Commencez

Astra Control Center

NetApp
June 07, 2024

Sommaire

Commencez . . . 1
- Exigences du centre de contrôle Astra . . . 1
- Démarrage rapide pour Astra Control Center . . . 5
- Présentation de l’installation . . . 7
- Configurer le centre de contrôle Astra . . . 29
- Foire aux questions pour Astra Control Center . . . 44

Commencez

Exigences du centre de contrôle Astra

Commencez par vérifier que votre environnement opérationnel, vos clusters d'applications, vos applications, vos licences et votre navigateur Web sont prêts.

De l'environnement opérationnel

Astra Control Center requiert l'un des types d'environnements opérationnels suivants :

- Red Hat OpenShift Container Platform 4.6.8, 4.7 ou 4.8
- Rancher 2.5
- Kubernetes 1.19 à 1.21 (dont 1.21.x)

Assurez-vous que l'environnement d'exploitation que vous choisissez d'héberger est conforme aux exigences de base en matière de ressources décrites dans la documentation officielle de l'environnement. Outre les exigences de l'environnement en matière de ressources, Astra Control Center requiert les ressources suivantes :

Composant	Conditions requises
Capacité de stockage ONTAP interne	300 Go au moins disponibles
Nœuds worker	Au moins 3 nœuds workers au total, avec 4 cœurs de processeurs et 12 Go de RAM chacun
Équilibrage de la charge	Type de service « LoadBalancer » disponible pour que le trafic d'entrée soit envoyé aux services du cluster d'environnement opérationnel
Résolution du FQDN	Méthode permettant de pointer le FQDN de Astra Control Center vers l'adresse IP à charge équilibrée
Astra Trident	• Astra Trident 21.04 ou version ultérieure installé et configuré si NetApp ONTAP version 9.5 ou ultérieure sera utilisé comme système de stockage back-end • Astra Trident 21.10.1 ou version ultérieure installé et configuré en cas d'utilisation de la version préliminaire du magasin de données d'Astra comme système de stockage back-end

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

- **Registre d'images**: Vous devez avoir un registre d'images privé Docker existant à laquelle vous pouvez pousser les images de construction d'Astra Control Center. Vous devez fournir l'URL du registre d'images où vous allez télécharger les images.
- **Configuration de l'Astra Trident / ONTAP** : le Centre de contrôle Astra requiert la création et la définition

d'une classe de stockage comme classe de stockage par défaut. Le centre de contrôle Astra prend en charge les pilotes ONTAP suivants fournis par Astra Trident :

- ontap-nas
- ontap-san
- ontap-san-économie

> Lors du clonage d'applications dans les environnements OpenShift, Astra Control Center doit permettre à OpenShift de monter des volumes et de modifier la propriété des fichiers. Pour cela, il faut configurer une policy d'exportation de volume ONTAP afin de permettre ces opérations. Pour ce faire, utilisez les commandes suivantes :
>
> 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 vous prévoyez d'ajouter un deuxième environnement opérationnel OpenShift comme ressource de calcul gérée, vous devez vous assurer que la fonctionnalité Snapshot de volume Astra Trident est activée. Pour activer et tester des copies Snapshot de volumes avec Astra Trident, "Consultez les instructions officielles de l'Astra Trident".

Configuration requise en cluster des applications

Astra Control Center a les exigences suivantes pour les clusters que vous prévoyez de gérer à partir d'Astra Control Center. Ces exigences s'appliquent également si le cluster que vous prévoyez de gérer est le cluster d'environnement opérationnel qui héberge Astra Control Center.

- La version la plus récente de Kubernetes "composant de snapshot-controller" est installé
- Découvrez Astra Trident "objet volumesnapshotclass" a été défini par un administrateur
- Une classe de stockage Kubernetes par défaut existe sur le cluster
- Au moins une classe de stockage est configurée pour utiliser Astra Trident

> Votre cluster d'applications doit disposer d'un `kubeconfig.yaml` fichier qui définit un seul *context* element. Consultez la documentation Kubernetes sur "informations sur la création de fichiers kubeconfig".

> Lors de la gestion des clusters d'applications dans un environnement Rancher, modifiez le contexte par défaut du cluster d'applications dans `kubeconfig` Fichier fourni par Rancher pour utiliser un contexte de plan de contrôle au lieu du contexte de serveur API Rancher. La charge est réduite sur le serveur API Rancher et les performances sont améliorées.

De gestion des applications

Astra Control présente les exigences de gestion des applications suivantes :

- **Licence** : pour gérer des applications à l'aide d'Astra Control Center, vous devez disposer d'une licence Astra Control Center.
- **Espaces de noms** : Astra Control exige qu'une application ne couvre pas plus d'un seul espace de noms,

mais qu'un espace de noms peut contenir plus d'une application.

- **StorageClass** : si vous installez explicitement une application avec une classe de stockage et que vous devez cloner l'application, le cluster cible pour l'opération de clonage doit avoir la classe de stockage spécifiée à l'origine. Le clonage d'une application avec une classe de stockage explicitement définie sur un cluster ne disposant pas de la même classe de stockage échouera.
- **Ressources Kubernetes** : les applications qui utilisent des ressources Kubernetes non collectées par Astra Control peuvent ne pas disposer de fonctionnalités complètes de gestion des données d'application. Astra Control collecte les ressources Kubernetes suivantes :
  - ClusterRole
  - ClusterRoleBinding
  - ConfigMap
  - CustomResourceDefinition
  - Ressource CustomResource
  - Ensemble de démonstrations
  - Déploiement
  - Déploiement.Config
  - Entrée
  - MutatingWebhook
  - Demande de volume persistant
  - Pod
  - Et de réplication
  - RoleBinding
  - Rôle
  - Itinéraire
  - Secret
  - Service
  - Compte de service
  - StatefulSet
  - ValidétingWebhook

Méthodes d'installation d'applications prises en charge

Astra Control prend en charge les méthodes d'installation d'application suivantes :

- **Fichier manifeste** : Astra Control prend en charge les applications installées à partir d'un fichier manifeste utilisant kubectl. Par exemple :

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

- **Helm 3** : si vous utilisez Helm pour installer des applications, Astra Control nécessite Helm version 3. La gestion et le clonage des applications installées avec Helm 3 (ou mises à niveau de Helm 2 à Helm 3) sont entièrement pris en charge. La gestion des applications installées avec Helm 2 n'est pas prise en charge.

- **Applications déployées par l'opérateur** : Astra Control prend en charge les applications installées avec des opérateurs de l'espace de noms. Les applications suivantes ont été validées pour ce modèle d'installation :
  - "Apache K8ssandra"
  - "IC Jenkins"
  - "Cluster Percona XtraDB"

> Un opérateur et l'application qu'il installe doivent utiliser le même espace de noms ; vous devrez peut-être modifier le fichier .yaml de déploiement pour que l'opérateur s'assure que c'est le cas.

Accès à Internet

Vous devez déterminer si vous avez un accès externe à Internet. Si ce n'est pas le cas, certaines fonctionnalités peuvent être limitées, comme la réception de données de surveillance et de metrics depuis NetApp Cloud Insights ou l'envoi de packs de support au "Site de support NetApp".

Licence

Astra Control Center requiert une licence Astra Control Center pour bénéficier de toutes les fonctionnalités. Obtenez une licence d'évaluation ou une licence complète auprès de NetApp. Sans licence, vous ne pourrez pas :

- Définir des applications personnalisées
- Créer des snapshots ou des clones d'applications existantes
- Configuration des règles de protection des données

Si vous voulez essayer Astra Control Center, vous pouvez "utilisez une licence d'essai gratuite de 90 jours".

