



Commencez

Astra Control Center

NetApp
June 07, 2024

Sommaire

- Commencez 1
 - Exigences du centre de contrôle Astra 1
 - Démarrage rapide pour Astra Control Center 5
 - Présentation de l'installation 7
 - Configurer le centre de contrôle Astra 62
 - Foire aux questions pour Astra Control Center 76

Commencez

= :allow-uri-read:

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](#)
- [Accès à Internet](#)
- [Licence](#)
- [Entrée pour les clusters Kubernetes sur site](#)
- [Configuration réseau requise](#)
- [Navigateurs Web pris en charge](#)
- [Exigences supplémentaires relatives aux clusters d'applications](#)
- [Exigences des clusters Google Anthos](#)
- [Configuration requise pour le cluster VMware Tanzu Kubernetes Grid](#)

De l'environnement opérationnel

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

- Cisco IKS avec Kubernetes 1.22
- Google Anthos 1.11 ou 1.12 (voir [Exigences des clusters Google Anthos](#))
- Rancher Kubernetes Engine (RKE) :
 - RKE 1.3.12 avec Rancher 2.6.5 et 2.6.6
 - RKE 1.3.13 avec Rancher 2.6.8
 - RKE 2 (v1.23,6+rke2r1) avec Rancher 2.6.5 et 2.6.6
 - RKE 2 (v1.24.x) avec Rancher 2.6.8
- Red Hat OpenShift Container Platform 4.8 à 4.11
- Kubernetes 1.23 vers 1.25 en amont (Astra Trident 22.10 ou version ultérieure requise pour Kubernetes 1.25)
- VMware Tanzu Kubernetes Grid : (voir [Configuration requise pour le cluster VMware Tanzu Kubernetes Grid](#))
 - VMware Tanzu Kubernetes Grid 1.5
 - VMware Tanzu Kubernetes Grid Integrated Edition 1.13 et 1.14

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
Extensions de processeur	Les processeurs de tous les nœuds de l'environnement d'hébergement doivent avoir des extensions AVX activées.
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
Astra Trident	Astra Trident 22.01 ou version ultérieure installé et configuré avec Astra Trident 22.07 ou plus récent installé pour la réplication d'applications basée sur SnapMirror Astra Trident 22.10 ou version ultérieure installée pour les clusters Kubernetes 1.25 (vous devez effectuer une mise à niveau vers Astra Trident 22.10 avant la mise à niveau vers Kubernetes 1.25).



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 :**
 - Vous devez configurer au moins une classe de stockage Astra Trident sur le cluster. Si une classe de stockage par défaut est configurée, assurez-vous qu'elle est la seule classe de stockage avec la désignation par défaut.
 - Assurez-vous que les nœuds workers de votre cluster sont configurés avec les pilotes de stockage appropriés afin que les pods puissent interagir avec le système de stockage back-end. Le centre de contrôle Astra prend en charge les pilotes ONTAP suivants fournis par Astra Trident :
 - ontap-nas
 - ontap-san
 - ontap-san-économie (non pris en charge pour la réplication d'applications)

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 version ultérieure des systèmes AFF, FAS et ASA
- Systèmes NetApp ONTAP 9.8 ou plus récents AFF, FAS et ASA pour la réplication d'applications basée sur SnapMirror
- NetApp ONTAP Select 9.5 ou version ultérieure
- NetApp ONTAP Select 9.8 ou version ultérieure pour la réplication d'applications basée sur SnapMirror
- NetApp Cloud Volumes ONTAP 9.5 ou version ultérieure

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

Pour vérifier si votre système ONTAP dispose des licences requises, reportez-vous à la section "[Gérer les licences ONTAP](#)".

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 configurer la licence, reportez-vous à la section "[utilisez une licence d'essai gratuite de 90 jours](#)".

Pour en savoir plus sur le fonctionnement des licences, reportez-vous à la section "[Licences](#)".

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](#)".

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'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.



L'équilibreur de charge doit utiliser une adresse IP située dans le même sous-réseau que les adresses IP du nœud de travail de l'Astra Control Center.



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.

Exigences supplémentaires relatives aux clusters d'applications

Gardez à l'esprit ces exigences si vous prévoyez d'utiliser ces caractéristiques du centre de contrôle Astra :

- **Configuration requise pour le cluster d'applications** : ["Exigences de gestion du cluster"](#)
 - * Exigences des applications gérées* : ["De gestion des applications"](#)
 - **Exigences supplémentaires pour la réplication d'applications** : ["Conditions préalables à la réplication"](#)

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.

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.
- 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"](#)

Et la suite

Afficher le ["démarrage rapide"](#) présentation.

Démarrage rapide pour Astra Control Center

Voici un aperçu 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.



Vérifiez la configuration des clusters Kubernetes

Assurez-vous que votre environnement répond à ces exigences.

Cluster Kubernetes

- "Assurez-vous que votre environnement répond aux exigences de votre environnement opérationnel"
- "Configuration de la détection d'entrée pour l'équilibrage de la charge sur les clusters Kubernetes sur site"

Intégration du stockage

- "Vérifiez que votre environnement inclut la version prise en charge d'Astra Trident"
- "Préparez les nœuds worker"
- "Configurer le système back-end de stockage Astra Trident"
- "Configurez des classes de stockage Astra Trident"
- "Installation du contrôleur de snapshot de volume Astra Trident"
- "Créer une classe de snapshot de volume"

Informations d'identification ONTAP

- "Configurez les identifiants ONTAP"

2

Téléchargez et installez Astra Control Center

Effectuez ces tâches d'installation.

- "Téléchargez Astra Control Center à partir de la page de téléchargement du site de support NetApp"
- Obtenez le fichier de licence NetApp :
 - "Si vous évaluez Astra Control Center, téléchargez le fichier de licence d'évaluation"
 - "Si vous avez déjà acheté Astra Control Center, générez votre fichier de licence"
- "Poser le centre de contrôle Astra"
- "Effectuez d'autres étapes de configuration facultatives"

3

Effectuez certaines tâches de configuration initiales

Réalisez certaines tâches de base pour démarrer.

- "Ajouter une licence"
- "Préparez votre environnement à la gestion du cluster"
- "Ajouter un cluster"
- "Ajout d'un système back-end"
- "Ajouter un godet"

4

Utilisez Astra Control Center

Après avoir terminé la configuration du centre de contrôle Astra, voici ce que vous pourriez faire ensuite. Vous pouvez utiliser l'interface utilisateur Astra Control (UI) ou le "[API de contrôle Astra](#)".

- ["Gérer des applications"](#)
- ["Protégez vos applications"](#): Configurer des stratégies de protection et répliquer, cloner et migrer des applications.
- ["Gestion des comptes"](#): Utilisateurs, rôles, LDAP, informations d'identification, etc
- ["Vous pouvez également vous connecter à Cloud Insights"](#): Permet d'afficher des mesures sur l'état de santé de votre système.

Pour en savoir plus

- ["API de contrôle Astra"](#)
- ["Mettez à niveau Astra Control Center"](#)
- ["Aidez-vous d'Astra Control"](#)

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"](#)

Selon votre environnement, une configuration supplémentaire peut être nécessaire après l'installation d'Astra Control Center :

- ["Configurer le centre de contrôle Astra après l'installation"](#)

Installer le centre de contrôle Astra en suivant la procédure standard

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

Autres procédures d'installation

- **Installer avec RedHat OpenShift OperatorHub**: Utilisez ceci ["autre procédure"](#) Pour installer Astra Control Center sur OpenShift à l'aide d'OperatorHub.
- **Installer dans le Cloud public avec Cloud Volumes ONTAP backend**: Utilisez ["ces procédures"](#) Pour installer Astra Control Center dans Amazon Web Services (AWS), Google Cloud Platform (GCP) ou Microsoft Azure avec un système de stockage principal Cloud Volumes ONTAP.

Pour une démonstration du processus d'installation d'Astra Control Center, reportez-vous à la section ["vidéo"](#).

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](#)".

- 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 ne tente pas d'installer son propre gestionnaire de certificat. Par défaut, Astra Control Center installe son propre gestionnaire de certificats lors de l'installation.

Description de la tâche

Le processus d'installation d'Astra Control Center vous aide à :

- Poser les composants Astra dans le `netapp-acc` (ou espace de nom personnalisé).
- Créez un compte d'administrateur propriétaire Astra Control par défaut.
- Définissez une adresse e-mail d'utilisateur administratif et un mot de passe de configuration initiale par défaut. Ce rôle de propriétaire est attribué à cet utilisateur pour la première connexion à l'interface utilisateur.
- Vérifiez que toutes les POD Astra Control Center sont en cours d'exécution.
- Installez l'interface utilisateur du centre de contrôle Astra.



Ne supprimez pas l'opérateur du centre de contrôle Astra (par exemple, `kubectl delete -f astra_control_center_operator_deploy.yaml`) À tout moment pendant l'installation ou le fonctionnement d'Astra Control Center pour éviter de supprimer les modules.

Étapes

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

- [Téléchargez et extrayez 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 extrayez Astra Control Center

1. Accédez au "[Page de téléchargement de l'évaluation Astra Control Center](#)" Sur le site de support NetApp.
2. Téléchargez le pack contenant Astra Control Center (`astra-control-center-[version].tar.gz`).
3. (Recommandé mais facultatif) Téléchargez le lot de certificats et de signatures pour Astra Control Center (`astra-control-center-certs-[version].tar.gz`) pour vérifier la signature du paquet :

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

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

La sortie s'affiche `Verified OK` une fois la vérification terminée.

4. Extraire les images du pack Astra Control Center :

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

Installez le plug-in NetApp Astra kubectl

Le plug-in de ligne de commande NetApp Astra kubectl 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 de plug-ins pour différentes architectures CPU et systèmes d'exploitation. Avant d'effectuer cette tâche, vous devez savoir quelle unité centrale et quel système d'exploitation vous possédez.

