Commencez

Astra Control Center

NetApp
November 21, 2023

Sommaire

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
- Systèmes back-end de stockage pris en charge
- Configuration requise en cluster des applications
- De gestion des applications
- Conditions préalables à la réplication
- Accès à Internet
- Licence
- Entrée pour les clusters Kubernetes sur site
- Configuration réseau requise
- Navigateurs Web pris en charge

De l'environnement opérationnel

Le centre de contrôle Astra a été validé pour les types d'environnements opérationnels suivants :

- Google Anthos 1.10 ou 1.11
- Kubernetes 1.22 à 1.24
- Rancher Kubernetes Engine (RKE) :
  - RKE 1.2.16 avec Rancher 2.5.12 et RKE 1.3.3 avec 2.6.3
  - RKE 2 (v1.23,6+rke2r2) avec Rancher 2.6.3
- Red Hat OpenShift Container Platform 4.8, 4.9 ou 4.10
- VMware Tanzu Kubernetes Grid 1.4 ou 1.5
- VMware Tanzu Kubernetes Grid Integrated Edition 1.12.2 ou 1.13

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 du système back-end	Au moins 500 Go disponibles
Nœuds worker	Au moins 3 nœuds workers au total, avec 4 cœurs de processeurs et 12 Go de RAM chacun
Adresse FQDN	Une adresse FQDN pour Astra Control Center

Composant	Conditions requises
Astra Trident	Astra Trident 21.10.1 ou version ultérieure installé et configuré avec Astra Trident 22.07 ou version ultérieure pour la réplication d'applications basée sur SnapMirror

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 pour le cluster VMware Tanzu Kubernetes Grid

Lorsque vous hébergez Astra Control Center sur un cluster VMware Tanzu Kubernetes Grid (TKG) ou Tanzu Kubernetes Grid Integrated Edition (TKGi), gardez à l'esprit les considérations suivantes.

- Désactivez la mise en œuvre par défaut des classes de stockage TKG ou TKGi sur les clusters d'applications devant être gérés par Astra Control. Vous pouvez le faire en modifiant le `TanzuKubernetesCluster` ressource sur le cluster d'espace de noms.
- Dans le cadre de l'installation d'Astra Control Center, les ressources suivantes sont créées dans un environnement restreint de politique de sécurité de pod (PSP) :
  - politique de sécurité des pods

    - Le rôle RBAC
    - RBAC RoleBinding les ressources RBAC et RoleBinding sont créées dans le `netapp-acc` espace de noms.
- Tenez compte des exigences spécifiques de l'Astra Trident lorsque vous déployez le centre de contrôle Astra dans un environnement TKG ou TKGi. Pour plus d'informations, reportez-vous à la section "Documentation Astra Trident".

> Le token de fichier de configuration VMware TKG et TKGi par défaut expire dix heures après le déploiement. Si vous utilisez des produits de la gamme Tanzu, vous devez générer un fichier de configuration de cluster Kubernetes Tanzu avec un jeton non expirant pour éviter les problèmes de connexion entre Astra Control Center et les clusters d'applications gérés. Pour obtenir des instructions, rendez-vous sur "Documentation produit relative au data Center VMware NSX-T"

Exigences des clusters Google Anthos

Lorsque vous hébergez Astra Control Center sur un cluster Google Anthos, notez que Google Anthos inclut par défaut l'équilibreur de charge MetalLB et le service de passerelle d'entrée Istio, vous permettant d'utiliser simplement les fonctionnalités d'entrée génériques d'Astra Control Center pendant l'installation. Voir "Configurer le centre de contrôle Astra" pour plus d'informations.

Systèmes back-end de stockage pris en charge

Astra Control Center prend en charge les systèmes back-end de stockage suivants.

- NetApp ONTAP 9.5 ou versions ultérieures, AFF et FAS
- NetApp ONTAP 9.8 ou versions ultérieures AFF et FAS pour la réplication d'applications basée sur SnapMirror
- NetApp Cloud Volumes ONTAP

Pour utiliser Astra Control Center, vérifiez que vous disposez des licences ONTAP suivantes, en fonction de ce que vous devez accomplir :

- FlexClone
- SnapMirror : en option. Elle est nécessaire uniquement pour la réplication vers des systèmes distants à l'aide de la technologie SnapMirror. Reportez-vous à la section "Informations sur la licence SnapMirror".
- Licence S3 : en option. Nécessaire uniquement pour les compartiments ONTAP S3

Vous pouvez vérifier si votre système ONTAP dispose des licences requises. Reportez-vous à la section "Gérer les licences ONTAP".

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
Cronjob	CustomResourceDefinition	Ressource CustomResource
Ensemble de démonstrations	Déploiement.Config	HorizontalPodAutoscaler
Entrée	MutatingWebhook	Stratégie réseau
Demande de volume persistant	Pod	PodPetitionBudget
PodTemplate	Et de réplication	Rôle
RoleBinding	Itinéraire	Secret
Service	Compte de service	StatefulSet
ValidétingWebhook		

Conditions préalables à la réplication

La réplication de l'application Astra Control exige que les conditions préalables suivantes soient respectées avant de commencer :

- Pour assurer une reprise après incident transparente, nous vous recommandons de déployer Astra Control Center dans un troisième domaine de pannes ou un troisième site secondaire.
- Le cluster Kubernetes hôte de l'application et un cluster Kubernetes de destination doivent être disponibles

et connectés à deux clusters ONTAP, idéalement dans des domaines ou sites de défaillance différents.

- Les clusters ONTAP et le SVM hôte doivent être associés. Voir "Présentation du cluster et de SVM peering".
- Le SVM distant associé doit être disponible auprès de Trident sur le cluster de destination.
- Trident version 22.07 ou supérieure doit exister sur les clusters ONTAP source et de destination.
- Les licences asynchrones ONTAP SnapMirror via le bundle protection des données doivent être activées sur les clusters ONTAP source et cible. Voir "Présentation des licences SnapMirror dans ONTAP".
- Lorsque vous ajoutez un système de stockage back-end ONTAP à Astra Control Center, appliquez les identifiants de l'utilisateur avec le rôle « admin » qui possède des méthodes d'accès `http` et `ontapi` Activé sur les deux clusters ONTAP. Voir "Gérer les comptes d'utilisateurs" pour en savoir plus.
- Les clusters Kubernetes source et destination et les clusters ONTAP doivent être gérés par Astra Control.

> Vous pouvez répliquer simultanément une autre application (exécutée sur l'autre cluster ou site) dans la direction opposée. Par exemple, les applications A, B, C peuvent être répliquées depuis Datacenter 1 vers Datacenter 2. Et les applications X, y, Z peuvent être répliquées depuis Datacenter 2 vers Datacenter 1.

Découvrez comment "Répliquez vos applications sur un système distant grâce à la technologie SnapMirror".