Type de service « LoadBalancer » pour les clusters Kubernetes sur site

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

Configuration réseau requise

L'environnement opérationnel qui héberge le centre de contrôle Astra communique avec les ports TCP suivants. Veillez à ce que ces ports soient autorisés par le biais de pare-feu et configurez des pare-feu pour autoriser tout trafic de sortie HTTPS provenant du réseau Astra. Certains ports nécessitent une connectivité entre l'environnement hébergeant le centre de contrôle Astra et chaque cluster géré (le cas échéant).

Source	Destination	Port	Protocole	Objectif
PC client	Centre de contrôle Astra	443	HTTPS	Accès à l'interface utilisateur/à l'API : assurez-vous que ce port est ouvert à la fois entre le cluster hébergeant Astra Control Center et chaque cluster géré
Consommateurs de metrics	Nœud de travail Astra Control Center	9090	HTTPS	Communication de données de metrics : assurez-vous que chaque cluster géré peut accéder à ce port sur le cluster hébergeant Astra Control Center (communication bidirectionnelle requise).
Centre de contrôle Astra	Service Cloud Insights hébergé	443	HTTPS	Communication avec Cloud Insights
Centre de contrôle Astra	Fournisseur de compartiments de stockage Amazon S3	443	HTTPS	Communications de stockage Amazon S3
Centre de contrôle Astra	Active IQ de NetApp	443	HTTPS	Communication avec NetApp ActiveIQ

Navigateurs Web pris en charge

Astra Control Center prend en charge les versions récentes de Firefox, Safari et Chrome avec une résolution minimale de 1280 x 720.

Et la suite

Afficher le "démarrage rapide" présentation.

Démarrage rapide pour Astra Control Center

Cette page offre un aperçu détaillé des étapes à suivre pour commencer à utiliser le centre de contrôle Astra. Les liens de chaque étape vous mènent à une page qui fournit plus de détails.

Essayez-le ! Si vous voulez essayer Astra Control Center, vous pouvez utiliser une licence d'évaluation de 90 jours. Voir "informations de licence" pour plus d'informations.

1 **Vérifiez la configuration des clusters Kubernetes**

- Astra fonctionne avec les clusters Kubernetes avec un système de stockage back-end ONTAP configuré

avec Trident ou un système back-end de stockage d'aperçu du magasin de données Astra.

- Les clusters doivent fonctionner correctement, avec au moins trois nœuds de travail en ligne.
- Le cluster doit exécuter Kubernetes.

"En savoir plus sur les exigences du centre de contrôle Astra".

2 Téléchargez et installez Astra Control Center

- Téléchargez Astra Control Center à partir du "Page de téléchargements de l'Astra Control Center du site de support NetApp".
- Installez Astra Control Center dans votre environnement local.

  Vous pouvez également installer Astra Control Center à l'aide de Red Hat OperatorHub.

- Découvrez votre configuration Trident sur le système back-end de stockage ONTAP. Ou, découvrez votre "Présentation d'Astra Data Store" clusters comme système de stockage back-end.

  Vous installez les images sur un registre OpenShift ou utilisez votre registre local.

"En savoir plus sur l'installation d'Astra Control Center".

3 Effectuez certaines tâches de configuration initiales

- Ajouter une licence.
- Ajoutez un cluster Kubernetes et Astra Control Center découverte des détails.
- Ajout d'un système back-end de stockage d'aperçu pour ONTAP ou Astra Data Store
- Vous pouvez également ajouter un compartiment de magasin d'objets qui stockera les sauvegardes de vos applications.

"En savoir plus sur le processus de configuration initiale".

4 Utilisez Astra Control Center

Après avoir terminé la configuration du centre de contrôle Astra, voici ce que vous pourriez faire :

- Gérer une application. "En savoir plus sur la gestion des applications".
- Vous pouvez également vous connecter à NetApp Cloud Insights pour afficher des mesures sur l'état de santé de votre système, la capacité et le débit dans l'interface utilisateur de l'Astra Control Center. "En savoir plus sur la connexion à Cloud Insights".

5 Continuez à partir de ce démarrage rapide

"Poser le centre de contrôle Astra".

Trouvez plus d'informations

- "Utilisez l'API de contrôle Astra"

Présentation de l'installation

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

- "Installer le centre de contrôle Astra en suivant la procédure standard"
- "(Si vous utilisez Red Hat OpenShift) installez Astra Control Center à l'aide d'OpenShift OperatorHub"

Installer le centre de contrôle Astra en suivant la procédure standard

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

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

Ce dont vous avez besoin

- "Avant de commencer l'installation, préparez votre environnement pour le déploiement d'Astra Control Center".
- S'assurer que tous les opérateurs du groupe d'instruments sont en état de fonctionnement et disponibles.

  Exemple OpenShift :

  ```
  oc get clusteroperators
  ```

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

  Exemple OpenShift :

  ```
  oc get apiservices
  ```

- Vous avez créé une adresse FQDN pour Astra Control Center dans votre data Center.

Description de la tâche

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

- Installe les composants Astra dans le `netapp-acc` (ou espace de nom personnalisé).
- Crée un compte par défaut.
- Définit une adresse e-mail d'utilisateur administratif par défaut et un mot de passe unique par défaut de `ACC-<UUID_of_installation>` Pour cet exemple de centre de contrôle Astra. Ce rôle est attribué à cet utilisateur dans le système et est nécessaire pour la première connexion à l'interface utilisateur.
- Vous aide à déterminer que toutes les POD Astra Control Center sont en cours d'exécution.
- Installe l'interface utilisateur Astra.

> Les commandes Podman peuvent être utilisées à la place des commandes Docker si vous utilisez le Podman de Red Hat au lieu de Docker Engine.

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

Étapes

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

- Téléchargez le pack Astra Control Center
- Déballez le bundle et modifiez le répertoire
- Ajoutez les images à votre registre local
- Configurez l'espace de noms et le secret pour les registres avec les exigences d'authentification
- Poser le conducteur du centre de commande Astra
- Configurer le centre de contrôle Astra
- Installation complète du centre de contrôle Astra et du conducteur
- Vérifiez l'état du système
- Connectez-vous à l'interface utilisateur du centre de contrôle Astra

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

Téléchargez le pack Astra Control Center

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

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

Déballez le bundle et modifiez le répertoire

1. Extraire les images :

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

2. Passez au répertoire Astra.

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

Ajoutez les images à votre registre local

1. Ajoutez les fichiers du répertoire d'images de l'Astra Control Center à votre registre local.

   Voir les exemples de scripts pour le chargement automatique des images ci-dessous.

   a. Connectez-vous à votre registre :

      Docker :

      ```
      docker login [your_registry_path]
      ```

      Podman :

      ```
      podman login [your_registry_path]
      ```

   b. Utilisez le script approprié pour charger les images, les marquer et pousser les images dans votre registre 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
```

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

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

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

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

   Réponse :

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

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

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

   Exemple de réponse :

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

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

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

   Exemple de réponse :

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

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

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

Réponse

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

Poser le conducteur du centre de commande Astra

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

```
vim astra_control_center_operator_deploy.yaml
```

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

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

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

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

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

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

2. Poser le conducteur du centre de commande Astra :

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

Exemple de réponse :

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

Configurer le centre de contrôle Astra

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

   Si d'autres personnalisations sont nécessaires pour votre environnement, vous pouvez l'utiliser `astra_control_center.yaml` En tant que CR alternatif. `astra_control_center_min.yaml` Est le CR par défaut et convient à la plupart des installations.

   ```
   vim astra_control_center_min.yaml
   ```

   Les propriétés configurées par le CR ne peuvent pas être modifiées après le déploiement initial du centre de contrôle Astra.

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

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

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

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

dans un plus tard.

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

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

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

g. (Facultatif) modifiez le `storageClass` Avantages pour une autre ressource de stockage Astra Trident, si nécessaire à votre installation.

h. (Pour les installations utilisant l'aperçu d'Astra Data Store) Découvrez ce problème connu pour "autres modifications requises" Au YAML.