Étapes

1. Répertoriez les binaires NetApp Astra kubectl disponibles et notez le nom du fichier dont vous avez besoin pour votre système d'exploitation et votre architecture de processeur :



La bibliothèque de plug-ins kubectl fait partie du bundle tar et est extraite dans le dossier `kubectl-astra`.

```
ls kubectl-astra/
```

2. Déplacez le bon binaire dans le chemin actuel et renommez-le `kubectl-astra`:

```
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. Accédez au répertoire racine du tarball. Vous devriez voir ce fichier et ce répertoire:

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

2. Envoyez les images du package dans le répertoire d'images Astra Control Center vers votre registre local. Effectuez les remplacements suivants avant d'exécuter le `push-images` commande :

- Remplacez `<BUNDLE_FILE>` par le nom du fichier bundle Astra Control (`acc.manifest.bundle.yaml`).
- Remplacer `<MY_FULL_REGISTRY_PATH>` par l'URL du référentiel Docker, par exemple `<a href="https://<docker-registry>" class="bare">https://<docker-registry>"`.
- Remplacez `<MY_REGISTRY_USER>` par le nom d'utilisateur.
- Remplacez `<MY_REGISTRY_TOKEN>` par un jeton autorisé pour le registre.

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

Podman

1. Accédez au répertoire racine du tarball. Vous devriez voir ce fichier et ce répertoire:

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

2. Connectez-vous à votre registre :

```
podman login <YOUR_REGISTRY>
```

3. Préparez et exécutez l'un des scripts suivants qui est personnalisé pour la version de Podman que vous utilisez. Remplacez `<MY_FULL_REGISTRY_PATH>` par l'URL de votre référentiel qui inclut tous les sous-répertoires.

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

```

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

```

Podman 3

```

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

```



Le chemin d'accès à l'image que le script crée doit ressembler aux éléments suivants, selon la configuration de votre registre : <https://netappdownloads.jfrog.io/docker-astra-control-prod/netapp/astra/acc/22.11.0-82/image:version>

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]
```



Avant de terminer l'installation, assurez-vous que votre KUBECONFIG pointe vers le groupe d'instruments où vous souhaitez installer le centre de contrôle Astra. Le KUBECONFIG ne peut contenir qu'un seul contexte.

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/22.11.0-82`).

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

Exemple de réponse :

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



Si vous supprimez l'espace de noms après la génération du secret, recréez l'espace de noms, puis régénérez le secret pour 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
```

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](#).

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

```
apiVersion: apps/v1
kind: Deployment
metadata:
```

```

labels:
  control-plane: controller-manager
name: acc-operator-controller-manager
namespace: netapp-acc-operator
spec:
  replicas: 1
  selector:
    matchLabels:
      control-plane: controller-manager
  strategy:
    type: Recreate
  template:
    metadata:
      labels:
        control-plane: controller-manager
    spec:
      containers:
        - args:
            - --secure-listen-address=0.0.0.0:8443
            - --upstream=http://127.0.0.1:8080/
            - --logtostderr=true
            - --v=10
          image: [your_registry_path]/kube-rbac-proxy:v4.8.0
          name: kube-rbac-proxy
          ports:
            - containerPort: 8443
              name: https
        - args:
            - --health-probe-bind-address=:8081
            - --metrics-bind-address=127.0.0.1:8080
            - --leader-elect
          env:
            - name: ACCOP_LOG_LEVEL
              value: "2"
            - name: ACCOP_HELM_INSTALLTIMEOUT
              value: 5m
          image: [your_registry_path]/acc-operator:[version x.y.z]
          imagePullPolicy: IfNotPresent
          livenessProbe:
            httpGet:
              path: /healthz
              port: 8081
              initialDelaySeconds: 15
              periodSeconds: 20
          name: manager
          readinessProbe:

```

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

3. 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.yaml`) pour créer des comptes, un support, un registre et d'autres configurations nécessaires :

```
vim astra_control_center.yaml
```



Un échantillon annoté YAML suit ces étapes.

2. Modifiez ou confirmez les paramètres suivants :

`<code>accountName</code>`

Réglage	Guidage	Type	Exemple
<code>accountName</code>	Modifiez le <code>accountName</code> Chaîne du nom que vous souhaitez associer au compte Astra Control Center. Il ne peut y avoir qu'un seul nom de compte.	chaîne	Exemple

`<code>astraVersion</code>`

Réglage	Guidage	Type	Exemple
<code>astraVersion</code>	La version d'Astra Control Center à déployer. Aucune action n'est nécessaire pour ce paramètre car la valeur sera pré-remplie.	chaîne	22.11.0-82

`<code>astraAddress</code>`

Réglage	Guidage	Type	Exemple
<code>astraAddress</code>	<p>Modifiez le <code>astraAddress</code> Chaîne sur le FQDN (recommandé) ou l'adresse IP que vous souhaitez utiliser dans votre navigateur pour accéder à Astra Control Center. Cette adresse définit la façon dont Astra Control Center se trouve dans votre centre de données et est le même FQDN ou l'adresse IP que vous avez fournie à partir de votre équilibreur de charge une fois que vous avez terminé "Exigences du centre de contrôle Astra". REMARQUE : ne pas utiliser <code>http://</code> ou <code>https://</code> dans l'adresse. Copier ce FQDN pour l'utiliser dans un plus tard.</p>	chaîne	<code>astra.example.com</code>

<code>autoSupport</code>

Vos sélections dans cette section déterminent si vous allez participer à l'application de support proactif de NetApp, à NetApp Active IQ et à l'endroit où les données seront envoyées. Une connexion Internet est requise (port 442) et toutes les données de support sont anonymisées.

Réglage	Utiliser	Guidage	Type	Exemple
<code>autoSupport.enrolled</code>	Soit <code>enrolled</code> ou <code>url</code> les champs doivent être sélectionnés	Changer <code>enrolled</code> Pour AutoSupport à <code>false</code> pour les sites sans connexion internet ou sans conservation <code>true</code> pour les sites connectés. Un réglage de <code>true</code> Les données anonymes peuvent être envoyées à NetApp pour bénéficier d'un support. La sélection par défaut est <code>false</code> Aucune donnée de support n'est envoyée à NetApp.	Booléen	<code>false</code> (cette valeur est la valeur par défaut)
<code>autoSupport.url</code>	Soit <code>enrolled</code> ou <code>url</code> les champs doivent être sélectionnés	Cette URL détermine l'emplacement d'envoi des données anonymes.	chaîne	https://support.netapp.com/asupprod/post/1.0/postAsup

<code>email</code>

Réglage	Guidage	Type	Exemple
email	Modifiez le email chaîne à l'adresse d'administrateur initiale par défaut. Copiez cette adresse e-mail pour l'utiliser dans un plus tard . Cette adresse e-mail sera utilisée comme nom d'utilisateur du compte initial pour se connecter à l'interface utilisateur et sera informée des événements dans Astra Control.	chaîne	admin@example.com

<code>firstName</code>

Réglage	Guidage	Type	Exemple
firstName	Prénom de l'administrateur initial par défaut associé au compte Astra. Le nom utilisé ici sera visible dans un en-tête de l'interface utilisateur après votre première connexion.	chaîne	SRE

<code>LastName</code>

Réglage	Guidage	Type	Exemple
lastName	Nom de l'administrateur initial par défaut associé au compte Astra. Le nom utilisé ici sera visible dans un en-tête de l'interface utilisateur après votre première connexion.	chaîne	Admin

<code>imageRegistry</code>

Vos sélections dans cette section définissent le registre d'images du conteneur qui héberge les images d'application Astra, l'opérateur du centre de contrôle Astra et le référentiel Helm d'Astra Control Center.

Réglage	Utiliser	Guidage	Type	Exemple
<code>imageRegistry.name</code>	Obligatoire	Nom du registre d'images dans lequel vous avez poussé les images dans le étape précédente . Ne pas utiliser <code>http://</code> ou <code>https://</code> dans le nom du registre.	chaîne	<code>example.registry.com/astra</code>
<code>imageRegistry.secret</code>	Obligatoire si la chaîne que vous avez entrée pour <code>imageRegistry.name</code> requires a secret. IMPORTANT: If you are using a registry that does not require authorization, you must delete this <code>secret</code> ligne comprise entre <code>imageRegistry</code> sinon, l'installation échouera.	Nom du secret Kubernetes utilisé pour s'authentifier auprès du registre d'images.	chaîne	<code>astra-registry-cred</code>

`<code>storageClass</code>`

Réglage	Guidage	Type	Exemple
storageClass	<p>Modifiez le <code>storageClass</code> valeur à partir de <code>ontap-gold</code> Sur une autre ressource de stockage Trident, tel que requis par votre installation. Lancer la commande <code>kubectl get sc</code> pour déterminer vos classes de stockage configurées existantes. L'une des classes de stockage basées sur Trident doit être saisie dans le fichier manifeste (<code>astra-control-center-<version>.manifest</code>) Et sera utilisé pour ASTRA PVS. Si elle n'est pas définie, la classe de stockage par défaut sera utilisée. REMARQUE : si une classe de stockage par défaut est configurée, assurez-vous qu'elle est la seule classe de stockage à avoir l'annotation par défaut.</p>	chaîne	ontap-gold

`<code>volumeReclaimPolicy</code>`

Réglage	Guidage	Type	Options
<code>volumeReclaimPolicy</code>	Cette règle définit la règle de récupération pour les volumes persistants d'Astra. Définition de cette règle sur <code>Retain</code> Conserve les volumes persistants après la suppression d'Astra. Définition de cette règle sur <code>Delete</code> supprime les volumes persistants après la suppression d'astra. Si cette valeur n'est pas définie, les PV sont conservés.	chaîne	<ul style="list-style-type: none">• <code>Retain</code> (Il s'agit de la valeur par défaut)• <code>Delete</code>

`<code>ingressType</code>`





Réglage	Guidage	Type	Options
ingressType	<p>Utilisez l'un des types d'entrées suivants</p> <pre>:*Generic*</pre> <p>(ingressType: "Generic") (Par défaut) utilisez cette option lorsque vous avez un autre contrôleur d'entrée en service ou que vous préférez utiliser votre propre contrôleur d'entrée. Après le déploiement du centre de contrôle Astra, vous devez configurer le "contrôleur d'entrée" Pour exposer Astra Control Center avec une URL AccTraefik</p> <pre>(ingressType: "AccTraefik")</pre> <p>Utilisez cette option si vous préférez ne pas configurer un contrôleur d'entrée. 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 »</p> <p>(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</p>	chaîne	<ul style="list-style-type: none"> • Generic (il s'agit de la valeur par défaut) • AccTraefik

`<code>astraResourcesScaler</code>`

Réglage	Guidage	Type	Options
<code>astraResourcesScaler</code>	<p>Options d'évolutivité pour les limites de ressources AstrakControlCenter. Par défaut, Astra Control Center se déploie avec des demandes de ressources définies pour la plupart des composants d'Astra. Avec cette configuration, la pile logicielle Astra Control Center est plus performante dans les environnements soumis à une charge et à une évolutivité accrues des applications. Cependant, dans les scénarios utilisant des grappes de développement ou de test plus petites, le champ CR <code>astraResourcesScaler</code> peut être réglé sur <code>Off</code>. Cela désactive les demandes de ressources et permet un déploiement sur les clusters plus petits.</p>	chaîne	<ul style="list-style-type: none">• Default (Il s'agit de la valeur par défaut)• Off

<code>crds</code>

Vos sélections dans cette section déterminent comment Astra Control Center doit traiter les CRD.

Réglage	Guidage	Type	Exemple
<code>crds.externalCertManager</code>	Si vous utilisez un gestionnaire de certificats externe, modifiez-le <code>externalCertManager</code> à <code>true</code> . La valeur par défaut <code>false</code> Provoque l'installation d'Astra Control Center de ses propres CRD de <code>cert Manager</code> lors de l'installation. Les CRDS sont des objets à l'échelle du cluster et leur installation peut avoir un impact sur d'autres parties du cluster. Vous pouvez utiliser cet indicateur pour signaler à Astra Control Center que ces CRD seront installés et gérés par l'administrateur de cluster en dehors du centre de contrôle Astra.	Booléen	<code>False</code> (cette valeur est la valeur par défaut)
<code>crds.externalTraefik</code>	Par défaut, Astra Control Center installe les CRD Traefik requis. Les CRDS sont des objets à l'échelle du cluster et leur installation peut avoir un impact sur d'autres parties du cluster. Vous pouvez utiliser cet indicateur pour signaler à Astra Control Center que ces CRD seront installés et gérés par l'administrateur de cluster en dehors du centre de contrôle Astra.	Booléen	<code>False</code> (cette valeur est la valeur par défaut)

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

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

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

Exemple de réponse :

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

Vérifiez l'état du système

Vous pouvez vérifier l'état du système à l'aide des commandes kubectl. Si vous préférez utiliser OpenShift, vous pouvez utiliser des commandes oc comparables pour les étapes de vérification.

Étapes

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

fluent-bit-ds-9rql1	1/1	Running	0
6m54s			
fluent-bit-ds-w5mp7	1/1	Running	0
6m54s			
graphql-server-7c7cc49776-jz2kn	1/1	Running	0
2m29s			
identity-87c59c975-9jpnf	1/1	Running	0
9m6s			
influxdb2-0	1/1	Running	0
13m			
keycloak-operator-84ff6d59d4-qcnmc	1/1	Running	0
7m1s			
krakend-cbf6c7df9-mdtzv	1/1	Running	0
2m30s			
license-5b888b78bf-plj6j	1/1	Running	0
9m32s			
login-ui-846b4664dd-fz8hv	1/1	Running	0
2m24s			
loki-0	1/1	Running	0
13m			
metrics-facade-779cc9774-n26rw	1/1	Running	0
9m18s			
monitoring-operator-974db78f-pkspq	2/2	Running	0
6m58s			
nats-0	1/1	Running	0
13m			
nats-1	1/1	Running	0
13m			
nats-2	1/1	Running	0
13m			
nautilus-7bdc7ddc54-49tfn	1/1	Running	0
7m50s			
nautilus-7bdc7ddc54-cwc79	1/1	Running	0
9m36s			
openapi-5584ff9f46-gbrdj	1/1	Running	0
9m17s			
openapi-5584ff9f46-z9mzk	1/1	Running	0
9m17s			
packages-bfc58cc98-lpxq9	1/1	Running	0
8m58s			
polaris-consul-consul-server-0	1/1	Running	0
13m			
polaris-consul-consul-server-1	1/1	Running	0
13m			
polaris-consul-consul-server-2	1/1	Running	0
13m			

polaris-keycloak-0 (6m15s ago) 6m56s	1/1	Running	3
polaris-keycloak-1 4m22s	1/1	Running	0
polaris-keycloak-2 3m41s	1/1	Running	0
polaris-keycloak-db-0 6m56s	1/1	Running	0
polaris-keycloak-db-1 4m23s	1/1	Running	0
polaris-keycloak-db-2 3m36s	1/1	Running	0
polaris-mongodb-0 13m	2/2	Running	0
polaris-mongodb-1 13m	2/2	Running	0
polaris-mongodb-2 12m	2/2	Running	0
polaris-ui-5ccff47897-8rzgh 2m33s	1/1	Running	0
polaris-vault-0 13m	1/1	Running	0
polaris-vault-1 13m	1/1	Running	0
polaris-vault-2 13m	1/1	Running	0
public-metrics-6cb7bfc49b-p54xm (8m29s ago) 9m31s	1/1	Running	1
storage-backend-metrics-5c77994586-kjn48 8m52s	1/1	Running	0
storage-provider-769fdc858c-62w54 8m54s	1/1	Running	0
task-service-9ffc484c5-kx9f4 (8m44s ago) 9m34s	1/1	Running	3
telegraf-ds-bphb9 6m54s	1/1	Running	0
telegraf-ds-rtsm2 6m54s	1/1	Running	0
telegraf-ds-s9h5h 6m54s	1/1	Running	0
telegraf-rs-lbpv7 6m54s	1/1	Running	0
telemetry-service-57cfb998db-zjx78 (8m40s ago) 9m26s	1/1	Running	1
tenancy-5d5dfbcf9f-vmboxh 9m5s	1/1	Running	0

