paloalto TECHDOCS

Déployer le pare-feu CN-Series dans le cloud et sur site

docs.paloaltonetworks.com

Contact Information

Corporate Headquarters:
Palo Alto Networks
3000 Tannery Way
Santa Clara, CA 95054
www.paloaltonetworks.com/company/contact-support

About the Documentation

- For the most recent version of this guide or for access to related documentation, visit the Technical Documentation portal docs.paloaltonetworks.com.
- To search for a specific topic, go to our search page docs.paloaltonetworks.com/search.html.
- Have feedback or questions for us? Leave a comment on any page in the portal, or write to us at documentation@paloaltonetworks.com.

Copyright

Last Revised

December 13, 2021

Table of Contents

Déployer le pare-feu CN-Series dans GKE.....5

Déploiement du pare-feu CN-Series en tant que service Kubernetes dans GKE.....6

Déploiement du pare-feu CN-Series en tant que DaemonSet dans GKE.....19

Déployer le pare-feu CN-Series dans OKE.....31

Déployer le pare-feu CN-Series en tant que service Kubernetes dans OKE.....33

Déployer le pare-feu CN-Series en tant que DaemonSet dans OKE..... 45

Déployer le pare-feu CN-Series dans EKS.....57

Déploiement du pare-feu CN-Series en tant que service Kubernetes dans AWS EKS..... 58

Déploiement du pare-feu CN-Series en tant que DaemonSet dans AWS EKS.....67

Déployer le pare-feu CN-Series à partir d'AWS Marketplace.....76

Déployer le pare-feu CN-Series en tant que service Kubernetes sur AliCloud (ACK)..... 85

Déployer CN-Series sur OpenShift.....107

Déployer CN-Series sur le hub de l'opérateur OpenShift..... 109

Déployer le pare-feu CN-Series dans GKE

Où puis-je utiliser ceci ?	De quoi ai-je besoin ?
• Déploiement CN-Series	• CN-Series 10.1.x or above Container Images • Panorama sous PAN-OS 10.1.x ou une version supérieure • Helm 3.6 or above version client pour le déploiement CN-Series à l'aide de Helm

Après avoir examiné les blocs de construction CN-Series et la présentation générale du flux de travail dans Sécuriser les environnements Kubernetes avec CN-Series , vous pouvez commencer à déployer les pare-feu CN-Series sur la plate-forme GKE pour sécuriser le trafic entre les conteneurs au sein du même cluster, ainsi qu'entre les conteneurs et d'autres types de charges de travail tels que les machines virtuelles et les serveurs bare-metal.

Vous avez besoin d'outils Kubernetes standard tels que kubectl ou Helm pour déployer et gérer vos applications, vos services pare-feu et vos clusters Kubernetes.

Pour plus d'informations, consultez Déployer des pare-feu CN-Series avec des graphiques et des modèles Helm. *Panorama n'est pas conçu pour être utilisé comme orchestrateur pour le déploiement et la gestion de clusters Kubernetes. Les modèles pour la gestion des clusters sont fournis par les fournisseurs de Kubernetes gérés. Palo Alto Networks fournit des modèles pris en charge par la communauté pour le déploiement CN-Series avec* Helm *et* Terraform.

- Déploiement du pare-feu CN-Series en tant que service Kubernetes dans GKE
- Déploiement du pare-feu CN-Series en tant que DaemonSet dans GKE

Avant de passer du déploiement de CN-Series en tant que DaemonSet à CN-Series en tant que service ou vice versa, vous devez supprimer et réappliquer `plugin-serviceaccount.yaml`. *Pour plus d'informations, consultez* Créer des comptes de service pour l'authentification des clusters.

- *Lorsque vous déployez CN-Series en tant que DaemonSet dans GKE, le* `pan-plugin-cluster-mode-secret` *ne doit pas exister.*
- *Lorsque vous déployez CN-Series en tant que service Kubernetes dans GKE, le* `pan-plugin-cluster-mode-secret` *doit être présent.*

Déploiement du pare-feu CN-Series en tant que service Kubernetes dans GKE

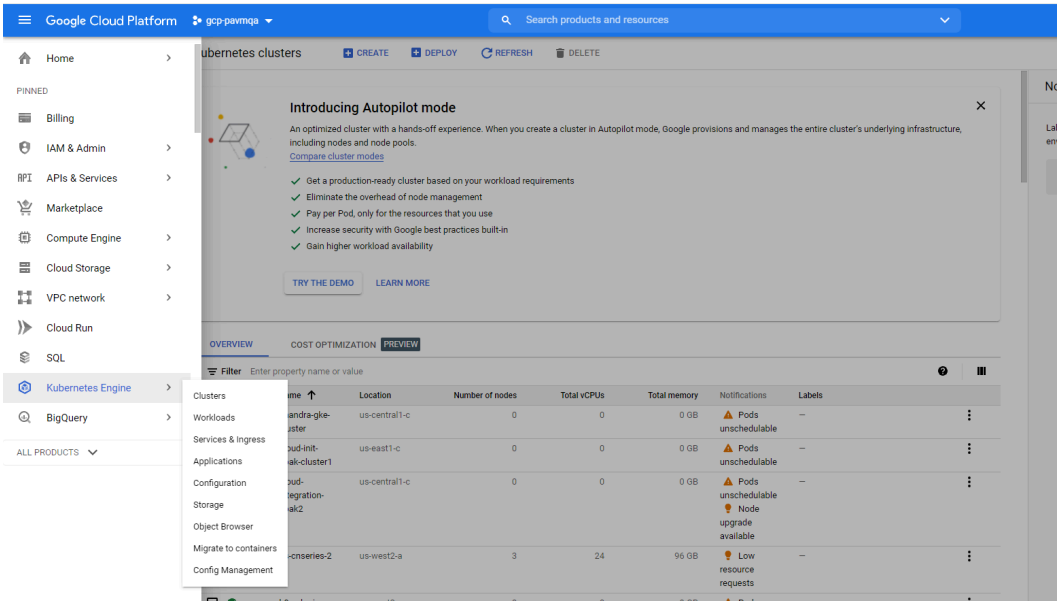
Où puis-je utiliser ceci ?	De quoi ai-je besoin ?
• Déploiement CN-Series	• CN-Series 10.1.x or above Container Images • Panorama sous PAN-OS 10.1.x ou une version supérieure • Helm 3.6 or above version client pour le déploiement CN-Series à l’aide de Helm

Effectuez la procédure suivante pour déployer le pare-feu CN-Series en tant que service Kubernetes.

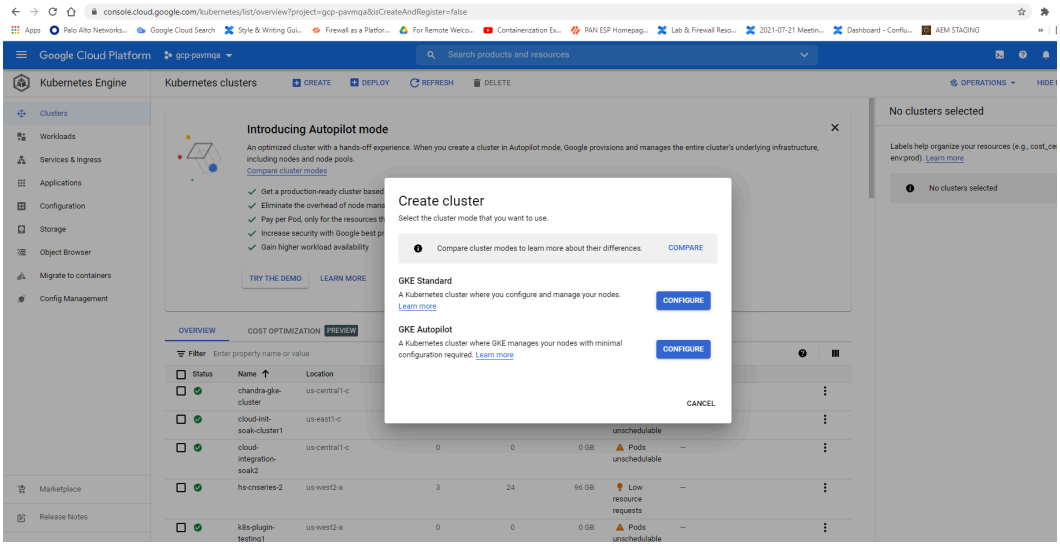
STEP 1 | Configurez votre cluster Kubernetes.

Pour créer un cluster dans GKE, procédez comme suit :

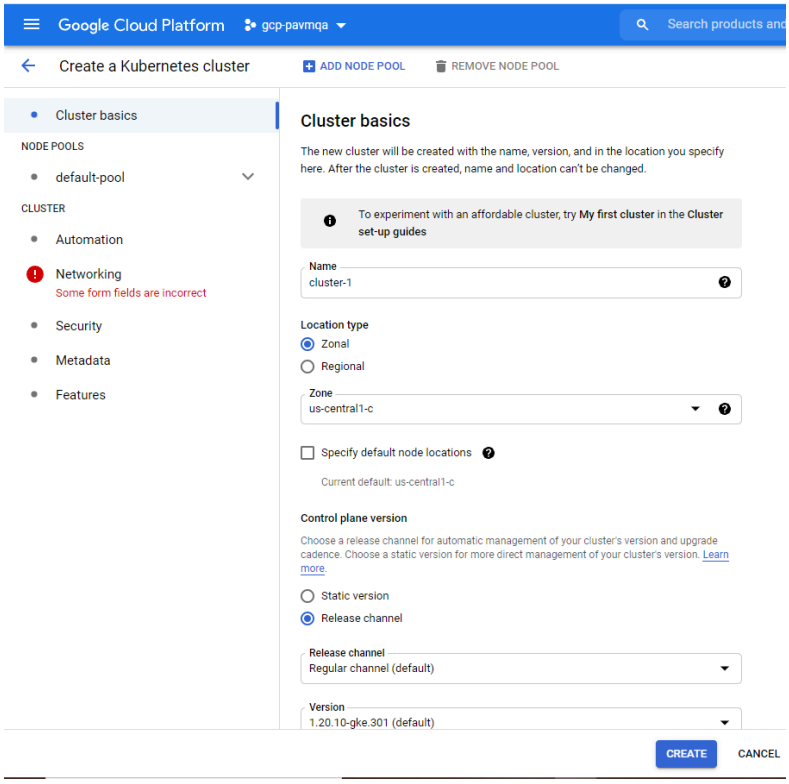
1. Cliquez sur le menu de navigation, accédez à **Kubernetes Engine (Moteur Kubernetes)**, puis sélectionnez **clusters**.



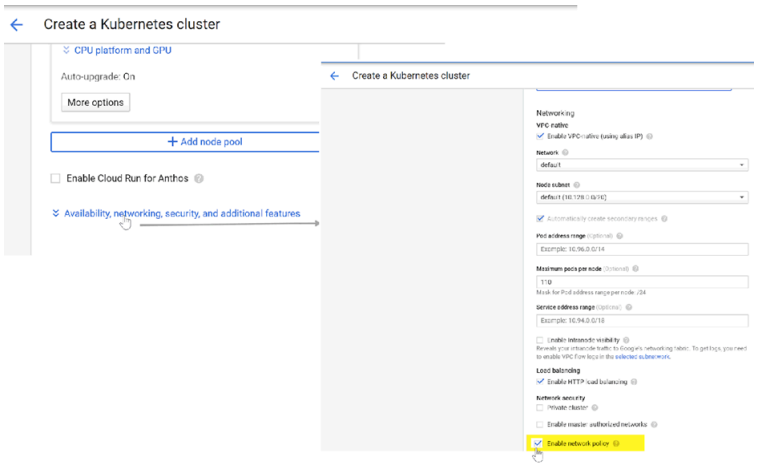
2. Cliquez sur **Create (Créer)**.
3. Sélectionnez la **GKE Standard (Norme GKE)** comme mode de cluster que vous souhaitez utiliser, puis cliquez sur **Configure (Configurer)**.



4. Saisissez les informations de base du cluster, notamment Nom, Version, Emplacement, Sous-réseau de nœud, puis cliquez sur **Create (Créer)**.



Si votre cluster se trouve sur GKE, assurez-vous d'activer l'API Network Policy de Kubernetes pour permettre à l'administrateur du cluster d'indiquer quels pods sont autorisés à communiquer entre eux. Cette API est requise pour permettre aux pods CN-NGFW et CN-MGMT de communiquer.



1. Vérifiez que le cluster dispose des ressources adéquates. La spécification par défaut du pool de nœuds GKE n'est pas adaptée au pare-feu CN-Series. Vous devez vous assurer que ce cluster dispose des conditions préalables de CN-Series pour prendre en charge le pare-feu :

   ```
   kubectl get nodes
   ```

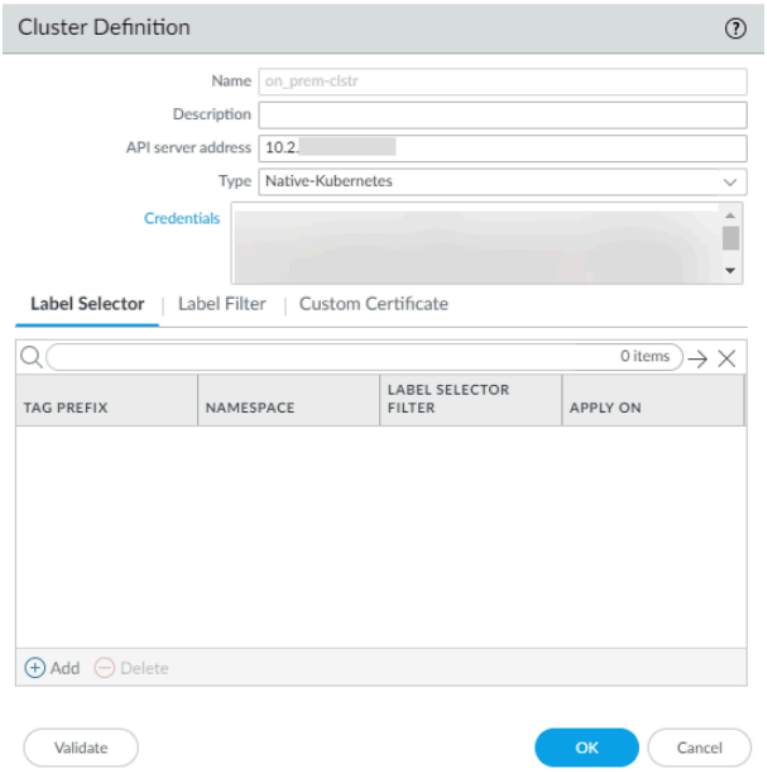
   ```
   kubectl describe node <node-name>
   ```


Affichez les informations sous l'en-tête Capacity (Capacité) dans la sortie de la commande pour voir le processeur et la mémoire disponibles sur le nœud spécifié.

L'allocation du processeur, de la mémoire et du stockage sur disque dépendra de vos besoins. Voir Performances et mise à l'échelle de CN-Series.

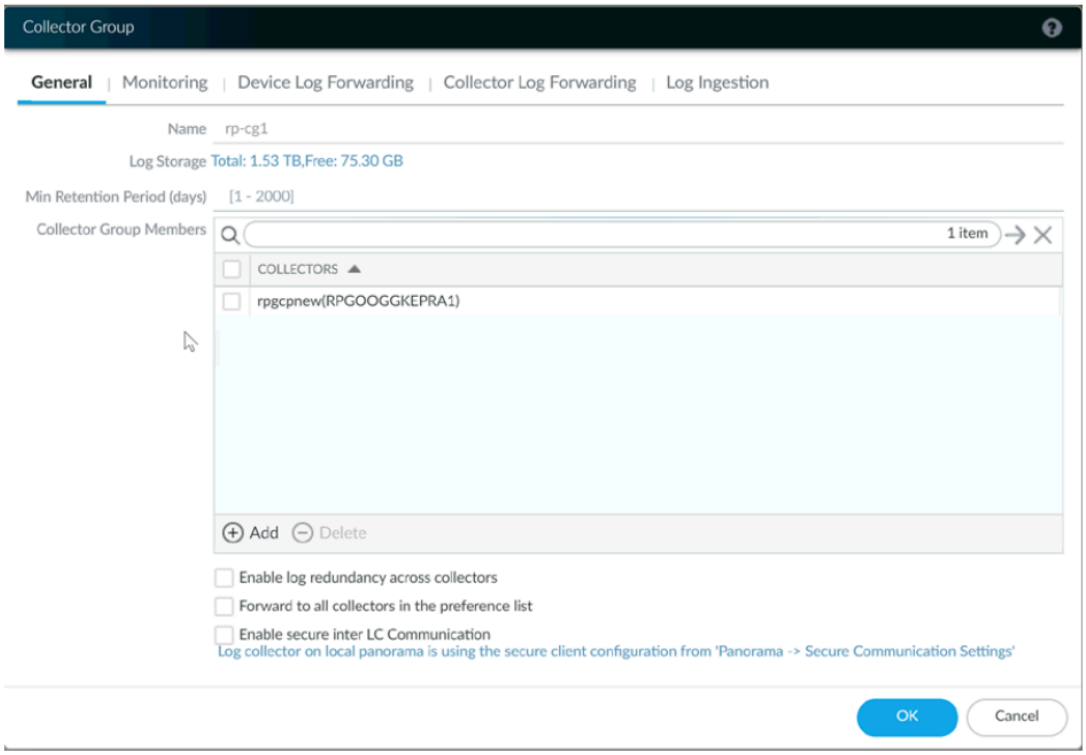
Assurez-vous d'avoir les informations suivantes :

- Collectez l'adresse IP du terminal pour configurer le serveur API sur Panorama.



  Panorama utilise cette adresse IP pour se connecter à votre cluster Kubernetes.

- Collectez le nom de la pile de modèles, le nom du groupe d'appareils, l'adresse IP Panorama et éventuellement le nom du groupe de collecteurs de journaux à partir de Panorama.



Pour plus d'informations, consultez Créer un groupe d'appareils parents et une pile de modèles.

- Collectez le code d'autorisation et l'ID et la valeur du code PIN d'enregistrement automatique.
- L'emplacement du conteneur d'images dans lequel vous avez téléchargé les images.