```
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"
```

Installation complète du centre de contrôle Astra et du conducteur

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

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

Exemple de réponse :

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

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

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

Exemple de réponse :

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

Vérifiez l'état du système

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

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

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

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

Exemple de réponse :

```
NAME                                    READY   STATUS    RESTARTS
AGE
acc-helm-repo-5f75c5f564-bzqmt          1/1     Running   0
11m
activity-6b8f7cccb9-mlrn4               1/1     Running   0
9m2s
api-token-authentication-6hznt          1/1     Running   0
8m50s
api-token-authentication-qpfgb          1/1     Running   0
8m50s
api-token-authentication-sqnb7          1/1     Running   0
8m50s
asup-5578bbdd57-dxkbp                   1/1     Running   0
9m3s
authentication-56bff4f95d-mspmq         1/1     Running   0
7m31s
bucketservice-6f7968b95d-9rrrl          1/1     Running   0
8m36s
cert-manager-5f6cf4bc4b-82khn           1/1     Running   0
6m19s
cert-manager-cainjector-76cf976458-sdrbc   1/1     Running   0
6m19s
```

```
cert-manager-webhook-5b7896bfd8-2n45j      1/1     Running   0
6m19s
cloud-extension-749d9f684c-8bdhq           1/1     Running   0
9m6s
cloud-insights-service-7d58687d9-h5tzw     1/1     Running   2
8m56s
composite-compute-968c79cb5-nv7l4          1/1     Running   0
9m11s
composite-volume-7687569985-jg9gg          1/1     Running   0
8m33s
credentials-5c9b75f4d6-nx9cz               1/1     Running   0
8m42s
entitlement-6c96fd8b78-zt7f8               1/1     Running   0
8m28s
features-5f7bfc9f68-gsjnl                  1/1     Running   0
8m57s
fluent-bit-ds-h88p7                        1/1     Running   0
7m22s
fluent-bit-ds-krhnj                        1/1     Running   0
7m23s
fluent-bit-ds-l5bjj                        1/1     Running   0
7m22s
fluent-bit-ds-lrclb                        1/1     Running   0
7m23s
fluent-bit-ds-s5t4n                        1/1     Running   0
7m23s
fluent-bit-ds-zpr6v                        1/1     Running   0
7m22s
graphql-server-5f5976f4bd-vbb4z            1/1     Running   0
7m13s
identity-56f78b8f9f-8h9p9                  1/1     Running   0
8m29s
influxdb2-0                                1/1     Running   0
11m
krakend-6f8d995b4d-5khkl                   1/1     Running   0
7m7s
license-5b5db87c97-jmxzc                   1/1     Running   0
9m
login-ui-57b57c74b8-6xtv7                  1/1     Running   0
7m10s
loki-0                                     1/1     Running   0
11m
monitoring-operator-9dbc9c76d-8znck        2/2     Running   0
7m33s
nats-0                                     1/1     Running   0
11m
```

```
nats-1                                      1/1     Running   0
10m
nats-2                                      1/1     Running   0
10m
nautilus-6b9d88bc86-h8kfb                   1/1     Running   0
8m6s
nautilus-6b9d88bc86-vn68r                   1/1     Running   0
8m35s
openapi-b87d77dd8-5dz9h                     1/1     Running   0
9m7s
polaris-consul-consul-5ljfb                 1/1     Running   0
11m
polaris-consul-consul-s5d5z                 1/1     Running   0
11m
polaris-consul-consul-server-0              1/1     Running   0
11m
polaris-consul-consul-server-1              1/1     Running   0
11m
polaris-consul-consul-server-2              1/1     Running   0
11m
polaris-consul-consul-twmpq                 1/1     Running   0
11m
polaris-mongodb-0                           2/2     Running   0
11m
polaris-mongodb-1                           2/2     Running   0
10m
polaris-mongodb-2                           2/2     Running   0
10m
polaris-ui-84dc87847f-zrg8w                 1/1     Running   0
7m12s
polaris-vault-0                             1/1     Running   0
11m
polaris-vault-1                             1/1     Running   0
11m
polaris-vault-2                             1/1     Running   0
11m
public-metrics-657698b66f-67pgt             1/1     Running   0
8m47s
storage-backend-metrics-6848b9fd87-w7x8r    1/1     Running   0
8m39s
storage-provider-5ff5868cd5-r9hj7           1/1     Running   0
8m45s
telegraf-ds-dw4hg                           1/1     Running   0
7m23s
telegraf-ds-k92gn                           1/1     Running   0
7m23s
```

```
telegraf-ds-mmxjl                          1/1     Running   0
7m23s
telegraf-ds-nhs8s                          1/1     Running   0
7m23s
telegraf-ds-rj7lw                          1/1     Running   0
7m23s
telegraf-ds-tqrkb                          1/1     Running   0
7m23s
telegraf-rs-9mwgj                          1/1     Running   0
7m23s
telemetry-service-56c49d689b-ffrzx         1/1     Running   0
8m42s
tenancy-767c77fb9d-g9ctv                   1/1     Running   0
8m52s
traefik-5857d87f85-7pmx8                   1/1     Running   0
6m49s
traefik-5857d87f85-cpxgv                   1/1     Running   0
5m34s
traefik-5857d87f85-lvmlb                   1/1     Running   0
4m33s
traefik-5857d87f85-t2xlk                   1/1     Running   0
4m33s
traefik-5857d87f85-v9wpf                   1/1     Running   0
7m3s
trident-svc-595f84dd78-zb816               1/1     Running   0
8m54s
vault-controller-86c94fbf4f-krttq          1/1     Running   0
9m24s
```

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

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

3. Lorsque tous les modules sont en cours d'exécution, vérifiez que l'installation a réussi en récupérant `AstraControlCenter` Instance installée par l'opérateur du centre de contrôle Astra.

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

4. Vérifier le `status.deploymentState` dans le champ de réponse pour le `Deployed` valeur. Si le déploiement a échoué, un message d'erreur s'affiche à la place.

   Vous utiliserez le `uuid` à l'étape suivante.

```
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. Pour obtenir le mot de passe unique que vous utiliserez lorsque vous vous connecterez à Astra Control Center, copiez le `status.uuid` valeur de la réponse à l'étape précédente. Le mot de passe est `ACC-` Suivi de la valeur UUID (`ACC-[UUID]` ou, dans cet exemple, `ACC-c49008a5-4ef1-4c5d-a53e-830daf994116`).

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

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

Étapes

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

   Vous pouvez créer un certificat personnalisé après la connexion.

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

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

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

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

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

Dépanner l'installation

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

Étapes

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

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

Et la suite

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

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

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

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

Ce dont vous avez besoin

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

```
oc get clusteroperators
```

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

```
oc get apiservices
```

- Vous avez créé une adresse FQDN pour Astra Control Center dans votre data Center.
- Vous disposez des autorisations nécessaires et de l'accès à Red Hat OpenShift Container Platform pour effectuer les étapes d'installation décrites.