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. Vous devez disposer d'une licence pour protéger vos applications et vos données. Reportez-vous à la section "Caractéristiques du centre de contrôle Astra" pour plus d'informations.

Vous pouvez 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".

Pour plus d'informations sur les licences requises pour les systèmes de stockage back-end ONTAP, reportez-vous à la "Systèmes back-end de stockage pris en charge".

Pour plus d'informations sur le fonctionnement des licences, reportez-vous à la section "Licences".

Entrée pour les clusters Kubernetes sur site

Vous pouvez choisir le type d'entrée de réseau utilisé par le centre de contrôle Astra. Par défaut, Astra Control Center déploie la passerelle Astra Control Center (service/trafik) comme ressource à l'échelle du cluster. Astra Control Center prend également en charge l'utilisation d'un équilibreur de charge de service, s'ils sont autorisés dans votre environnement. Si vous préférez utiliser un équilibreur de charge de service et que vous n'en avez pas encore configuré, vous pouvez utiliser l'équilibreur de charge 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.

> Si vous hébergez Astra Control Center sur un cluster Kubernetes Grid de Tanzu, utilisez le `kubectl get nsxlbmonitors -A` commande pour voir si un moniteur de service est déjà configuré pour accepter le trafic d'entrée. S'il en existe un, vous ne devez pas installer MetalLB, car le moniteur de service existant remplacera toute nouvelle configuration d'équilibreur de charge.

Pour plus d'informations, voir "Configurer l'entrée pour l'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).

> Vous pouvez déployer Astra Control Center dans un cluster Kubernetes à double pile, et Astra Control Center peut gérer les applications et les systèmes back-end de stockage qui ont été configurés pour un fonctionnement à double pile. Pour plus d'informations sur la configuration requise pour les clusters à double pile, consultez le "Documentation Kubernetes".

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	NetApp AutoSupport	443	HTTPS	Communication avec NetApp AutoSupport

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 ONTAP configuré par Trident

ou avec un système back-end de stockage 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 le "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.

  Vous pouvez également installer Astra Control Center avec un système de stockage back-end Cloud Volumes ONTAP.

En savoir plus sur "Installation du centre de contrôle Astra".

3 Effectuez certaines tâches de configuration initiales

- Ajoutez une licence Astra Control et toutes les licences ONTAP compatibles.
- Ajoutez un cluster Kubernetes et Astra Control Center découverte des détails.
- Ajouter un système back-end de stockage ONTAP.
- 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 "gérer des applications".
- Protégez les applications en configurant des stratégies de protection pour les applications, en répliquant les applications sur des systèmes distants et en migrant les applications. En savoir plus "protégez vos applications".
- Gérez les comptes (utilisateurs, rôles, LDAP pour l'authentification des utilisateurs, informations d'identification, connexions au référentiel, etc.). En savoir plus "gérer les utilisateurs".
- 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 "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 avec un système de stockage back-end Cloud Volumes ONTAP"

Installer le centre de contrôle Astra en suivant la procédure standard

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

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

Ce dont vous avez besoin

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

```
kubectl get clusteroperators
```

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

```
kubectl get apiservices
```

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

Description de la tâche

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

- Installe les composants Astra dans le `netapp-acc` (ou espace de nom personnalisé).
- Crée un compte par défaut.

- Définit une adresse e-mail d'utilisateur administratif par défaut et un mot de passe unique par défaut. Ce rôle propriétaire est attribué à cet utilisateur dans le système qui est nécessaire pour la première connexion à l'interface utilisateur.
- Vous aide à déterminer que toutes les POD Astra Control Center sont en cours d'exécution.
- Installe l'interface utilisateur Astra.

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

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

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

Étapes

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

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

Téléchargez et déballez le pack Astra Control Center

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

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

4. Extraire les images :

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

Installez le plug-in NetApp Astra kubectl

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

Ce dont vous avez besoin

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

Étapes

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

```
ls kubectl-astra/
```

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

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

Ajoutez les images à votre registre local

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

Docker

1. Passez au répertoire Astra :

```
cd acc
```

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

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

Podman

1. Connectez-vous à votre registre :

```
podman login [your_registry_path]
```

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

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

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

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

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

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

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

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

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

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

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

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

   Réponse :

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

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

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

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

Exemple de réponse :

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

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

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

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

Exemple de réponse :

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

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

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

Réponse

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

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

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

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

Poser le conducteur du centre de commande Astra

1. Modifier le répertoire :

```
cd manifests
```

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

```
vim astra_control_center_operator_deploy.yaml
```

> Un échantillon annoté YAML suit ces étapes.

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

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

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

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

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

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

3. Poser le conducteur du centre de commande Astra :

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

Exemple de réponse :

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

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

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

Configurer le centre de contrôle Astra

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

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

```
vim astra_control_center_min.yaml
```

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

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

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

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

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

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

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

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

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

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

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

j. Effectuez l'une des opérations suivantes :

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

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

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

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

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

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

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

Installation complète du centre de contrôle Astra et du conducteur

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

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

   Exemple de réponse :

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

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

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

   Exemple de réponse :

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

Vérifiez l'état du système

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

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

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

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

Exemple de réponse

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

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

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

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

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

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

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

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

Réponse :

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

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

Configurer l'entrée pour l'équilibrage de charge

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

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

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

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

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

Ce dont vous avez besoin

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

Étapes pour l'entrée Istio

1. Configurer l'entrée Istio.

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

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

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

   Exemple de commande :

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

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

   Exemple de commande :

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

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

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

   Résultat :

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

Pour le nouveau schéma v1, suivez cet exemple :

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

Résultat :

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

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

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

Réponse :

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

Étapes du contrôleur d'entrée Nginx

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

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

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

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

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

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

Étapes du contrôleur d'entrée OpenShift

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

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

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

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

Étapes

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

> Vous pouvez créer un certificat personnalisé après la connexion.

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

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

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

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

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

Dépanner l'installation

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

Étapes

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

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

Et la suite

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

=
:allow-uri-read:

Comprendre les restrictions de la stratégie de sécurité du pod

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

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

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

PSP installé par Astra Control Center

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

PSP créé lors de l'installation

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

La nouvelle règle et les objets ont les attributs suivants :

```
kubectl get psp

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

kubectl get role

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

kubectl get rolebinding

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

PSP créé pendant les opérations de sauvegarde

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

La nouvelle règle et les objets ont les attributs suivants :

```
kubectl get psp

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

kubectl get role

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

kubectl get rolebinding

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

PSP créé lors de la gestion du cluster

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

La nouvelle règle et les objets ont les attributs suivants :

```
kubectl get psp

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

kubectl get role

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

kubectl get rolebinding

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

Activer la communication réseau entre les espaces de noms

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

Étapes

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

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

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

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

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

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

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

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

Supprimer les limitations de ressources

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

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

Étapes

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

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

Réponse :

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

2. Supprimez tous les quotas de ressources par nom :

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

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

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

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

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

Réponse :

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

4. Supprimez les plages de limite par nom :

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

=
:allow-uri-read:

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

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

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

Ce dont vous avez besoin

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