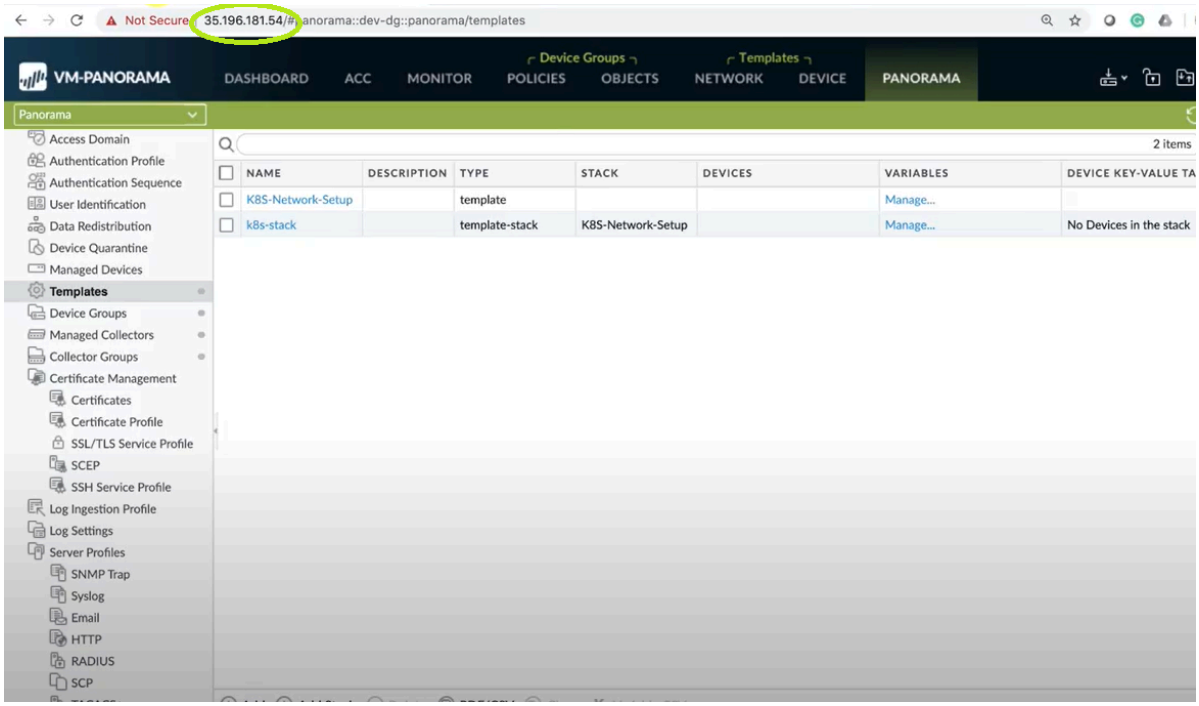
STEP 2 | (facultatif) Si vous avez configuré un certificat personnalisé dans le plug-in Kubernetes pour Panorama, vous devez créer le secret de certificat en exécutant la commande suivante. Ne modifiez pas le nom de fichier de ca.crt. Le volume des certificats personnalisés dans pan-cn-mgmt.yaml et pan-cn-ngfw.yaml est facultatif.

```
kubectl -n kube-system crée un secret générique custom-ca --from-file=ca.crt
```

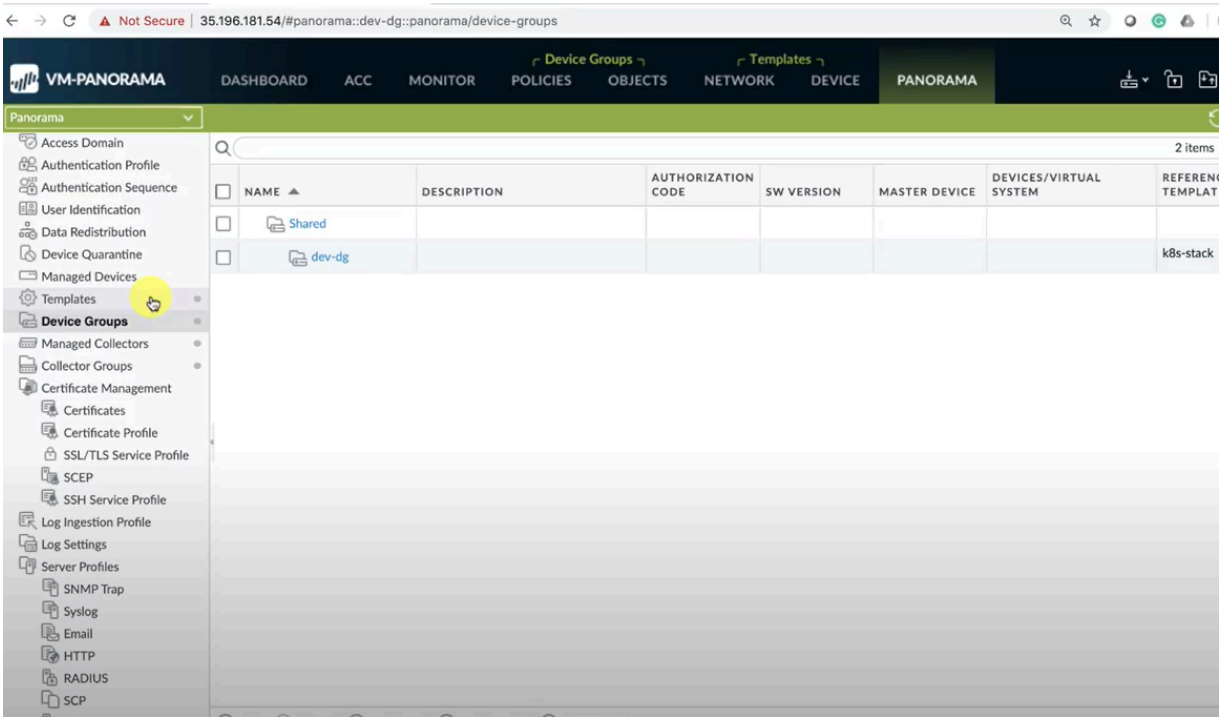

STEP 3 | Modifiez les fichiers YAML afin de fournir les détails nécessaires au déploiement des pare-feu CN-Series.

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: pan-mgmt-config
  namespace: kube-system
data:
  PAN_OPERATION_MODE: "daemonset"
  PAN_SERVICE_NAME: "pan-mgmt"
  # Panorama settings
  PAN_PANORAMA_IP: "35.196.181.54"
  PAN_PANORAMA_AUTH_KEY: [redacted]
  PAN_DEVICE_GROUP: "dev-dg"
  PAN_TEMPLATE: "k8s-stack"
#Non-mandatory parameters
  PAN_PANORAMA_CGNAME: "rp-cg1"
  #PAN_CERTIFICATE: ""
  #PAN_CERTKEYFILE: ""
  #PAN_CERTPASSPHRASE: ""
```

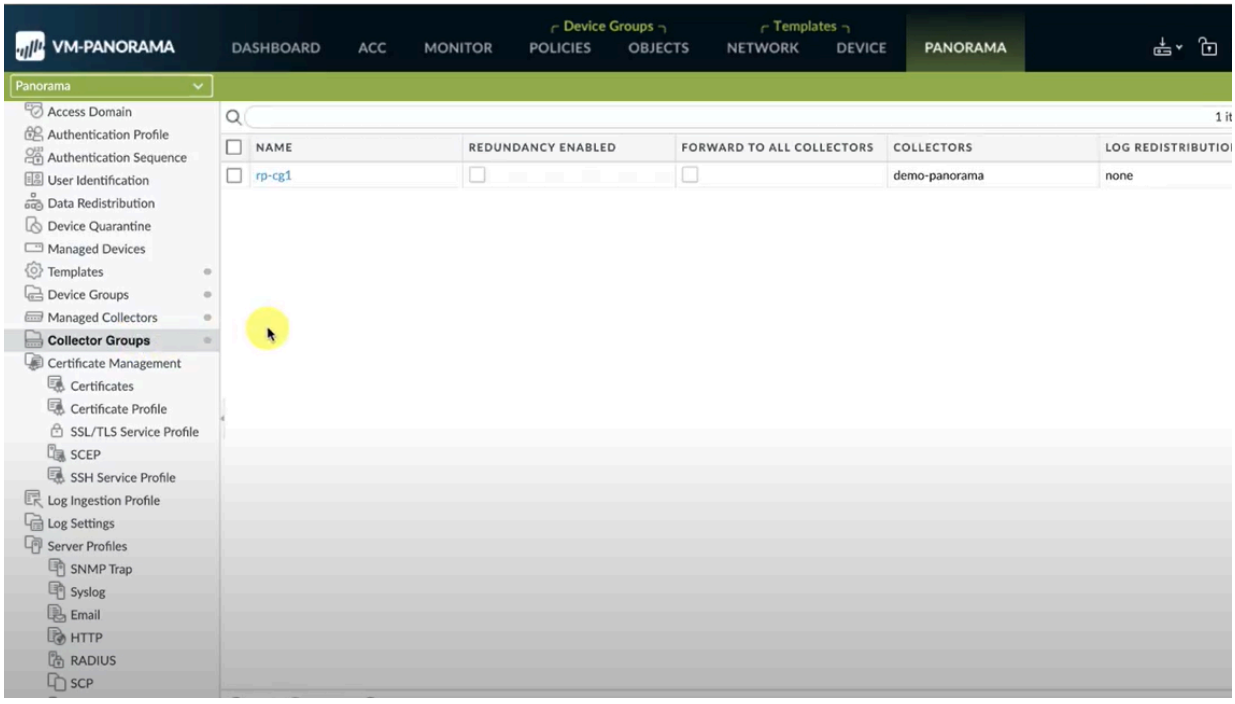
Vous devez vous assurer que la valeur du paramètre `PAN_PANORAMA_IP` sur votre fichier YAML correspond à votre adresse IP Panorama réelle, comme indiqué dans le diagramme ci-dessous :



Vous devez vous assurer que la valeur des paramètres de `PAN_DEVICE_GROUP` et `PAN_TEMPLATE` sur votre fichier YAML correspond au nom du groupe d'appareils et de la pile de modèles que vous avez créés sur Panorama, comme indiqué dans le diagramme ci-dessous :



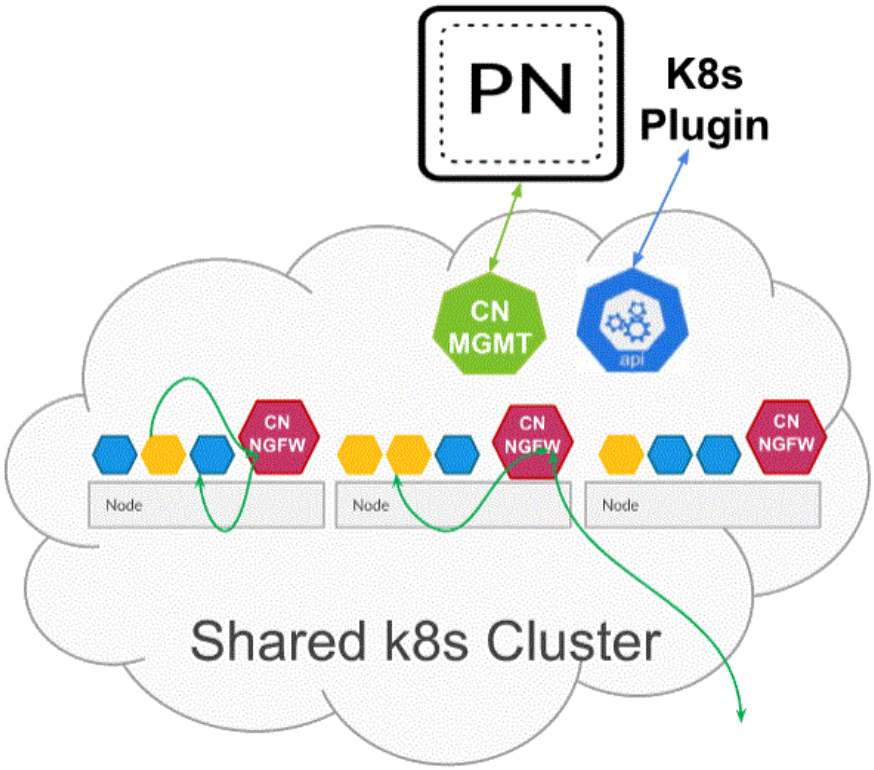
Vous devez vous assurer que la valeur du paramètre `PAN_PANORAMA_CG_NAME` est identique au nom du collecteur de journaux que vous avez créé.



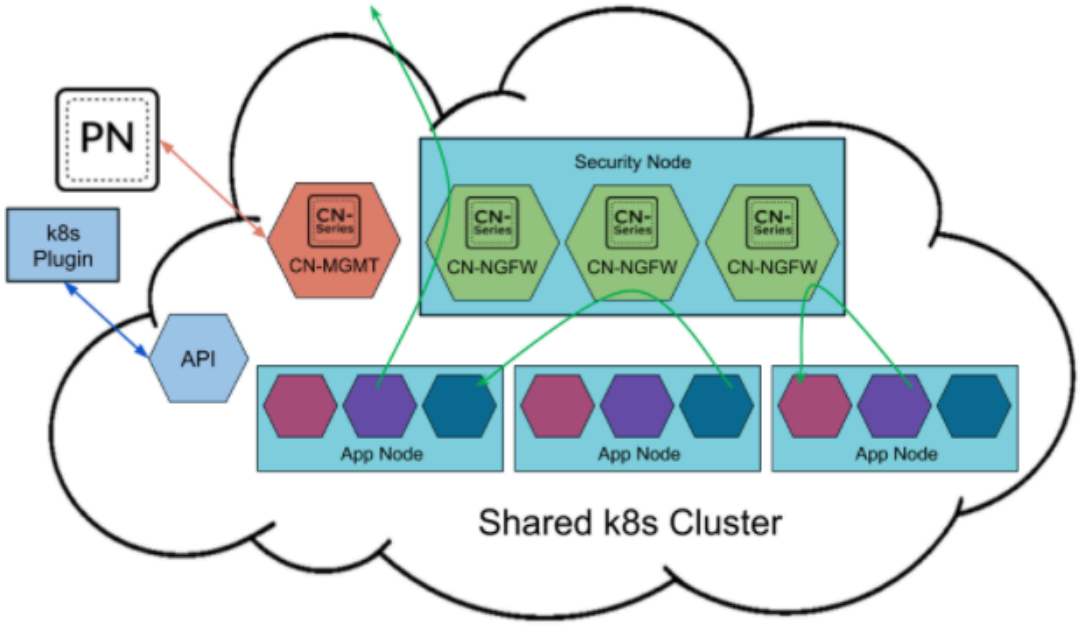
Pour plus d'informations, consultez Paramètres modifiables dans les fichiers yaml de déploiement CN-Series.

STEP 4 | Si vous utilisez la mise à l'échelle automatique dans votre environnement Kubernetes, consultez Activer la mise à l'échelle horizontale du pod.

STEP 5 | Déployez le service CN-NGFW. Effectuez les étapes suivantes :



Lorsqu'elles sont déployées en tant que service Kubernetes, les instances du pod CN-NGFW peuvent être déployées sur des nœuds de sécurité et le trafic du pod d'application est redirigé vers une instance CN-NGFW disponible pour inspection et application.



1. Vérifiez que vous avez créé le compte de service à l'aide du fichier pan-cni-serviceaccount.yaml.

   Consultez Création de comptes de service pour l'authentification des clusters.
2. Utilisez Kubectl pour exécuter le fichier pan-cni-configmap.yaml.


```
kubectl apply -f pan-cni-configmap.yaml
```

3. Utilisez kubectl pour exécuter le fichier pan-cn-ngfw-svc.yaml.

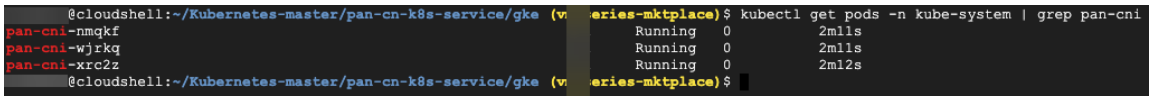
```
kubectl apply -f pan-cn-ngfw-svc.yaml
```

Ce fichier yaml doit être déployé avant pan-cni.yaml.

4. Utilisez Kubectl pour exécuter le fichier pan-cni.yaml.

```
kubectl apply -f pan-cni.yaml
```

5. Vérifiez que vous avez modifié les fichiers YAML pan-cni et pan-cni-configmap.
6. Exécutez la commande suivante et vérifiez que votre sortie est similaire à l'exemple suivant.



STEP 6 | Déployez le StatefulSet CN-MGMT.

Par défaut, le plan de gestion est déployé comme un StatefulSet qui garantie la tolérance aux pannes. Jusqu'à 30 pods pare-feu CN-NGFW peuvent se connecter à un StatefulSet CN-MGMT.

1. (Requis uniquement pour les PV provisionnés statiquement) Déployez les volumes persistants (PV) pour le StatefulSet CN-MGMT.
   1. Créez les répertoires qui correspondent aux noms des volumes locaux définis dans le fichier pan-cn-pv-local.yaml.

      Vous avez besoin de six (6) répertoires sur au moins 2 nœuds esclaves. Connectez-vous à chaque nœud esclave sur lequel le StatefulSet CN-MGMT sera déployé pour créer les répertoires. Par exemple, pour créer des répertoires nommés /mnt/pan-local1 vers /mnt/pan-local6, utilisez la commande :

      ```
      mkdir -p /mnt/pan-local1 /mnt/pan-local2 /mnt/pan-local3 /
      mnt/pan-local4 /mnt/pan-local5 /mnt/pan-local6
      ```

   2. Modifiez pan-cn-pv-local.yaml.

      Faites correspondre le nom d'hôte sous `nodeaffinity`, et vérifiez que vous avez modifié les répertoires que vous avez créés ci-dessus dans `spec.local.path` puis déployez le fichier pour créer une nouvelle storage class pan-local-storage et des PV locaux.
2. Vérifiez que vous avez modifié les fichiers YAML pan-cn-mgmt et pan-cn-mgmt-configmap

   Exemple de pan-cn-mgmt-configmap de l'EKS.

   ```
   apiVersion: v1 kind: ConfigMap metadata: name: pan-mgmt-
   config namespace: kube-system data: PAN_SERVICE_NAME:
   pan-mgmt-svc PAN_MGMT_SECRET: pan-mgmt-secret # Panorama
   settings PAN_PANORAMA_IP: "<panorama-IP>" PAN_DEVICE_GROUP:
   "<panorama-device-group>" PAN_TEMPLATE_STACK: "<panorama-
   template-stack>" PAN_CGNAME: "<panorama-collector-
   ```



```
group>" # ctnr mode: "k8s-service", "k8s-ilbservice"
 PAN_CTNR_MODE_TYPE: "k8s-service" #Non-mandatory parameters #
 Recommended to have same name as the cluster name provided in
 Panorama Kubernetes plugin - helps with easier identification
 of pods if managing multiple clusters with same Panorama
 #CLUSTER_NAME: "<Cluster name>" #PAN_PANORAMA_IP2: "" #
 Comment out to use CERTs otherwise PSK for IPSec between pan-
mgmt and pan-ngfw #IPSEC_CERT_BYPASS: "" # No values needed
 # Override auto-detect of jumbo-frame mode and force enable
 system-wide #PAN_JUMBO_FRAME_ENABLED: "true" # Start MGMT
 pod with GTP enabled. For complete functionality, need GTP #
 enable at Panorama as well. #PAN_GTP_ENABLED: "true" # Enable
 high feature capacities. These need high memory for MGMT pod
 and # higher/matching memory than specified below for NGFW
 pod. #PAN_NGFW_MEMORY="6Gi" #PAN_NGFW_MEMORY="40Gi" # For
 enabling faster datapath - AF_XDP, default is AF_PACKETV2.
 This requires kernel support. #PAN_DATA_MODE: "next-gen" #HPA
 params #PAN_CLOUD: "EKS" #PAN_NAMESPACE_EKS: "EKSNamespace"
 #PUSH_INTERVAL: "15" #time interval to publish metrics to AWS
 cloudwatch
```

Exemple de pan-cn-mgmt.yaml

```
initContainers: - name: pan-mgmt-init image: <your-private-
registry-image-path>
```