Étapes

- Téléchargez le pack Astra Control Center
- Déballez le bundle et modifiez le répertoire
- Ajoutez les images à votre registre local
- Recherchez la page d'installation de l'opérateur
- Poser l'opérateur
- Poser le centre de contrôle Astra

Téléchargez le pack Astra Control Center

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

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

Déballez le bundle et modifiez le répertoire

1. Extraire les images :

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

2. Passez au répertoire Astra.

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

Ajoutez les images à votre registre local

1. Ajoutez les fichiers du répertoire d'images de l'Astra Control Center à votre registre local.

   Voir les exemples de scripts pour le chargement automatique des images ci-dessous.

   a. Connectez-vous à votre registre :

      Docker :

      ```
      docker login [your_registry_path]
      ```

      Podman :

      ```
      podman login [your_registry_path]
      ```

   b. Utilisez le script approprié pour charger les images, les marquer et pousser les images dans votre registre local :

      Docker :

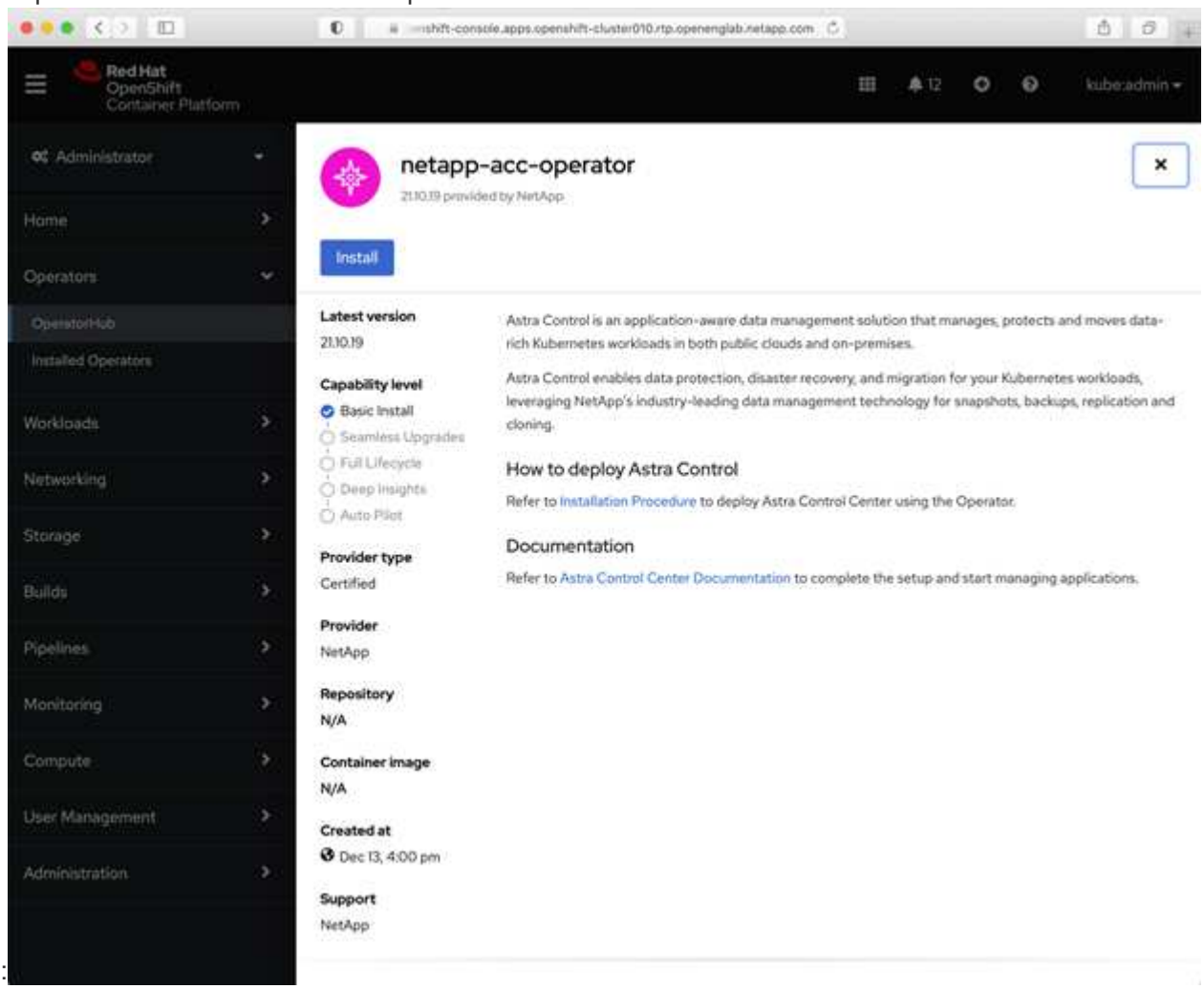
      ```
      export REGISTRY=[Docker_registry_path]
      for astraImageFile in $(ls images/*.tar) ; do
        # Load to local cache. And store the name of the loaded image
      trimming the 'Loaded images: '
        astraImage=$(docker load --input ${astraImageFile} | sed 's/Loaded
      image: //')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

Recherchez la page d'installation de l'opérateur

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

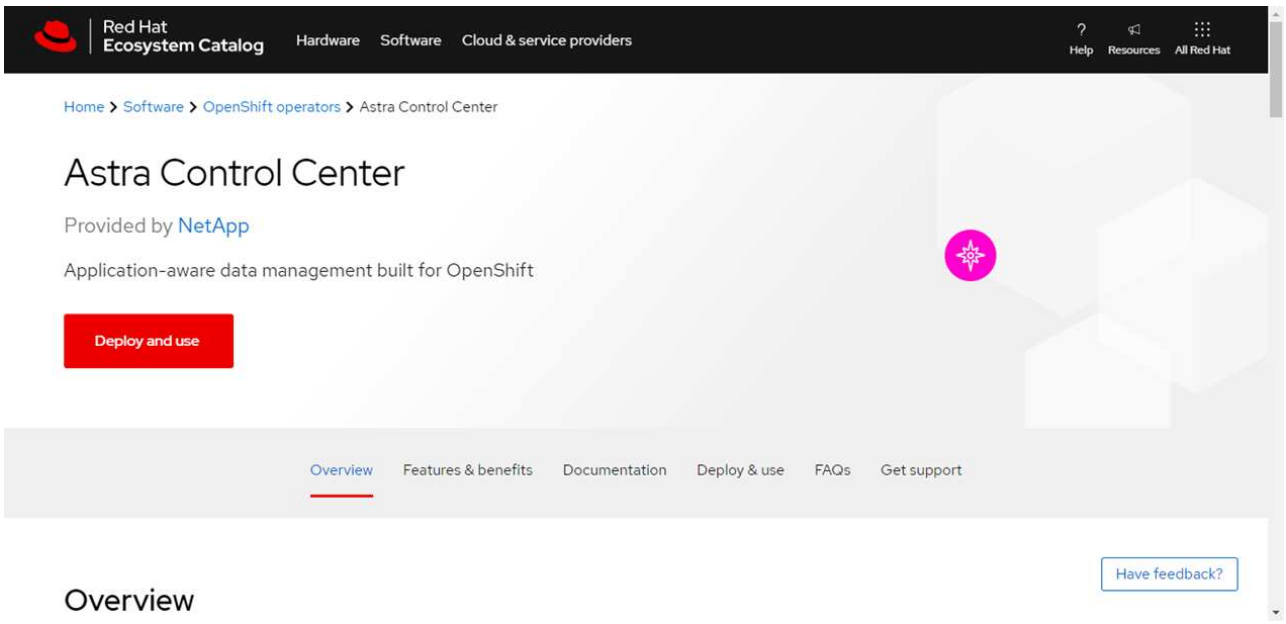


     i. Connectez-vous à l'interface utilisateur de OpenShift Container Platform.
     ii. Dans le menu latéral, sélectionnez **Operators > OperatorHub**.

iii. Sélectionnez l'opérateur du centre de contrôle Astra NetApp.

iv. Sélectionnez **installer**.