```
oc get clusteroperators
```

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

est `true`) :

```
oc get apiservices
```

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

Étapes

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

Téléchargez et déballez le pack Astra Control Center

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

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

4. Extraire les images :

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

Installez le plug-in NetApp Astra kubectl

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

Ce dont vous avez besoin

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

Étapes

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

```
ls kubectl-astra/
```

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

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

Ajoutez les images à votre registre local

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

Docker

1. Passez au répertoire Astra :

```
cd acc
```

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

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

Podman

1. Connectez-vous à votre registre :

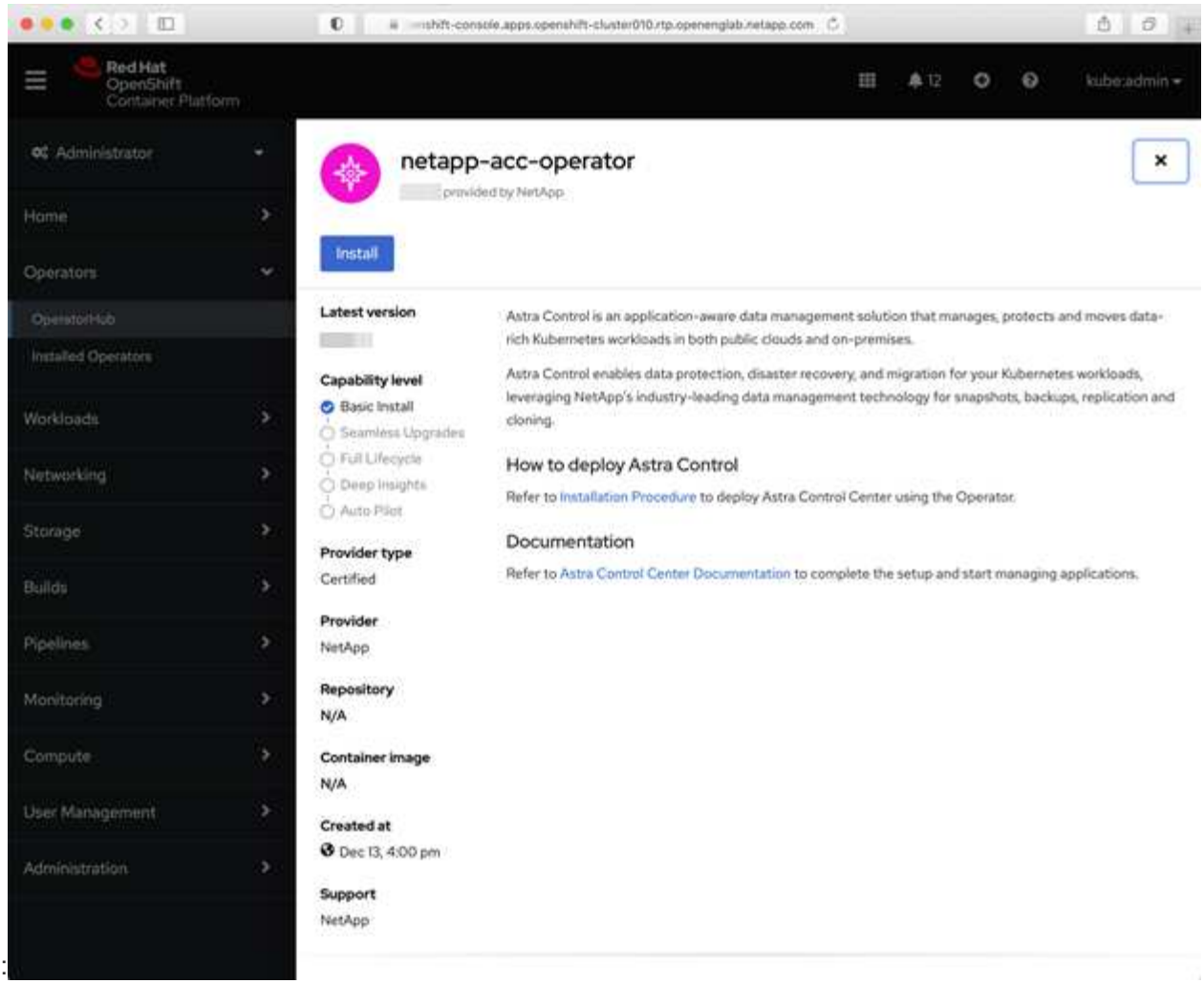
```
podman login [your_registry_path]
```

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

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

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

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

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

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

Recherchez la page d'installation de l'opérateur

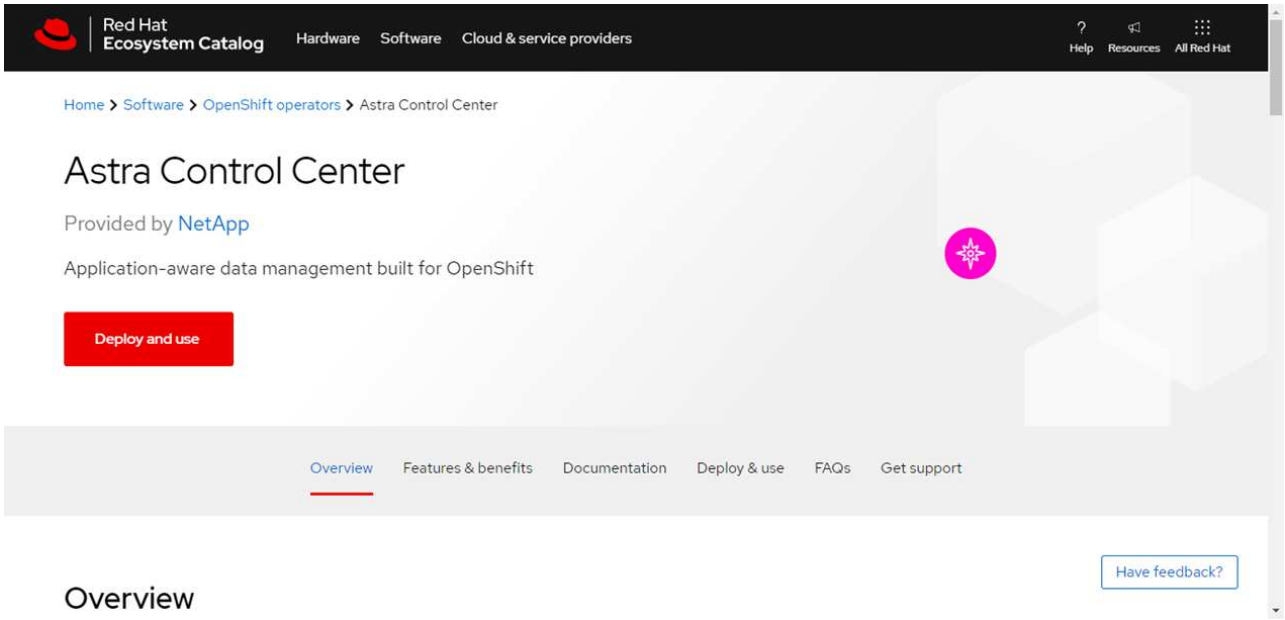
1. Effectuez l'une des procédures suivantes pour accéder à la page d'installation de l'opérateur :
   - Depuis la console Web Red Hat OpenShift



:

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

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



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

ii. Sélectionnez **déployer et utiliser**.

Poser l'opérateur

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

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

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

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

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

   c. Sélectionnez **installer**.

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

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

Poser le centre de contrôle Astra

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

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

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

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

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

3. Sélectionnez `Create`.

Et la suite

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

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

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

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

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

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

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

Déploiement d'Astra Control Center dans Amazon Web Services

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

Ce dont vous avez besoin pour AWS

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

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

Exigences de l'environnement opérationnel pour AWS

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

- Red Hat OpenShift Container Platform 4.8

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

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

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

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

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

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

Présentation du déploiement pour AWS

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

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

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

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

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

Voir "Identifiants AWS initiaux".

Installez un cluster Red Hat OpenShift sur AWS

Installez un cluster Red Hat OpenShift Container Platform sur AWS.

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

Configurez AWS

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