```
 containers: - name: pan-mgmt image: <your-private-registry-
image-path> terminationMessagePolicy: FallbackToLogsOnError
```

3. Utilisez Kubectl pour exécuter les fichiers yaml.

   kubectl apply -f pan-cn-mgmt-configmap.yaml

   kubectl apply -f pan-cn-mgmt-slot-crd.yaml

   kubectl apply -f pan-cn-mgmt-slot-cr.yaml

   kubectl apply -f pan-cn-mgmt-secret.yaml

   kubectl apply -f pan-cn-mgmt.yaml

   Vous devez exécuter pan-mgmt-serviceaccount.yaml, uniquement si vous n’aviez pas déjà terminé la création de compte de service pour l’authentification de cluster.

4. Vérifiez que les pods CN-MGMT sont opérationnels en exécutant la commande suivante :

   kubectl get pods -l app=pan-mgmt -n kube-system

   Cela prend environ 5-6 minutes.

STEP 7 | Déployez les pods CN-NGFW.

1. Vérifiez que vous avez modifié les fichiers YAML comme indiqué dans PAN-CN-NGFW-CONFIGMAP et PAN-CN-NGFW.

   containers: - name: pan-ngfw-container image: <your-private-registry-image-path>

2. Utilisez l'application Kubectl pour exécuter le fichier pan-cn-ngfw-configmap.yaml.

   kubectl apply -f pan-cn-ngfw-configmap.yaml

3. Utilisez l'application Kubectl pour exécuter le pan-cn-ngfw.yaml.

   kubectl apply -f pan-cn-ngfw.yaml

4. Vérifiez que les pods CN-NGFW sont en cours d'exécution.

   kubectl get pods -n kube-system -l app=pan-ngfw -o wide

STEP 8 | Activez l'autoscaling horizontal des pods en procédant comme suit :

1. Déployez l'adaptateur de pilote de pile de mesures personnalisées dans votre cluster CN-Series. Le nom du cluster doit être fourni via un secret K8s.
2. Téléchargez les fichiers yaml HPA spécifiques à GKE à partir du référentiel GitHub de Palo Alto Networks.
3. Si votre CN-MGMT est déployé dans un espace de noms personnalisé, mettez à jour pan-cn-adapater.yaml avec l'espace de noms personnalisé. L'espace de noms par défaut est **kube-system**.
4. Mettez à jour les paramètres HPA dans le fichier pan-cn-mgmt-configmap.yaml spécifique à GKE.

   ```
   #PAN_CLOUD: "GKE"
   ```

   ```
   #HPA_NAME: "<name>" #nom unique pour identifier la
    ressource hpa par espace de noms ou par locataire
   ```

   ```
   #PUSH_INTERVAL: "15" #intervalle de temps pour publier les
    métriques sur stackdriver
   ```

5. Modifiez les fichiers **pan-cn-hpa-dp.yaml** et **pan-cn-hpa-mp.yaml** avec HPA_NAME (remplacez par le nom) tel que mis à jour dans le fichier pan-cn-mgmt-configmap.yaml ci-dessus et mettez à jour la métrique en fonction de quel HPA doit être déclenché.

   1. Entrez le nombre minimal et maximal de réplicas.
   2. (Facultatif) Modifiez les valeurs de fréquence de mise à l'échelle et de montée en puissance en fonction de votre déploiement. Si vous ne modifiez pas ces valeurs, les valeurs par défaut sont utilisées.
   3. (Facultatif) Modifiez la valeur de seuil pour chaque métrique que vous souhaitez utiliser pour la mise à l'échelle. Si vous ne modifiez pas ces valeurs, les valeurs par défaut sont utilisées.

4. Enregistrez les modifications.

6. Déployez les fichiers yaml HPA. Les fichiers doivent être déployés dans l'ordre décrit ci-dessous.

   1. Utiliser Kubectl pour exécuter le fichier pan-cn-adapter.yaml

      ```
      kubectl apply -f pan-cn-adapter.yaml
      ```

   2. Utiliser Kubectl pour exécuter le fichier pan-cn-crole.yaml

      ```
      kubectl apply -f pan-cn-crole.yaml
      ```

   3. Utiliser Kubectl pour exécuter le fichier pan-cn-hpa-dp.yaml

      ```
      kubectl apply -f pan-cn-hpa-dp.yaml
      ```

   4. Utiliser Kubectl pour exécuter le fichier pan-cn-hpa-mp.yaml

      ```
      kubectl apply -f pan-cn-hpa-mp.yaml
      ```

7. Vérifiez votre déploiement.

   - Utilisez kubectl pour vérifier que le pod d'adaptateur de mesures personnalisées dans l'espace de noms de mesures personnalisées.

     ```
     kubectl get pods -n custom-metrics
     ```

   - Utilisez kubectl pour rechercher la ressource HPA.

     ```
     kubectl get hpa -n kube-system
     ```

     ```
     kubectl describe hpa <hpa-name> -n kube-system
     ```

Pour plus d'informations, consultez Activer la mise à l'échelle automatique horizontale du pod sur CN-Series.

STEP 9 | Vérifiez que vous pouvez voir CN-MGMT, le CN-NGFW et le PAN-CNI sur le cluster Kubernetes.

```
kubectl -n kube-system get pods
```

STEP 10 | Annotez l'application yaml ou l'espace de noms afin que le trafic de leurs nouveaux pods soit redirigé vers le pare-feu.

Vous devez ajouter l'annotation suivante pour rediriger le trafic vers le CN-NGFW pour inspection :

```
annotations: paloaltonetworks.com/firewall: pan-fw
```

Par exemple, pour tous les nouveaux pods dans l'espace de noms « par défaut :

```
kubectl annotate namespace default paloaltonetworks.com/
firewall=pan-fw
```

Sur certaines plateformes, les pods de l'application peuvent démarrer lorsque le pan-cni n'est pas actif dans la chaîne de plug-ins CNI. Pour éviter de tels scénarios, vous devez spécifier les volumes comme indiqué ici dans le pod d'application YAML.

```
volumes: - name: pan-cni-ready hostPath: path: /var/log/
pan-appinfo/pan-cni-ready type: Répertoire
```


STEP 11 | Déployez votre application dans le cluster.

Déploiement du pare-feu CN-Series en tant que DaemonSet dans GKE

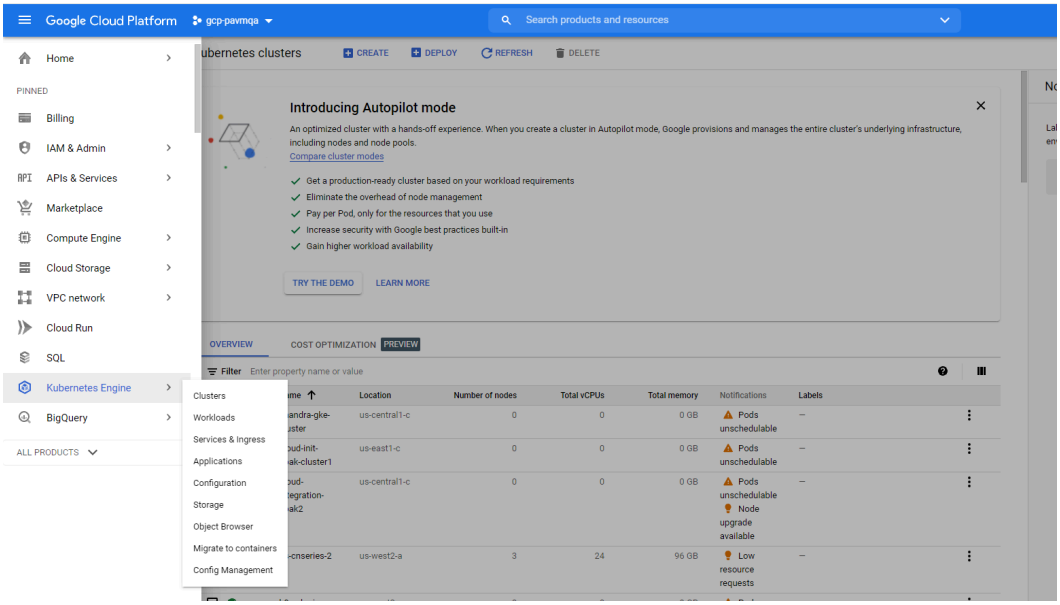
Où puis-je utiliser ceci ?	De quoi ai-je besoin ?
• Déploiement CN-Series	• CN-Series 10.1.x or above Container Images • Panorama sous PAN-OS 10.1.x ou une version supérieure • Helm 3.6 or above version client pour le déploiement CN-Series à l’aide de Helm

Effectuez la procédure suivante pour déployer le pare-feu CN-Series en tant que Daemonset dans la plateforme GKE :

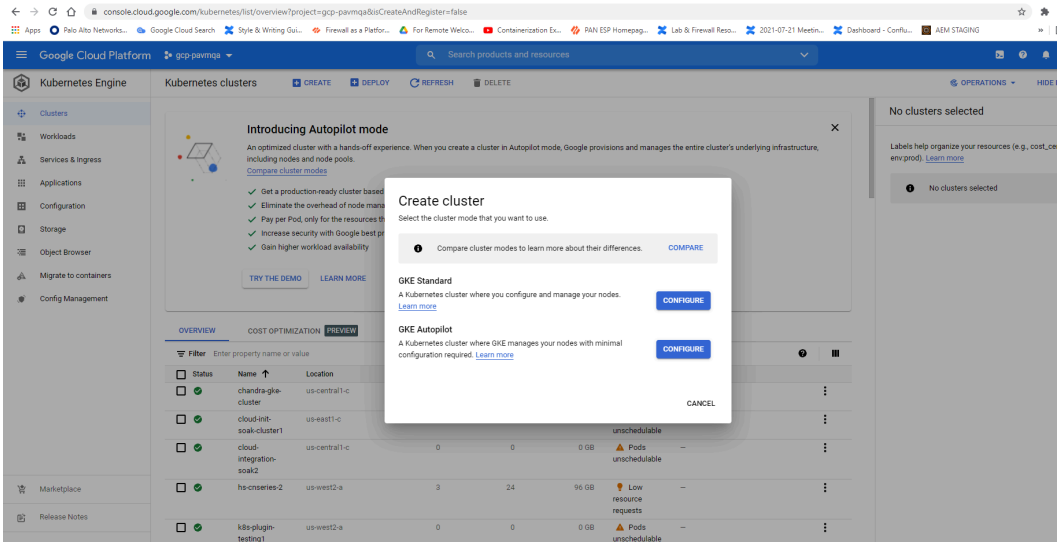
STEP 1 | Configurez votre cluster Kubernetes.

Pour créer un cluster dans GKE, procédez comme suit :

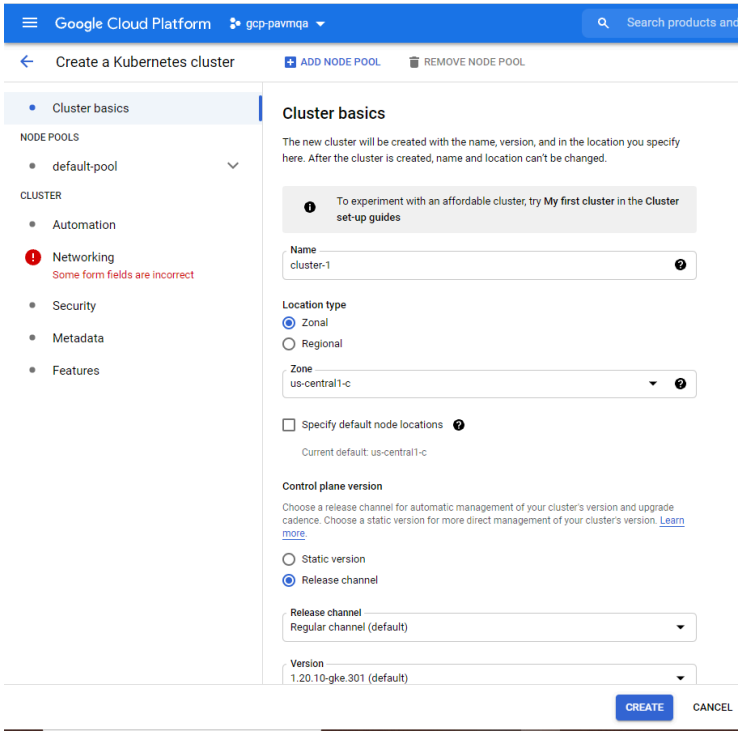
1. Cliquez sur le menu de navigation, accédez à **Kubernetes Engine (Moteur Kubernetes)**, puis sélectionnez **clusters**.



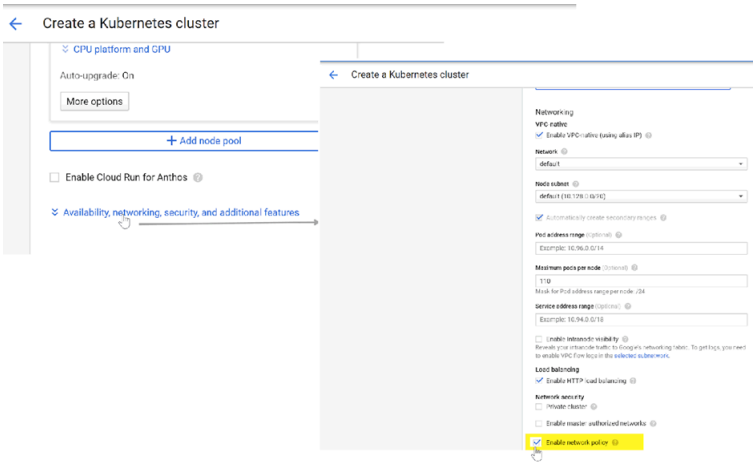
2. Cliquez sur **Create (Créer)**.
3. Sélectionnez la **GKE Standard (Norme GKE)** comme mode de cluster que vous souhaitez utiliser, puis cliquez sur **Configure (Configurer)**.



4. Saisissez les informations de base du cluster, notamment Nom, Version, Emplacement, Sous-réseau de nœud, puis cliquez sur **Create (Créer)**.



Si votre cluster se trouve sur GKE, assurez-vous d'activer l'API Network Policy de Kubernetes pour permettre à l'administrateur du cluster d'indiquer quels pods sont autorisés à communiquer entre eux. Cette API est requise pour permettre aux pods CN-NGFW et CN-MGMT de communiquer.



Vérifiez que le cluster dispose des ressources adéquates. Assurez-vous que ce cluster dispose des exigences système de CN-Series pour prendre en charge le pare-feu.

`kubectl get nodes`

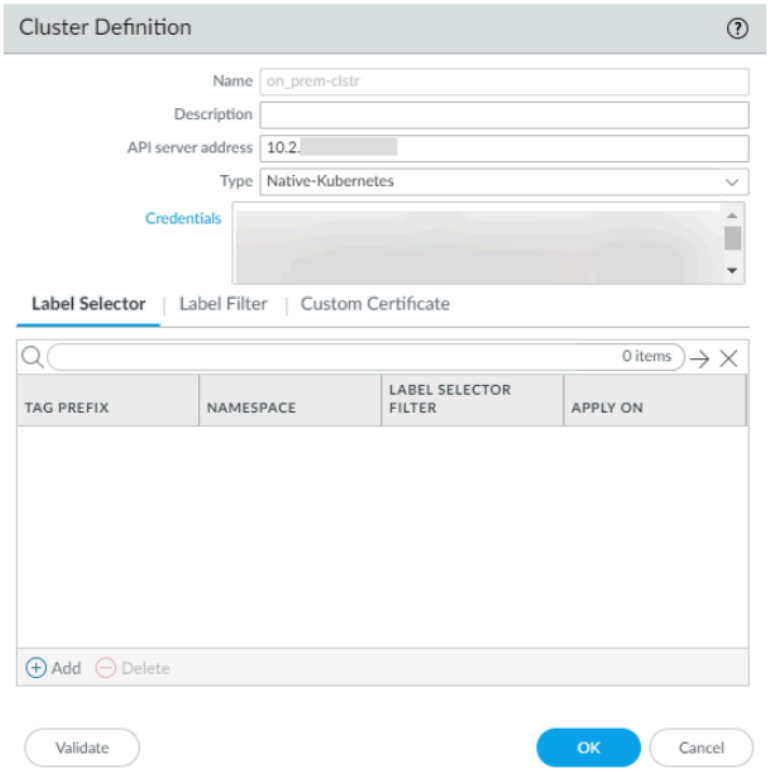
`kubectl describe node <node-name>`

Affichez les informations sous l'en-tête Capacity (Capacité) dans la sortie de la commande pour voir le processeur et la mémoire disponibles sur le nœud spécifié.

L’allocation du processeur, de la mémoire et du stockage sur disque dépendra de vos besoins. Voir Performances et évolutivité de CN-Series.

Assurez-vous d’avoir les informations suivantes :

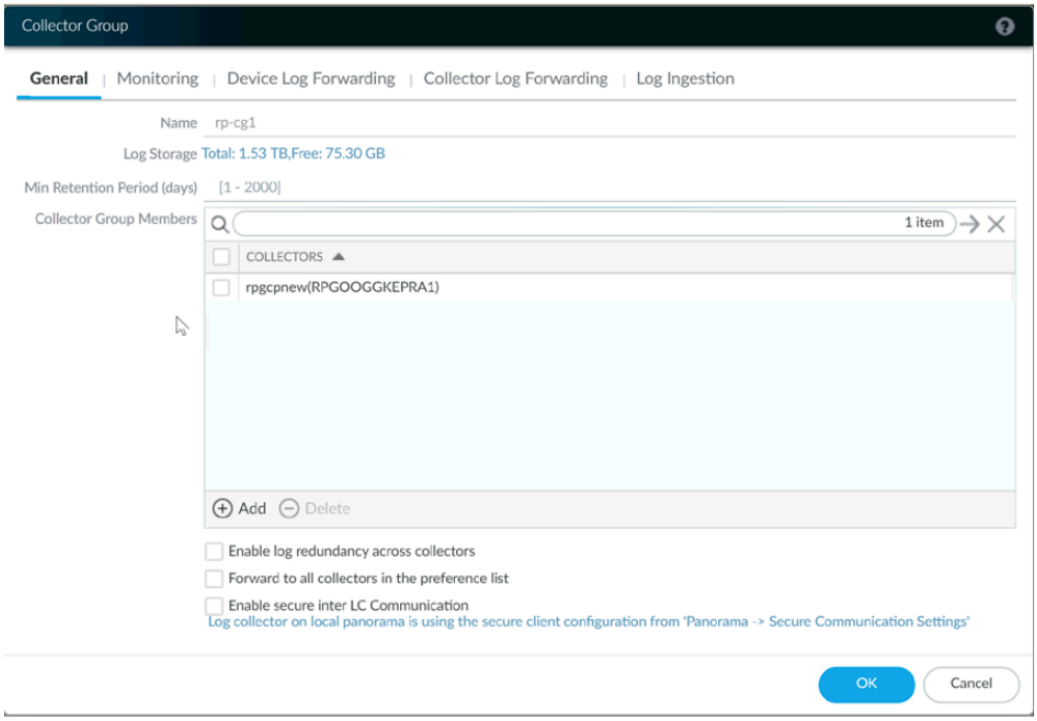
- Collectez l’adresse IP du terminal pour configurer le serveur API sur Panorama.



  Panorama utilise cette adresse IP pour se connecter à votre cluster Kubernetes.

  Pour plus d’informations, consultez Configurer le plug-in Kubernetes pour la surveillance des clusters.

- Collectez le nom de la pile de modèles, le nom du groupe d’appareils, l’adresse IP Panorama et éventuellement le nom du groupe du collecteur de journaux à partir de Panorama.



Pour plus d'informations, consultez Créer un groupe d'appareils parents et une pile de modèles.

- Collectez le code d'autorisation et l'ID et la valeur du code PIN d'enregistrement automatique.
- L'emplacement du conteneur d'images dans lequel vous avez téléchargé les images.

STEP 2 | (facultatif) Si vous avez configuré un certificat personnalisé dans le plug-in Kubernetes pour Panorama, vous devez créer le secret de certificat en exécutant la commande suivante. Ne modifiez pas le nom de fichier de ca.crt. Le volume des certificats personnalisés dans pan-cn-mgmt.yaml et pan-cn-ngfw.yaml est facultatif.

