	Comunicación del almacenamiento de Amazon S3
Astra Control Center	Active IQ de NetApp	443	HTTPS	Comunicación ActiveIQ de NetApp

Exploradores web compatibles

Astra Control Center es compatible con las versiones recientes de Firefox, Safari y Chrome con una resolución mínima de 1280 x 720.

El futuro

Vea la "inicio rápido" descripción general.

Inicio rápido para Astra Control Center

Esta página ofrece una descripción general de alto nivel de los pasos necesarios para empezar con Astra Control Center. Los vínculos de cada paso le llevan a una página que proporciona más detalles.

¡Pruébela! Si desea probar Astra Control Center, puede utilizar una licencia de evaluación de 90 días. Consulte "información sobre licencias" para obtener más detalles.

1 Revise los requisitos del clúster de Kubernetes

- Astra funciona con clústeres de Kubernetes con un back-end de almacenamiento de ONTAP configurado para Trident o un back-end de almacenamiento previo al almacén de datos de Astra.
- Los clústeres deben ejecutarse en buen estado, con al menos tres nodos de trabajo en línea.
- El clúster debe ejecutar Kubernetes.

"Más información sobre los requisitos de Astra Control Center".

2 Descargue e instale Astra Control Center

- Descargue Astra Control Center desde "Página de descargas del Centro de control de Astra del sitio de soporte de NetApp".
- Instale Astra Control Center en su entorno local.

  Opcionalmente, instale Astra Control Center utilizando Red Hat OperatorHub.

- Descubra su configuración de Trident respaldada por el back-end de almacenamiento de ONTAP. O bien, descubra su "Vista previa de Astra Data Store" los clústeres como back-end de almacenamiento.

  Instala las imágenes en un registro de OpenShift o utiliza el registro local.

"Más información sobre la instalación de Astra Control Center".

3 Complete algunas tareas de configuración inicial

- Añadir una licencia.
- Añada un clúster de Kubernetes y Astra Control Center descubre los detalles.
- Añada un back-end de almacenamiento de vista previa del almacén de datos de ONTAP o Astra.
- Opcionalmente, agregue un bucket de almacén de objetos que almacenará las copias de seguridad de la aplicación.

"Obtenga más información acerca del proceso de configuración inicial".

4 Utilice Astra Control Center

Cuando termine de configurar Astra Control Center, a continuación puede hacer lo siguiente:

- Gestionar una aplicación. "Más información sobre cómo gestionar aplicaciones".
- De manera opcional, conéctese a Cloud Insights de NetApp para mostrar métricas sobre el estado del sistema, la capacidad y el rendimiento dentro de la IU del centro de control de Astra. "Obtenga más información sobre cómo conectarse a Cloud Insights".

5 Continuar desde este Inicio rápido

"Instalar Astra Control Center".

Obtenga más información

- "Utilice la API Astra Control"

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 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
  ```

- Ha creado una dirección FQDN para Astra Control Center en su centro de datos.

Acerca de esta tarea

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

- Instala los componentes de Astra en `netapp-acc` (o espacio de nombres 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.

> Los comandos de Podman se pueden utilizar en lugar de los comandos de Docker si está utilizando Podman de Red Hat en lugar de Docker Engine.

> 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`

Pasos

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

- Descargue el paquete Astra Control Center
- Desembale el paquete y cambie el directorio
- 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
- Inicie sesión en la interfaz de usuario de Astra Control Center

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

Descargue 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
```

Desembale el paquete y cambie el directorio

1. Extraiga las imágenes:

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

2. Cambie al directorio Astra.

```
cd astra-control-center-[version]
```

Agregue las imágenes al registro local

1. 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
```

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 añadido con otro recurso Astra Trident StorageClass si así lo requiere su instalación.

h. (Para instalaciones que utilizan la vista previa de Astra Data Store) Consulte este problema conocido para "cambios adicionales necesarios" A la AYLMA.

```
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"
```

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-zb8l6               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
```

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. Compruebe la `status.deploymentState` en la respuesta para `Deployed` valor. Si la implementación no se realizó correctamente, aparece en su lugar un mensaje de error.

Utilizará la `uuid` en el siguiente paso.

```
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: ""
```

5. Para obtener la contraseña única que utilizará cuando inicie sesión en Astra Control Center, copie la `status.uuid` valor de la respuesta en el paso anterior. La contraseña es `ACC-` Seguido del valor UUID (`ACC-[UUID]` o, en este ejemplo, `ACC-c49008a5-4ef1-4c5d-a53e-830daf994116`).

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 el paquete Astra Control Center
- Desembale el paquete y cambie el directorio
- Agregue las imágenes al registro local
- Busque la página de instalación del operador
- Instale el operador
- Instalar Astra Control Center

Descargue 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 desde "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
```

Desembale el paquete y cambie el directorio

1. Extraiga las imágenes:

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

2. Cambie al directorio Astra.