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



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

ii. Sélectionnez **déployer et utiliser**.

Poser l'opérateur

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

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

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

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

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

   c. Sélectionnez **installer**.

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

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

Poser le centre de contrôle Astra

1. Depuis la console dans la vue détaillée du conducteur du centre de contrôle Astra, sélectionnez `Create instance` Dans la section API fournies.
2. Complétez le `Create AstraControlCenter` champ de formulaire :

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

3. Sélectionnez `Create`.

Et la suite

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

Configurer le centre de contrôle Astra

Astra Control Center prend en charge et surveille ONTAP et Astra Data Store en tant que système back-end de stockage. Après avoir installé Astra Control Center, connectez-vous à l'interface utilisateur et modifiez votre mot de passe, vous devez configurer une licence, ajouter des clusters, gérer le stockage et ajouter des compartiments.

Tâches

- Ajoutez une licence pour Astra Control Center
- Ajouter un cluster
- Ajout d'un système back-end
- Ajouter un godet

Ajoutez une licence pour Astra Control Center

Vous pouvez ajouter une nouvelle licence à l'aide de l'interface utilisateur ou de "API" Pour bénéficier de toutes les fonctionnalités de l'Astra Control Center. Sans licence, votre utilisation d'Astra Control Center se limite à la gestion des utilisateurs et à l'ajout de nouveaux clusters.

Ce dont vous avez besoin

Lorsque vous avez téléchargé Astra Control Center à partir du "Site de support NetApp", Vous avez également téléchargé le fichier de licence NetApp (NLF). Assurez-vous d'avoir accès à ce fichier de licence.

Pour mettre à jour une évaluation existante ou une licence complète, voir "Mettre à jour une licence existante".

Ajoutez une licence d'évaluation ou complète

Les licences Astra Control Center mesurent les ressources CPU avec des unités de processeur Kubernetes. La licence doit tenir compte des ressources CPU attribuées aux nœuds workers de tous les clusters Kubernetes gérés. Avant d'ajouter une licence, vous devez obtenir le fichier de licence (NLF) du "Site de support NetApp".

Vous pouvez également essayer Astra Control Center avec une licence d'évaluation qui vous permet d'utiliser Astra Control Center pendant 90 jours à compter de la date de téléchargement de la licence. Vous pouvez vous inscrire pour une version d'évaluation gratuite en vous inscrivant "ici".

Si votre installation dépasse le nombre de processeurs sous licence, Astra Control Center vous empêche de gérer de nouvelles applications. Une alerte s'affiche lorsque la capacité est dépassée.

Étapes

1. Connectez-vous à l'interface utilisateur du centre de contrôle Astra.
2. Sélectionnez **compte** > **Licence**.
3. Sélectionnez **Ajouter licence**.
4. Accédez au fichier de licence (NLF) que vous avez téléchargé.
5. Sélectionnez **Ajouter licence**.

La page **Account** > **License** affiche les informations de licence, la date d'expiration, le numéro de série de licence, l'ID de compte et les unités UC utilisées.

Si vous disposez d'une licence d'évaluation, veillez à stocker votre identifiant de compte afin d'éviter toute perte de données en cas d'échec du Centre de contrôle Astra si vous n'envoyez pas d'ASUP.

Ajouter un cluster

Pour commencer à gérer vos applications, ajoutez un cluster Kubernetes et gérez-le comme une ressource de calcul. Il faut ajouter un cluster pour découvrir vos applications Kubernetes pour Astra Control Center. Pour la prévisualisation du Data Store d'Astra, vous devez ajouter le cluster d'applications Kubernetes qui contient des applications qui utilisent des volumes provisionnés par l'aperçu d'Astra Data Store.

Nous vous recommandons de gérer le cluster qu'Astra Control Center déploie en premier avant d'ajouter d'autres clusters à Astra Control Center. La gestion du cluster initial est nécessaire pour envoyer les données Kubemetrics et les données associées au cluster pour les mesures et le dépannage. Vous pouvez utiliser la fonction **Ajouter un cluster** pour gérer un cluster avec Astra Control Center.

Lorsque Astra Control gère un cluster, il assure le suivi de la classe de stockage par défaut du cluster. Si vous modifiez la classe de stockage à l'aide de `kubectl` Contrôle Astra rétablit le changement. Pour modifier la classe de stockage par défaut d'un cluster géré par Astra Control, utilisez l'une des méthodes suivantes :

- Utilisez l'API de contrôle Astra `PUT /managedClusters` Et attribuez une classe de stockage par défaut différente à l' `DefaultStorageClass` paramètre
- Supprimez le cluster de la gestion Astra Control et ajoutez-le à nouveau avec une classe de stockage par défaut différente sélectionnée

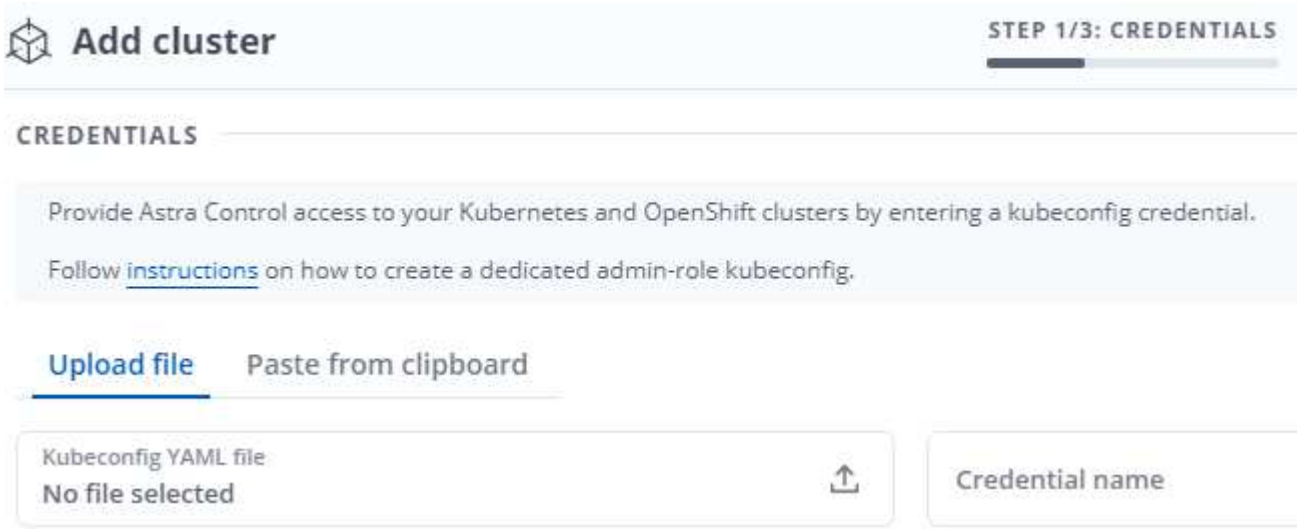
Ce dont vous avez besoin, 8217;ll

Avant d'ajouter un cluster, vérifiez et effectuez les opérations nécessaires "tâches préalables".