```

traefik-7b87c4c474-jmcp2      1/1      Running   0
2m24s
traefik-7b87c4c474-t9k8x     1/1      Running   0
2m24s
trident-svc-c78f5b6bd-nwdsq  1/1      Running   0
9m22s
vault-controller-55bbc96668-c6425 1/1      Running   0
11m
vault-controller-55bbc96668-lq9n9 1/1      Running   0
11m
vault-controller-55bbc96668-rfkgg 1/1      Running   0
11m

```

2. (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 du cluster était indiqué dans les journaux, vous pouvez essayer de nouveau l'enregistrement via le ["Ajout du flux de travail du 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 de configuration initial que vous utiliserez lorsque vous vous connectez à Astra Control Center :

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

Réponse :

NAME	UUID	VERSION	ADDRESS
READY			
astra	9aa5fdae-4214-4cb7-9976-5d8b4c0ce27f	22.11.0-82	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. Ces procédures fournissent des exemples de configuration pour un contrôleur d'entrée si vous avez utilisé la valeur par défaut de `ingressType: "Generic"` Dans la ressource personnalisée Astra Control Center (`astra_control_center.yaml`). Vous n'avez pas besoin d'utiliser cette procédure si vous avez spécifié `ingressType: "AccTraefik"` Dans la ressource personnalisée Astra Control Center (`astra_control_center.yaml`).

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.

Les étapes de configuration varient en fonction du type de contrôleur d'entrée utilisé. Le centre de contrôle Astra prend en charge de nombreux types de contrôleurs d'entrée. Ces procédures de configuration fournissent des exemples pour les types de contrôleurs d'entrée suivants :

- 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.

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. Déployer une ressource d'entrée dans le `netapp-acc` (ou nom personnalisé) de l'espace de noms utilisant le type de ressource `v1` pour un schéma (`istio-Ingress.yaml` est utilisé dans cet exemple) :

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

5. Appliquer les modifications :

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

6. Vérifier l'état de l'entrée :

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

Réponse :

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

7. Terminer l'installation du centre de contrôle Astra.

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 nom personnalisé) de l'espace de noms utilisant le type de ressource `v1` pour un schéma (`nginx-Ingress.yaml` est utilisé dans cet exemple) :

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

3. Appliquer les modifications :

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



NetApp recommande d'installer le contrôleur nginx en tant que déploiement plutôt qu'en tant que `daemonSet`.

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, saisissez le nom de domaine complet (y compris le `https://` prefix) que vous avez utilisé dans `astraAddress` dans le `astra_control_center.yaml` CR quand [Vous avez installé Astra Control Center](#).
2. Acceptez les certificats auto-signés si 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.yaml` CR quand [Vous avez installé Astra Control Center](#), suivi du mot de passe de configuration initiale (`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.



S'il s'agit de votre première connexion et que vous oubliez le mot de passe et qu'aucun autre compte d'utilisateur administratif n'a encore été créé, contactez ["Support NetApp"](#) pour obtenir de l'aide sur la récupération des mots de

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. Recherchez 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 deploy/acc-operator-controller-manager -n netapp-acc-operator -c manager -f
```

Et la suite

- (Facultatif) en fonction de votre environnement, effectuez l'installation complète après l'installation "[étapes de configuration](#)".
- Terminez le déploiement en effectuant le processus "[tâches de configuration](#)".

=
: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. Utilisez 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

- * Conditions préalables à l'environnement remplies* : "[Avant de commencer l'installation, préparez votre environnement pour le déploiement d'Astra Control Center](#)".
- **Opérateurs de grappe et services API sains** :
 - Depuis votre cluster OpenShift, assurez-vous que tous les opérateurs de clusters sont en état sain :

```
oc get clusteroperators
```

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

```
oc get apiservices
```

- **Adresse FQDN** : obtention d'une adresse FQDN pour Astra Control Center dans votre centre de données.
- **Autorisations OpenShift** : obtenez les autorisations nécessaires et l'accès à Red Hat OpenShift Container Platform pour effectuer les étapes d'installation décrites.
- **Cert Manager configuré** : 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 gestionnaire de certificat. Par défaut, Astra Control Center installe son propre gestionnaire de certificats lors de l'installation.
- **Contrôleur d'entrée Kubernetes** : si vous disposez d'un contrôleur d'entrée Kubernetes qui gère l'accès externe aux services, comme l'équilibrage de charge dans un cluster, vous devez le configurer pour l'utiliser avec Astra Control Center :

- a. Créer l'espace de noms de l'opérateur :

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

- b. "[Terminez l'installation](#)" pour votre type de contrôleur d'entrée.