```
kubectl -n kube-system crée un secret générique custom-ca --from-file=ca.crt
```

STEP 3 | Modifiez les fichiers YAML afin de fournir les détails nécessaires au déploiement des pare-feu CN-Series.

Vous devez remplacer le chemin d'accès de l'image dans les fichiers YAML pour inclure le chemin d'accès à votre répertoire privé Google Container et fournir les paramètres requis. Pour plus d'informations, consultez Paramètres modifiables dans les fichiers yaml de déploiement CN-Series.

STEP 4 | Déployez le DaemonSet CNI.

Le conteneur CNI est déployé comme un DaemonSet (un pod par nœud) et il crée deux interfaces sur le pod CN-NGFW pour chaque application déployée sur le nœud. Lorsque vous utilisez les commandes kubectl pour exécuter les fichiers YAML pan-cni, il devient une partie de la chaîne de service sur chaque nœud.

1. Le pare-feu CN-Series nécessite trois comptes de service avec les autorisations minimales qui l'autorisent à communiquer avec les ressources de votre cluster Kubernetes. Vous devez créer

Création d’un compte de service pour l’authentification du cluster CN-Series et vérifier que vous avez créé le compte de service à l’aide du fichier pan-cni-serviceaccount.yaml.

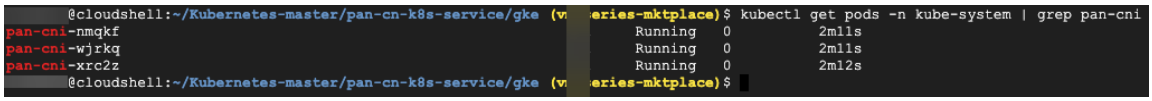
2. Utilisez Kubectl pour exécuter le fichier pan-cni-configmap.yaml.

   ```
   kubectl apply -f pan-cni-configmap.yaml
   ```

3. Utilisez Kubectl pour exécuter le fichier pan-cni.yaml.

   ```
   kubectl apply -f pan-cni.yaml
   ```

4. Vérifiez que vous avez modifié les fichiers YAML pan-cni et pan-cni-configmap.
5. Exécutez la commande suivante et vérifiez que votre sortie est similaire à l’exemple suivant.



STEP 5 | Déployez le StatefulSet CN-MGMT.

Par défaut, le plan de gestion est déployé comme un StatefulSet qui garantie la tolérance aux pannes. Jusqu'à 30 pods pare-feu CN-NGFW peuvent se connecter à un StatefulSet CN-MGMT.

1. Vérifiez que vous avez modifié les fichiers YAML pan-cn-mgmt et pan-cn-mgmt-configmap.

   Exemple de pan-cn-mgmt-configmap

```
name: pan-mgmt-config
```

```
metadata:
```

```
namespace: kube-system
```

```
data:
```

```
PAN_SERVICE_NAME: pan-mgmt-svc
```

```
PAN_MGMT_SECRET: pan-mgmt-secret
```

```
#Paramètres de Panorama :
```

```
PAN_PANORAMA_IP: "x.y.z.a"
```

```
PAN_DEVICE_GROUP: "dg-1"
```

```
PAN_TEMPLATE_STACK: "temp-stack-1"
```

```
PAN_CGNAME: "CG-GKE"
```

```
Paramètres non obligatoires
```

```
#Il est recommandé d'avoir le même nom que le nom du
 cluster fourni dans l'extension Kubernetes de Panorama.
```



```
 L’identification des pods sera plus facile en cas de gestion
 de plusieurs clusters avec le même Panorama
```

```
#CLUSTER_NAME: "<Cluster name>"
```

```
#PAN_PANORAMA_IP2: ""
```

```
#Commentaire pour utiliser les CERT sinon PSK pour IPSec entre
 pan-mgmt et pan-ngfw
```

```
#IPSEC_CERT_BYPASS: ""
```

```
#Aucune valeur nécessaire
```

```
#Remplacer la détection automatique du mode
 jumbo-frame et forcer l’activation à l’échelle du
 système#PAN_JUMBO_FRAME_ENABLED: "true"
```

```
#Démarrer le pod MGMT avec GTP activé. Pour une fonctionnalité
 complète, vous devez également activer GTP dans Panorama.
```

```
#PAN_GTP_ENABLED: "true"
```

```
#Activer les capacités de fonctionnalités élevées. Celles-
ci nécessitent une mémoire élevée pour le pod MGMT et une
```

```
 mémoire supérieure/correspondant à celle spécifiée ci-dessous
 pour le pod NGFW.
```

```
#Ceci nécessite la prise en charge du noyau et le pod NGFW
 s'exécutant avec des privilèges : true
```

```
#PAN_NGFW_MEMORY: "42Gi"
```

Exemple de pan-cn-mgmt.yaml

```
initContainers:
```

```
 - name: pan-mgmt-init
```

```
image : <your-private-registry-image-path>
```

```
conteneurs : - nom : pan-mgmt
```

```
image : <your-private-registry-image-path>
```

```
terminationMessagePolicy: FallbackToLogsOnError
```

2. Utilisez Kubectl pour exécuter les fichiers yaml.

   kubectl apply -f pan-cn-mgmt-configmap.yaml

   kubectl apply -f pan-cn-mgmt-secret.yaml

   kubectl apply -f pan-cn-mgmt.yaml

   Vous devez exécuter pan-mgmt-serviceaccount.yaml, uniquement si vous n'aviez pas déjà terminé la création de comptes de service pour l'authentification de cluster.

3. Vérifiez que les pods CN-MGMT sont activés.

   Cela prend environ 5-6 minutes.

   Utilisez **kubectl get pods -l app=pan-mgmt -n kube-system**

   NOM PRÊT ÉTAT REDÉMARRE AGEpan-mgmt-sts-0 1/1

   Fonctionnement 0 27hpan-mgmt-sts-1 1/1 Exécution 0 27h

STEP 6 | Déployez les pods CN-NGFW.

Par défaut, le pod CN-NGFW du plan de données du pare-feu est déployé comme un DaemonSet. Une instance du pod CN-NFGW peut sécuriser le trafic pour un maximum de 30 pods d'application sur un nœud.

1. Vérifiez que vous avez modifié les fichiers YAML comme indiqué dans PAN-CN-NGFW-CONFIGMAP et PAN-CN-NGFW.

   containers: - name: pan-ngfw-container image: <your-private-registry-image-path>

2. Utilisez l'application Kubectl pour exécuter le fichier pan-cn-ngfw-configmap.yaml.

   kubectl apply -f pan-cn-ngfw-configmap.yaml

3. Utilisez l'application Kubectl pour exécuter le pan-cn-ngfw.yaml.

   kubectl apply -f pan-cn-ngfw.yaml

4. Vérifiez que tous les pods CN-NGFW sont en cours d'exécution (un par nœud dans votre cluster)

   Il s'agit d'un exemple de résultat provenant d'un cluster de 4 nœuds sur site.

   kubectl get pods -n kube-system -l app=pan-ngfw -o wide

   ```
   NAME READY STATUS RESTARTS AGE IP NODE NOMINATED NODE READINESS GATES

   pan-ngfw-ds-8g5xb 1/1 Running 0 27h 10.233.71.113 rk-k8-node-1 <none> <none>

   pan-ngfw-ds-qsrm6 1/1 Running 0 27h 10.233.115.189 rk-k8-vm-worker-1 <none> <none>

   pan-ngfw-ds-vqk7z 1/1 Running 0 27h 10.233.118.208 rk-k8-vm-worker-3 <none> <none>

   pan-ngfw-ds-zncqg 1/1 Running 0 27h 10.233.91.210 rk-k8-vm-worker-2 <none> <none>
   ```


STEP 7 | Vérifiez que vous pouvez voir CN-MGMT, le CN-NGFW et le PAN-CNI sur le cluster Kubernetes.

```
kubectl -n kube-system get pods
0 27hpan-cni-5fhbg 1/1 En cours d'exécution
0 27hpan-cni-9j4rs 1/1 En cours d'exécution
0 27hpan-cni-ddwb4 1/1 En cours d'exécution
0 27hpan-cni-fwfrk 1/1 En cours d'exécution
0 27hpan-cni-h57lm 1/1 En cours d'exécution
0 27hpan-cni-h57lm 1/1 En cours d'exécution
0 27hpan-cni-j62rk 1/1 En cours d'exécution
0 27hpan-cni-lmxdz 1/1 En cours d'exécution
0 27hpan-mgmt-sts-0 1/1 En cours d'exécution
0 27hpan-mgmt-sts-1 1/1 En cours d'exécution
0 27hpan-ngfw-ds-8g5xb 1/1 En cours d'exécution
27hpan-ngfw-ds-qsrm6 1/1 En cours d'exécution
0 27hpan-ngfw-ds-vqk7z 1/1 En cours d'exécution
0 27hpan-ngfw-ds-zncqg 1/1 En cours d'exécution
```

STEP 8 | Annotez l'application yaml ou l'espace de noms afin que le trafic de leurs nouveaux pods soit redirigé vers le pare-feu.

Vous devez ajouter l'annotation suivante pour rediriger le trafic vers le CN-NGFW pour inspection :

```
annotations: paloaltonetworks.com/firewall: pan-fw
```

Par exemple, pour tous les nouveaux pods dans l'espace de noms « par défaut :

```
kubectl annotate namespace default paloaltonetworks.com/
firewall=pan-fw
```

Sur certaines plateformes, les pods de l'application peuvent démarrer lorsque le pan-cni n'est pas actif dans la chaîne de plug-ins CNI. Pour éviter de tels scénarios, vous devez spécifier les volumes comme indiqué ici dans le pod d'application YAML.