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

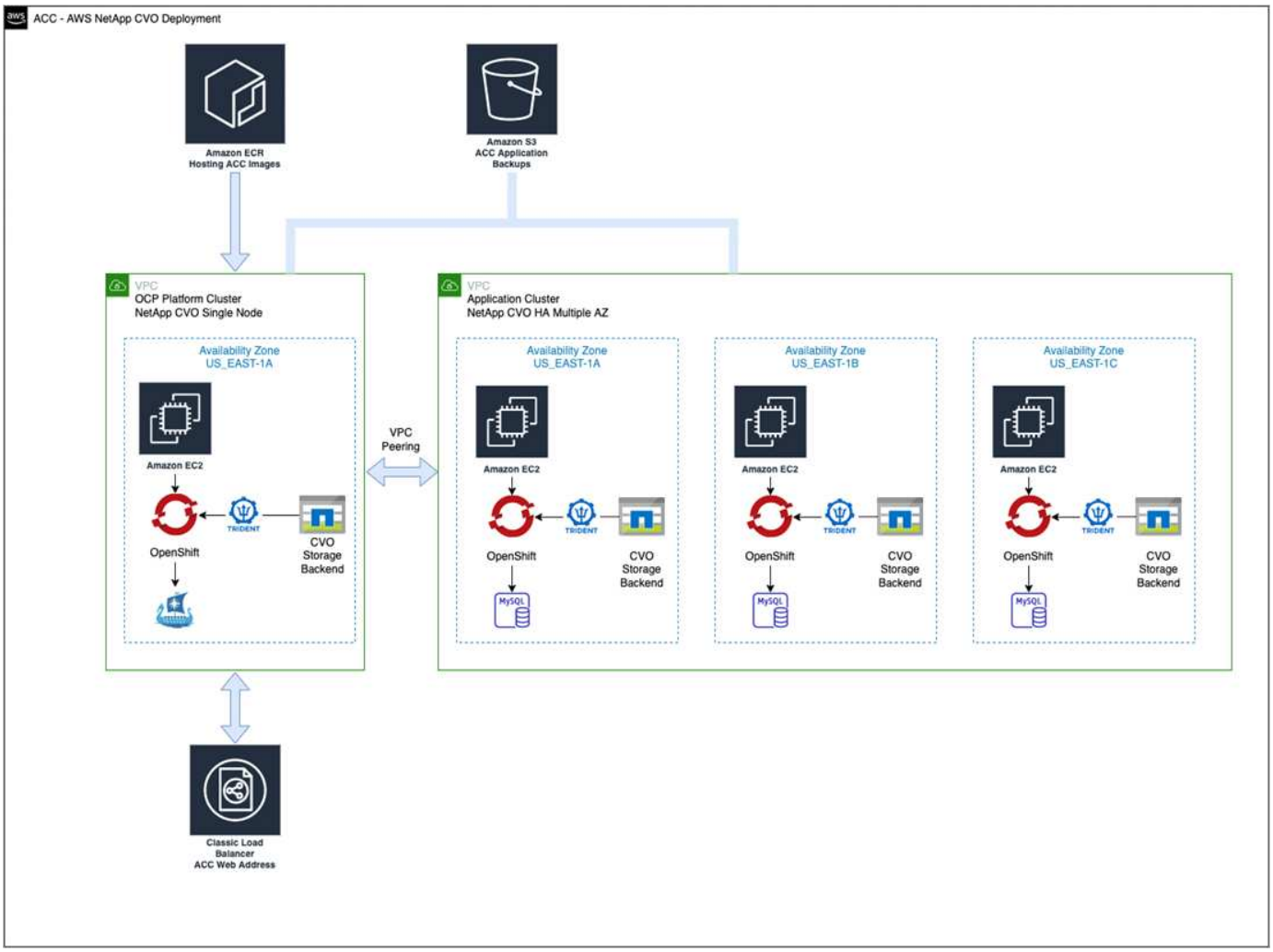
1. Créez un réseau virtuel AWS.
2. Vérifiez les instances de calcul EC2. Il peut s'agir d'un serveur bare Metal ou de machines virtuelles dans AWS.
3. Si le type d'instance ne correspond pas déjà aux exigences de ressources minimales Astra pour les nœuds maîtres et workers, modifiez le type d'instance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

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

6. Poussez les images ACC dans le registre défini.

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

Voici un exemple de déploiement AWS :



Configurez NetApp Cloud Manager

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

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

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

Étapes

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

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

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

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

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

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

Poser le centre de contrôle Astra

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

> AWS utilise le type de compartiment S3 générique.

Déployez Astra Control Center dans Google Cloud Platform

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

Éléments requis pour GCP

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

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

Exigences de l'environnement opérationnel pour GCP

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

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

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

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

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

Présentation du déploiement pour GCP

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

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

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

Installez un cluster Red Hat OpenShift sur GCP

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

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

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

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

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

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

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

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

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

Voir "Identifiants et autorisations GCP initiaux".

Configurez GCP

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

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

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

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

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

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

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

   Exemple :

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

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

8. Configurer les zones DNS.

Configurez NetApp Cloud Manager

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

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

Ce dont vous avez besoin

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

Étapes

1. Ajoutez vos identifiants à Cloud Manager. Voir "Ajout de comptes GCP".
2. Ajoutez un connecteur pour GCP.
   a. Choisissez GCP comme fournisseur.
   b. Entrez les identifiants GCP. Voir "Création d'un connecteur dans GCP à partir de Cloud Manager".
   c. S'assurer que le connecteur est en marche et basculer vers ce connecteur.

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

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

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

Poser le centre de contrôle Astra

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

> GCP utilise le type de compartiment S3 générique.

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

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

Déploiement d'Astra Control Center dans Microsoft Azure

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

Ce dont vous avez besoin pour Azure

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

- Licence Astra Control Center. Voir "Exigences de licence d'Astra Control Center".
- "Découvrez les exigences d'Astra Control Center".
- Compte NetApp Cloud Central