Étapes

- [Téléchargez et extrayez 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 extrayez Astra Control Center

1. Accédez au "[Page de téléchargement de l'évaluation Astra Control Center](#)" Sur le site de support NetApp.
2. Téléchargez le pack contenant Astra Control Center (`astra-control-center-[version].tar.gz`).
3. (Recommandé mais facultatif) Téléchargez le lot de certificats et de signatures pour Astra Control Center (`astra-control-center-certs-[version].tar.gz`) pour vérifier la signature du paquet :

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

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

La sortie s'affiche `Verified OK` une fois la vérification terminée.

4. Extraire les images du pack Astra Control Center :

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

Installez le plug-in NetApp Astra kubectl

Le plug-in de ligne de commande NetApp Astra kubectl 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 de plug-ins pour différentes architectures CPU et systèmes d'exploitation. Avant d'effectuer cette tâche, vous devez savoir quelle unité centrale et quel système d'exploitation vous possédez.

Étapes

1. Répertoriez les binaires NetApp Astra kubectl disponibles et notez le nom du fichier dont vous avez besoin pour votre système d'exploitation et votre architecture de processeur :



La bibliothèque de plug-ins kubectl fait partie du bundle tar et est extraite dans le dossier kubectl-astra.

```
ls kubectl-astra/
```

2. Déplacez le bon binaire dans le chemin actuel et renommez-le kubectl-astra:

```
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. Accédez au répertoire racine du tarball. Vous devriez voir ce fichier et ce répertoire:

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

2. Envoyez les images du package dans le répertoire d'images Astra Control Center vers votre registre local. Effectuez les remplacements suivants avant d'exécuter le `push-images` commande :

- Remplacez `<BUNDLE_FILE>` par le nom du fichier bundle Astra Control (`acc.manifest.bundle.yaml`).
- Remplacer `<MY_FULL_REGISTRY_PATH>` par l'URL du référentiel Docker, par exemple "`<a href="https://<docker-registry>" class="bare">https://<docker-registry>"`".
- Remplacez `<MY_REGISTRY_USER>` par le nom d'utilisateur.
- Remplacez `<MY_REGISTRY_TOKEN>` par un jeton autorisé pour le registre.

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

Podman

1. Accédez au répertoire racine du tarball. Vous devriez voir ce fichier et ce répertoire:

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

2. Connectez-vous à votre registre :

```
podman login <YOUR_REGISTRY>
```

3. Préparez et exécutez l'un des scripts suivants qui est personnalisé pour la version de Podman que vous utilisez. Remplacez `<MY_FULL_REGISTRY_PATH>` par l'URL de votre référentiel qui inclut tous les sous-répertoires.

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