```
volumes: - name: pan-cni-ready hostPath: path: /var/log/
pan-appinfo/pan-cni-ready type: Répertoire
```

STEP 9 | Déployez votre application dans le cluster.

Déployer le pare-feu CN-Series dans OKE

Où puis-je utiliser ceci ?	De quoi ai-je besoin ?
• Déploiement CN-Series	• CN-Series 10.1.x or above Container Images • Panorama sous PAN-OS 10.1.x ou une version supérieure • Helm 3.6 or above version client pour le déploiement CN-Series à l'aide de Helm

Oracle Kubernetes Engine (OKE) est un service OCI qui vous permet de déployer un cluster Kubernetes. Vous pouvez désormais déployer le pare-feu CN-Series dans le cluster OKE en tant que daemonset et en kubernetes en tant que service.

Après avoir examiné les blocs de construction CN-Series et la présentation générale du flux de travail dans Sécuriser les environnements Kubernetes avec CN-Series , vous pouvez commencer à déployer les pare-feu CN-Series sur la plate-forme OKE pour sécuriser le trafic entre les conteneurs au sein du même cluster, ainsi qu'entre les conteneurs et d'autres types de charges de travail tels que les machines virtuelles et les serveurs bare-metal.

Vous avez besoin d'outils Kubernetes standard tels que kubectl ou Helm pour déployer et gérer vos applications, vos services pare-feu et vos clusters Kubernetes.

Pour plus d'informations, consultez Déployer des pare-feu CN-Series avec des graphiques et des modèles Helm*. Panorama n'est pas conçu pour être utilisé comme orchestrateur pour le déploiement et la gestion de clusters Kubernetes. Les modèles pour la gestion des clusters sont fournis par les fournisseurs de Kubernetes gérés. Palo Alto Networks fournit des modèles pris en charge par la communauté pour le déploiement CN-Series avec* Helm *et* Terraform.

- Déployer le pare-feu CN-Series en tant que service Kubernetes dans OKE
- Déployer le pare-feu CN-Series en tant que DaemonSet dans OKE

Avant de passer du déploiement de CN-Series en tant que DaemonSet à CN-Series en tant que service ou vice versa, vous devez supprimer et réappliquer `plugin-serviceaccount.yaml`. *Pour plus d'informations, consultez* Créer des comptes de service pour l'authentification des clusters.

- *Lorsque vous déployez CN-Series en tant que DaemonSet dans OKE, le* `pan-plugin-cluster-mode-secret` *ne doit pas exister.*
- *Lorsque vous déployez CN-Series en tant que service Kubernetes, le*`pan-plugin-cluster-mode-secret` *doit être présent.*

Déployer le pare-feu CN-Series en tant que service Kubernetes dans OKE

Où puis-je utiliser ceci ?	De quoi ai-je besoin ?
• Déploiement CN-Series	• CN-Series 10.1.x or above Container Images • Panorama sous PAN-OS 10.1.x ou une version supérieure • Helm 3.6 or above version client pour le déploiement CN-Series à l’aide de Helm

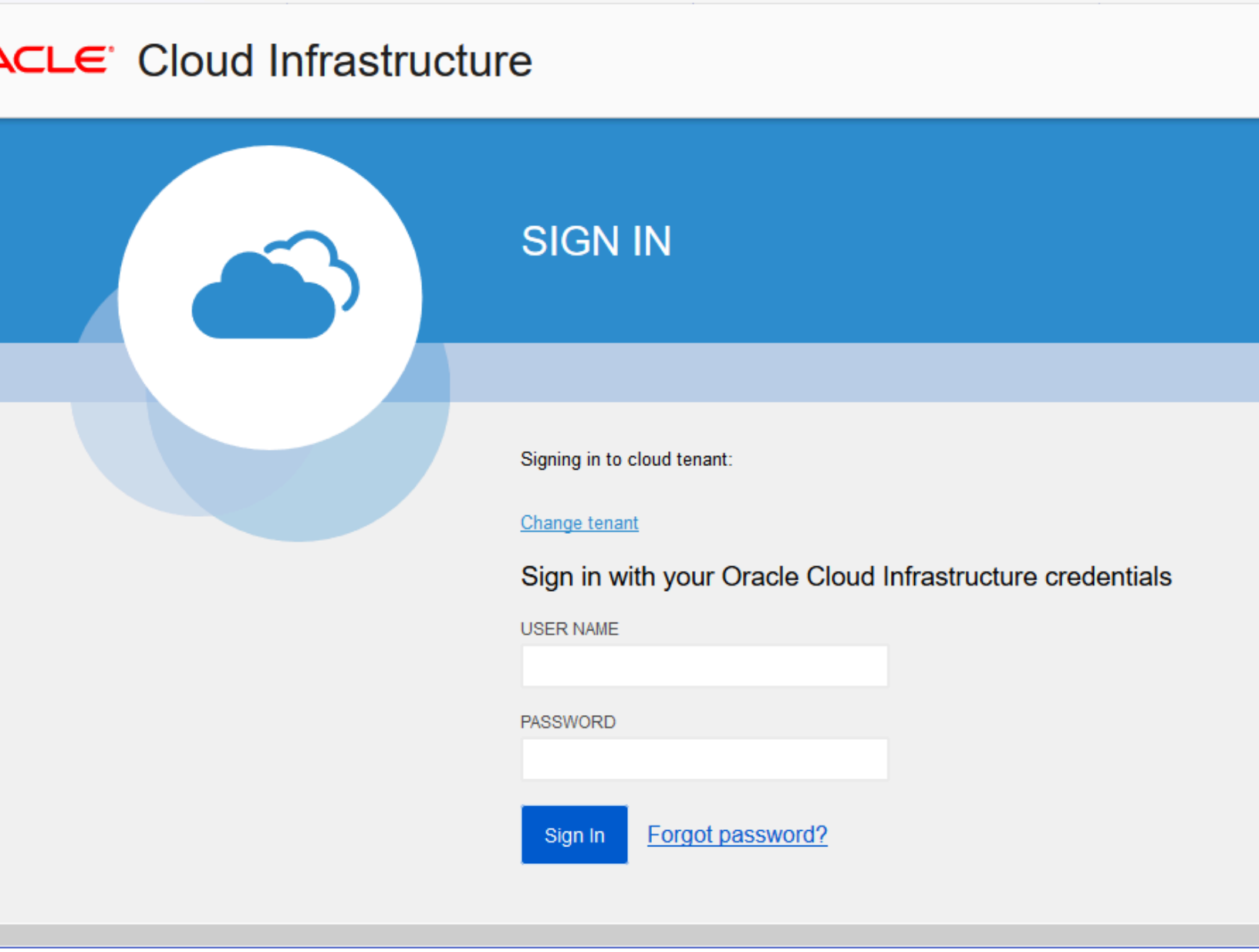
Effectuez la procédure suivante pour déployer le pare-feu CN-Series en tant que service Kubernetes dans la plateforme OKE.

Le système d’exploitation Oracle Linux 8.5 est le seul environnement qualifié pour le déploiement du pare-feu CN-Series dans OKE.

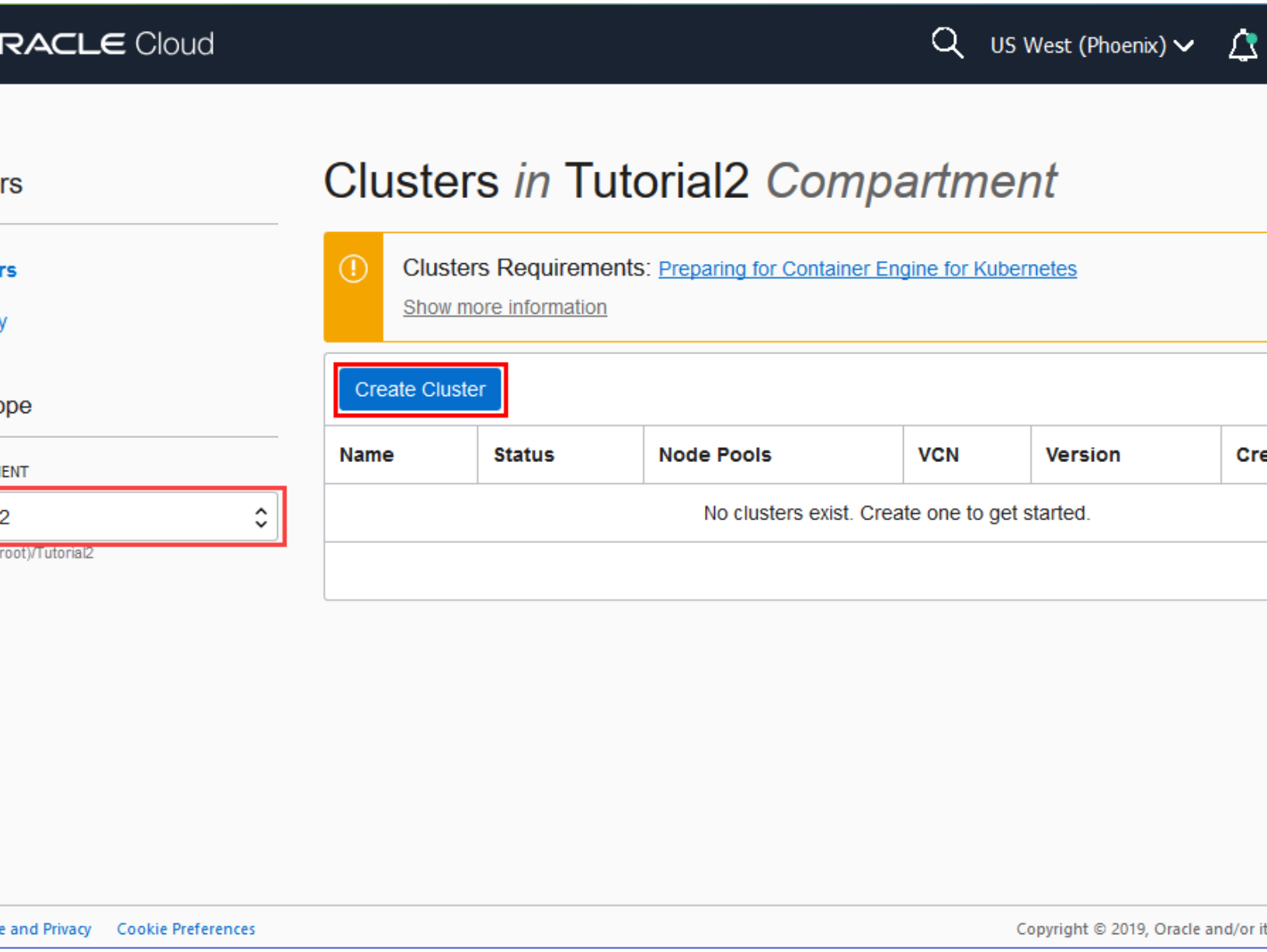
STEP 1 | Configurez votre cluster Kubernetes.

Pour créer un cluster dans OKE, procédez comme suit :

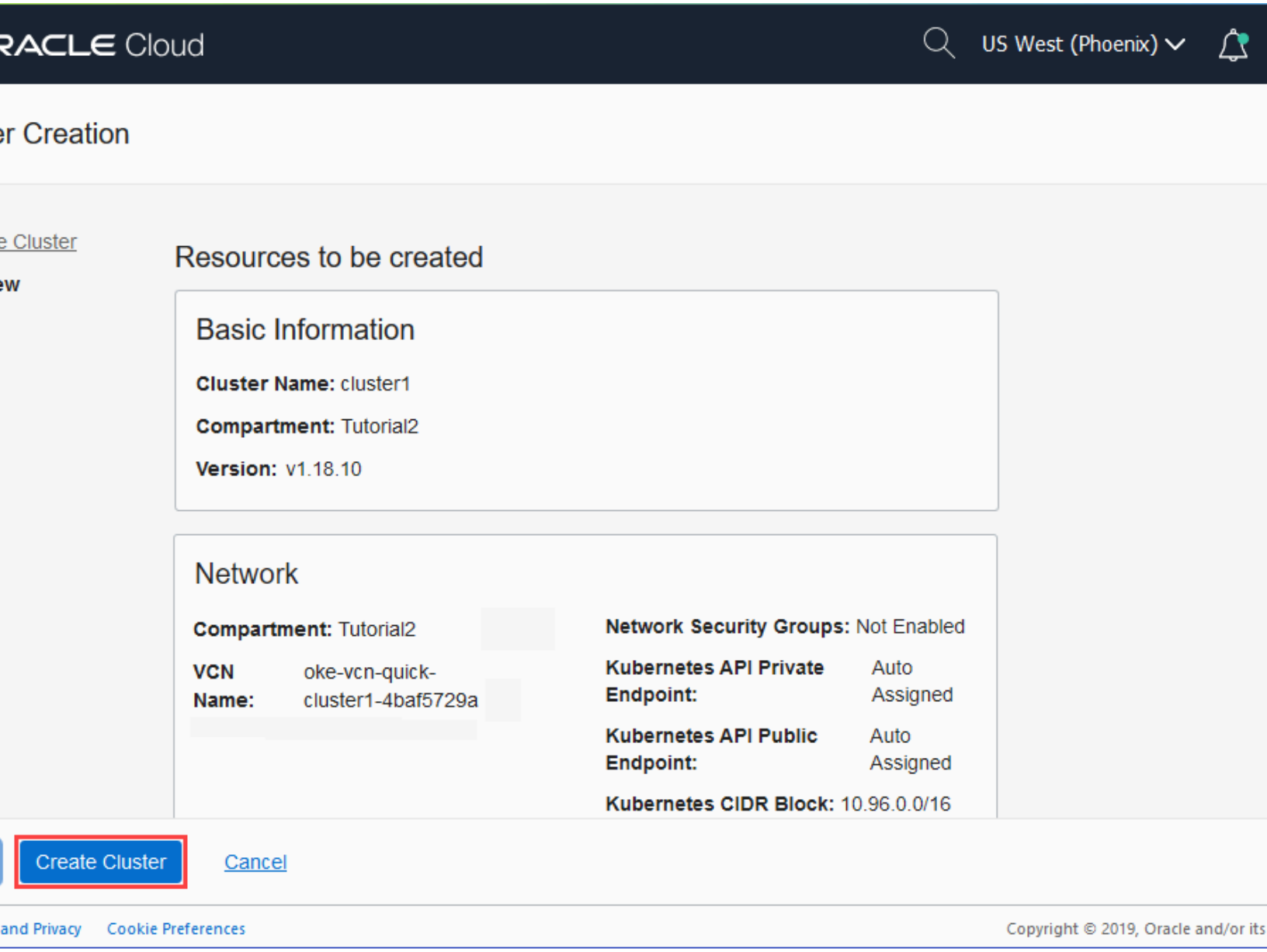
1. Connectez-vous à Oracle Cloud Infrastructure.



2. Cliquez sur le menu de navigation, accédez à **Under Solutions and Platform (Solutions et plate-forme inférieures)**, puis cliquez sur **Developer Services (Services aux développeurs)**.
3. Cliquez sur **Kubernetes Clusters (Clusters Kubernetes)**.
4. Sélectionnez un compartiment et cliquez sur **Create Cluster (Créer un cluster)**.



5. Dans la boîte de dialogue Create Cluster (Créer un cluster), cliquez sur **Custom Create (Création personnalisée)**, puis sur **Launch Workflow (Lancer le flux de production)**.
6. Sur la page **Create Cluster (Créer un cluster)**, entrez le **Name (Nom)** du cluster et d'autres détails.
7. Cliquez sur **Next (Suivant)** pour vérifier les détails que vous avez saisis pour le nouveau cluster.
8. Sur la page Réviser, cliquez sur **Create Cluster (Créer un cluster)**.



1. Vous devez vous assurer que ce cluster dispose des conditions préalables de CN-Series pour prendre en charge le pare-feu :

   `kubectl get nodes`

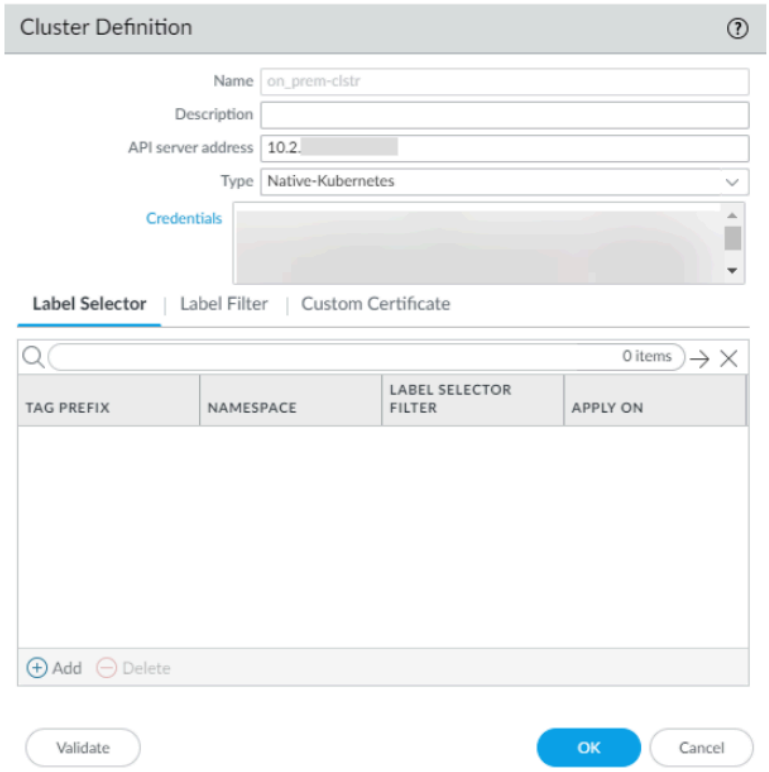
   `kubectl describe node <node-name>`

   Affichez les informations sous l'en-tête Capacity (Capacité) dans la sortie de la commande pour voir le processeur et la mémoire disponibles sur le nœud spécifié.

   L'allocation du processeur, de la mémoire et du stockage sur disque dépendra de vos besoins. Voir Performances et mise à l'échelle de CN-Series.

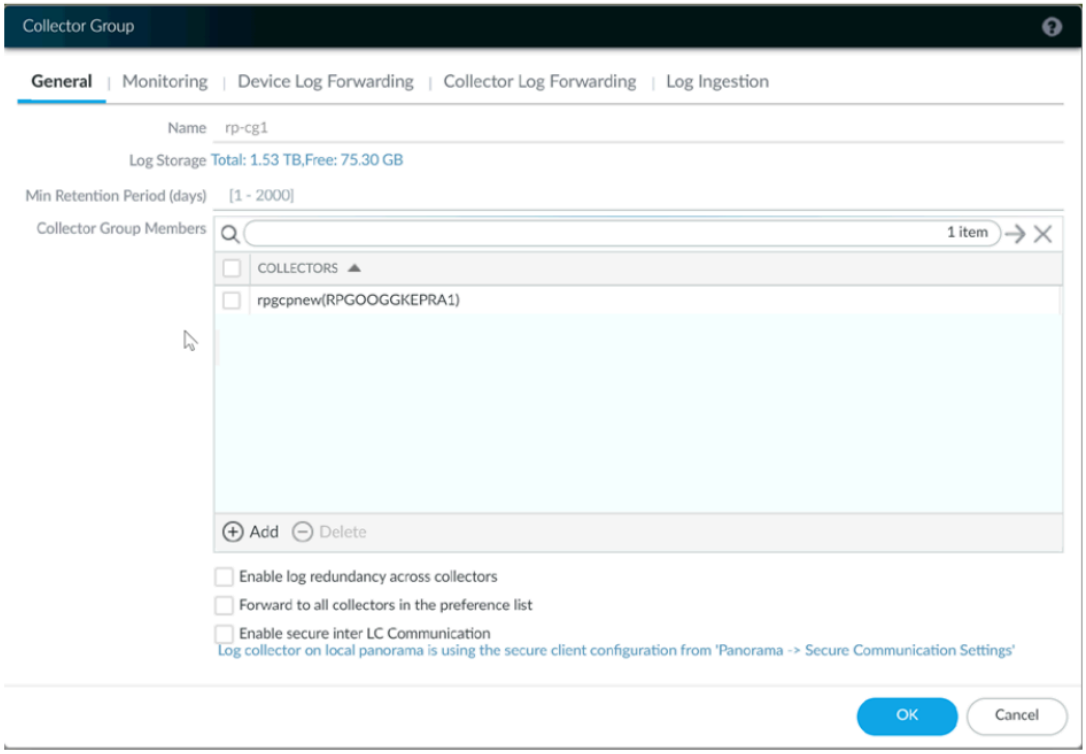
   Assurez-vous d'avoir les informations suivantes :

   - Collectez l'adresse IP du terminal pour configurer le serveur API sur Panorama.



Panorama utilise cette adresse IP pour se connecter à votre cluster Kubernetes.

- Collectez le nom de la pile de modèles, le nom du groupe d'appareils, l'adresse IP Panorama et éventuellement le nom du groupe de collecteurs de journaux à partir de Panorama.



Pour plus d'informations, consultez Créer un groupe d'appareils parents et une pile de modèles.

- Collectez le code d'autorisation et l'ID et la valeur du code PIN d'enregistrement automatique.
- Préparez l'emplacement du répertoire du conteneur d'images dans lequel vous avez téléchargé les images.

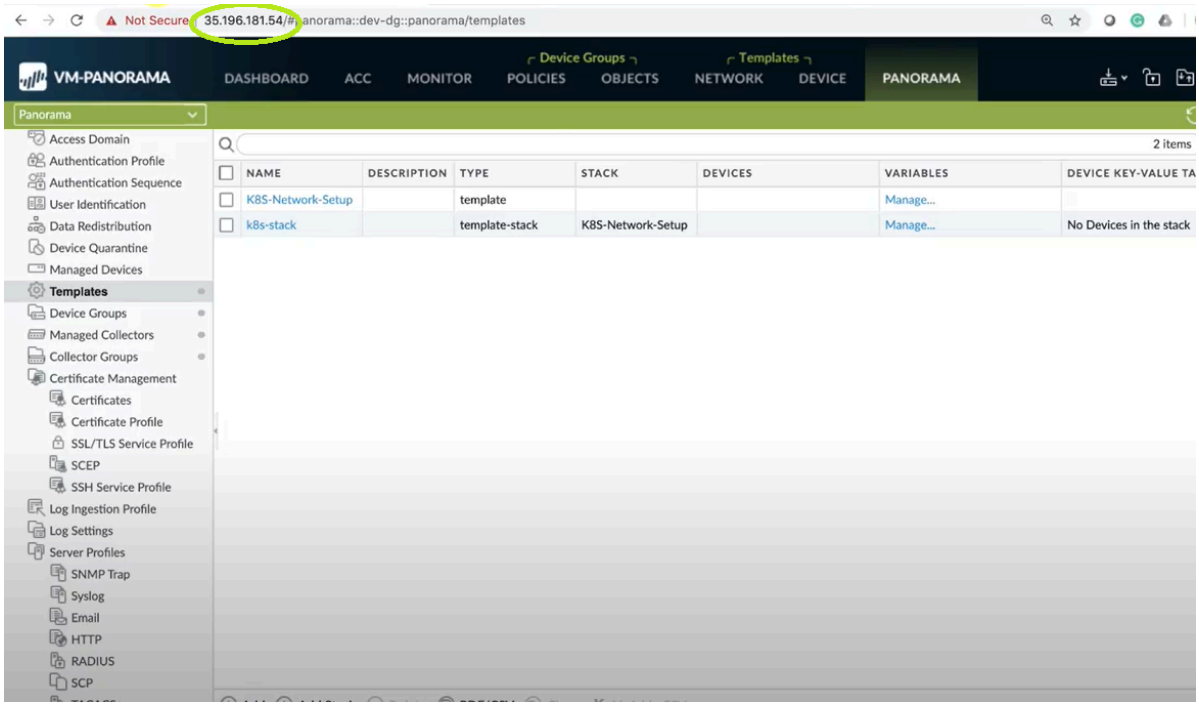
STEP 2 | (facultatif) Si vous avez configuré un certificat personnalisé dans le plug-in Kubernetes pour Panorama, vous devez créer le secret de certificat en exécutant la commande suivante. Ne modifiez pas le nom de fichier de ca.crt. Le volume des certificats personnalisés dans pan-cn-mgmt.yaml et pan-cn-ngfw.yaml est facultatif.

```
kubectl -n kube-system crée un secret générique custom-ca --from-file=ca.crt
```

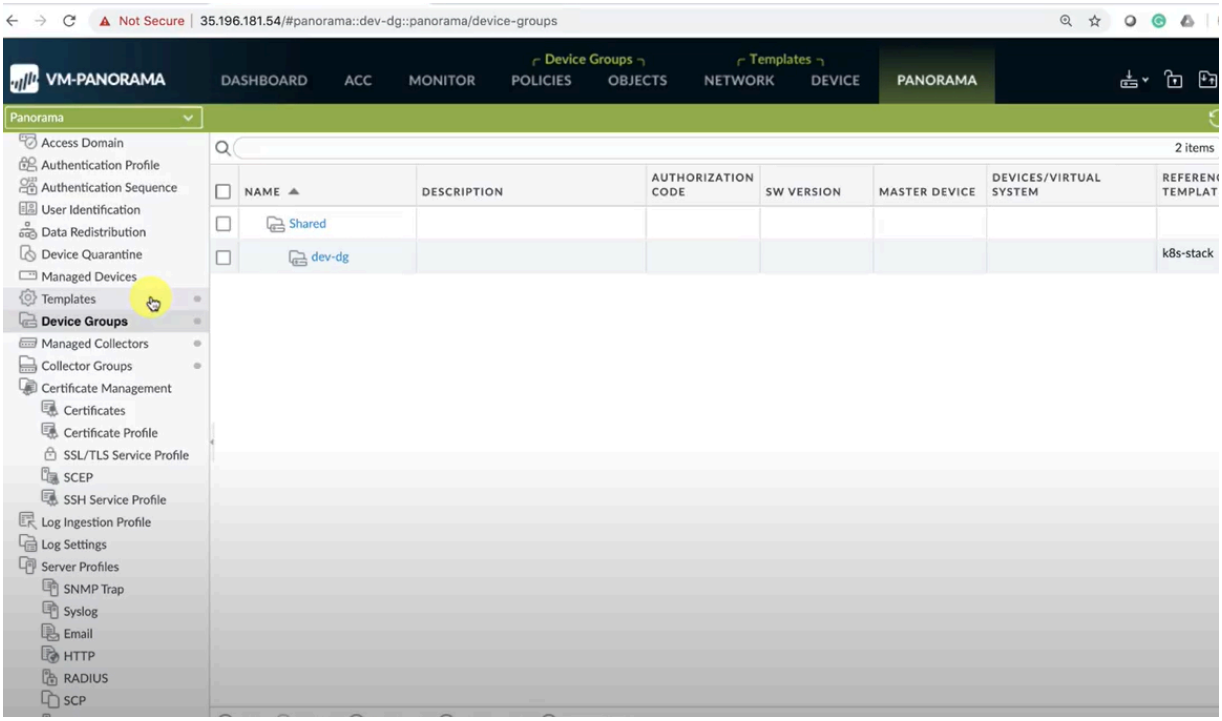

STEP 3 | Modifiez les fichiers YAML afin de fournir les détails nécessaires au déploiement des pare-feu CN-Series.

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: pan-mgmt-config
  namespace: kube-system
data:
  PAN_OPERATION_MODE: "daemonset"
  PAN_SERVICE_NAME: "pan-mgmt"
  # Panorama settings
  PAN_PANORAMA_IP: "35.196.181.54"
  PAN_PANORAMA_AUTH_KEY: [redacted]
  PAN_DEVICE_GROUP: "dev-dg"
  PAN_TEMPLATE: "k8s-stack"
#Non-mandatory parameters
  PAN_PANORAMA_CGNAME: "rp-cg1"
  #PAN_CERTIFICATE: ""
  #PAN_CERTKEYFILE: ""
  #PAN_CERTPASSPHRASE: ""
```

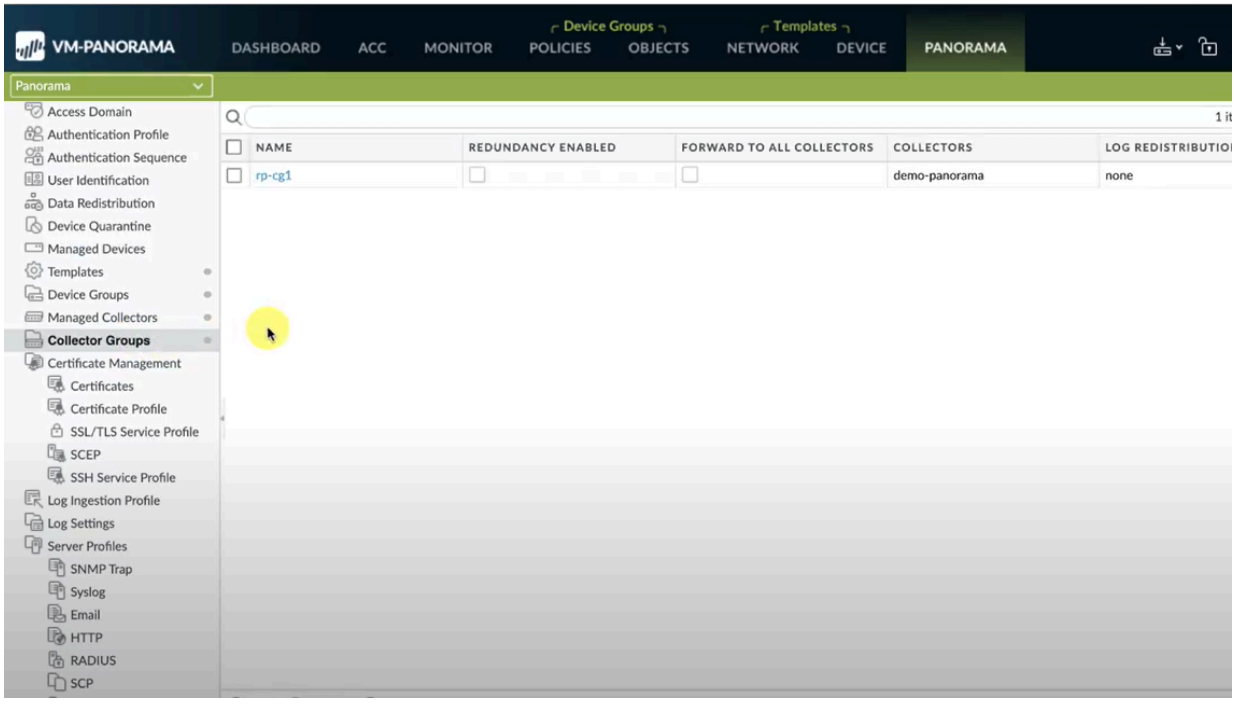
Vous devez vous assurer que la valeur du paramètre `PAN_PANORAMA_IP` sur votre fichier YAML correspond à votre adresse IP Panorama réelle, comme indiqué dans le diagramme ci-dessous :