```
cd astra-control-center-[version]
```

Agregue las imágenes al registro local

1. 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:

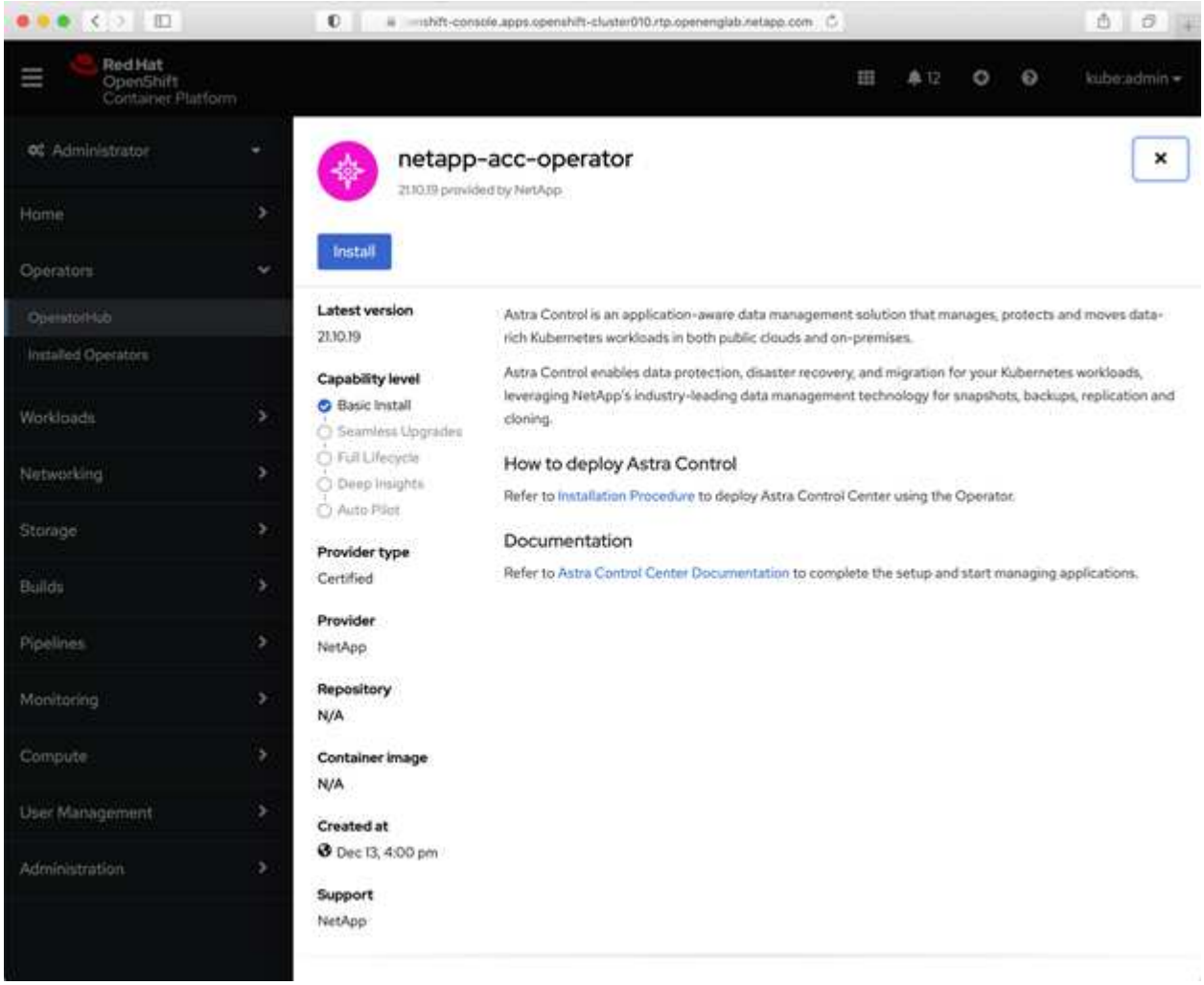
      ```
      export REGISTRY=[Docker_registry_path]
      for astraImageFile in $(ls images/*.tar) ; do
        # Load to local cache. And store the name of the loaded image
      trimming the 'Loaded images: '
        astraImage=$(docker load --input ${astraImageFile} | sed 's/Loaded
      image: //')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



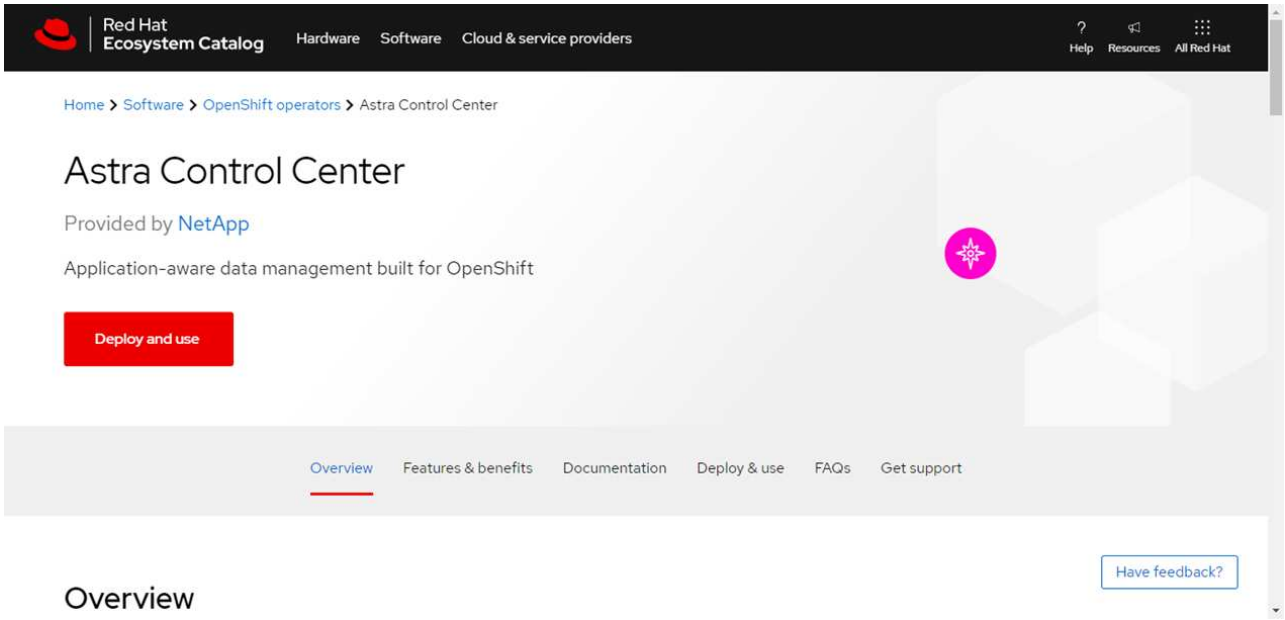
     i. Inicie sesión en la IU de OpenShift Container Platform.

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".

Configure Astra Control Center

Astra Control Center admite y supervisa ONTAP y Astra Data Store como back-end de almacenamiento. Después de instalar Astra Control Center, inicie sesión en la interfaz de usuario y cambie la contraseña, le interesa configurar una licencia, añadir clústeres, gestionar el almacenamiento y añadir bloques.

Tareas

- Agregue una licencia de Astra Control Center
- Añada el clúster
- Añada un back-end de almacenamiento
- Añadir un bucket

Agregue una licencia de Astra Control Center

Puede añadir una licencia nueva con la interfaz de usuario o. "API" Para obtener todas las funciones de Astra Control Center. Sin una licencia, el uso de Astra Control Center se limita a gestionar usuarios y agregar nuevos clústeres.

Lo que necesitará

Al descargar Astra Control Center desde "Sitio de soporte de NetApp"También puede descargar el archivo de licencia de NetApp (NLF). Asegúrese de tener acceso a este archivo de licencia.

Para actualizar una evaluación existente o una licencia completa, consulte "Actualizar una licencia existente".

Añada una licencia completa o de evaluación

Las licencias de Astra Control Center miden los recursos de CPU mediante unidades de CPU de Kubernetes. La licencia debe tener en cuenta los recursos de CPU asignados a los nodos de trabajo de todos los clústeres de Kubernetes gestionados. Antes de agregar una licencia, debe obtener el archivo de licencia (NLF) de "Sitio de soporte de NetApp".

También puede probar Astra Control Center con una licencia de evaluación, que le permite utilizar Astra Control Center durante 90 días a partir de la fecha de descarga de la licencia. Puede inscribirse para obtener una prueba gratuita registrándose "aquí".

Si su instalación crece para superar el número de unidades CPU con licencia, Astra Control Center le impide gestionar nuevas aplicaciones. Se muestra una alerta cuando se supera la capacidad.

Pasos

1. Inicie sesión en la interfaz de usuario de Astra Control Center.
2. Seleccione **cuenta** > **Licencia**.
3. Seleccione **Agregar licencia**.
4. Busque el archivo de licencia (NLF) que descargó.
5. Seleccione **Agregar licencia**.

La página **cuenta** > **Licencia** muestra la información de la licencia, la fecha de caducidad, el número de serie de la licencia, el ID de cuenta y las unidades de CPU utilizadas.

Si dispone de una licencia de evaluación, asegúrese de almacenar su ID de cuenta para evitar la pérdida de datos en caso de que se produzca un error en Astra Control Center si no envía los ASUP.

Añada el clúster

Para comenzar a gestionar las aplicaciones, añada un clúster de Kubernetes y gestiónelo como un recurso de computación. Es necesario añadir un clúster para Astra Control Center para descubrir sus aplicaciones Kubernetes. Para la vista previa de Astra Data Store, queremos añadir el clúster de aplicaciones de Kubernetes que contiene aplicaciones que utilizan volúmenes aprovisionados mediante una vista previa del almacén de datos de Astra.