```

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

```

Podman 3

```

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

```



Le chemin d'accès à l'image que le script crée doit ressembler aux éléments suivants, selon la configuration de votre registre : <https://netappdownloads.jfrog.io/docker-astra-control-prod/netapp/astra/acc/22.11.0-82/image:version>

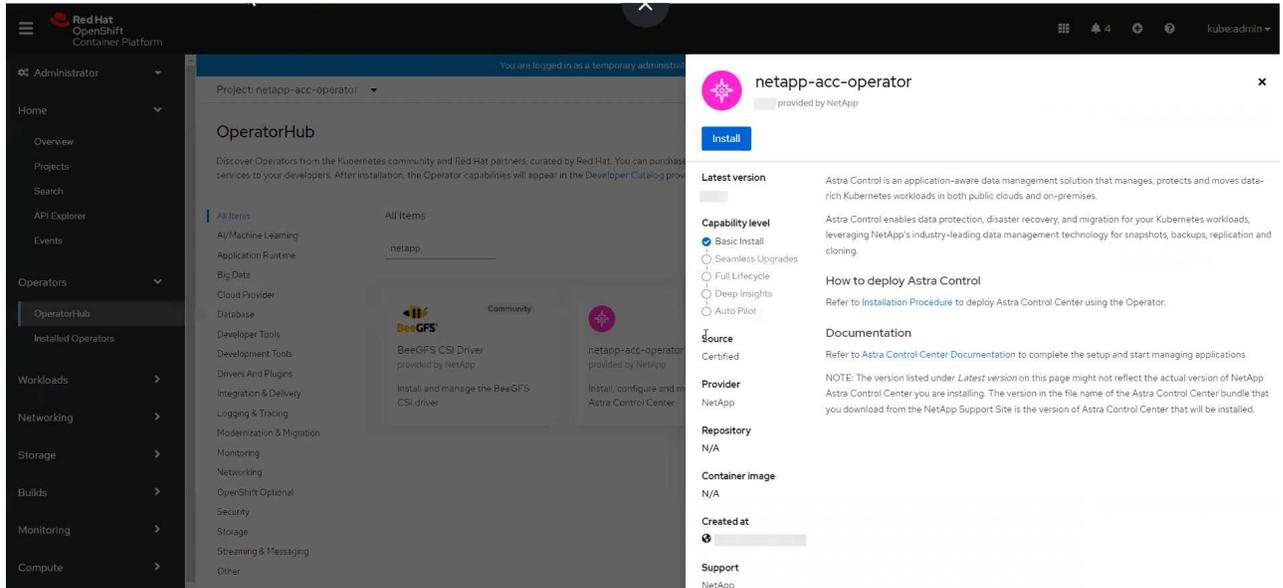
Recherchez la page d'installation de l'opérateur

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**.

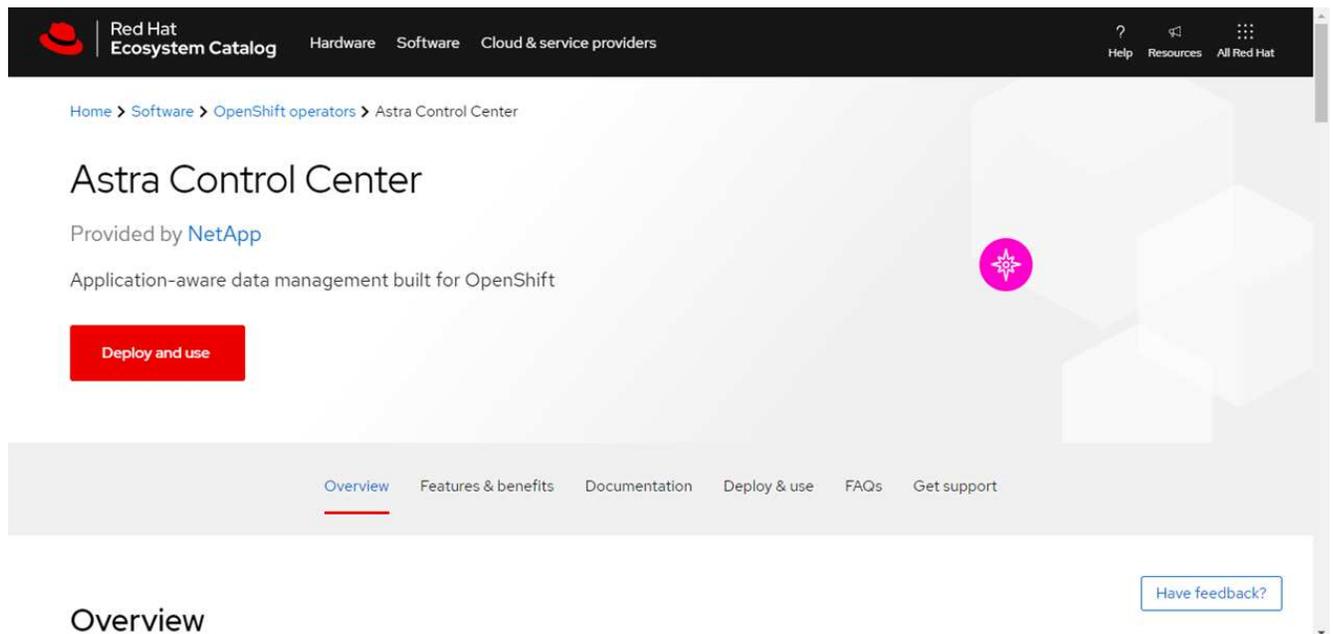
iii. Recherchez et sélectionnez l'opérateur NetApp Astra Control Center.



◦ 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**.

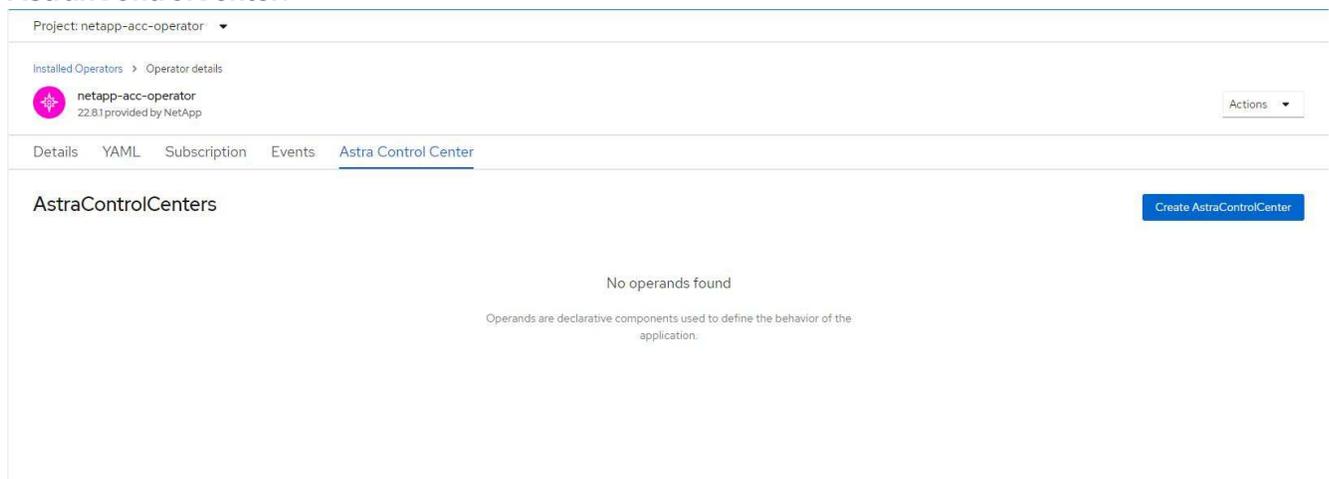


Si vous avez sélectionné une stratégie d'approbation manuelle, vous serez invité à approuver le plan d'installation manuelle pour cet opérateur.

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

Poser le centre de contrôle Astra

1. Dans la console de l'onglet **Astra Control Center** de l'opérateur Astra Control Center, sélectionnez **Create AstrakControlCenter**.



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

- Conservez ou ajustez le nom du centre de contrôle Astra.
- Ajouter des étiquettes pour le centre de contrôle Astra.
- Activez ou désactivez Auto support. Il est recommandé de conserver la fonctionnalité Auto support.
- Saisissez le nom de domaine complet ou l'adresse IP d'Astra Control Center. N'entrez pas `http://` ou `https://` dans le champ d'adresse.
- Entrez la version Astra Control Center, par exemple 22.04.1.
- Entrez un nom de compte, une adresse e-mail et un nom d'administrateur.
- Choisir une règle de récupération de volume de `Retain`, `Recycle`, ou `Delete`. La valeur par défaut est `Retain`.
- Sélectionnez le type d'entrée :

▪ **Generic** (`ingressType: "Generic"`) (Par défaut)

Utilisez cette option si vous avez un autre contrôleur d'entrée en service ou si vous préférez utiliser votre propre contrôleur d'entrée. Après le déploiement du centre de contrôle Astra, vous devez configurer le "contrôleur d'entrée" Pour exposer Astra Control Center avec une URL.

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

Utilisez cette option lorsque vous préférez ne pas configurer de contrôleur d'entrée. 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"](#).

- a. Dans **image Registry**, entrez le chemin d'accès au registre d'images du conteneur local. N'entrez pas `http://` ou `https://` dans le champ d'adresse.
- b. Si vous utilisez un registre d'images qui nécessite une authentification, saisissez le secret d'image.



Si vous utilisez un registre qui nécessite une authentification, [créez un secret sur le cluster](#).

- c. Entrez le prénom de l'administrateur.
- d. Configurer l'évolutivité des ressources.
- e. Indiquez la classe de stockage par défaut.



Si une classe de stockage par défaut est configurée, assurez-vous qu'elle est la seule classe de stockage qui possède l'annotation par défaut.

- f. Définissez les préférences de gestion de CRD.

3. Sélectionnez la vue YAML pour vérifier les paramètres sélectionnés.
4. Sélectionnez `Create`.

Créer un secret de registre

Si vous utilisez un registre qui nécessite une authentification, créez un secret sur le cluster OpenShift et entrez le nom secret dans le `Create AstraControlCenter` champ de formulaire.

1. Créez un espace de noms pour l'opérateur du centre de contrôle Astra :

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

2. Créez un secret dans ce namespace :

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



Astra Control prend uniquement en charge les secrets de registre Docker.

3. Renseignez les champs restants dans [Le champ de formulaire Create AstrakControlCenter](#).

Et la suite

Complétez le "[les étapes restantes](#)" Pour vérifier que le centre de contrôle Astra est correctement installé, configurez un contrôleur d'entrée (en option) et connectez-vous à l'interface utilisateur. De plus, vous devez effectuer cette opération "[tâches de configuration](#)" une fois l'installation terminée.

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.

Outre les exigences de l'environnement en matière de ressources, Astra Control Center requiert les ressources suivantes :

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 BlueXP, anciennement 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	<p>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.</p> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p>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.</p> </div>

Composant	Conditions requises
Configuration d'Astra Trident et ONTAP	<p>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 ONTAP suivantes qui sont créées lorsque vous importez votre cluster Kubernetes dans NetApp BlueXP (anciennement Cloud Manager). Découvrez Astra Trident :</p> <ul style="list-style-type: none"> • <code>vsaworkingenvironment-<>-ha-nas</code> <code>csi.trident.netapp.io</code> • <code>vsaworkingenvironment-<>-ha-san</code> <code>csi.trident.netapp.io</code> • <code>vsaworkingenvironment-<>-single-nas</code> <code>csi.trident.netapp.io</code> • <code>vsaworkingenvironment-<>-single-san</code> <code>csi.trident.netapp.io</code>



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. [Configuration de NetApp BlueXP pour AWS.](#)
5. [Installer Astra Control Center pour AWS.](#)

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 BlueXP (anciennement 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.



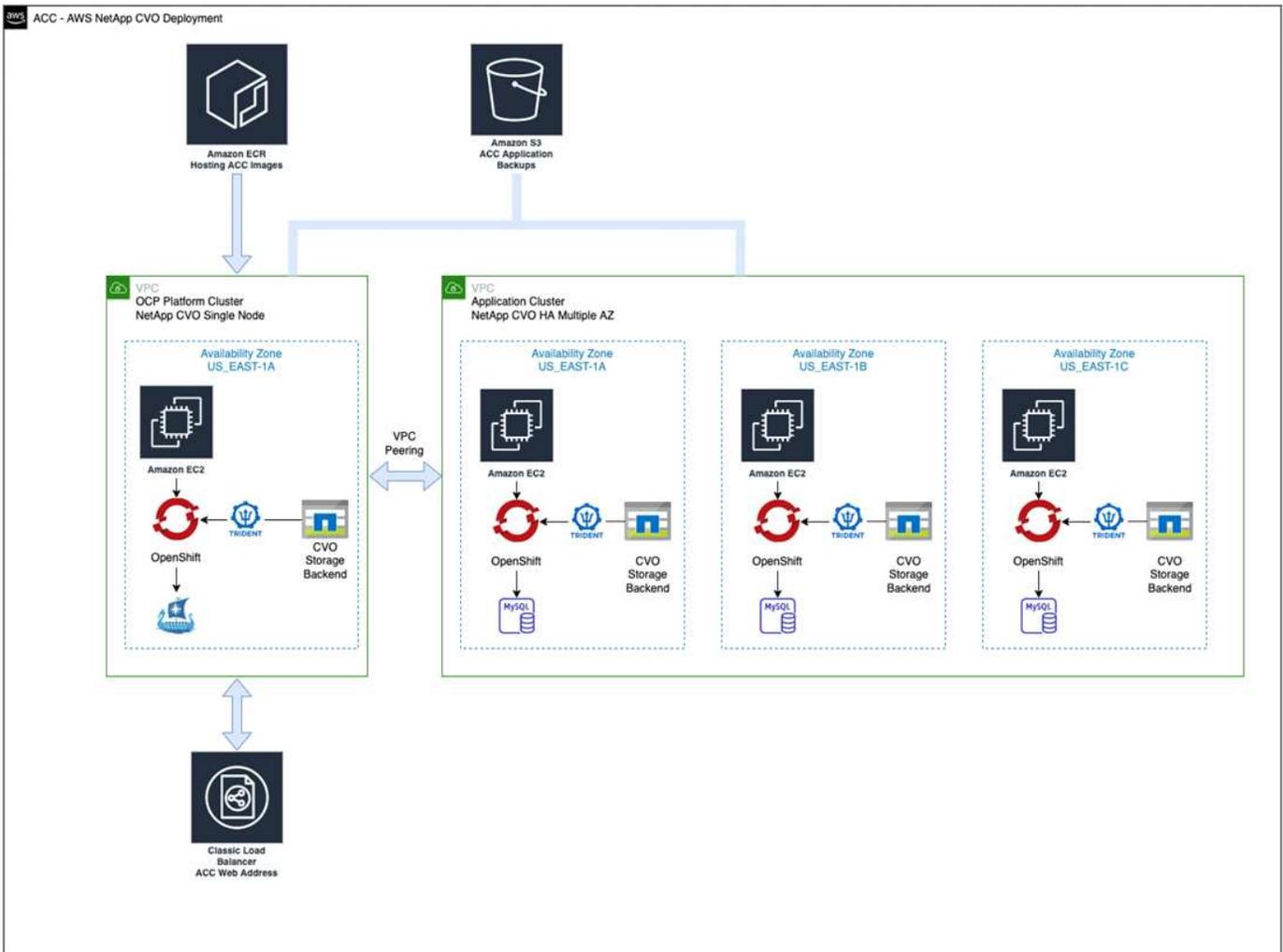
Si vous ne créez pas d'ECR, le centre de contrôle Astra ne peut pas accéder aux données de surveillance à partir d'un cluster contenant Cloud Volumes ONTAP avec un back-end AWS. Le problème survient lorsque le cluster que vous essayez de découvrir et de gérer à l'aide d'Astra Control Center n'a pas accès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



Configuration de NetApp BlueXP pour AWS

Avec NetApp BlueXP (anciennement Cloud Manager), créez un espace de travail, ajoutez un connecteur à AWS, créez un environnement de travail et importez le cluster.

Suivez la documentation BlueXP 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 BlueXP"](#)

Étapes

1. Ajoutez vos informations d'identification à BlueXP.
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.

Installer Astra Control Center pour AWS

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.

Outre les exigences de l'environnement en matière de ressources, Astra Control Center requiert les ressources suivantes :

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 BlueXP, anciennement 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 backend
Registre d'images	<p>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.</p> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;">  <p>Vous devez activer l'accès anonyme pour extraire les images Restic pour les sauvegardes.</p> </div>
Configuration d'Astra Trident et ONTAP	<p>Avec Astra Control Center, il est nécessaire de créer une classe de stockage et de la définir comme classe de stockage par défaut. Astra Control Center prend en charge les classes de stockage Kubernetes ONTAP suivantes qui sont créées lorsque vous importez votre cluster Kubernetes dans NetApp BlueXP. Découvrez Astra Trident :</p> <ul style="list-style-type: none"> • <code>vsaworkingenvironment-<>-ha-nas</code> <code>csi.trident.netapp.io</code> • <code>vsaworkingenvironment-<>-ha-san</code> <code>csi.trident.netapp.io</code> • <code>vsaworkingenvironment-<>-single-nas</code> <code>csi.trident.netapp.io</code> • <code>vsaworkingenvironment-<>-single-san</code> <code>csi.trident.netapp.io</code>



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. [Configuration de NetApp BlueXP pour GCP.](#)
6. [Installer Astra Control Center pour GCP.](#)

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 BlueXP (anciennement 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.

Configuration de NetApp BlueXP pour GCP

À l'aide de NetApp BlueXP (anciennement Cloud Manager), créez un espace de travail, ajoutez un connecteur à GCP, créez un environnement de travail et importez le cluster.

Suivez la documentation BlueXP 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 informations d'identification à BlueXP. 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 BlueXP](#)".
 - 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.

Installer Astra Control Center pour GCP

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.

Outre les exigences de l'environnement en matière de ressources, Astra Control Center requiert les ressources suivantes :

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 BlueXP)	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	<p>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.</p> <div style="display: flex; align-items: center;">  <p>Vous devez activer l'accès anonyme pour extraire les images Restic pour les sauvegardes.</p> </div>

Composant	Conditions requises
Configuration d'Astra Trident et ONTAP	<p>Avec Astra Control Center, il est nécessaire de créer une classe de stockage et de la définir comme classe de stockage par défaut. Astra Control Center prend en charge les classes de stockage Kubernetes ONTAP suivantes qui sont créées lorsque vous importez votre cluster Kubernetes dans NetApp BlueXP. Découvrez Astra Trident :</p> <ul style="list-style-type: none"> • <code>vsaworkingenvironment-⟨⟩-ha-nas</code> <code>csi.trident.netapp.io</code> • <code>vsaworkingenvironment-⟨⟩-ha-san</code> <code>csi.trident.netapp.io</code> • <code>vsaworkingenvironment-⟨⟩-single-nas</code> <code>csi.trident.netapp.io</code> • <code>vsaworkingenvironment-⟨⟩-single-san</code> <code>csi.trident.netapp.io</code>



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. [Configuration de NetApp BlueXP \(anciennement Cloud Manager\) pour Azure.](#)
6. [Installer et configurer Astra Control Center pour Azure.](#)

Installez un cluster Red Hat OpenShift sur Azure

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

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

- ["Installation du cluster OpenShift sur Azure"](#).
- ["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 BlueXP.

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. Configurez 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 $AZ_ACR_REGISTRY/$image
    docker push $AZ_ACR_REGISTRY/$image
done < astra-control-center-22.04.41.manifest
```