- Si vous utilisez OCP, Red Hat OpenShift Container Platform (OCP) 4.8

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

Exigences de l'environnement opérationnel pour Azure

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

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

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

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

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

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

Présentation du déploiement pour Azure

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

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

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

Installez un cluster Red Hat OpenShift sur Azure

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

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

Créez des groupes de ressources Azure

Créez au moins un groupe de ressources Azure.

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

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

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

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

Voir "Identifiants et autorisations Azure".

Configurez Azure

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

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

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

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

   Exemple :

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

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

10. Configurer les zones DNS.

Configurez NetApp Cloud Manager

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

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

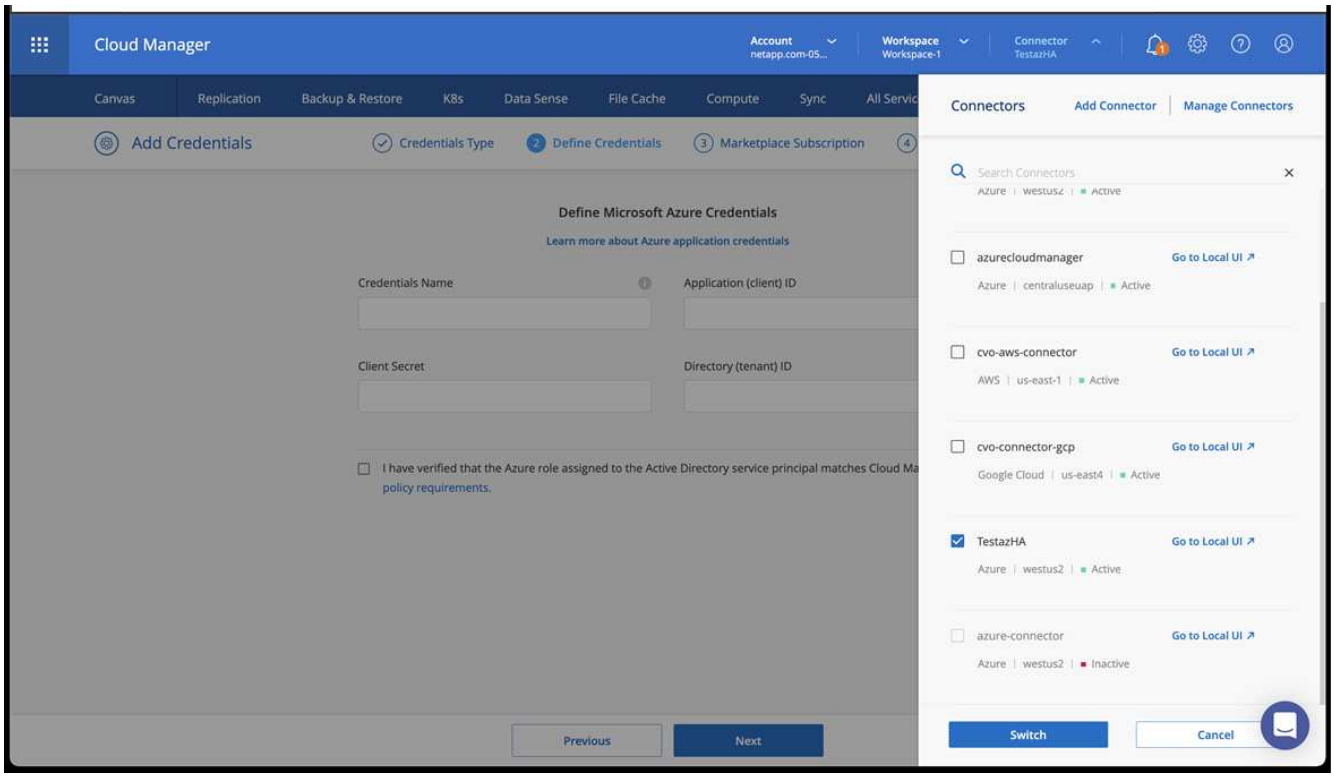
Ce dont vous avez besoin

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

Étapes

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

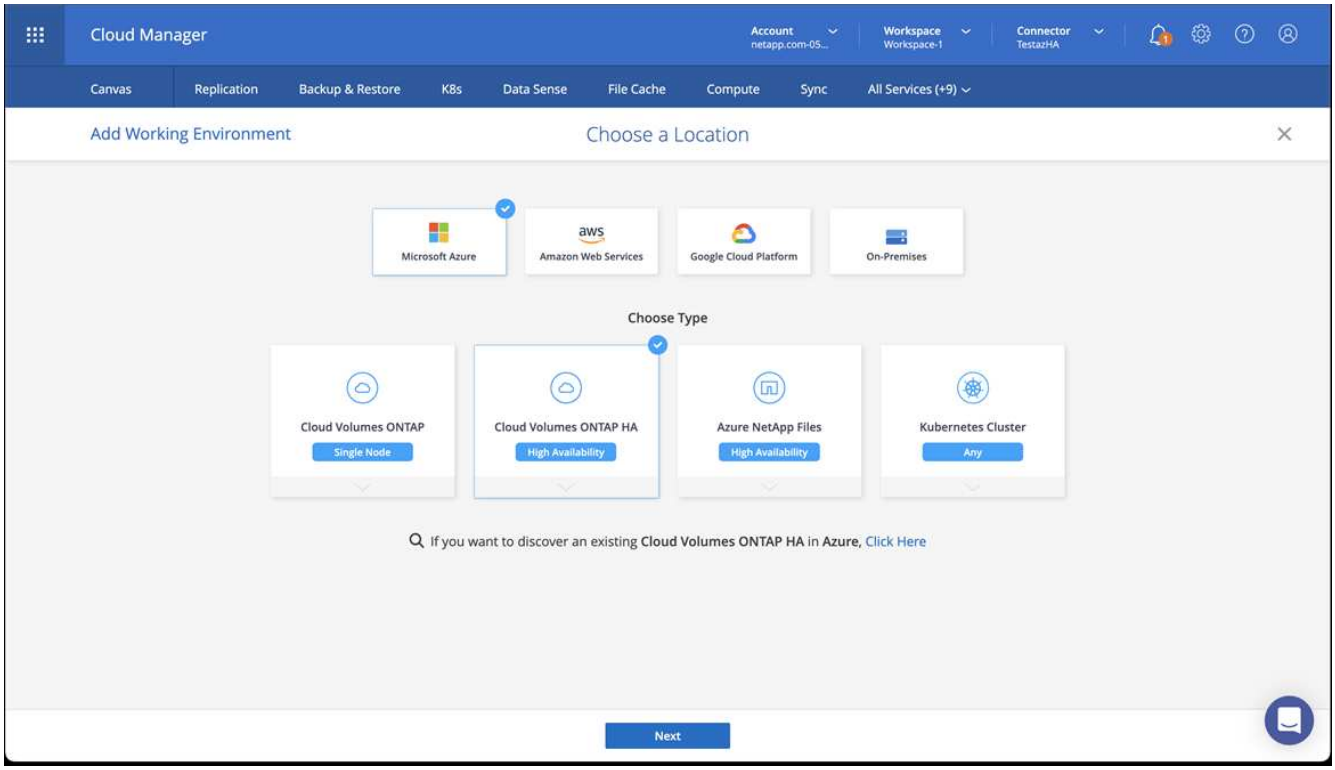
      Voir "Création d'un connecteur dans Azure à partir de Cloud Manager".
3. S'assurer que le connecteur est en marche et basculer vers ce connecteur.



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

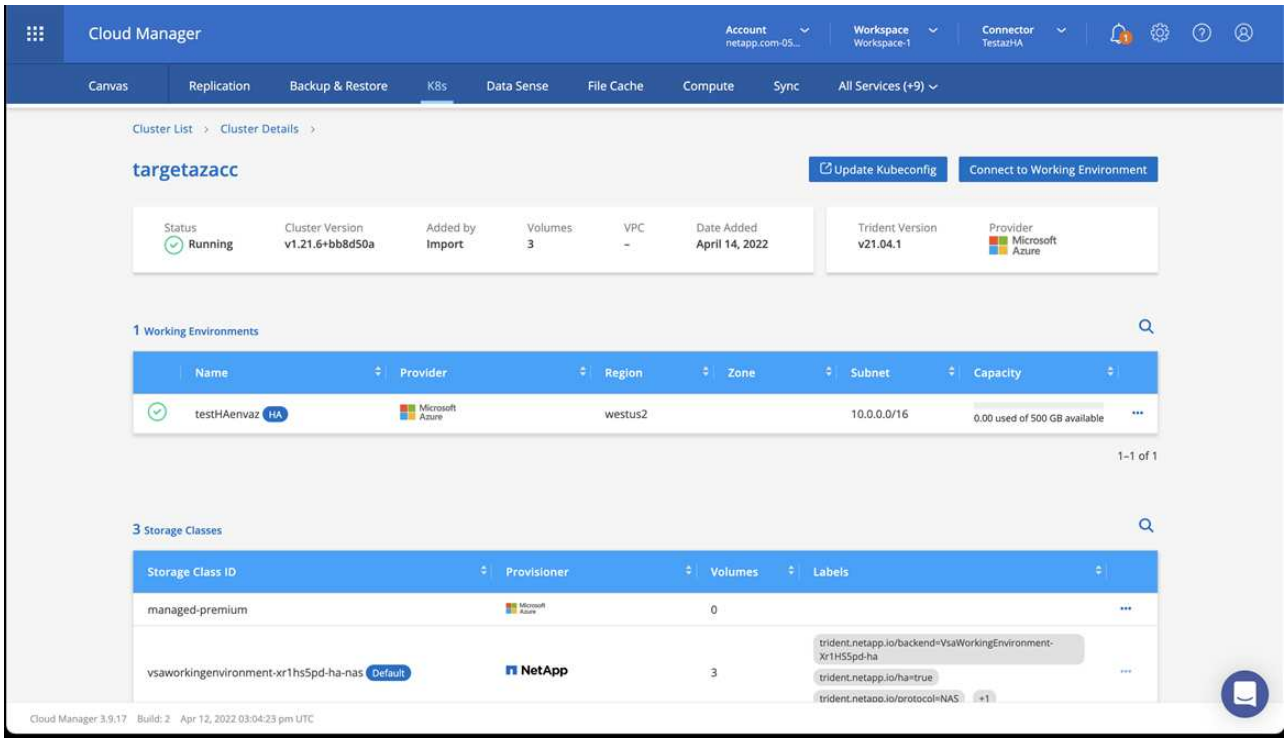
   a. Emplacement : « Microsoft Azure ».

   b. Type : « Cloud Volumes ONTAP HA ».



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

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



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

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

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

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

Installer et configurer le centre de contrôle Astra

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

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

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.

Pour plus d'informations sur le calcul des licences, reportez-vous à la section "Licences".

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

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.

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.

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. Avec Astra Data Store, vous pouvez ajouter le cluster d'applications Kubernetes qui contient des applications qui utilisent des volumes provisionnés par 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 conserve 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.
- Utilisez l'interface utilisateur Web Astra Control pour attribuer une classe de stockage par défaut différente. Voir Modifiez la classe de stockage par défaut.

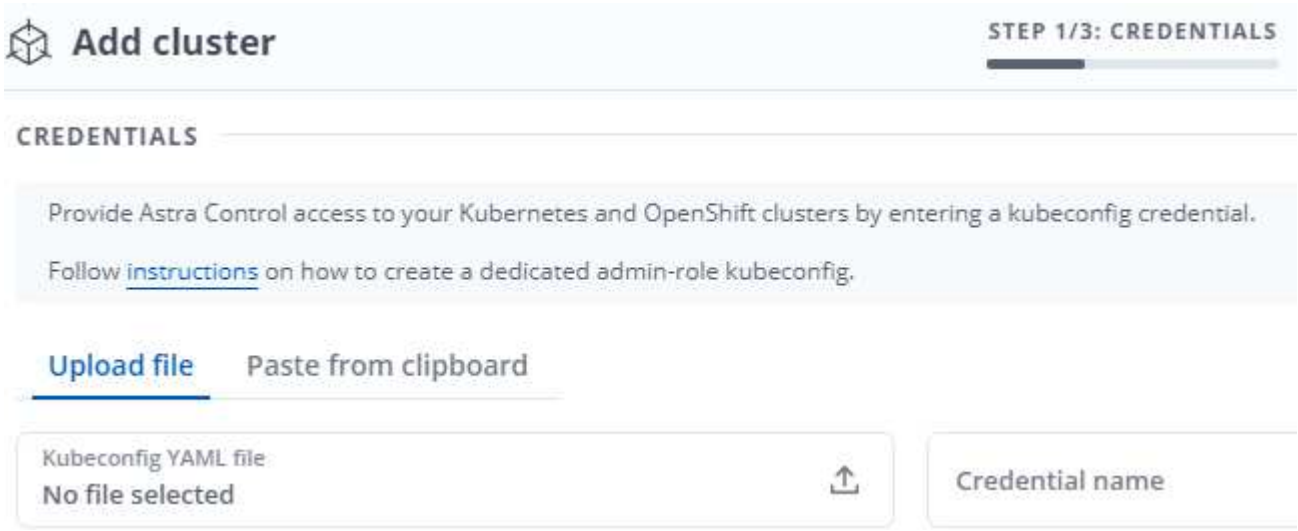
Ce dont vous avez besoin

- 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**.

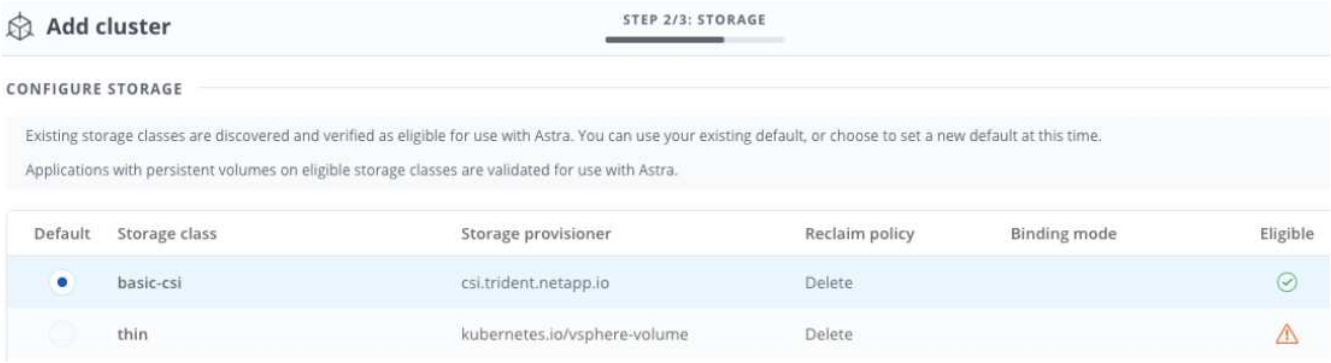


   Si vous créez la vôtre `kubeconfig` fichier, vous ne devez définir que **un** élément de contexte dans celui-ci. Voir "Documentation Kubernetes" pour plus d'informations sur la création `kubeconfig` fichiers.

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. Vous pouvez déployer un système back-end de stockage sur un cluster géré ou utiliser un système back-end existant.

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.

Il vous faudra pour déployer un data Store Astra

- Vous avez ajouté votre cluster d'applications Kubernetes et le cluster de calcul sous-jacent.

> Lorsque vous ajoutez votre cluster d'applications Kubernetes pour Astra Data Store et qu'il est géré par Astra Control, 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.