Le recomendamos que Astra Control Center gestione el clúster en el que se implementa primero antes de añadir otros clústeres a Astra Control Center para su gestión. Disponer del cluster inicial en administración es necesario para enviar datos Kubemetrics y datos asociados al cluster para mediciones y resolución de problemas. Puede utilizar la función **Agregar clúster** para administrar un clúster con Astra Control Center.

Cuando Astra Control gestiona un clúster, realiza un seguimiento del StorageClass predeterminado del clúster. Si se cambia el tipo de almacenamiento con `kubectl` Comandos, Control Astra revierte el cambio. Para cambiar el StorageClass predeterminado en un clúster gestionado por Astra Control, utilice uno de los siguientes métodos:

- Utilice la API Astra Control `PUT /managedClusters` Endpoint y asigne otro tipo de almacenamiento predeterminado con el `DefaultStorageClass` parámetro
- Quite el clúster de Astra Control Management y vuelva a añadirlo con una opción StorageClass predeterminada diferente seleccionada

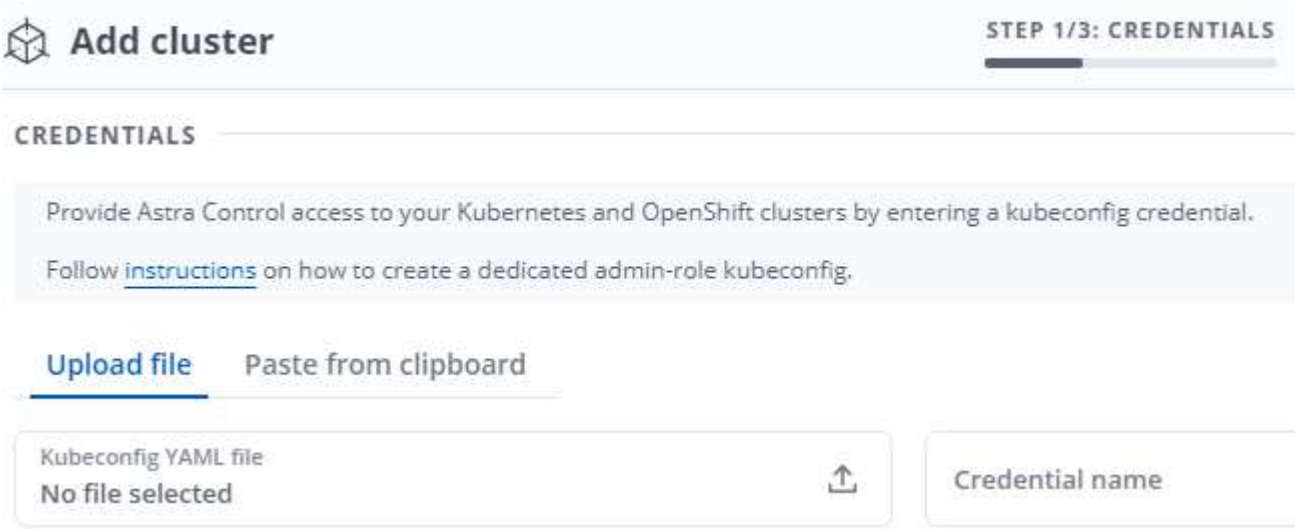
Lo que necesite.#8217;lo necesitará

Antes de añadir un clúster, revise y realice la operación necesaria "requisitos previos".

Pasos

1. En **Dashboard** de la interfaz de usuario de Astra Control Center, seleccione **Agregar** en la sección Clusters.
2. En la ventana **Agregar clúster** que se abre, cargue un `kubeconfig.yaml` archivar o pegar el contenido de un `kubeconfig.yaml` archivo.

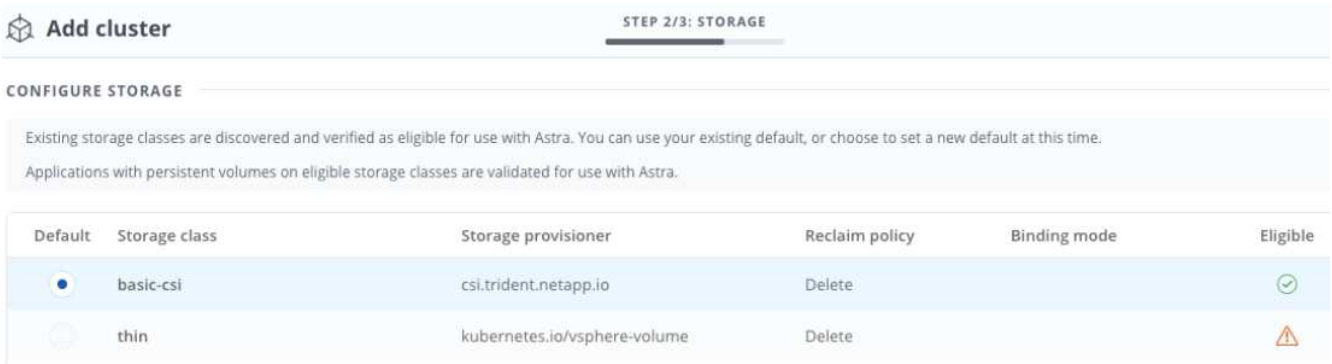
   La `kubeconfig.yaml` el archivo debe incluir **sólo la credencial de cluster para un cluster**.



   Si crea el suyo propio `kubeconfig` file, debe definir sólo un elemento de contexto **uno** en él. Consulte "Documentación de Kubernetes" para obtener información acerca de cómo crear `kubeconfig` archivos.

3. Introduzca un nombre de credencial. De forma predeterminada, el nombre de las credenciales se completa automáticamente como nombre del clúster.
4. Seleccione **Configurar almacenamiento**.
5. Seleccione la clase de almacenamiento que se va a utilizar para este clúster de Kubernetes y seleccione **Review**.

Debe seleccionar una clase de almacenamiento de Trident con el respaldo del almacenamiento de ONTAP o el almacén de datos Astra.



6. Revise la información y si todo parece bien, seleccione **Agregar clúster**.

Resultado

El clúster entra en el estado **detectando** y luego cambia a **ejecutando**. Ha añadido correctamente un clúster de Kubernetes y ahora lo gestiona en Astra Control Center.

Después de agregar un clúster para administrarlo en Astra Control Center, puede que el operador de supervisión tarde unos minutos en implementar. Hasta entonces, el icono de notificación se vuelve rojo y registra un evento **Comprobación de estado del agente de supervisión fallida**. Puede ignorar esto porque el problema se resuelve cuando Astra Control Center obtiene el estado correcto. Si el problema no se resuelve en unos minutos, vaya al clúster y ejecute `oc get pods -n netapp-monitoring` como punto de partida. Deberá consultar los registros del operador de supervisión para depurar el problema.

Añada un back-end de almacenamiento

Puede añadir un back-end de almacenamiento para que Astra Control pueda gestionar sus recursos. Gestionar los clústeres de almacenamiento en Astra Control como back-end de almacenamiento le permite obtener vínculos entre los volúmenes persistentes (VP) y el back-end de almacenamiento, así como mediciones de almacenamiento adicionales.