10. Configurer les zones DNS.

Configuration de NetApp BlueXP (anciennement Cloud Manager) pour Azure

À l'aide de BlueXP (anciennement Cloud Manager), créez un espace de travail, ajoutez un connecteur à Azure, créez un environnement de travail et importez le cluster.

Suivez la documentation BlueXP pour effectuer les étapes suivantes. Voir "[Mise en route de BlueXP dans Azure](#)".

Ce dont vous avez besoin

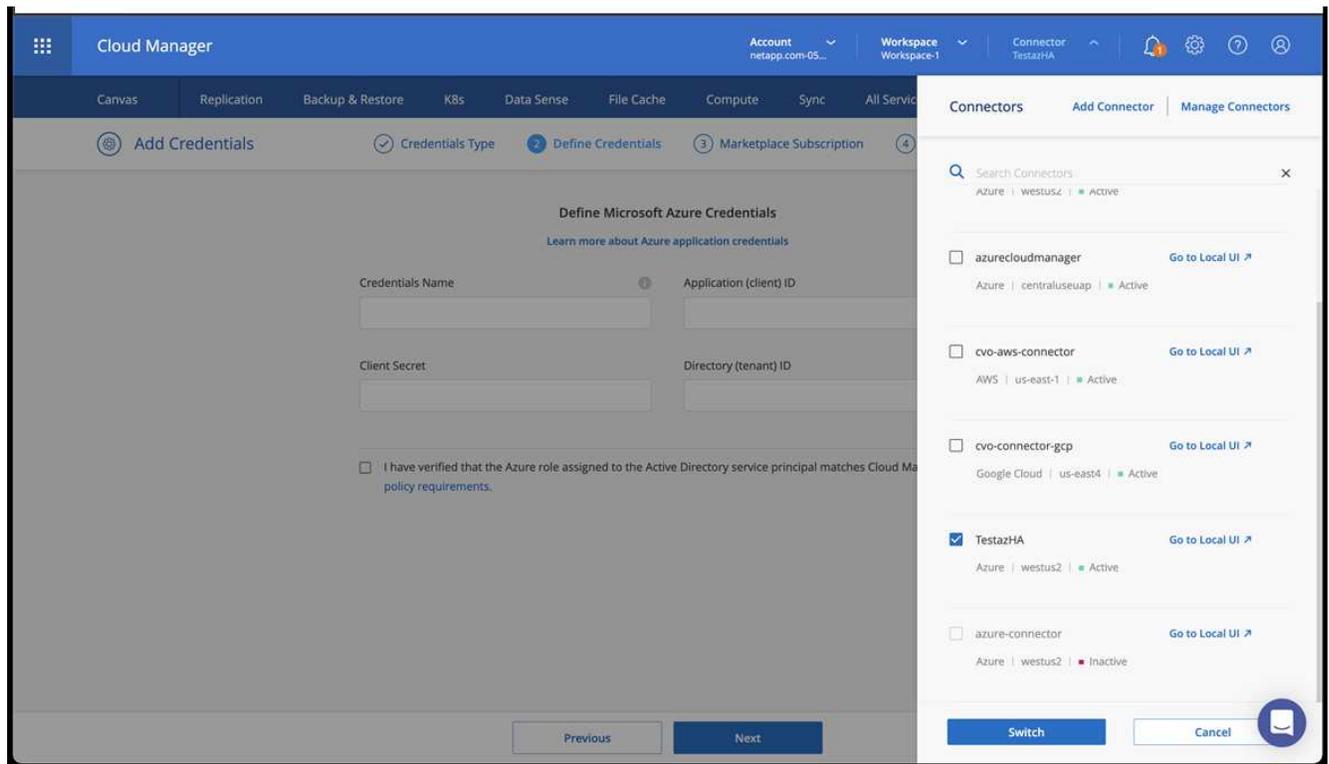
Accès au compte Azure avec les autorisations IAM et les rôles requis

Étapes

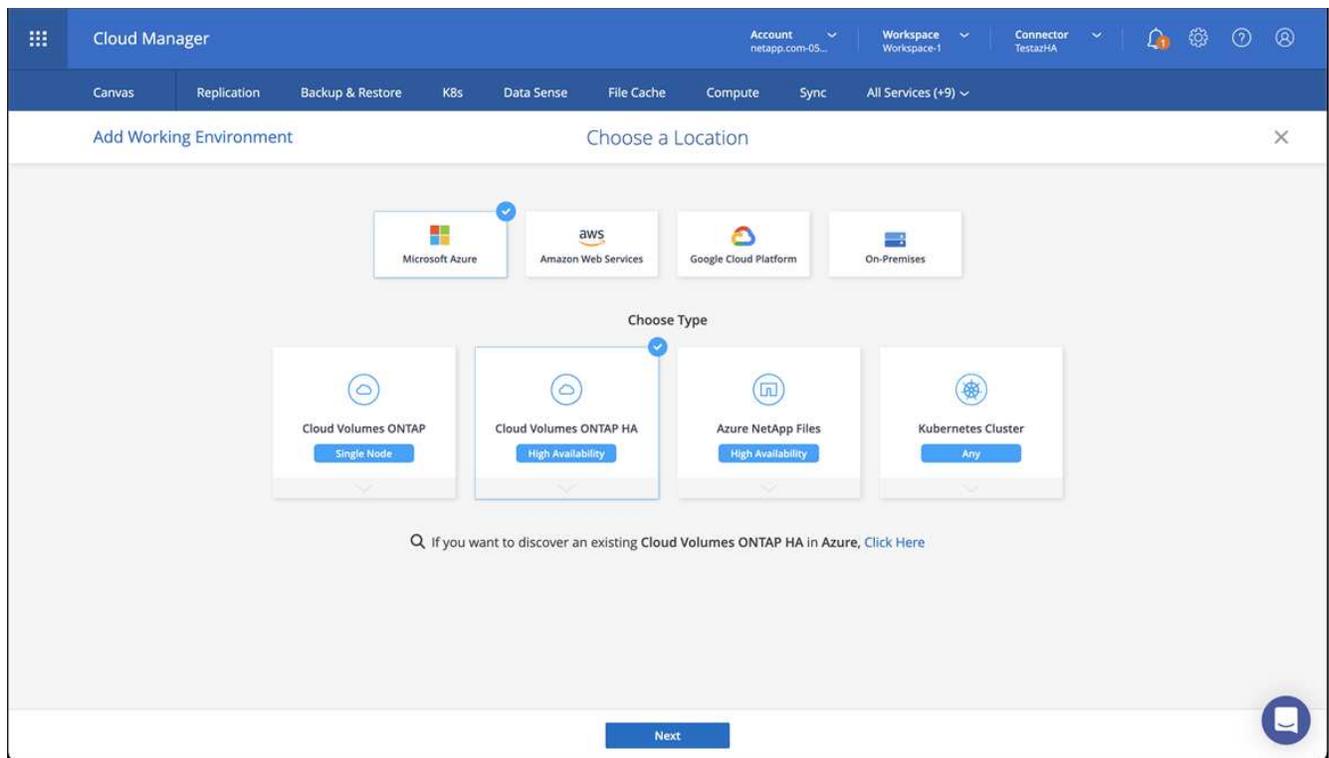
1. Ajoutez vos informations d'identification à BlueXP.
2. Ajoutez un connecteur pour Azure. Voir "[Politiques BlueXP](#)".
 - a. Choisissez **Azure** comme fournisseur.
 - b. Vous pouvez entrer les identifiants Azure, notamment l'ID de l'application, le secret client et l'ID du répertoire (locataire).

Voir "[Création d'un connecteur dans Azure à partir de BlueXP](#)".