Il vous faudra de nouveaux déploiements de data Store Astra

- Vous avez "a chargé la version du pack d'installation que vous envisagez de déployer" À un endroit accessible à Astra Control.
- Vous avez ajouté le cluster Kubernetes que vous souhaitez utiliser pour le déploiement.
- Vous avez téléchargé le Licence Astra Data Store Pour votre déploiement vers un emplacement accessible à Astra Control.

Options

- Déploiement des ressources de stockage
- Utiliser un système back-end existant

Déploiement des ressources de stockage

Vous pouvez déployer un nouveau magasin de données Astra et gérer le stockage back-end associé.

Étapes

1. Naviguer dans le tableau de bord ou le menu Backends :
   - Dans **Dashboard** : dans le Résumé des ressources, sélectionnez un lien dans le volet stockage arrière-plans et sélectionnez **Ajouter** dans la section Backends.
   - À partir de **Backends** :
     i. Dans la zone de navigation de gauche, sélectionnez **Backends**.
     ii. Sélectionnez **Ajouter**.
2. Sélectionnez l'option de déploiement **Astra Data Store** dans l'onglet **Deploy**.
3. Sélectionnez le package de magasin de données Astra à déployer :
   a. Entrez un nom pour l'application de magasin de données Astra.
   b. Choisissez la version d'Astra que vous voulez déployer.

   > Si vous n'avez pas encore téléchargé la version que vous avez l'intention de déployer, vous pouvez utiliser l'option **Ajouter un paquet** ou quitter l'assistant et utiliser "gestion des packages" pour télécharger le pack d'installation.

4. Sélectionnez une licence Astra Data Store que vous avez déjà téléchargée ou utilisez l'option **Ajouter une licence** pour télécharger une licence à utiliser avec l'application.

> Les licences Astra Data Store avec autorisation complète sont associées à votre cluster Kubernetes, et ces clusters associés doivent apparaître automatiquement. S'il n'y a pas de cluster géré, vous pouvez sélectionner l'option **Ajouter un cluster** pour en ajouter un à Astra Control Management. Pour les licences Astra Data Store, si aucune association n'a été établie entre la licence et le cluster, vous pouvez définir cette association à la page suivante de l'assistant.

5. Si vous n'avez pas ajouté de cluster Kubernetes à la gestion Astra Control, vous devez le faire depuis la page **cluster Kubernetes**. Sélectionnez un cluster existant dans la liste ou sélectionnez **Ajouter le cluster sous-jacent** pour ajouter un cluster à la gestion Astra Control.
6. Sélectionnez une taille de modèle pour le cluster Kubernetes qui fournira les ressources pour le magasin de données Astra. Vous pouvez choisir l'une des options suivantes :
   - Si vous le souhaitez `Recommended Kubernetes worker node requirements`, sélectionnez un modèle de grande à petite en fonction de ce que votre licence autorise.
   - Si vous le souhaitez `Custom Kubernetes worker node requirements`, sélectionnez le nombre de cœurs et la mémoire totale que vous souhaitez pour chaque nœud de cluster. Vous pouvez également afficher le nombre de nœuds éligibles qui répondent à vos critères de sélection pour les cœurs et la mémoire.

> Lorsque vous choisissez un modèle, sélectionnez des nœuds de grande taille avec plus de mémoire et de cœurs pour des charges de travail plus importantes ou un nombre plus important de nœuds pour des charges de travail plus petites. Vous devez sélectionner un modèle en fonction de ce que votre licence autorise. Chaque option de modèle recommandée indique le nombre de nœuds éligibles qui répondent au modèle de modèle pour la mémoire, les cœurs et la capacité de chaque nœud.

7. Configurez les nœuds :
   a. Ajoutez une étiquette de nœud pour identifier le pool de nœuds de travail qui prend en charge ce cluster de magasin de données Astra.

> L'étiquette doit être ajoutée à chaque nœud du cluster qui sera utilisé pour le déploiement du magasin de données Astra avant le début du déploiement, sinon le déploiement échouera.

   b. Configurez la capacité (Gio) par nœud manuellement ou sélectionnez la capacité maximale de nœud autorisée.
   c. Configurez un nombre maximum de nœuds autorisés dans le cluster ou autorisez le nombre maximum de nœuds sur le cluster.
8. (Licences complètes de l'Astra Data Store uniquement) Entrez la clé de l'étiquette que vous souhaitez utiliser pour les domaines de protection.

> Créez au moins trois étiquettes uniques pour la clé pour chaque nœud. Par exemple, si votre clé est `astra.datastore.protection.domain`, vous pouvez créer les étiquettes suivantes : `astra.datastore.protection.domain=domain1` ,`astra.datastore.protection.domain=domain2`, et `astra.datastore.protection.domain=domain3`.