Puede añadir un back-end de almacenamiento detectado navegando por las peticiones desde el menú Panel o backends.

Lo que necesitará

- Ya tienes "se añadió un clúster" Y es gestionado por Astra Control.

  El clúster gestionado tiene un backend compatible conectado a él que puede ser detectado por Astra Control.

- Para instalaciones de previsualización de Astra Data Store: Has añadido tu clúster de aplicaciones de Kubernetes.

> Después de añadir el clúster de aplicaciones Kubernetes para Astra Data Store, el clúster aparece como `unmanaged` en la lista de back-ends detectados. A continuación, debe añadir el clúster informático que contiene Astra Data Store y es la base para el clúster de aplicaciones de Kubernetes. Puede hacerlo desde **Backends** en la interfaz de usuario. Seleccione el menú Actions para el clúster, seleccione `Manage`, y. "añada el clúster". Tras el estado del clúster de `unmanaged` Los cambios en el nombre del clúster de Kubernetes, puede continuar con la adición de un back-end.

Pasos

1. Debe realizar una de las siguientes acciones:
   - Desde **Panel**:
     i. En la sección backend de almacenamiento del panel, seleccione **gestionar**.
     ii. En la sección Resumen de recursos del panel > Gestión de fondo de almacenamiento, seleccione **Agregar**.
   - Desde **Backends**:
     i. En el área de navegación de la izquierda, seleccione **Backends**.
     ii. Seleccione **gestionar**.
2. Realice una de las siguientes acciones según el tipo de backend:
   - **Almacén de datos Astra**:
     i. Seleccione la ficha **Astra Data Store**.
     ii. Seleccione el clúster de cálculo administrado y seleccione **Siguiente**.
     iii. Confirme los detalles del back-end y seleccione **Administrar el back-end de almacenamiento**.
   - **ONTAP**:
     i. Introduzca las credenciales de administración de ONTAP y seleccione **Revisión**.
     ii. Confirme los detalles del backend y seleccione **Administrar**.

   El back-end aparece en `available` estado en la lista con información resumida.

> Es posible que deba actualizar la página para que se muestre el back-end.

Añadir un bucket

Añadir proveedores de bloques de almacenamiento de objetos es esencial si desea realizar backups de sus aplicaciones y del almacenamiento persistente o si desea clonar aplicaciones entre clústeres. Astra Control almacena estas copias de seguridad o clones en los bloques de almacenamiento de objetos que defina.

Cuando se agrega un bloque, Astra Control Marca un bloque como el indicador de segmento predeterminado. El primer bloque que crea se convierte en el bloque predeterminado.

No necesita un bucket si va a clonar la configuración de sus aplicaciones y el almacenamiento persistente en el mismo clúster.

Utilice cualquiera de los siguientes tipos de bloques:

- ONTAP S3 de NetApp

- StorageGRID S3 de NetApp
- Genérico S3

> Aunque Astra Control Center es compatible con Amazon S3 como proveedor de cubos de S3 genérico, Astra Control Center podría no admitir todos los proveedores de almacenes de objetos que afirman que Amazon es compatible con S3.

Para obtener instrucciones sobre cómo añadir cubos con la API Astra Control, consulte "Información sobre API y automatización de Astra".

Pasos

1. En el área de navegación de la izquierda, seleccione **Cuchos**.
   a. Seleccione **Agregar**.
   b. Seleccione el tipo de bloque.

   > Cuando agregue un bloque, seleccione el proveedor de segmento correcto y proporcione las credenciales correctas para ese proveedor. Por ejemplo, la interfaz de usuario acepta ONTAP S3 de NetApp como tipo y acepta credenciales de StorageGRID; sin embargo, esto hará que se produzcan errores en todos los futuros backups de aplicaciones y restauraciones usando este bucket.