Étapes

1. Dans **Dashboard** de l'interface utilisateur du Centre de contrôle Astra, sélectionnez **Add** dans la section clusters.
2. Dans la fenêtre **Ajouter un cluster** qui s'ouvre, chargez un `kubeconfig.yaml` classez le contenu d'un `kubeconfig.yaml` fichier.

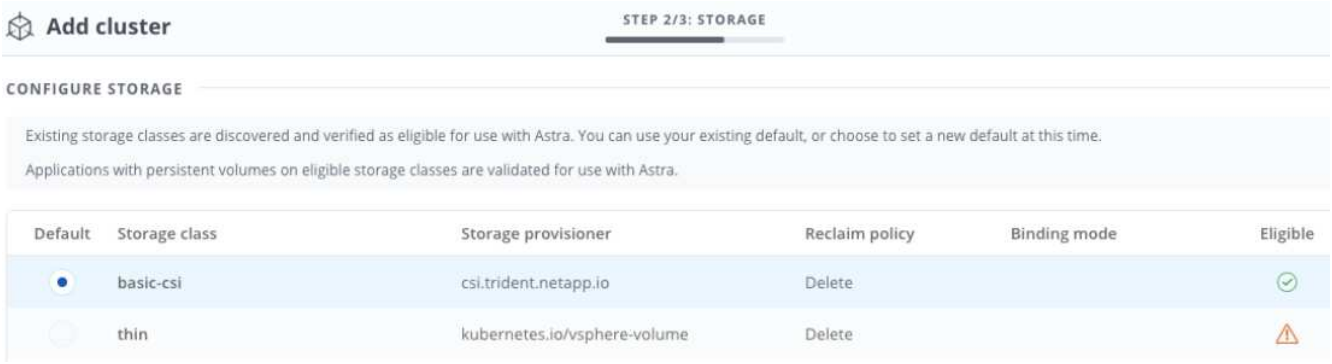
Le `kubeconfig.yaml` le fichier doit inclure **uniquement les informations d'identification du cluster pour un cluster**.



Si vous créez la vôtre `kubeconfig` fichier, vous ne devez définir que **un** élément de contexte dans celui-ci. Voir "Documentation Kubernetes" pour plus d'informations sur la création `kubeconfig` fichiers.

3. Indiquez un nom d'identification. Par défaut, le nom des identifiants est automatiquement renseigné comme nom du cluster.
4. Sélectionnez **configurer le stockage**.
5. Sélectionnez la classe de stockage à utiliser pour ce cluster Kubernetes et sélectionnez **Review**.

Nous vous recommandons de sélectionner une classe de stockage Trident avec le stockage ONTAP ou le magasin de données Astra.



6. Vérifiez les informations, et si tout semble bien, sélectionnez **Ajouter cluster**.

Résultat

Le cluster passe à l'état **découverte**, puis à **en cours d'exécution**. Vous avez ajouté un cluster Kubernetes et gérez-le dans Astra Control Center.

Une fois que vous avez ajouté un cluster à gérer dans Astra Control Center, le déploiement de l'opérateur de surveillance peut prendre quelques minutes. En attendant, l'icône notification devient rouge et consigne un événement **échec de la vérification de l'état de l'agent de surveillance**. Vous pouvez ignorer cela car le problème résout lorsque le centre de contrôle Astra obtient le statut correct. Si le problème ne résout pas le problème en quelques minutes, accédez au cluster, puis exécutez-le `oc get pods -n netapp-monitoring` comme point de départ. Vous devrez consulter les journaux de l'opérateur de surveillance pour déboguer le problème.

Ajout d'un système back-end

Vous pouvez ajouter un système de stockage back-end pour qu'Astra Control puisse gérer ses ressources. La gestion des clusters de stockage d'Astra Control en tant que backend de stockage vous permet d'obtenir des liens entre les volumes persistants (PVS) et le back-end de stockage, ainsi que des metrics de stockage supplémentaires.

Vous pouvez ajouter un stockage back-end découvert en parcourant les invites du tableau de bord ou du menu Backends.

Ce dont vous avez besoin

- Vous avez "ajout d'un cluster" Il est géré par Astra Control.

Le cluster géré possède un back-end pris en charge qui peut être découvert par Astra Control.

- Pour les installations de prévisualisation d'Astra Data Store, vous avez ajouté votre cluster d'applications Kubernetes.

Après avoir ajouté votre cluster d'applications Kubernetes pour Astra Data Store, le cluster apparaît comme `unmanaged` dans la liste des systèmes back-end découverts. Vous devez ensuite ajouter le cluster de calcul qui contient Astra Data Store et qui intègre le cluster d'applications Kubernetes. Vous pouvez le faire à partir de **Backends** dans l'interface utilisateur. Sélectionnez le menu actions du cluster, puis `Manage`, et "ajouter le cluster". Après l'état du cluster de `unmanaged` Modifications au nom du cluster Kubernetes, vous pouvez procéder à l'ajout d'un back-end.

Étapes

1. Effectuez l'une des opérations suivantes :
   - Depuis **Dashboard** :
     i. Dans la section backend de stockage du tableau de bord, sélectionnez **gérer**.
     ii. Dans la section Dashboard Resource Summary > Storage backend, sélectionnez **Add**.
   - À partir de **Backends** :
     i. Dans la zone de navigation de gauche, sélectionnez **Backends**.
     ii. Sélectionnez **gérer**.
2. Effectuez l'une des opérations suivantes en fonction de votre type de système back-end :
   - **Magasin de données Astra**:
     i. Sélectionnez l'onglet **Astra Data Store**.
     ii. Sélectionnez le cluster de calcul géré et sélectionnez **Suivant**.
     iii. Confirmez les détails du back-end et sélectionnez **gérer le back-end de stockage**.
   - **ONTAP** :
     i. Entrez les informations d'identification administrateur ONTAP et sélectionnez **Revue**.
     ii. Confirmez les détails du back-end et sélectionnez **gérer**.

Le back-end apparaît dans `available` état dans la liste avec des informations récapitulatives.

Vous devrez peut-être actualiser la page pour que le back-end apparaisse.

Ajouter un godet

Il est essentiel d'ajouter des fournisseurs de compartiments de stockage objet pour sauvegarder les applications et le stockage persistant ou pour cloner les applications entre les clusters. Astra Control stocke les sauvegardes ou les clones dans les compartiments de magasin d'objets que vous définissez.

Lorsque vous ajoutez un godet, Astra Control marque un godet comme indicateur de compartiment par défaut. Le premier compartiment que vous créez devient le compartiment par défaut.

Il n'est pas nécessaire de cloner la configuration de vos applications et le stockage persistant vers le même cluster.

Utiliser l'un des types de godet suivants :

- NetApp ONTAP S3
- NetApp StorageGRID S3

- S3 générique

> Bien qu'Astra Control Center prenne en charge Amazon S3 en tant que fournisseur de compartiments S3 génériques, Astra Control Center peut ne pas prendre en charge tous les fournisseurs de magasins d'objets qui affirment la prise en charge d'Amazon S3.

Pour plus d'informations sur l'ajout de compartiments à l'aide de l'API de contrôle Astra, reportez-vous à la section "Informations sur l'automatisation et les API d'Astra".