Vous devez vous assurer que la valeur des paramètres de `PAN_DEVICE_GROUP` et `PAN_TEMPLATE` sur votre fichier YAML correspond au nom du groupe d'appareils et de la pile de modèles que vous avez créés sur Panorama, comme indiqué dans le diagramme ci-dessous :

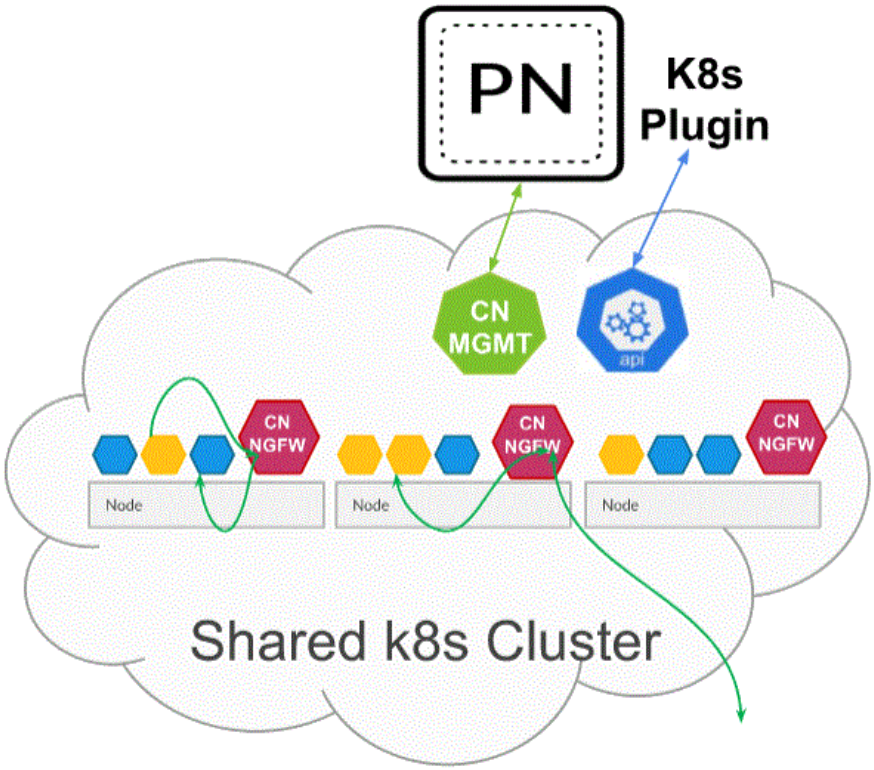


Vous devez vous assurer que la valeur du paramètre `PAN_PANORAMA_CG_NAME` est identique au nom du collecteur de journaux que vous avez créé.



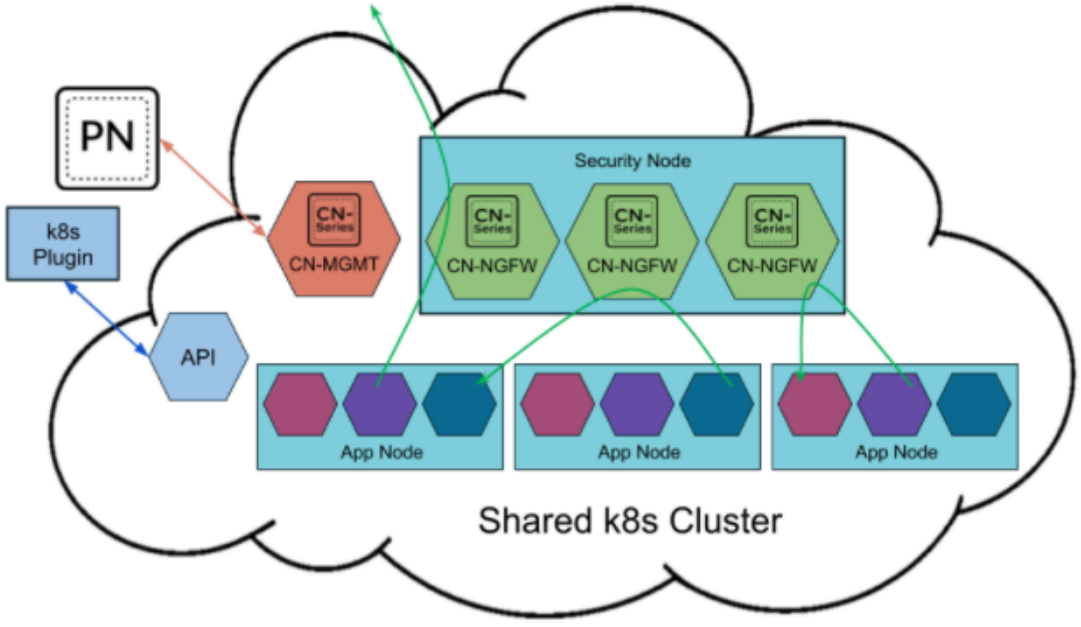
Pour plus d'informations, consultez Paramètres modifiables dans les fichiers yaml de déploiement CN-Series.

STEP 4 | Déployez le service CN-NGFW. Effectuez les étapes suivantes :



Lorsqu'elles sont déployées en tant que service Kubernetes, les instances du pod CN-NGFW peuvent être déployées sur des nœuds de sécurité et le trafic du pod d'application est redirigé vers une instance CN-NGFW disponible pour inspection et application.

Lors du déploiement du pare-feu CN-Series dans OKE en tant que service Kubernetes, vous pouvez utiliser les fichiers yaml du dossier natif pan-cn-k8s-service.



1. Vérifiez que vous avez créé le compte de service à l'aide du fichier pan-cni-serviceaccount.yaml.

   Consultez Création de comptes de service pour l'authentification des clusters.
2. Utilisez Kubectl pour exécuter le fichier pan-cni-configmap.yaml.

   kubectl apply -f pan-cni-configmap.yaml
3. Utilisez kubectl pour exécuter le fichier pan-cn-ngfw-svc.yaml.

   kubectl apply -f pan-cn-ngfw-svc.yaml

   Ce fichier yaml doit être déployé avant pan-cni.yaml.
4. Utilisez Kubectl pour exécuter le fichier pan-cni.yaml.

   kubectl apply -f pan-cni.yaml
5. Vérifiez que vous avez modifié les fichiers YAML pan-cni et pan-cni-configmap.
6. Exécutez la commande suivante et vérifiez que votre sortie est similaire à l'exemple suivant.

```
@cloudshell:~/Kubernetes-master/pan-cn-k8s-service/gke (v  eries-mktplace)$ kubectl get pods -n kube-system | grep pan-cni
pan-cni-nmqkf                                              Running   0          2m11s
pan-cni-wjrkq                                              Running   0          2m11s
pan-cni-xrc2z                                              Running   0          2m12s
@cloudshell:~/Kubernetes-master/pan-cn-k8s-service/gke (v  eries-mktplace)$
```

STEP 5 | Déployez le StatefulSet CN-MGMT.

Par défaut, le plan de gestion est déployé comme un StatefulSet qui garantie la tolérance aux pannes. Jusqu'à 30 pods pare-feu CN-NGFW peuvent se connecter à un StatefulSet CN-MGMT.

1. Vérifiez que vous avez modifié les fichiers YAML pan-cn-mgmt et pan-cn-mgmt-configmap.

   Exemple de pan-cn-mgmt-configmap d'OKE.

```
apiVersion: v1 kind: ConfigMap metadata: name: pan-mgmt-
config namespace: kube-system data: PAN_SERVICE_NAME:
 pan-mgmt-svc PAN_MGMT_SECRET: pan-mgmt-secret # Panorama
 settings PAN_PANORAMA_IP: "<panorama-IP>" PAN_DEVICE_GROUP:
 "<panorama-device-group>" PAN_TEMPLATE_STACK: "<panorama-
template-stack>" PAN_CGNAME: "<panorama-collector-group>"
 PAN_CTNR_MODE_TYPE: "k8s-service" #Non-mandatory parameters #
 Recommended to have same name as the cluster name provided in
 Panorama Kubernetes plugin - helps with easier identification
 of pods if managing multiple clusters with same Panorama
 #CLUSTER_NAME: "<Cluster name>" #PAN_PANORAMA_IP2: "" #
 Comment out to use CERTs otherwise PSK for IPSec between pan-
mgmt and pan-ngfw #IPSEC_CERT_BYPASS: "" # No values needed
 # Override auto-detect of jumbo-frame mode and force enable
 system-wide #PAN_JUMBO_FRAME_ENABLED: "true" # Start MGMT
 pod with GTP enabled. For complete functionality, need GTP #
 enable at Panorama as well. #PAN_GTP_ENABLED: "true" # Enable
 high feature capacities. These need high memory for MGMT pod
 and # higher/matching memory than specified below for NGFW
 pod. # Refer to the system requirements documentation to see
 the max supported NGFW CPU size # supported for each memory
```



```
 profile. #PAN_NGFW_MEMORY: "6.5Gi" #PAN_NGFW_MEMORY: "48Gi"
 #PAN_NGFW_MEMORY: "56Gi"
```

Exemple pan-cn-mgmt-dynamic-pv.yaml

```
initContainers: - name: pan-mgmt-init image: <your-private-
registry-image-path> command: ["/usr/bin/pan_start.sh"]
 imagePullPolicy: Toujours
```

```
 containers: - name: pan-mgmt image: <your-private-registry-
image-path> terminationMessagePolicy: FallbackToLogsOnError
```

2. Utilisez Kubectl pour exécuter les fichiers yaml.

   kubectl apply -f pan-cn-mgmt-configmap.yaml

   kubectl apply -f pan-cn-mgmt-slot-crd.yaml

   kubectl apply -f pan-cn-mgmt-slot-cr.yaml

   kubectl apply -f pan-cn-mgmt-secret.yaml

   kubectl apply -f pan-cn-mgmt-dynamic-pv.yaml

   Vous devez exécuter pan-mgmt-serviceaccount.yaml, uniquement si vous n'aviez pas déjà terminé la création de compte de service pour l'authentification de cluster.

3. Vérifiez que les pods CN-MGMT sont opérationnels en exécutant la commande suivante :

   kubectl get pods -l app=pan-mgmt -n kube-system

   Cela prend environ 5-6 minutes.

STEP 6 | Déployez les pods CN-NGFW.

1. Vérifiez que vous avez modifié les fichiers YAML comme indiqué dans PAN-CN-NGFW-CONFIGMAP et PAN-CN-NGFW.

   containers: - name: pan-ngfw-container image: <your-private-registry-image-path>

2. Utilisez l'application Kubectl pour exécuter le fichier pan-cn-ngfw-configmap.yaml.

   kubectl apply -f pan-cn-ngfw-configmap.yaml

3. Utilisez l'application Kubectl pour exécuter le pan-cn-ngfw.yaml.

   kubectl apply -f pan-cn-ngfw.yaml

4. Vérifiez que les pods CN-NGFW sont en cours d'exécution.

   kubectl get pods -n kube-system -l app=pan-ngfw -o wide

STEP 7 | Vérifiez que vous pouvez voir CN-MGMT, le CN-NGFW et le PAN-CNI sur le cluster Kubernetes.

kubectl -n kube-system get pods

STEP 8 | Annotez l'application yaml ou l'espace de noms afin que le trafic de leurs nouveaux pods soit redirigé vers le pare-feu.

Vous devez ajouter l'annotation suivante pour rediriger le trafic vers le CN-NGFW pour inspection :

```
annotations: paloaltonetworks.com/firewall: pan-fw
```

Par exemple, pour tous les nouveaux pods dans l'espace de noms « par défaut :

```
kubectl annotate namespace default paloaltonetworks.com/
firewall=pan-fw
```

Sur certaines plateformes, les pods de l'application peuvent démarrer lorsque le pan-cni n'est pas actif dans la chaîne de plug-ins CNI. Pour éviter de tels scénarios, vous devez spécifier les volumes comme indiqué ici dans le pod d'application YAML.

```
volumes: - name: pan-cni-ready hostPath: path: /var/log/
pan-appinfo/pan-cni-ready type: Répertoire
```

STEP 9 | Déployez votre application dans le cluster.

Déployer le pare-feu CN-Series en tant que DaemonSet dans OKE

Où puis-je utiliser ceci ?	De quoi ai-je besoin ?
• Déploiement CN-Series	• CN-Series 10.2.x or above Container Images • Panorama sous PAN-OS 10.2.x ou version supérieure • Helm 3.6 or above version client pour le déploiement CN-Series à l'aide de Helm

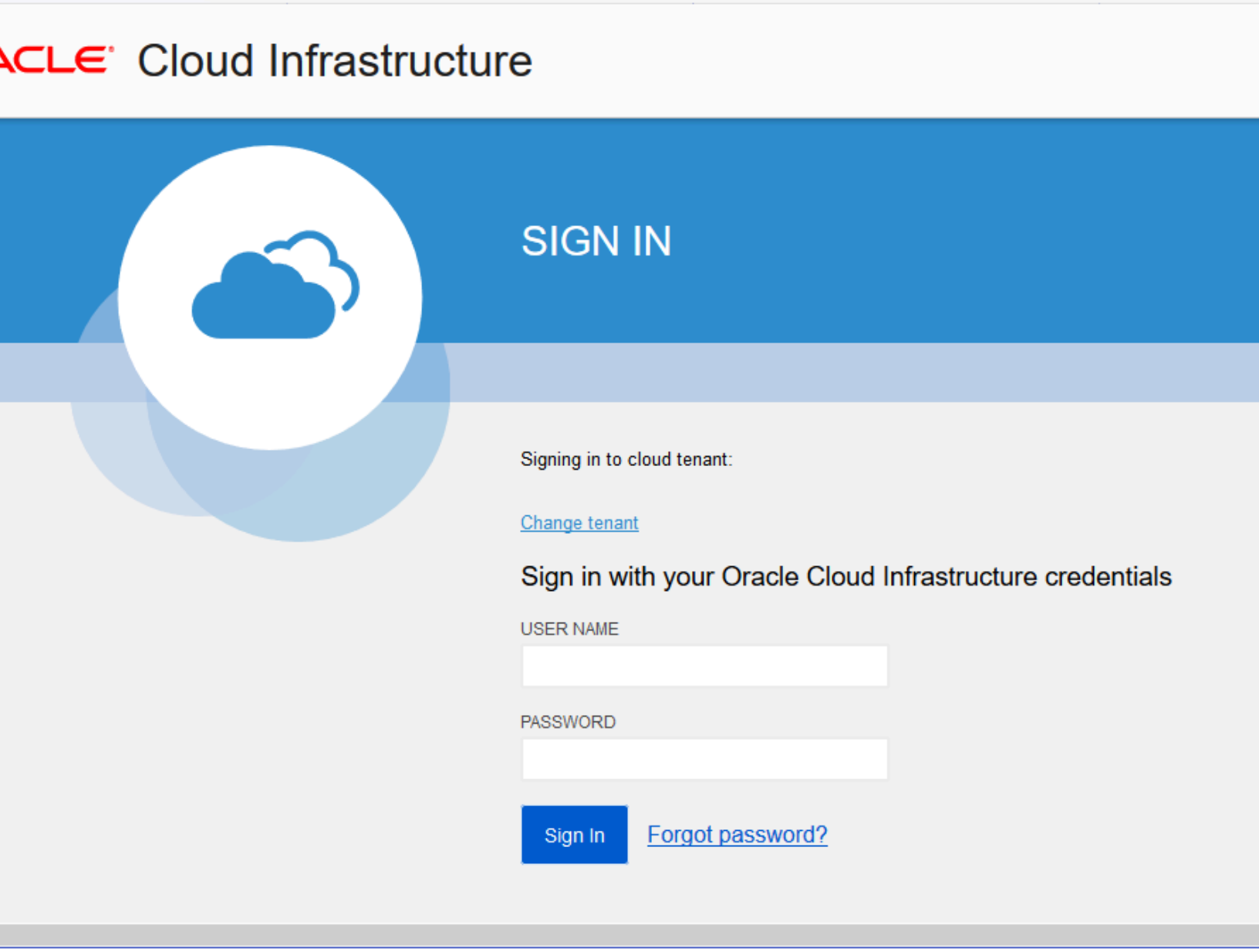
Effectuez la procédure suivante pour déployer le pare-feu CN-Series en tant que Daemonset dans la plateforme OKE :

Le système d'exploitation Oracle Linux 8.5 est le seul environnement qualifié pour le déploiement du pare-feu CN-Series dans OKE.

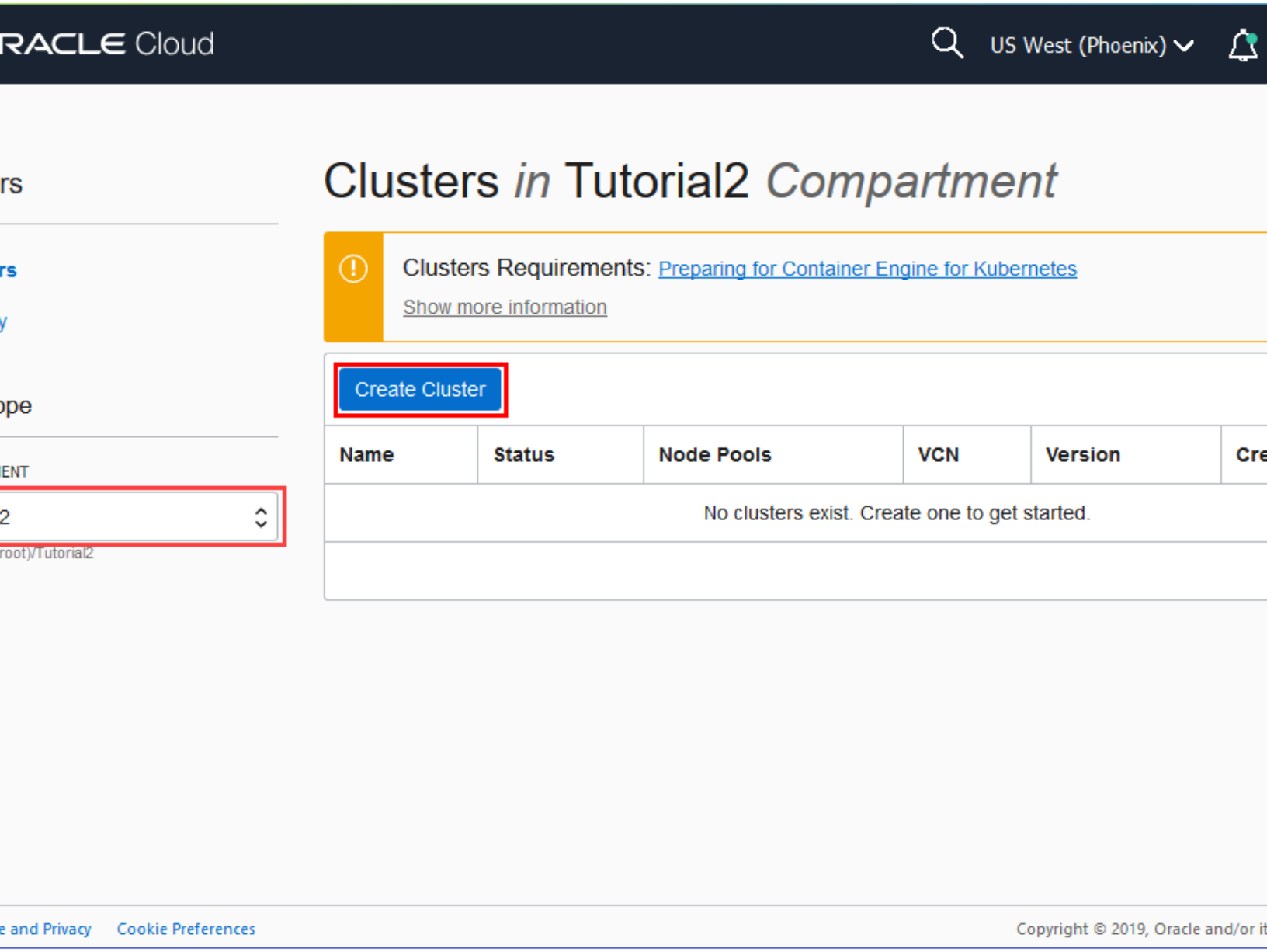
STEP 1 | Configurez votre cluster Kubernetes.