   c. Cree un nuevo nombre de bloque o introduzca un nombre de bloque existente y una descripción opcional.

   > El nombre del bloque y la descripción aparecen como una ubicación de copia de seguridad que puede elegir más tarde al crear una copia de seguridad. El nombre también aparece durante la configuración de la política de protección.

   d. Introduzca el nombre o la dirección IP del extremo de S3.
   e. Si desea que este bloque sea el bloque predeterminado para todos los backups, compruebe la `Make this bucket the default bucket for this private cloud` opción.

   > Esta opción no aparece para el primer bloque que cree.

   f. Continúe añadiendo información sobre credenciales.

Añada credenciales de acceso de S3

Añada credenciales de acceso de S3 en cualquier momento.

Pasos

1. En el cuadro de diálogo Cuchos, seleccione la ficha **Agregar** o **utilizar existente**.
   a. Introduzca un nombre para la credencial que la distingue de otras credenciales en Astra Control.
   b. Escriba el identificador de acceso y la clave secreta pegando el contenido del portapapeles.

El futuro

Ahora que ha iniciado sesión y agregado clústeres a Astra Control Center, está listo para empezar a utilizar las funciones de gestión de datos de aplicaciones de Astra Control Center.

- "Gestionar usuarios"
- "Inicie la gestión de aplicaciones"
- "Proteja sus aplicaciones"
- "Clone aplicaciones"
- "Gestionar notificaciones"
- "Conéctese a Cloud Insights"
- "Agregue un certificado TLS personalizado"

Obtenga más información

- "Utilice la API Astra Control"
- "Problemas conocidos"

Requisitos previos para añadir un clúster

Debe asegurarse de que se cumplan las condiciones previas antes de añadir un clúster. También debe ejecutar las comprobaciones de elegibilidad para asegurarse de que su clúster esté listo para ser agregado a Astra Control Center.

Qué necesitará antes de añadir un clúster

- Uno de los siguientes tipos de clústeres:
  - Clústeres que ejecutan OpenShift 4.6, 4.7 o 4.8, y que tienen StorageClasses Astra respaldados por el almacén de datos Astra o ONTAP 9.5 o posterior
  - Clústeres que ejecutan Rancher 2.5
  - Clústeres que ejecutan entre 1.19 y 1.21 de Kubernetes (incluida 1.21.x)

    Asegúrese de que los clústeres tienen uno o más nodos de trabajo con al menos 1 GB de RAM disponibles para ejecutar los servicios de telemetría.

    > Si tiene pensado añadir un segundo clúster OpenShift 4.6, 4.7 o 4.8 como un recurso informático gestionado, debe asegurarse de que la función de Snapshot de volumen de Astra Trident esté habilitada. Consulte la Astra Trident oficial "instrucciones" Para habilitar y probar Volume Snapshots con Astra Trident.

- El superusuario y el ID de usuario establecidos en el sistema ONTAP de respaldo para realizar copias de seguridad y restaurar aplicaciones con Astra Control Center. Ejecute el siguiente comando en la línea de comandos de la ONTAP:
  ```
  export-policy rule modify -vserver <storage virtual machine name> -policyname
  <policy name> -ruleindex 1 -superuser sysm --anon 65534
  ```
- Una Astra Trident `volumesnapshotclass` objeto definido por un administrador. Vea la Astra Trident "instrucciones" Para habilitar y probar Volume Snapshots con Astra Trident.
- Compruebe que solo tiene una única clase de almacenamiento predeterminada definida para el clúster de Kubernetes.

Ejecutar las comprobaciones de elegibilidad

Ejecute las siguientes comprobaciones de elegibilidad para asegurarse de que su clúster esté listo para ser agregado a Astra Control Center.

Pasos

1. Compruebe la versión de Trident.