Étapes

1. Dans la zone de navigation de gauche, sélectionnez **godets**.
   a. Sélectionnez **Ajouter**.
   b. Sélectionner le type de godet.

   > Lorsque vous ajoutez un compartiment, sélectionnez le fournisseur approprié et fournissez les identifiants appropriés pour ce fournisseur. Par exemple, l'interface utilisateur accepte NetApp ONTAP S3 comme type et accepte les identifiants StorageGRID. Toutefois, toutes les futures sauvegardes et restaurations des applications à l'aide de ce compartiment échoueront.

   c. Créer un nouveau nom de compartiment ou saisir un nom de compartiment existant et une description facultative.

   > Le nom et la description du compartiment apparaissent comme un emplacement de sauvegarde que vous pouvez choisir ultérieurement lors de la création d'une sauvegarde. Ce nom apparaît également lors de la configuration de la règle de protection.

   d. Entrez le nom ou l'adresse IP du terminal S3.
   e. Si vous souhaitez que ce compartiment soit utilisé comme compartiment par défaut pour toutes les sauvegardes, vérifiez le `Make this bucket the default bucket for this private cloud` option.

   > Cette option n'apparaît pas pour le premier compartiment que vous créez.

   f. Continuez en ajoutant informations d'identification.

Ajoutez des identifiants d'accès S3

Ajoutez les identifiants d'accès S3 à tout moment.

Étapes

1. Dans la boîte de dialogue compartiments, sélectionnez l'onglet **Ajouter** ou **utiliser existant**.
   a. Saisissez un nom pour l'identifiant qui le distingue des autres identifiants dans Astra Control.
   b. Saisissez l'ID d'accès et la clé secrète en collant le contenu dans le presse-papiers.

Et la suite ?

Maintenant que vous vous êtes connecté et que vous avez ajouté des clusters à Astra Control Center, vous

pouvez commencer à utiliser les fonctions de gestion des données applicatives d'Astra Control Center.

- "Gérer les utilisateurs"
- "Commencez à gérer les applications"
- "Protégez vos applications"
- "Clonage des applications"
- "Gérer les notifications"
- "Connectez-vous à Cloud Insights"
- "Ajouter un certificat TLS personnalisé"

Trouvez plus d'informations

- "Utilisez l'API de contrôle Astra"
- "Problèmes connus"

Conditions préalables à l'ajout d'un cluster

Assurez-vous que les conditions préalables sont remplies avant d'ajouter un cluster. Vous devez également effectuer les vérifications d'admissibilité pour vous assurer que votre grappe est prête à être ajoutée au Centre de contrôle Astra.

Ce dont vous avez besoin avant d'ajouter un cluster

- Un des types de clusters suivants :
  - Clusters qui exécutent OpenShift 4.6, 4.7 ou 4.8 avec des classes de stockage Astra Trident basées sur le data Store d'Astra, ou sur ONTAP 9.5 ou version ultérieure
  - Clusters exécutant Rancher 2.5
  - Clusters qui exécutent Kubernetes 1.19 vers 1.21 (dont 1.21.x)

    Assurez-vous que vos clusters disposent d'un ou plusieurs nœuds workers avec au moins 1 Go de RAM disponible pour l'exécution des services de télémétrie.

    Si vous prévoyez d'ajouter un deuxième cluster OpenShift 4.6, 4.7 ou 4.8 en tant que ressource de calcul gérée, assurez-vous que la fonctionnalité Snapshot de volume Astra Trident est activée. Découvrez l'Astra Trident officielle "instructions" Pour activer et tester des copies Snapshot de volume avec Astra Trident.

- Le superutilisateur et l'ID utilisateur définis sur le système ONTAP de sauvegarde pour sauvegarder et restaurer des applications avec le Centre de contrôle Astra. Exécutez la commande suivante dans la ligne de commande ONTAP :
  `export-policy rule modify -vserver <storage virtual machine name> -policyname <policy name> -ruleindex 1 -superuser sysm --anon 65534`
- Découvrez Astra Trident `volumesnapshotclass` objet défini par un administrateur. Découvrez Astra Trident "instructions" Pour activer et tester des copies Snapshot de volume avec Astra Trident.
- Assurez-vous de n'avoir qu'une seule classe de stockage par défaut définie pour votre cluster Kubernetes.

Effectuer des vérifications d'éligibilité

Effectuez les contrôles d'éligibilité suivants pour vous assurer que votre grappe est prête à être ajoutée au Centre de contrôle Astra.

Étapes

1. Vérifiez la version de Trident.

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

Si Trident est présent, vous voyez des valeurs de sortie similaires à celles illustrées dans l'exemple suivant :

```
NAME      VERSION
trident   21.04.0
```

Si Trident n'existe pas, vous voyez des résultats similaires à ce qui suit :

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

> Si Trident n'est pas installé ou si la version installée n'est pas la dernière, vous devez installer la dernière version de Trident avant de continuer. Voir la "Documentation Trident" pour obtenir des instructions.

2. Vérifiez si les classes de stockage utilisent les pilotes Trident pris en charge. Le nom de provisionnement doit être `csi.trident.netapp.io`. Voir l'exemple suivant :

```
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
```

Créez un kubeconfig. Rôle admin

Avant d'effectuer les étapes suivantes, assurez-vous que vous disposez des éléments suivants sur votre machine :

- `kubectl` v1.19 ou version ultérieure installé
- Un kubeconfig actif avec des droits d'administrateur de cluster pour le contexte actif

Étapes

1. Créer un compte de service comme suit :

   a. Créez un fichier de compte de service appelé `astracontrol-service-account.yaml`.

      Ajustez le nom et l'espace de noms selon vos besoins. Si des modifications sont apportées ici, vous devez appliquer les mêmes modifications dans les étapes suivantes.

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

+

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

   a. Appliquer le compte de service :

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

2. Accordez des autorisations d'administration du cluster comme suit :

   a. Créer un `ClusterRoleBinding` fichier appelé `astracontrol-clusterrolebinding.yaml`.

      Ajustez les noms et espaces de noms modifiés lors de la création du compte de service, le cas échéant.

```
<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. Appliquer la liaison de rôle de cluster :

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

3. Indiquez les secrets du compte de service, en les remplaçant `<context>` avec le contexte approprié pour votre installation :

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

La fin de la sortie doit ressembler à ce qui suit :

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

Les indices pour chaque élément dans `secrets` la matrice commence par 0. Dans l'exemple ci-dessus, l'index de `astracontrol-service-account-dockercfg-vhz87` serait 0 et l'index pour `astracontrol-service-account-token-r59kr` serait 1. Dans votre résultat, notez l'index du nom du compte de service qui contient le mot "jeton".

4. Générez le kubeconfig comme suit :

a. Créer un `create-kubeconfig.sh` fichier. Remplacement `TOKEN_INDEX` au début du script suivant avec la valeur correcte.

```
<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. Source des commandes à appliquer à votre cluster Kubernetes.

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

5. (**Facultatif**) Renommer le kubeconfig en un nom significatif pour votre grappe. Protéger les informations d'identification du cluster.

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

Et la suite ?

Maintenant que vous avez vérifié que les conditions préalables sont remplies, vous êtes prêt à "ajouter un cluster".

Trouvez plus d'informations

- "Documentation Trident"
- "Utilisez l'API de contrôle Astra"

Ajouter un certificat TLS personnalisé

Vous pouvez supprimer le certificat TLS auto-signé existant et le remplacer par un certificat TLS signé par une autorité de certification (AC).