Pour créer un cluster dans OKE, procédez comme suit :

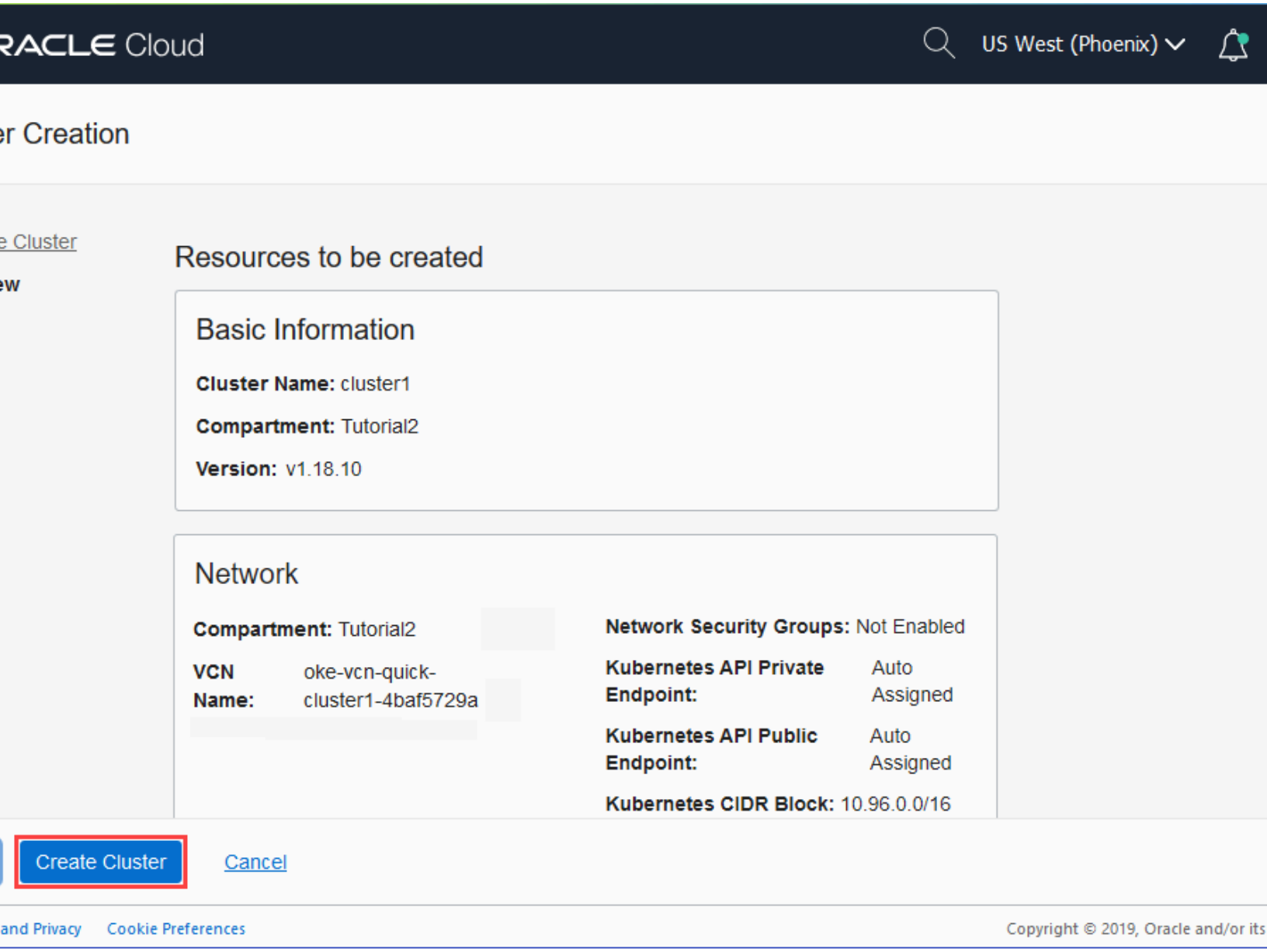
1. Connectez-vous à Oracle Cloud Infrastructure.



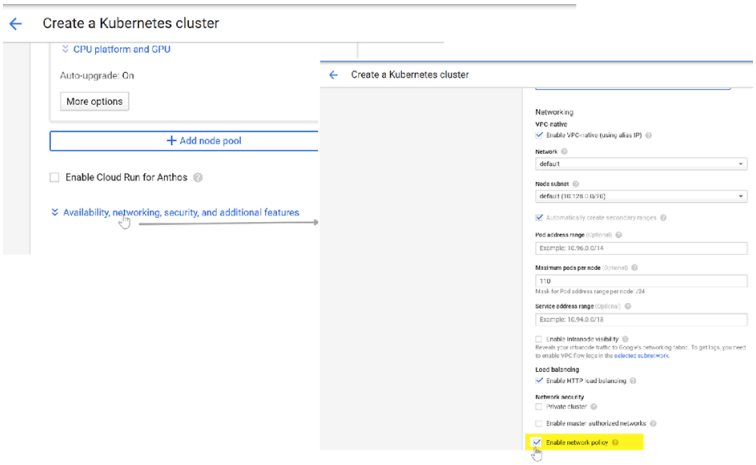
2. Cliquez sur le menu de navigation, accédez à **Under Solutions and Platform (Solutions et plate-forme inférieures)**, puis cliquez sur **Developer Services (Services aux développeurs)**.
3. Cliquez sur **Kubernetes Clusters (Clusters Kubernetes)**.
4. Sélectionnez un compartiment et cliquez sur **Create Cluster (Créer un cluster)**.



5. Dans la boîte de dialogue Create Cluster (Créer un cluster), cliquez sur **Custom Create (Création personnalisée)**, puis sur **Launch Workflow (Lancer le flux de production)**.
6. Sur la page **Create Cluster (Créer un cluster)**, entrez le **Name (Nom)** du cluster et d'autres détails.
7. Cliquez sur **Next (Suivant)** pour vérifier les détails que vous avez saisis pour le nouveau cluster.
8. Sur la page Réviser, cliquez sur **Create Cluster (Créer un cluster)**.



Si votre cluster se trouve sur OKE, assurez-vous d'activer l'API Network Policy de Kubernetes pour permettre à l'administrateur du cluster d'indiquer quels pods sont autorisés à communiquer entre eux. Cette API est requise pour permettre aux pods CN-NGFW et CN-MGMT de communiquer.



Vérifiez que le cluster dispose des ressources adéquates. Assurez-vous que ce cluster dispose des conditions préalables de CN-Series pour prendre en charge le pare-feu.

`kubectl get nodes`

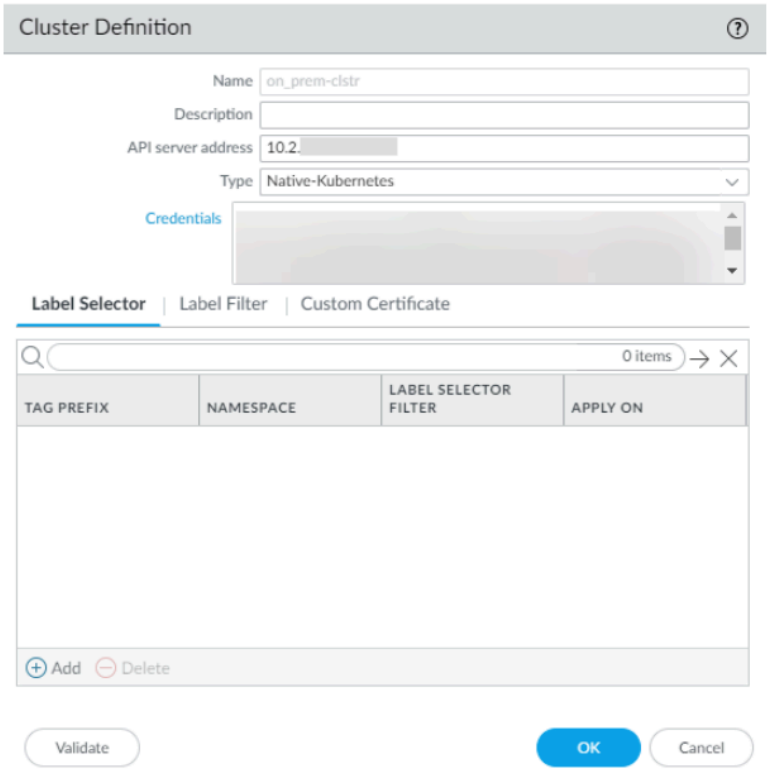
`kubectl describe node <node-name>`

Affichez les informations sous l'en-tête Capacity (Capacité) dans la sortie de la commande pour voir le processeur et la mémoire disponibles sur le nœud spécifié.

L'allocation du processeur, de la mémoire et du stockage sur disque dépendra de vos besoins. Voir Performances et évolutivité de CN-Series.

Assurez-vous d'avoir les informations suivantes :

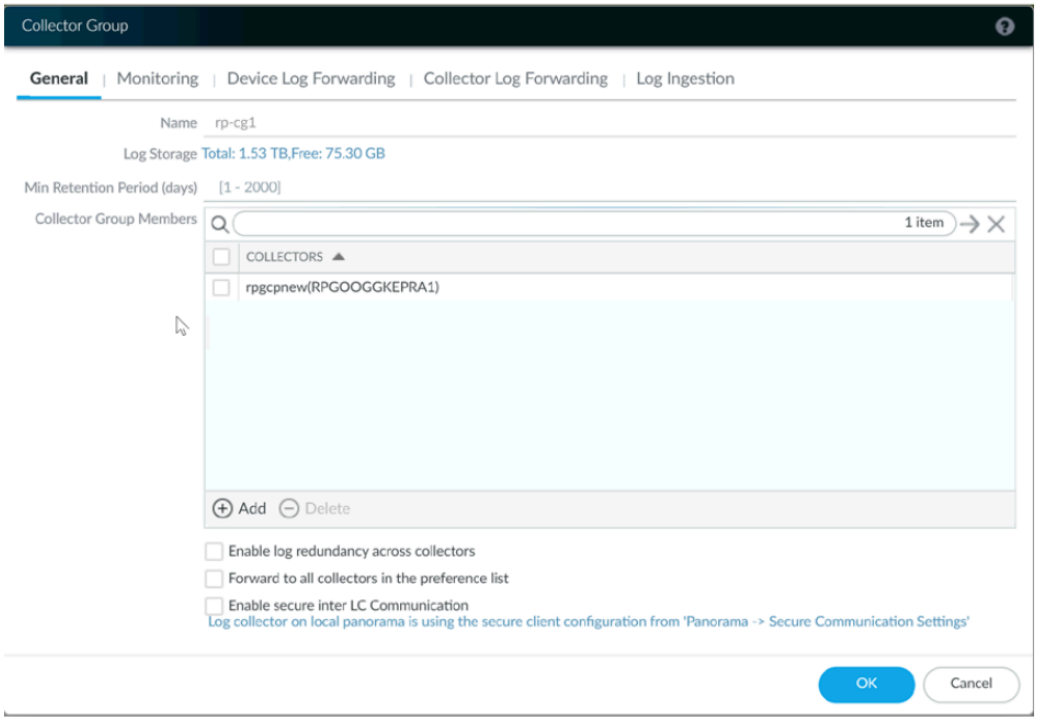
- Collectez l'adresse IP du terminal pour configurer le serveur API sur Panorama.



Panorama utilise cette adresse IP pour se connecter à votre cluster Kubernetes.

Pour plus d'informations, consultez Configurer le plug-in Kubernetes pour la surveillance des clusters.

- Collectez le nom de la pile de modèles, le nom du groupe d'appareils, l'adresse IP Panorama et éventuellement le nom du groupe du collecteur de journaux à partir de Panorama.



Pour plus d'informations, consultez Créer un groupe d'appareils parents et une pile de modèles.

- Collectez le code d'autorisation et l'ID et la valeur du code PIN d'enregistrement automatique.
- L'emplacement du conteneur d'images dans lequel vous avez téléchargé les images.

STEP 2 | (facultatif) Si vous avez configuré un certificat personnalisé dans le plug-in Kubernetes pour Panorama, vous devez créer le secret de certificat en exécutant la commande suivante. Ne modifiez pas le nom de fichier de ca.crt. Le volume des certificats personnalisés dans pan-cn-mgmt.yaml et pan-cn-ngfw.yaml est facultatif.

```
kubectl -n kube-system crée un secret générique custom-ca --from-file=ca.crt
```

STEP 3 | Modifiez les fichiers YAML afin de fournir les détails nécessaires au déploiement des pare-feu CN-Series.

Vous devez remplacer le chemin d'accès de l'image dans les fichiers YAML pour inclure le chemin d'accès à votre répertoire privé Google Container et fournir les paramètres requis. Pour plus d'informations, consultez Paramètres modifiables dans les fichiers yaml de déploiement CN-Series.

STEP 4 | Déployez le DaemonSet CNI.

Le conteneur CNI est déployé comme un DaemonSet (un pod par nœud) et il crée deux interfaces sur le pod CN-NGFW pour chaque application déployée sur le nœud. Lorsque vous utilisez les commandes

kubectl pour exécuter les fichiers YAML pan-cni, il devient une partie de la chaîne de service sur chaque nœud.

Lors du déploiement du pare-feu CN-Series sur OKE en tant que Daemonset, vous pouvez utiliser les fichiers yaml du dossier natif pan-cn-k8s-daemonset.

1. Le pare-feu CN-Series nécessite trois comptes de service avec les autorisations minimales qui l'autorisent à communiquer avec les ressources de votre cluster Kubernetes. Vous devez créer des comptes de service pour l'authentification de cluster avec CN-Series et vérifier que vous avez créé le compte de service à l'aide de pan-cni-serviceaccount.yaml.
2. Utilisez Kubectl pour exécuter le fichier pan-cni-configmap.yaml.

   kubectl apply -f pan-cni-configmap.yaml
3. Utilisez Kubectl pour exécuter le fichier pan-cni.yaml.

   kubectl apply -f pan-cni.yaml
4. Vérifiez que vous avez modifié les fichiers YAML pan-cni et pan-cni-configmap.
5. Exécutez la commande suivante et vérifiez que votre sortie est similaire à l'exemple suivant.

STEP 5 | Déployez le StatefulSet CN-MGMT.

Par défaut, le plan de gestion est déployé comme un StatefulSet qui garantie la tolérance aux pannes. Jusqu'à 30 pods pare-feu CN-NGFW peuvent se connecter à un StatefulSet CN-MGMT.

1. Vérifiez que vous avez modifié les fichiers YAML pan-cn-mgmt et pan-cn-mgmt-configmap.

   Exemple de pan-cn-mgmt-configmap

```
apiVersion: v1 kind: ConfigMap metadata: name: pan-mgmt-
config namespace: kube-system data: PAN_SERVICE_NAME:
 pan-mgmt-svc PAN_MGMT_SECRET: pan-mgmt-secret # Panorama
 settings PAN_PANORAMA_IP: "<panorama-IP>" PAN_DEVICE_GROUP:
 "<panorama-device-group>" PAN_TEMPLATE_STACK: "<panorama-
template-stack>" PAN_CGNAME: "<panorama-collector-group>"#Non-
mandatory parameters # Recommended to have same name as
 the cluster name provided in Panorama Kubernetes plugin
 - helps with easier identification of pods if managing
 multiple clusters with same Panorama #CLUSTER_NAME: "<Cluster
 name>" #PAN_PANORAMA_IP2: "" # Comment out to use CERTs
 otherwise PSK for IPSec between pan-mgmt and pan-ngfw
 #IPSEC_CERT_BYPASS: "" # No values needed # Override auto-
detect of jumbo-frame mode and force enable system-wide
 #PAN_JUMBO_FRAME_ENABLED: "true" # Start MGMT pod with GTP
 enabled. For complete functionality, need GTP # enable at
 Panorama as well. #PAN_GTP_ENABLED: "true" # Enable high
 feature capacities. These need high memory for MGMT pod and
 # higher/matching memory than specified below for NGFW pod.
 # Refer to the system requirements documentation to see
```



```
 the max supported NGFW CPU size # supported for each memory
 profile. #PAN_NGFW_MEMORY: "6.5Gi" #PAN_NGFW_MEMORY: "48Gi"
 #PAN_NGFW_MEMORY: "56Gi"
```

Exemple pan-cn-mgmt-dynamic-pv.yaml

```
initContainers: - name: pan-mgmt-init image: <your-private-
registry-image-path>
```

```
 containers: - name: pan-mgmt image: <your-private-registry-
image-path> terminationMessagePolicy: FallbackToLogsOnError
```

2. Utilisez Kubectl pour exécuter les fichiers yaml.

   kubectl apply -f pan-cn-mgmt-configmap.yaml

   kubectl apply -f pan-cn-mgmt-secret.yaml

   kubectl apply -f pan-cn-mgmt-dynamic-pv.yaml

   Vous devez exécuter pan-mgmt-serviceaccount.yaml, uniquement si vous n'aviez pas déjà terminé la création de comptes de service pour l'authentification de cluster avec CN-Series.

3. Vérifiez que les pods CN-MGMT sont activés.

   Cela prend environ 5-6 minutes.

   Utilisez **kubectl get pods -l app=pan-mgmt -n kube-system**

   NOM PRÊT ÉTAT REDÉMARRE AGEpan-mgmt-sts-0 1/1

   Fonctionnement 0 27hpan-mgmt-sts-1 1/1 Exécution 0 27h

STEP 6 | Déployez les pods CN-NGFW.

Par défaut, le pod CN-NGFW du plan de données du pare-feu est déployé comme un DaemonSet. Une instance du pod CN-NFGW peut sécuriser le trafic pour un maximum de 30 pods d'application sur un nœud.