```
kubectl get tridentversions -n trident
```

Si Trident existe, se muestra una salida similar a la siguiente:

```
NAME      VERSION
trident   21.04.0
```

Si Trident no existe, se muestra un resultado similar al siguiente:

```
error: the server doesn't have a resource type "tridentversions"
```

> Si Trident no está instalado o la versión instalada no es la más reciente, debe instalar la versión más reciente de Trident antes de continuar. Consulte "Documentación de Trident" si desea obtener instrucciones.

2. Compruebe si las clases de almacenamiento están usando los controladores de Trident compatibles. El nombre del aprovisionador debe ser `csi.trident.netapp.io`. Consulte el siguiente ejemplo:

```
kubectl get sc
NAME                    PROVISIONER                    RECLAIMPOLICY
VOLUMEBINDINGMODE   ALLOWVOLUMEEXPANSION   AGE
ontap-gold (default)    csi.trident.netapp.io          Delete
Immediate           true                   5d23h
thin                    kubernetes.io/vsphere-volume   Delete
Immediate           false                  6d
```

Cree una imagen de rol administrativo

Asegúrese de que dispone de lo siguiente en su máquina antes de realizar los pasos siguientes:

- `kubectl` v1.19 o posterior instalado
- Una imagen marcada activa con los derechos de administrador del clúster para el contexto activo

Pasos

1. Cree una cuenta de servicio del siguiente modo:

a. Cree un archivo de cuenta de servicio llamado `astracontrol-service-account.yaml`.

   Ajuste el nombre y el espacio de nombres según sea necesario. Si se realizan cambios aquí, debe aplicar los mismos cambios en los pasos siguientes.

```
<strong>astracontrol-service-account.yaml</strong>
```

+

```
apiVersion: v1
kind: ServiceAccount
metadata:
  name: astracontrol-service-account
  namespace: default
```

a. Aplicar la cuenta de servicio:

```
kubectl apply -f astracontrol-service-account.yaml
```

2. Conceda permisos de administrador del clúster de la siguiente manera:

   a. Cree un `ClusterRoleBinding` archivo llamado `astracontrol-clusterrolebinding.yaml`.

      Ajuste los nombres y espacios de nombres modificados al crear la cuenta de servicio según sea necesario.

```
<strong>astracontrol-clusterrolebinding.yaml</strong>
```

+

```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
  name: astracontrol-admin
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: ClusterRole
  name: cluster-admin
subjects:
- kind: ServiceAccount
  name: astracontrol-service-account
  namespace: default
```

a. Aplique el enlace de roles del clúster:

```
kubectl apply -f astracontrol-clusterrolebinding.yaml
```

3. Enumere los secretos de la cuenta de servicio, reemplazando `<context>` con el contexto correcto para su instalación:

```
kubectl get serviceaccount astracontrol-service-account --context
<context> --namespace default -o json
```

El final de la salida debe ser similar a lo siguiente:

```
"secrets": [
{ "name": "astracontrol-service-account-dockercfg-vhz87"},
{ "name": "astracontrol-service-account-token-r59kr"}
]
```

Los índices de cada elemento de la `secrets` la matriz comienza con 0. En el ejemplo anterior, el índice para `astracontrol-service-account-dockercfg-vhz87` sería 0 y el índice para `astracontrol-service-account-token-r59kr` sería 1. En la salida, anote el índice del nombre de la cuenta de servicio que contiene la palabra "token".

4. Genere la kubeconfig de la siguiente manera:

a. Cree un `create-kubeconfig.sh` archivo. Sustituya `TOKEN_INDEX` al principio de la secuencia de comandos siguiente con el valor correcto.

```
<strong>create-kubeconfig.sh</strong>
```

```
# Update these to match your environment.
# Replace TOKEN_INDEX with the correct value
# from the output in the previous step. If you
# didn't change anything else above, don't change
# anything else here.

SERVICE_ACCOUNT_NAME=astracontrol-service-account
NAMESPACE=default
NEW_CONTEXT=astracontrol
KUBECONFIG_FILE='kubeconfig-sa'

CONTEXT=$(kubectl config current-context)

SECRET_NAME=$(kubectl get serviceaccount ${SERVICE_ACCOUNT_NAME} \
  --context ${CONTEXT} \
```

```
  --namespace ${NAMESPACE} \
  -o jsonpath='{.secrets[TOKEN_INDEX].name}')
TOKEN_DATA=$(kubectl get secret ${SECRET_NAME} \
  --context ${CONTEXT} \
  --namespace ${NAMESPACE} \
  -o jsonpath='{.data.token}')

TOKEN=$(echo ${TOKEN_DATA} | base64 -d)

# Create dedicated kubeconfig
# Create a full copy
kubectl config view --raw > ${KUBECONFIG_FILE}.full.tmp

# Switch working context to correct context
kubectl --kubeconfig ${KUBECONFIG_FILE}.full.tmp config use-context
${CONTEXT}

# Minify
kubectl --kubeconfig ${KUBECONFIG_FILE}.full.tmp \
  config view --flatten --minify > ${KUBECONFIG_FILE}.tmp

# Rename context
kubectl config --kubeconfig ${KUBECONFIG_FILE}.tmp \
  rename-context ${CONTEXT} ${NEW_CONTEXT}

# Create token user
kubectl config --kubeconfig ${KUBECONFIG_FILE}.tmp \
  set-credentials ${CONTEXT}-${NAMESPACE}-token-user \
  --token ${TOKEN}