9. Configurez le réseau de gestion :

a. Saisissez une adresse IP de gestion pour la gestion interne du magasin de données Astra qui se trouve sur le même sous-réseau que les adresses IP du nœud de travail.
b. Choisissez d'utiliser la même carte réseau à la fois pour les réseaux de gestion et de données ou de les configurer séparément.
c. Entrez le pool d'adresses IP du réseau de données, le masque de sous-réseau et la passerelle pour l'accès au stockage.

10. Vérifiez la configuration et sélectionnez **Deploy** pour commencer l'installation.

Résultat

Après une installation réussie, le système back-end apparaît dans `available` état dans la liste des systèmes back-end avec des informations de performance actives.

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

Utiliser un système back-end existant

Vous pouvez intégrer un système back-end de stockage ONTAP ou Astra dans la gestion du centre de contrôle d'Astra.

Étapes

1. Naviguer dans le tableau de bord ou le menu Backends :
   - Dans **Dashboard** : dans le Résumé des ressources, sélectionnez un lien dans le volet stockage arrière-plans et sélectionnez **Ajouter** dans la section Backends.
   - À partir de **Backends** :
     i. Dans la zone de navigation de gauche, sélectionnez **Backends**.
     ii. Sélectionnez **Manage** sur un back-end découvert à partir du cluster géré ou sélectionnez **Add** pour gérer un back-end existant supplémentaire.
2. Sélectionnez l'onglet **utiliser existant**.
3. Effectuez l'une des opérations suivantes en fonction de votre type de système back-end :
   - **Magasin de données Astra**:
     i. Sélectionnez **Astra Data Store**.
     ii. Sélectionnez le cluster de calcul géré et sélectionnez **Suivant**.
     iii. Confirmez les détails du back-end et sélectionnez **Ajouter le back-end de stockage**.
   - **ONTAP** :
     i. Sélectionnez **ONTAP** et sélectionnez **Suivant**.
     ii. Saisissez l'adresse IP de gestion du cluster ONTAP et les identifiants d'administrateur.

     > L'utilisateur dont vous saisissez ici les informations d'identification doit disposer du `ontapi` Méthode d'accès de connexion utilisateur activée dans ONTAP System Manager sur le cluster ONTAP. Si vous prévoyez d'utiliser la réplication SnapMirror, activez les méthodes d'accès `ontapi` et `http` Pour l'utilisateur sur les deux clusters ONTAP. Voir "Gérer les comptes d'utilisateurs" pour en savoir plus.

     iii. Sélectionnez **Revue**.
     iv. Confirmez les détails du back-end et sélectionnez **Ajouter le back-end de stockage**.

Résultat

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

> Amazon Web Services (AWS) et Google Cloud Platform (GCP) utilisent le type de compartiment S3 générique.

- Microsoft Azure

> 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.

- Microsoft Azure

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.

Modifiez la classe de stockage par défaut

Vous pouvez modifier la classe de stockage par défaut d'un cluster.

Étapes

1. Dans l'interface utilisateur Web Astra Control Center, sélectionnez **clusters**.
2. Sur la page **clusters**, sélectionnez le cluster que vous souhaitez modifier.
3. Sélectionnez l'onglet **stockage**.
4. Sélectionnez la catégorie **classes de stockage**.
5. Sélectionnez le menu **actions** pour la classe de stockage que vous souhaitez définir par défaut.
6. Sélectionnez **définir comme valeur par défaut**.

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

Assurez-vous que votre cluster répond aux exigences décrites dans "Configuration requise en cluster des applications".

> 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.

- Les classes de stockage Astra Trident sont configurées avec un "système back-end pris en charge" (requis pour tout type de cluster)
- 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>
```

+

```yaml
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. (Facultatif) si votre cluster utilise une politique de sécurité de pod restrictive qui ne permet pas la création de pod privilégié ou l'exécution des processus dans les conteneurs de pod en tant qu'utilisateur racine, créez une politique de sécurité de pod personnalisée pour le cluster qui permet à Astra Control de créer et de gérer des pods. Pour obtenir des instructions, reportez-vous à la section "Créez une stratégie de sécurité de pod personnalisée".
3. 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>
```

+

```yaml
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
```