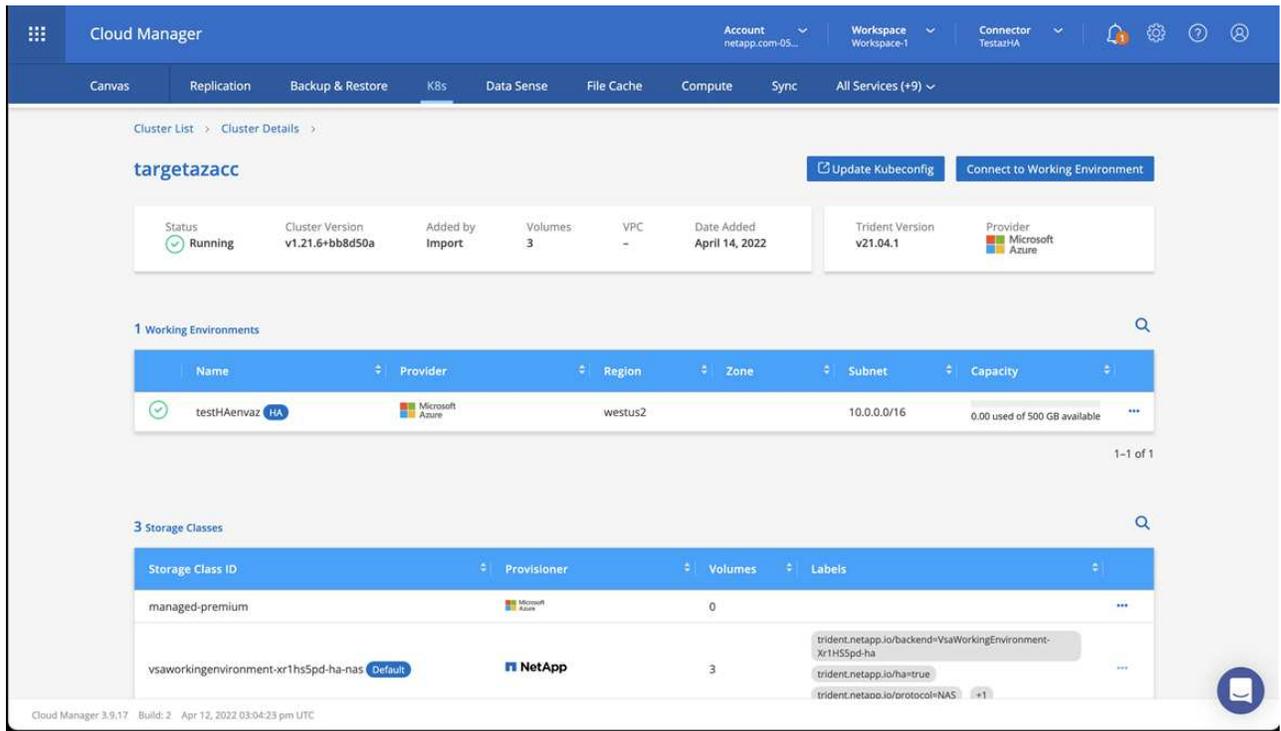
3. S'assurer que le connecteur est en marche et basculer vers ce connecteur.



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 Astra Control Center pour Azure

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](#)".

=

:allow-uri-read:

Configurer le centre de contrôle Astra

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](#)

- [Préparez votre environnement à la gestion des clusters avec Astra Control](#)
- [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 Astra Control ou ["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.

Les licences Astra Control Center mesurent les ressources CPU avec des unités de processeur Kubernetes et représentent les ressources de processeur attribuées aux nœuds de travail de tous les clusters Kubernetes gérés. Les licences dépendent de l'utilisation des processeurs virtuels. Pour plus d'informations sur le calcul des licences, reportez-vous à la section ["Licences"](#).



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.



Pour mettre à jour une évaluation existante ou une licence complète, reportez-vous à la section ["Mettre à jour une licence existante"](#).

Ce dont vous avez besoin

- Accès à une instance Astra Control Center récemment installée.
- Autorisations de rôle d'administrateur.
- A ["Fichier de licence NetApp"](#) (NLF).

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 et que vous n'envoyez pas de données à AutoSupport, assurez-vous de stocker votre identifiant de compte pour éviter toute perte de données en cas de défaillance d'Astra Control Center.

Préparez votre environnement à la gestion des clusters avec Astra Control

Avant d'ajouter un cluster, assurez-vous que les conditions préalables suivantes sont remplies. Vous devez également effectuer des vérifications d'admissibilité pour vous assurer que votre cluster est prêt à être ajouté au Centre de contrôle Astra et créer des rôles pour la gestion du cluster.

Ce dont vous avez besoin

- Assurez-vous que les nœuds workers de votre cluster sont configurés avec les pilotes de stockage appropriés afin que les pods puissent interagir avec le système de stockage back-end.
- Votre environnement répond au "de l'environnement opérationnel" Pour Astra Trident et Astra Control Center.
- Une version d'Astra Trident "Pris en charge par Astra Control Center" est installé :



C'est possible "Déployez Astra Trident" Utilisation de l'opérateur Trident (manuellement ou à l'aide du graphique Helm) ou `tridentctl`. Avant d'installer ou de mettre à niveau Astra Trident, consultez le "systèmes front-end, systèmes back-end et configurations hôte pris en charge".

- **Système back-end de stockage Trident configuré** : au moins un système back-end de stockage Trident Astra doit être configuré "configuré" sur le cluster.
 - **Classes de stockage Trident configurées** : au moins une classe de stockage Astra Trident doit être "configuré" sur le cluster. Si une classe de stockage par défaut est configurée, assurez-vous qu'elle est la seule classe de stockage qui possède l'annotation par défaut.
 - **Contrôleur de snapshot de volume Astra Trident et classe de snapshot de volume installés et configurés** : le contrôleur de snapshot de volume doit être "installé" Il est ainsi possible de créer des snapshots dans Astra Control. Au moins un Astra Trident `VolumeSnapshotClass` a été "configuration" par un administrateur.
- **Kubeconfig accessible**: Vous avez accès au "cluster kubeconfig" qui ne comprend qu'un seul élément de contexte.
 - **Informations d'identification ONTAP** : vous avez besoin d'informations d'identification ONTAP et d'un superutilisateur et d'un ID utilisateur définis sur le système ONTAP de sauvegarde pour sauvegarder et restaurer des applications avec le Centre de contrôle Astra.

Exécutez les commandes suivantes dans la ligne de commande ONTAP :

```
export-policy rule modify -vserver <storage virtual machine name>
-policyname <policy name> -ruleindex 1 -superuser sys
export-policy rule modify -vserver <storage virtual machine name>
-policyname <policy name> -ruleindex 1 -anon 65534
```

- **Rancher uniquement**: Lorsque vous gérez des clusters d'applications dans un environnement Rancher, modifiez le contexte par défaut du cluster d'applications dans le fichier kubeconfig fourni par Rancher pour utiliser un contexte de plan de contrôle au lieu du contexte du serveur d'API Rancher. La charge est réduite sur le serveur API Rancher et les performances sont améliorées.

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       22.10.0
```

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. Reportez-vous à la ["Documentation Trident"](#) pour obtenir des instructions.

2. Assurez-vous que les pods fonctionnent :

```
kubectl get pods -n trident
```

3. Déterminez 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
```

Exemple de réponse :

```
NAME          PROVISIONER          RECLAIMPOLICY
VOLUMEBINDINGMODE  ALLOWVOLUMEEXPANSION  AGE
ontap-gold (default)  csi.trident.netapp.io  Delete          Immediate
true                5d23h
```

Créez un nombre limité de rôles de cluster kubeconfig

Vous pouvez éventuellement créer un rôle d'administrateur limité pour Astra Control Center. Il ne s'agit pas d'une procédure requise pour la configuration du centre de contrôle Astra. Cette procédure permet de créer un kubeconfig distinct qui limite les autorisations de contrôle Astra sur les clusters qu'il gère.

Ce dont vous avez besoin

Assurez-vous que vous disposez des éléments suivants pour le cluster que vous souhaitez gérer avant d'effectuer la procédure suivante :

- kubectl v1.23 ou version ultérieure installée
- Accès kubectl au cluster que vous souhaitez ajouter et gérer avec Astra Control Center



Pour cette procédure, il n'est pas nécessaire d'avoir un accès kubectl au cluster qui exécute Astra Control Center.

- Un kubeconfig actif pour le cluster que vous avez l'intention de gérer avec des droits d'administrateur de cluster pour le contexte actif

Étapes

1. Créer un compte de service :

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

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

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

+

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

- a. Appliquer le compte de service :

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

2. Créez un rôle de cluster limité avec le minimum d'autorisations nécessaires à la gestion d'un cluster par Astra Control :

- a. Créer un `ClusterRole` fichier appelé `astra-admin-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>astra-admin-account.yaml</strong>
```

+

```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  name: astra-admin-account
rules:

# Get, List, Create, and Update all resources
# Necessary to backup and restore all resources in an app
- apiGroups:
```

```

- '*'
resources:
- '*'
verbs:
- get
- list
- create
- patch

# Delete Resources
# Necessary for in-place restore and AppMirror failover
- apiGroups:
- ""
- apps
- autoscaling
- batch
- crd.projectcalico.org
- extensions
- networking.k8s.io
- policy
- rbac.authorization.k8s.io
- snapshot.storage.k8s.io
- trident.netapp.io
resources:
- configmaps
- cronjobs
- daemonsets
- deployments
- horizontalpodautoscalers
- ingresses
- jobs
- namespaces
- networkpolicies
- persistentvolumeclaims
- poddisruptionbudgets
- pods
- podtemplates
- podsecuritypolicies
- replicaset
- replicationcontrollers
- replicationcontrollers/scale
- rolebindings
- roles
- secrets
- serviceaccounts
- services

```

```

- statefulsets
- tridentmirrorrelationships
- tridentnapshotinfos
- volumesnapshots
- volumesnapshotcontents
verbs:
- delete

# Watch resources
# Necessary to monitor progress
- apiGroups:
  - ""
  resources:
  - pods
  - replicationcontrollers
  - replicationcontrollers/scale
  verbs:
  - watch

# Update resources
- apiGroups:
  - ""
  - build.openshift.io
  - image.openshift.io
  resources:
  - builds/details
  - replicationcontrollers
  - replicationcontrollers/scale
  - imagestreams/layers
  - imagestreamtags
  - imagedtags
  verbs:
  - update

# Use PodSecurityPolicies
- apiGroups:
  - extensions
  - policy
  resources:
  - podsecuritypolicies
  verbs:
  - use

```

a. Appliquer le rôle de cluster :