# Set context to use token user
kubectl config --kubeconfig ${KUBECONFIG_FILE}.tmp \
  set-context ${NEW_CONTEXT} --user ${CONTEXT}-${NAMESPACE}-token-
user

# Set context to correct namespace
kubectl config --kubeconfig ${KUBECONFIG_FILE}.tmp \
  set-context ${NEW_CONTEXT} --namespace ${NAMESPACE}

# Flatten/minify kubeconfig
kubectl config --kubeconfig ${KUBECONFIG_FILE}.tmp \
  view --flatten --minify > ${KUBECONFIG_FILE}

# Remove tmp
rm ${KUBECONFIG_FILE}.full.tmp
rm ${KUBECONFIG_FILE}.tmp
```

b. Origen de los comandos para aplicarlos al clúster de Kubernetes.

```
source create-kubeconfig.sh
```

5. (**opcional**) cambie el nombre de la kubeconfig por un nombre significativo para el clúster. Proteja las credenciales del clúster.

```
chmod 700 create-kubeconfig.sh
mv kubeconfig-sa.txt YOUR_CLUSTER_NAME_kubeconfig
```

El futuro

Ahora que ha comprobado que se cumplen los requisitos previos, está listo "añadir un clúster".

Obtenga más información

- "Documentación de Trident"
- "Utilice la API Astra Control"

Agregue un certificado TLS personalizado

Puede quitar el certificado TLS autofirmado existente y reemplazarlo con un certificado TLS firmado por una entidad de certificación (CA).

Lo que necesitará

- Clúster Kubernetes con Astra Control Center instalado
- Acceso administrativo a un shell de comandos en el clúster para ejecutar `kubectl` comandos
- Archivos de claves privadas y certificados de la CA

Quite el certificado autofirmado

Quite el certificado TLS autofirmado existente.

1. Con SSH, inicie sesión en el clúster Kubernetes que aloja Astra Control Center como usuario administrativo.
2. Busque el secreto TLS asociado con el certificado actual mediante el comando siguiente, reemplazo `<ACC-deployment-namespace>` Con el espacio de nombres de puesta en marcha de Astra Control Center:

```
kubectl get certificate -n <ACC-deployment-namespace>
```

3. Elimine el secreto y certificado instalados actualmente con los comandos siguientes:

```
kubectl delete cert cert-manager-certificates -n <ACC-deployment-
namespace>
kubectl delete secret secure-testing-cert -n <ACC-deployment-namespace>
```

Añadir un nuevo certificado

Añada un nuevo certificado TLS firmado por una CA.

1. Utilice el siguiente comando para crear el nuevo secreto TLS con la clave privada y los archivos de certificado de la CA, reemplazando los argumentos entre paréntesis <> con la información adecuada:

```
kubectl create secret tls <secret-name> --key <private-key-filename>
--cert <certificate-filename> -n <ACC-deployment-namespace>
```

2. Utilice el siguiente comando y el ejemplo para editar el archivo de definición de recursos personalizados (CRD) del clúster y cambiar el `spec.selfSigned` valor a. `spec.ca.secretName` Para hacer referencia al secreto TLS que ha creado anteriormente:

```
kubectl edit clusterissuers.cert-manager.io/cert-manager-certificates -n
<ACC-deployment-namespace>
....

#spec:
#  selfSigned: {}

spec:
  ca:
    secretName: <secret-name>
```

3. Utilice el siguiente comando y el resultado de ejemplo para validar que los cambios son correctos y que el clúster está listo para validar certificados, sustituir `<ACC-deployment-namespace>` Con el espacio de nombres de puesta en marcha de Astra Control Center:

```
kubectl describe clusterissuers.cert-manager.io/cert-manager-
certificates -n <ACC-deployment-namespace>
....

Status:
  Conditions:
    Last Transition Time:  2021-07-01T23:50:27Z
    Message:               Signing CA verified
    Reason:                KeyPairVerified
    Status:                True
    Type:                  Ready
Events:                    <none>
```

4. Cree el `certificate.yaml` archivo utilizando el ejemplo siguiente, reemplazando los valores de marcador de posición entre corchetes <> con la información apropiada:

```
apiVersion: cert-manager.io/v1
kind: Certificate
metadata:
  name: <certificate-name>
  namespace: <ACC-deployment-namespace>
spec:
  secretName: <certificate-secret-name>
  duration: 2160h # 90d
  renewBefore: 360h # 15d
  dnsNames:
  - <astra.dnsname.example.com> #Replace with the correct Astra Control
Center DNS address
  issuerRef:
    kind: ClusterIssuer
    name: cert-manager-certificates
```

5. Cree el certificado con el comando siguiente:

```
kubectl apply -f certificate.yaml
```

6. Con el siguiente comando y el resultado de ejemplo, valide que el certificado se ha creado correctamente y con los argumentos especificados durante la creación (como nombre, duración, plazo de renovación y nombres DNS).

```
kubectl describe certificate -n <ACC-deployment-namespace>
....

Spec:
  Dns Names:
    astra.example.com
  Duration:  125h0m0s
  Issuer Ref:
    Kind:        ClusterIssuer
    Name:        cert-manager-certificates
  Renew Before:  61h0m0s
  Secret Name:   <certificate-secret-name>
Status:
  Conditions:
    Last Transition Time:  2021-07-02T00:45:41Z
    Message:               Certificate is up to date and has not expired
    Reason:                Ready
    Status:                True
    Type:                  Ready
  Not After:               2021-07-07T05:45:41Z
  Not Before:              2021-07-02T00:45:41Z
  Renewal Time:            2021-07-04T16:45:41Z
  Revision:                1
Events:                    <none>
```

7. Edite la opción Ingress CRD TLS para que apunte al nuevo secreto de certificado utilizando el siguiente comando y ejemplo, reemplazando los valores de marcador de posición entre paréntesis <> con la información adecuada:

```
kubectl edit ingressroutes.traefik.containo.us -n <ACC-deployment-
namespace>
....