4. 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".

5. 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
```

6. (**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.

Créez une stratégie de sécurité de pod personnalisée

Astra Control doit créer et gérer des pods Kubernetes sur les clusters qu'il gère. Si votre cluster utilise une politique de sécurité de pod restrictive qui ne permet pas la création de pod privilégié ou l'exécution des processus dans les conteneurs de pod en tant qu'utilisateur racine, vous devez créer une stratégie de sécurité de pod moins restrictive pour permettre à Astra Control de créer et de gérer ces pods.

Étapes

1. Créez une politique de sécurité du pod pour le cluster qui est moins restrictive par défaut et enregistrez-la dans un fichier. Par exemple :

```yaml
apiVersion: policy/v1beta1
kind: PodSecurityPolicy
metadata:
  name: astracontrol
  annotations:
    seccomp.security.alpha.kubernetes.io/allowedProfileNames: '*'
spec:
  privileged: true
  allowPrivilegeEscalation: true
  allowedCapabilities:
  - '*'
  volumes:
  - '*'
  hostNetwork: true
  hostPorts:
  - min: 0
    max: 65535
  hostIPC: true
  hostPID: true
  runAsUser:
    rule: 'RunAsAny'
  seLinux:
    rule: 'RunAsAny'
  supplementalGroups:
    rule: 'RunAsAny'
  fsGroup:
    rule: 'RunAsAny'
```

2. Créez un nouveau rôle pour la stratégie de sécurité du pod.

```bash
kubectl-admin create role psp:astracontrol \
    --verb=use \
    --resource=podsecuritypolicy \
    --resource-name=astracontrol
```

3. Lier le nouveau rôle au compte de service.

```bash
kubectl-admin create rolebinding default:psp:astracontrol \
    --role=psp:astracontrol \
    --serviceaccount=astracontrol-service-account:default
```

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.

Licences

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 est-il automatiquement désinstallé d'un cluster lorsque je le dégère ? lorsque vous dégérez un cluster depuis Astra Control Center, Trident n'est pas automatiquement désinstallé du cluster. Pour désinstaller Trident, vous devez procéder comme ça "Suivez ces étapes dans la documentation Trident".

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 pods malsains affectent-ils la gestion des applications? si une application gérée possède des pods dans un état malsain, Astra Control ne peut pas créer de nouvelles sauvegardes et de nouveaux clones.

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.