```
kubectl apply -f astra-admin-account.yaml
```

3. Créer la liaison de rôle cluster pour le rôle cluster vers le compte de service :

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

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

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

+

```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
  name: astracontrol-admin
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: ClusterRole
  name: astra-admin-account
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 :

- 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 pour nommer votre cluster.

```
mv kubeconfig-sa 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](#).

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.



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.

Ce dont vous avez besoin

- Avant d'ajouter un cluster, vérifiez et effectuez les opérations nécessaires [tâches préalables](#).

Étapes

1. Naviguer à partir du menu Tableau de bord ou clusters :
 - Dans **Dashboard**, sélectionnez **Add** dans le volet clusters.
 - Dans la zone de navigation de gauche, sélectionnez **clusters**, puis **Ajouter un cluster** à partir de la page 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. Reportez-vous à la section "[Documentation Kubernetes](#)" pour plus d'informations sur la création `kubeconfig` fichiers. Si vous avez créé un `kubeconfig` pour un rôle de cluster limité à l'aide de [le processus ci-dessus](#), assurez-vous de télécharger ou de coller ce `kubeconfig` dans cette étape.

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



Vous devez sélectionner une classe de stockage Trident soutenue par le stockage ONTAP.

6. Passez en revue les informations, et si tout semble bien, sélectionnez **Ajouter**.

Résultat

Le cluster passe à l'état **découverte**, puis passe à **sain**. Vous gérez maintenant le cluster avec 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 ONTAP à Astra Control Center pour gérer ses ressources.

La gestion des clusters de stockage d'Astra Control en tant que backend de stockage vous permet d'obtenir des liens entre les volumes persistants (PVS) et le back-end de stockage, ainsi que des metrics de stockage supplémentaires.

Étapes

1. Dans la zone de navigation gauche du tableau de bord, sélectionnez **Backends**.
2. Effectuez l'une des opérations suivantes :
 - **Nouveaux systèmes back-end** : sélectionnez **Ajouter** pour gérer un back-end existant, sélectionnez **ONTAP**, puis **Suivant**.
 - **Backends découverts** : dans le menu actions, sélectionnez **Manage** sur un back-end découvert à partir du cluster géré.
3. Saisissez l'adresse IP de gestion du cluster ONTAP et les identifiants d'administrateur. Les identifiants doivent être identifiants au niveau du cluster.



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, appliquez les identifiants de l'utilisateur au rôle « admin », qui dispose des méthodes d'accès `ontapi` et `http`, Sur les clusters ONTAP source et destination. Reportez-vous à la section "[Gérer les comptes utilisateur dans la documentation ONTAP](#)" pour en savoir plus.

4. Sélectionnez **Suivant**.
5. Confirmez les détails du back-end et sélectionnez **gérer**.

Résultat

Le back-end s'affiche dans le `Healthy` état dans la liste avec des informations récapitulatives.



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

Ajouter un godet

Vous pouvez ajouter un compartiment à l'aide de l'interface utilisateur Astra Control ou "API". 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.

Si vous clonez la configuration de vos applications et le stockage persistant vers le même cluster, il n'est pas nécessaire d'utiliser un compartiment dans Astra Control. La fonctionnalité de copie Snapshot des applications ne nécessite pas de compartiment.

Ce dont vous avez besoin

- Compartiment accessible depuis vos clusters gérés par Astra Control Center.
- Identifiants pour le compartiment.
- Un godet des types suivants :
 - NetApp ONTAP S3
 - NetApp StorageGRID S3
 - Microsoft Azure
 - S3 générique



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



Bien qu'Astra Control Center prenne en charge Amazon S3 en tant que fournisseur de compartiments 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.

Étapes

1. Dans la zone de navigation de gauche, sélectionnez **godets**.
2. Sélectionnez **Ajouter**.
3. 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.

4. Saisissez 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 plus tard lors de la création d'une sauvegarde. Ce nom apparaît également lors de la configuration de la règle de protection.

5. Entrez le nom ou l'adresse IP du terminal S3.
6. Sous **Sélectionner les informations d'identification**, choisissez l'onglet **Ajouter** ou **utiliser l'onglet existant**.
 - Si vous avez choisi **Ajouter**:
 - i. Saisissez un nom pour l'identifiant qui le distingue des autres identifiants dans Astra Control.
 - ii. Saisissez l'ID d'accès et la clé secrète en collant le contenu dans le presse-papiers.
 - Si vous avez choisi **utiliser existant**:
 - i. Sélectionnez les informations d'identification existantes à utiliser avec le compartiment.

7. Sélectionnez Add.



Lorsque vous ajoutez un godet, Astra Control marque un godet avec l'indicateur de compartiment par défaut. Le premier compartiment que vous créez devient le compartiment par défaut. Au fur et à mesure que vous ajoutez des compartiments, vous pourrez décider plus tard "[définir un autre compartiment par défaut](#)".

Et la suite ?

Maintenant que vous êtes connecté et que vous avez ajouté des clusters à Astra Control Center, vous êtes prêt à utiliser les fonctionnalités de gestion des données applicatives d'Astra Control Center.

- "[Gérez les utilisateurs et les rôles locaux](#)"
- "[Commencez à gérer les applications](#)"
- "[Protégez vos applications](#)"
- "[Gérer les notifications](#)"
- "[Connectez-vous à Cloud Insights](#)"
- "[Ajouter un certificat TLS personnalisé](#)"
- "[Modifiez la classe de stockage par défaut](#)"

Trouvez plus d'informations

- "[Utilisez l'API de contrôle Astra](#)"
- "[Problèmes connus](#)"

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` du fichier de ressource personnalisée `astra_control_Center.yaml` 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.yaml` CR.

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

1. Dans le menu de navigation de gauche, sélectionnez **compte > Licence**.
2. Sélectionnez **Ajouter licence**.
3. 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 dégrèrer correctement un cluster?

1. "[Gérez les applications avec Astra Control](#)".
2. "[Dégrèrer 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 dégèrez 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

Mon application utilise plusieurs PVS. ASTRA Control prendra-t-il des snapshots et des sauvegardes de ces volumes persistants ?

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.

Informations sur le copyright

Copyright © 2024 NetApp, Inc. Tous droits réservés. Imprimé aux États-Unis. Aucune partie de ce document protégé par copyright ne peut être reproduite sous quelque forme que ce soit ou selon quelque méthode que ce soit (graphique, électronique ou mécanique, notamment par photocopie, enregistrement ou stockage dans un système de récupération électronique) sans l'autorisation écrite préalable du détenteur du droit de copyright.

Les logiciels dérivés des éléments NetApp protégés par copyright sont soumis à la licence et à l'avis de non-responsabilité suivants :

CE LOGICIEL EST FOURNI PAR NETAPP « EN L'ÉTAT » ET SANS GARANTIES EXPRESSES OU TACITES, Y COMPRIS LES GARANTIES TACITES DE QUALITÉ MARCHANDE ET D'ADÉQUATION À UN USAGE PARTICULIER, QUI SONT EXCLUES PAR LES PRÉSENTES. EN AUCUN CAS NETAPP NE SERA TENU POUR RESPONSABLE DE DOMMAGES DIRECTS, INDIRECTS, ACCESSOIRES, PARTICULIERS OU EXEMPLAIRES (Y COMPRIS L'ACHAT DE BIENS ET DE SERVICES DE SUBSTITUTION, LA PERTE DE JOUISSANCE, DE DONNÉES OU DE PROFITS, OU L'INTERRUPTION D'ACTIVITÉ), QUELLES QU'EN SOIENT LA CAUSE ET LA DOCTRINE DE RESPONSABILITÉ, QU'IL S'AGISSE DE RESPONSABILITÉ CONTRACTUELLE, STRICTE OU DÉLICTEUELLE (Y COMPRIS LA NÉGLIGENCE OU AUTRE) DÉCOULANT DE L'UTILISATION DE CE LOGICIEL, MÊME SI LA SOCIÉTÉ A ÉTÉ INFORMÉE DE LA POSSIBILITÉ DE TELS DOMMAGES.

NetApp se réserve le droit de modifier les produits décrits dans le présent document à tout moment et sans préavis. NetApp décline toute responsabilité découlant de l'utilisation des produits décrits dans le présent document, sauf accord explicite écrit de NetApp. L'utilisation ou l'achat de ce produit ne concède pas de licence dans le cadre de droits de brevet, de droits de marque commerciale ou de tout autre droit de propriété intellectuelle de NetApp.

Le produit décrit dans ce manuel peut être protégé par un ou plusieurs brevets américains, étrangers ou par une demande en attente.

LÉGENDE DE RESTRICTION DES DROITS : L'utilisation, la duplication ou la divulgation par le gouvernement sont sujettes aux restrictions énoncées dans le sous-paragraphe (b)(3) de la clause Rights in Technical Data-Noncommercial Items du DFARS 252.227-7013 (février 2014) et du FAR 52.227-19 (décembre 2007).

Les données contenues dans les présentes se rapportent à un produit et/ou service commercial (tel que défini par la clause FAR 2.101). Il s'agit de données propriétaires de NetApp, Inc. Toutes les données techniques et tous les logiciels fournis par NetApp en vertu du présent Accord sont à caractère commercial et ont été exclusivement développés à l'aide de fonds privés. Le gouvernement des États-Unis dispose d'une licence limitée irrévocable, non exclusive, non cessible, non transférable et mondiale. Cette licence lui permet d'utiliser uniquement les données relatives au contrat du gouvernement des États-Unis d'après lequel les données lui ont été fournies ou celles qui sont nécessaires à son exécution. Sauf dispositions contraires énoncées dans les présentes, l'utilisation, la divulgation, la reproduction, la modification, l'exécution, l'affichage des données sont interdits sans avoir obtenu le consentement écrit préalable de NetApp, Inc. Les droits de licences du Département de la Défense du gouvernement des États-Unis se limitent aux droits identifiés par la clause 252.227-7015(b) du DFARS (février 2014).

Informations sur les marques commerciales

NETAPP, le logo NETAPP et les marques citées sur le site <http://www.netapp.com/TM> sont des marques déposées ou des marques commerciales de NetApp, Inc. Les autres noms de marques et de produits sont des marques commerciales de leurs propriétaires respectifs.