Ce dont vous avez besoin

- Cluster Kubernetes avec Astra Control Center installé
- Accès administratif à un shell de commande sur le cluster à exécuter `kubectl` commandes
- Clé privée et fichiers de certificat de l'autorité de certification

Supprimez le certificat auto-signé

Supprimez le certificat TLS auto-signé existant.

1. Avec SSH, connectez-vous au cluster Kubernetes qui héberge Astra Control Center en tant qu'utilisateur administratif.
2. Recherchez le code secret TLS associé au certificat en cours à l'aide de la commande suivante, remplacement `<ACC-deployment-namespace>` Avec l'espace de noms de déploiement d'Astra Control Center :

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

3. Supprimez le certificat et le secret actuellement installés à l'aide des commandes suivantes :

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

Ajoutez un nouveau certificat

Ajoutez un nouveau certificat TLS signé par une autorité de certification.

1. Utilisez la commande suivante pour créer le nouveau secret TLS avec la clé privée et les fichiers de certificat de l'autorité de certification, en remplaçant les arguments entre parenthèses <> par les informations appropriées :

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

2. Utilisez la commande et l'exemple suivants pour modifier le fichier CRD (Custom Resource Definition) du cluster et modifier `spec.selfSigned` valeur à `spec.ca.secretName` Pour consulter le secret TLS créé précédemment :

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

#spec:
#  selfSigned: {}

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

3. Utilisez la commande suivante et exemple de résultat pour vérifier que les modifications sont correctes et le cluster est prêt à valider les certificats, en remplaçant `<ACC-deployment-namespace>` Avec l'espace de noms de déploiement d'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. Créer le `certificate.yaml` fichier avec l'exemple suivant, en remplaçant les valeurs de paramètre fictif entre parenthèses <> par les informations appropriées :

```
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. Créez le certificat à l'aide de la commande suivante :

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

6. À l'aide de la commande et de l'exemple de sortie suivants, vérifiez que le certificat a été créé correctement et avec les arguments que vous avez spécifiés lors de la création (tels que le nom, la durée, la date limite de renouvellement et les noms 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. Modifiez l'option Ingress CRD TLS pour pointer vers votre nouveau secret de certificat à l'aide de la commande suivante et de l'exemple, en remplaçant les valeurs de paramètre fictif entre parenthèses <> par les informations appropriées :

```
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. À l'aide d'un navigateur Web, accédez à l'adresse IP de déploiement d'Astra Control Center.
9. Vérifiez que les détails du certificat correspondent aux détails du certificat que vous avez installé.
10. Exportez le certificat et importez le résultat dans le gestionnaire de certificats de votre navigateur Web.

Foire aux questions pour Astra Control Center

Cette FAQ peut vous aider si vous cherchez juste une réponse rapide à une question.

Présentation

Les sections suivantes fournissent des réponses à des questions supplémentaires que vous pourriez vous poser lorsque vous utilisez le centre de contrôle Astra. Pour plus de précisions, veuillez contacter astra.feedback@netapp.com

Accès au centre de contrôle Astra

Qu'est-ce que l'URL de contrôle Astra?

Astra Control Center utilise l'authentification locale et une URL spécifique à chaque environnement.

Pour l'URL, dans un navigateur, entrez le nom de domaine complet (FQDN) que vous avez défini dans le champ spec.astraAddress dans le fichier astra_control_Center_min.yaml personnalisé Resource definition (CRD) lorsque vous avez installé Astra Control Center. L'e-mail est la valeur que vous avez définie dans le champ spec.email de l'astra_Control_Center_min.yaml CRD.

J'utilise la licence d'évaluation. Comment puis-je passer à la licence complète?

Vous pouvez facilement passer à une licence complète en obtenant le fichier de licence NetApp (NLF).

Étapes

- Dans le menu de navigation de gauche, sélectionnez **compte** > **Licence**.
- Sélectionnez **Ajouter licence**.
- Naviguez jusqu'au fichier de licence que vous avez téléchargé et sélectionnez **Ajouter**.

J'utilise la licence d'évaluation. Puis-je toujours gérer les applications ?

Oui, vous pouvez tester la fonctionnalité de gestion des applications avec la licence d'évaluation.

Enregistrement des clusters Kubernetes

J'ai besoin d'ajouter des nœuds workers à mon cluster Kubernetes après avoir ajouté Astra Control. Que dois-je faire?

De nouveaux nœuds workers peuvent être ajoutés aux pools existants. Elles seront automatiquement découvertes par Astra Control. Si les nouveaux nœuds ne sont pas visibles dans Astra Control, vérifiez si les nouveaux nœuds de travail exécutent le type d'image pris en charge. Vous pouvez également vérifier l'état de santé des nouveaux nœuds workers à l'aide de la `kubectl get nodes` commande.

Comment puis-je dégérer correctement un cluster?

1. "Gérez les applications avec Astra Control".
2. "Dégérer le cluster à partir d'Astra Control".

Que se passe-t-il pour mes applications et données après avoir retiré le cluster Kubernetes d'Astra Control?

La suppression d'un cluster d'Astra Control ne modifie pas la configuration du cluster (applications et stockage persistant). Toute restauration de snapshots ou de sauvegardes Astra Control effectuée sur ce cluster sera indisponible. Les sauvegardes de stockage persistant créées par Astra Control restent dans le contrôle d'Astra, mais elles sont indisponibles pour les restaurations.

> Retirez toujours un cluster d'Astra Control avant de le supprimer par d'autres méthodes. La suppression d'un cluster à l'aide d'un autre outil alors qu'il est toujours géré par Astra Control peut causer des problèmes pour votre compte Astra Control.

NetApp Trident sera-t-il désinstallé lorsque je retire un cluster Kubernetes d'Astra Control ?

Trident ne sera pas désinstallé d'un cluster lorsque vous le supprimez d'Astra Control.

La gestion des applications

Astra Control peut-il déployer une application?

Astra Control ne déploie pas d'applications. Les applications doivent être déployées en dehors d'Astra Control.

Que se passe-t-il pour les applications après que je les ai cessent de les gérer à partir d'Astra Control?

Toutes les sauvegardes ou tous les instantanés existants seront supprimés. Les applications et les données restent disponibles. Les opérations de gestion des données ne seront pas disponibles pour les applications non gérées ni pour les sauvegardes ou snapshots qui y appartiennent.

Astra Control peut-il gérer une application qui se trouve sur un système de stockage autre que NetApp?

Non Astra Control peut découvrir des applications qui utilisent un stockage autre que NetApp, mais il ne peut pas gérer une application qui utilise un stockage non NetApp.

Devrais-je gérer Astra Control lui-même? non, vous ne devriez pas gérer Astra Control lui-même parce qu'il s'agit d'une "application système".

Les opérations de gestion des données

Il y a des instantanés dans mon compte que je n'ai pas créés. D'où viennent-ils?

Dans certains cas, Astra Control crée automatiquement un snapshot dans le cadre d'un processus de sauvegarde, de clonage ou de restauration.

Mon application utilise plusieurs PVS. ASTRA Control prendra-t-il des instantanés et des sauvegardes de toutes ces ESV?

Oui. Une opération d'instantané sur une application par Astra Control inclut un instantané de tous les volumes persistants liés aux demandes de volume persistant de l'application.

Puis-je gérer les instantanés pris par Astra Control directement via une interface ou un stockage objet différent?

Non Les copies Snapshot et les sauvegardes effectuées par Astra Control ne peuvent être gérées qu'avec Astra Control.