# tls:
#    options:
#      name: default
#    secretName: secure-testing-cert
#    store:
#      name: default

 tls:
    options:
      name: default
    secretName: <certificate-secret-name>
    store:
      name: default
```

8. Con un explorador web, vaya a la dirección IP de implementación de Astra Control Center.
9. Compruebe que los detalles del certificado coinciden con los detalles del certificado que ha instalado.
10. Exporte el certificado e importe el resultado en el administrador de certificados en su navegador web.

Preguntas frecuentes para Astra Control Center

Estas preguntas frecuentes pueden ser de ayuda si solo está buscando una respuesta rápida a una pregunta.

Descripción general

En las siguientes secciones se ofrecen respuestas a algunas preguntas adicionales que puede encontrar a medida que utiliza Astra Control Center. Para obtener más aclaraciones, por favor, diríjase a astra.feedback@netapp.com

Acceso a Astra Control Center

- ¿Cuál es la URL de Astra Control?*

Astra Control Center utiliza autenticación local y una dirección URL específica para cada entorno.

Para la dirección URL, en un explorador, introduzca el nombre de dominio completo (FQDN) establecido en el campo spec.astraAddress del archivo astra_control_Center_min.yaml custom resource definition (CRD) cuando instaló Astra Control Center. El mensaje de correo electrónico es el valor que se ha establecido en el campo SPEC.Email del astra_control_Center_min.ylma CRD.

Estoy utilizando la licencia de Evaluación. ¿Cómo puedo cambiar a la licencia completa?

Si desea cambiar fácilmente a una licencia completa, obtenga el archivo de licencia de NetApp (NLF).

- Pasos*

- En la navegación de la izquierda, seleccione **cuenta** > **Licencia**.
- Seleccione **Agregar licencia**.
- Busque el archivo de licencia que ha descargado y seleccione **Agregar**.

Estoy utilizando la licencia de Evaluación. ¿Puedo seguir gestionando aplicaciones?

Sí, puede comprobar la funcionalidad de administración de aplicaciones con la licencia de evaluación.

Registrar clústeres de Kubernetes

Necesito añadir nodos de trabajo a mi clúster Kubernetes después de añadir a Astra Control. ¿Qué debo hacer?

Los nodos de trabajo nuevos se pueden agregar a los pools existentes. Estos serán descubiertos automáticamente por Astra Control. Si los nuevos nodos no están visibles en Astra Control, compruebe si los nuevos nodos de trabajo están ejecutando el tipo de imagen admitido. También puede verificar el estado de los nuevos nodos de trabajo mediante el `kubectl get nodes` comando.

¿Cómo descontrolo correctamente un clúster?

1. "Desgestione las aplicaciones desde Astra Control".
2. "Desgestione el clúster desde Astra Control".

¿Qué ocurre con mis aplicaciones y datos después de eliminar el clúster Kubernetes de Astra Control?

La eliminación de un clúster de Astra Control no realizará ningún cambio en la configuración del clúster (aplicaciones y almacenamiento persistente). Las instantáneas de Astra Control o las copias de seguridad tomadas de las aplicaciones en ese clúster no estarán disponibles para restaurar. Los backups de almacenamiento persistentes creados por Astra Control permanecen en Astra Control, pero no están disponibles para la restauración.

> Quite siempre un clúster de Astra Control antes de eliminarlo mediante cualquier otro método. La eliminación de un clúster con otra herramienta mientras Astra Control sigue gestionando puede causar problemas para su cuenta Astra Control.

¿Se desinstala NetApp Trident cuando extrizo un clúster de Kubernetes de Astra Control?

Trident no se desinstala de un clúster cuando lo elimina de Astra Control.

Gestionar aplicaciones

- ¿Puede Astra Control implementar una aplicación?*

Astra Control no implementa aplicaciones. Las aplicaciones deben implementarse fuera de Astra Control.

¿Qué sucede con las aplicaciones después de dejar de administrarlas desde Astra Control?

Se eliminarán todos los backups o las snapshots existentes. Las aplicaciones y los datos siguen estando disponibles. Las operaciones de administración de datos no estarán disponibles para aplicaciones no administradas o para cualquier copia de seguridad o copia Snapshot que pertenezcan a él.

¿Puede Astra Control gestionar una aplicación que utiliza un almacenamiento que no sea de NetApp?

No Aunque Astra Control puede detectar aplicaciones que utilizan almacenamiento de terceros, no puede gestionar una aplicación que utilice almacenamiento de terceros.

¿Debo administrar Astra Control mismo? no, no debería gestionar Astra Control por sí mismo porque es una "app del sistema".

Operaciones de gestión de datos

- hay instantáneas en mi cuenta que no creé. ¿de dónde vienen?*

En algunas situaciones, Astra Control creará automáticamente una instantánea como parte de un proceso de backup, clonado o restauración.

Mi aplicación utiliza varios VP. ¿Tomará Astra Control instantáneas y copias de seguridad de todas estas EVs?

Sí. Una operación de instantánea en una aplicación de Astra Control incluye una instantánea de todos los VP vinculados a las RVP de la aplicación.

¿Puedo gestionar las instantáneas tomadas por Astra Control directamente a través de una interfaz o almacenamiento de objetos diferente?

No Las copias Snapshot y las copias de seguridad realizadas por Astra Control solo se pueden gestionar con Astra Control.