1. Vérifiez que vous avez modifié les fichiers YAML comme indiqué dans PAN-CN-NGFW-CONFIGMAP et PAN-CN-NGFW.

   containers: - name: pan-ngfw-container image: <your-private-registry-image-path>

2. Utilisez l'application Kubectl pour exécuter le fichier pan-cn-ngfw-configmap.yaml.

   kubectl apply -f pan-cn-ngfw-configmap.yaml

3. Utilisez l'application Kubectl pour exécuter le pan-cn-ngfw.yaml.

   kubectl apply -f pan-cn-ngfw.yaml

4. Vérifiez que tous les pods CN-NGFW sont en cours d'exécution (un par nœud dans votre cluster)

   Il s'agit d'un exemple de résultat provenant d'un cluster de 4 nœuds sur site.

   kubectl get pods -n kube-system -l app=pan-ngfw -o wide

   ```
   NAME READY STATUS RESTARTS AGE IP NODE NOMINATED NODE READINESS GATES

   pan-ngfw-ds-8g5xb 1/1 Running 0 27h 10.233.71.113 rk-k8-node-1 <none> <none>

   pan-ngfw-ds-qsrm6 1/1 Running 0 27h 10.233.115.189 rk-k8-vm-worker-1 <none> <none>

   pan-ngfw-ds-vqk7z 1/1 Running 0 27h 10.233.118.208 rk-k8-vm-worker-3 <none> <none>

   pan-ngfw-ds-zncqg 1/1 Running 0 27h 10.233.91.210 rk-k8-vm-worker-2 <none> <none>
   ```


STEP 7 | Vérifiez que vous pouvez voir CN-MGMT, le CN-NGFW et le PAN-CNI sur le cluster Kubernetes.

```
kubectl -n kube-system get pods
0 27hpan-cni-5fhbg 1/1 En cours d'exécution
0 27hpan-cni-9j4rs 1/1 En cours d'exécution
0 27hpan-cni-ddwb4 1/1 En cours d'exécution
0 27hpan-cni-fwfrk 1/1 En cours d'exécution
0 27hpan-cni-h57lm 1/1 En cours d'exécution
0 27hpan-cni-h57lm 1/1 En cours d'exécution
0 27hpan-cni-j62rk 1/1 En cours d'exécution
0 27hpan-cni-lmxdz 1/1 En cours d'exécution
0 27hpan-mgmt-sts-0 1/1 En cours d'exécution
0 27hpan-mgmt-sts-1 1/1 En cours d'exécution
0 27hpan-ngfw-ds-8g5xb 1/1 En cours d'exécution
27hpan-ngfw-ds-qsrm6 1/1 En cours d'exécution
0 27hpan-ngfw-ds-vqk7z 1/1 En cours d'exécution
0 27hpan-ngfw-ds-zncqg 1/1 En cours d'exécution
```

STEP 8 | Annotez l'application yaml ou l'espace de noms afin que le trafic de leurs nouveaux pods soit redirigé vers le pare-feu.

Vous devez ajouter l'annotation suivante pour rediriger le trafic vers le CN-NGFW pour inspection :

```
annotations: paloaltonetworks.com/firewall: pan-fw
```

Par exemple, pour tous les nouveaux pods dans l'espace de noms « par défaut :

```
kubectl annotate namespace default paloaltonetworks.com/
firewall=pan-fw
```

Sur certaines plateformes, les pods de l'application peuvent démarrer lorsque le pan-cni n'est pas actif dans la chaîne de plug-ins CNI. Pour éviter de tels scénarios, vous devez spécifier les volumes comme indiqué ici dans le pod d'application YAML.

```
volumes: - name: pan-cni-ready hostPath: path: /var/log/
pan-appinfo/pan-cni-ready type: Répertoire
```

STEP 9 | Déployez votre application dans le cluster.

Déployer le pare-feu CN-Series dans EKS

Où puis-je utiliser ceci ?	De quoi ai-je besoin ?
• Déploiement CN-Series	• CN-Series 10.1.x or above Container Images • Panorama sous PAN-OS 10.1.x ou une version supérieure • Helm 3.6 or above version client pour le déploiement CN-Series à l'aide de Helm

Après avoir examiné les blocs de construction CN-Series et la présentation générale du flux de travail dans Sécuriser les environnements Kubernetes avec CN-Series , vous pouvez commencer à déployer les pare-feu CN-Series sur la plate-forme AWS EKS pour sécuriser le trafic entre les conteneurs au sein du même cluster, ainsi qu'entre les conteneurs et d'autres types de charges de travail tels que les machines virtuelles et les serveurs bare-metal.

Vous avez besoin d'outils Kubernetes standard tels que kubectl ou Helm pour déployer et gérer vos applications, vos services pare-feu et vos clusters Kubernetes.

Pour plus d'informations, consultez Déployer des pare-feu CN-Series avec des graphiques et des modèles Helm. *Panorama n'est pas conçu pour être utilisé comme orchestrateur pour le déploiement et la gestion de clusters Kubernetes. Les modèles pour la gestion des clusters sont fournis par les fournisseurs de Kubernetes gérés. Palo Alto Networks fournit des modèles pris en charge par la communauté pour le déploiement CN-Series avec* Helm *et* Terraform.

- Déploiement du pare-feu CN-Series en tant que service Kubernetes dans AWS EKS
- Déploiement du pare-feu CN-Series en tant que DaemonSet dans AWS EKS
- Déployer le pare-feu CN-Series à partir d'AWS Marketplace

Avant de passer du déploiement de CN-Series en tant que DaemonSet à CN-Series en tant que service ou vice versa, vous devez supprimer et réappliquer `plugin-serviceaccount.yaml`. *Pour plus d'informations, consultez* Créer des comptes de service pour l'authentification des clusters.

- *Lorsque vous déployez CN-Series en tant que DaemonSet dans EKS, le* `pan-plugin-cluster-mode-secret` *ne doit pas exister.*
- *Lorsque vous déployez CN-Series en tant que service Kubernetes dans EKS, le* `pan-plugin-cluster-mode-secret` *doit être présent.*

Déploiement du pare-feu CN-Series en tant que service Kubernetes dans AWS EKS

Où puis-je utiliser ceci ?	De quoi ai-je besoin ?
• Déploiement CN-Series	• CN-Series 10.1.x or above Container Images • Panorama sous PAN-OS 10.1.x ou une version supérieure • Helm 3.6 or above version client pour le déploiement CN-Series avec Helm

Effectuez la procédure suivante pour déployer le pare-feu CN-Series en tant que service Kubernetes.

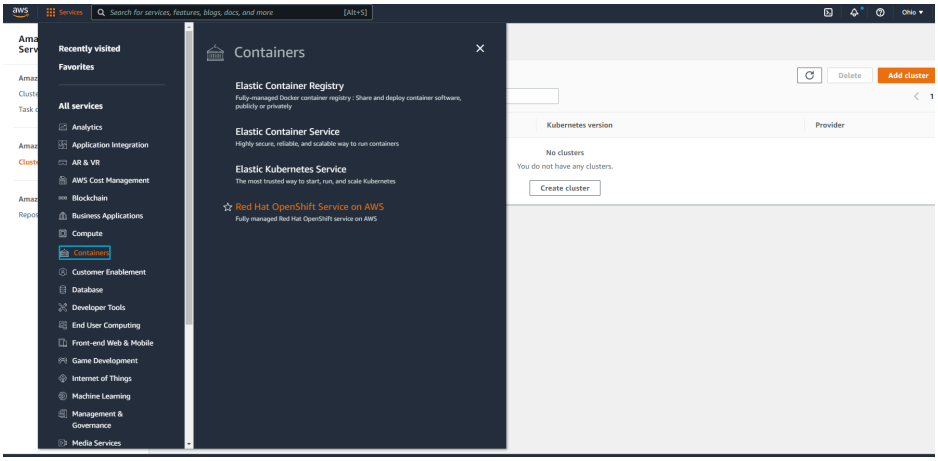
Avant de commencer, assurez-vous que la version du fichier YAML CN-Series est compatible avec la version PAN-OS.

- PAN-OS 10.1.2 ou version ultérieure nécessite YAML 2.0.2
- PAN-OS 10.1.0 et 10.1.1 nécessitent YAML 2.0.0 ou 2.0.1

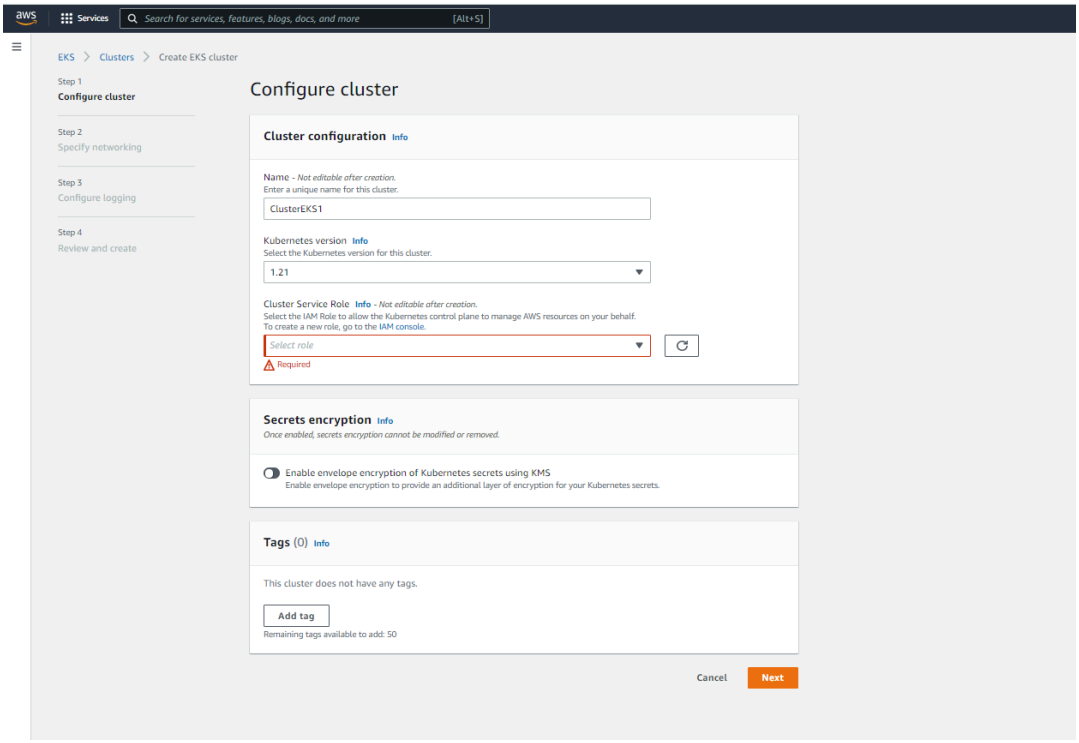
STEP 1 | Configurez votre cluster Kubernetes.

Pour créer un cluster dans AWS EKS, procédez comme suit :

1. Cliquez sur le menu de navigation **Services**, accédez à **Containers (Conteneurs)->Elastic Kubernetes Service (Service Elastic Kubernetes)**.



2. Cliquez sur **Create Cluster (Créer un cluster)**.
3. Renseignez les détails requis, puis cliquez sur **Create (Créer)**.



1. Vérifiez que le cluster dispose des ressources adéquates. Assurez-vous que ce cluster dispose des conditions préalables de CN-Series pour prendre en charge le pare-feu :

```
kubectl get nodes
```

```
kubectl describe node <node-name>
```


Affichez les informations sous l'en-tête Capacity (Capacité) dans la sortie de la commande pour voir le processeur et la mémoire disponibles sur le nœud spécifié.

L'allocation du processeur, de la mémoire et du stockage sur disque dépendra de vos besoins. Voir Performances et mise à l'échelle de CN-Series.

Assurez-vous d'avoir les informations suivantes :

- Collectez l'adresse IP du terminal pour configurer le serveur API sur Panorama. Panorama utilise cette adresse IP pour se connecter à votre cluster Kubernetes.
- Collectez le nom de la pile de modèles, le nom du groupe d'appareils, l'adresse IP Panorama et éventuellement le nom du groupe du collecteur de journaux à partir de Panorama.
- Collectez le code d'autorisation et l'ID et la valeur du code PIN d'enregistrement automatique.
- L'emplacement du conteneur d'images dans lequel vous avez téléchargé les images.

STEP 2 | (facultatif) Si vous avez configuré un certificat personnalisé dans le plug-in Kubernetes pour Panorama, vous devez créer le secret de certificat en exécutant la commande suivante. Ne modifiez pas le nom de fichier de ca.crt. Le volume des certificats personnalisés dans pan-cn-mgmt.yaml et pan-cn-ngfw.yaml est facultatif.

```
kubectl -n kube-system crée un secret générique custom-ca --from-file=ca.crt
```

STEP 3 | Modifiez les fichiers YAML afin de fournir les détails nécessaires au déploiement des pare-feu CN-Series.

Vous devez remplacer le chemin d'accès de l'image dans les fichiers YAML pour inclure le chemin d'accès à votre registre privé et fournir les paramètres requis. Pour plus d'informations, consultez Paramètres modifiables dans les fichiers yaml de déploiement CN-Series.

STEP 4 | Mettez à jour la classe de stockage. Pour prendre en charge CN-Series déployé sur AWS Outpost, vous devez utiliser le pilote de stockage aws-ebs-csi-driver, qui garantit qu'Outpost extrait les volumes d'Outpost lors de la création dynamique de volume persistant (PV).

1. Appliquez le yaml suivant.

   ```
   kubectl apply -k "github.com/kubernetes-sigs/aws-ebs-csi-driver/deploy/kubernetes/overlays/stable/?ref=release-0.10"
   ```

2. Vérifiez que le contrôleur ebs-sc est en cours d'exécution.

   ```
   kubectl -n kube-system get pods
   ```

3. Mettez à jour pan-cn-storage-class.yaml pour qu'il corresponde à l'exemple ci-dessous.

   ```
   apiVersion: v1 kind: StorageClass apiVersion: storage.k8s.io/v1 metadata: name: ebs-sc provisioner: ebs.csi.aws.com
    volumeBindingMode: WaitForFirstConsumer parameters: type: gp2
   ```

4. Ajoutez **`storageClassName: ebs-sc`** à pan-cn-mgmt.yaml aux emplacements indiqués ci-dessous.

   ```
   volumeClaimTemplates: - metadata: name: panlogs spec:
    #storageClassName: pan-cn-storage-class //For better disk
   ```



```
iops performance for logging accessModes: [ "ReadWriteOnce" ]
storageClassName: ebs-sc // resources: requests: storage:
20Gi # change this to 200Gi while using storageClassName
for better disk iops - metadata: name: varlogpan spec:
#storageClassName: pan-cn-storage-class //For better disk
iops performance for dp logs accessModes: [ "ReadWriteOnce" ]
storageClassName: ebs-sc resources: requests: storage: 20Gi #
change this to 200Gi while using storageClassName for better
disk iops - metadata: name: varcores spec: accessModes:
[ "ReadWriteOnce" ] storageClassName: ebs-sc resources:
requests: storage: 2Gi - metadata: name: panplugincfg spec:
accessModes: [ "ReadWriteOnce" ] storageClassName: ebs-sc
resources: requests: storage: 1Gi - metadata: name: panconfig
spec: accessModes: [ "ReadWriteOnce" ] storageClassName:
ebs-sc resources: requests: storage: 8Gi - metadata:
name: panplugins spec: accessModes: [ "ReadWriteOnce" ]
storageClassName: ebs-sc resources: requests: storage: 200Mi
```

STEP 5 | Si vous utilisez l'autoscaling dans votre environnement Kubernetes, procédez comme suit :

1. Déployez l'adaptateur Amazon CloudWatch Metrics pour Kubernetes dans votre cluster CN-Series en tant que service. Vous devez autoriser CloudWatch à accéder complètement aux deux rôles IAM associés à vos pods et clusters Kubernetes. Pour publier les métriques personnalisées dans CloudWatch, le rôle des nœuds esclaves doit disposer de la politique gérée par AWS **CloudWatchAgentServerPolicy** afin que la fonction HPA puisse les récupérer.
2. Téléchargez les fichiers yaml HPA spécifiques à EKS à partir du référentiel GitHub de Palo Alto Networks.
3. Si votre CN-MGMT est déployé dans un espace de noms personnalisé, mettez à jour pan-cn-adapater.yaml avec l'espace de noms personnalisé. L'espace de noms par défaut est **kube-system**.

4. Modifiez **`pan-cn-hpa-dp.yaml`** et **`pan-cn-hpa-mp.yaml`**.
   1. Entrez le nombre minimal et maximal de réplicas.
   2. (Facultatif) Modifiez les valeurs de fréquence de mise à l'échelle et de montée en puissance en fonction de votre déploiement. Si vous ne modifiez pas ces valeurs, les valeurs par défaut sont utilisées.
   3. Copiez la section suivante pour chaque métrique que vous souhaitez utiliser pour la mise à l'échelle.

      ```
      - type: Pods pods: metric: name: pansessionactive target: type:
       AverageValue averageValue: 30
      ```

   4. Modifiez le nom de la métrique que vous souhaitez utiliser et définissez **`averageValue`** sur le seuil décrit dans le tableau ci-dessus. Si vous ne modifiez pas ces valeurs, les valeurs par défaut sont utilisées.
   5. Enregistrez vos modifications.

      Pour plus d'informations, consultez Autoscaling horizontal des pods.
5. Déployez les fichiers yaml HPA. Les fichiers doivent être déployés dans l'ordre décrit ci-dessous.
   1. Utiliser Kubectl pour exécuter le fichier pan-cn-adapter.yaml

      ```
      kubectl apply -f pan-cn-adapter.yaml
      ```

   2. Utiliser Kubectl pour exécuter le fichier pan-cn-externalmetrics.yaml

      ```
      kubectl apply -f pan-cn-externalmetrics.yaml
      ```

   3. Utiliser Kubectl pour exécuter le fichier pan-cn-hpa-dp.yaml

      ```
      kubectl apply -f pan-cn-hpa-dp.yaml
      ```

   4. Utiliser Kubectl pour exécuter le fichier pan-cn-hpa-mp.yaml

      ```
      kubectl apply -f pan-cn-hpa-mp.yaml
      ```
6. Vérifiez votre déploiement.

   Utilisez kubectl pour vérifier que le pod d'adaptateur de mesures personnalisées dans l'espace de noms de mesures personnalisées.

   ```
   kubectl get pods -n custom-metrics
   ```

   Utilisez kubectl pour rechercher la ressource HPA.

   ```
   kubectl get hpa -n kube-system
   ```

   ```
   kubectl describe hpa <hpa-name> -n kube-system
   ```


STEP 6 | Déployez le service CN-NGFW.

1. Vérifiez que vous avez créé le compte de service à l'aide du fichier pan-cni-serviceaccount.yaml.

   Consultez Création de comptes de service pour l'authentification des clusters.

2. Utilisez Kubectl pour exécuter le fichier pan-cni-configmap.yaml.

   kubectl apply -f pan-cni-configmap.yaml

3. Utilisez kubectl pour exécuter le fichier pan-cn-ngfw-svc.yaml.

   kubectl apply -f pan-cn-ngfw-svc.yaml

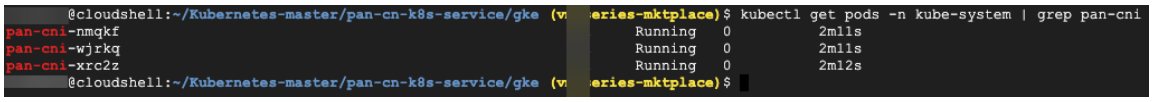
   Ce fichier yaml doit être déployé avant pan-cni.yaml.

4. Utilisez Kubectl pour exécuter le fichier pan-cni.yaml.

   kubectl apply -f pan-cni.yaml

5. Vérifiez que vous avez modifié les fichiers YAML pan-cni et pan-cni-configmap.
6. Exécutez la commande suivante et vérifiez que votre sortie est similaire à l'exemple suivant.

   kubectl get pods -n kube-system | grep pan-cni



STEP 7 | Déployez le StatefulSet CN-MGMT.

Par défaut, le plan de gestion est déployé comme un StatefulSet qui garantie la tolérance aux pannes. Jusqu'à 30 pods pare-feu CN-NGFW peuvent se connecter à un StatefulSet CN-MGMT.

1. (Requis uniquement pour les PV provisionnés statiquement) Déployez les volumes persistants (PV) pour le StatefulSet CN-MGMT.

   1. Créez les répertoires qui correspondent aux noms des volumes locaux définis dans le fichier pan-cn-pv-local.yaml.

   Vous avez besoin de six (6) répertoires sur au moins 2 nœuds esclaves. Connectez-vous à chaque nœud esclave sur lequel le StatefulSet CN-MGMT sera déployé pour créer les

répertoires. Par exemple, pour créer des répertoires nommés /mnt/pan-local1 vers /mnt/pan-local6, utilisez la commande :

```
mkdir -p /mnt/pan-local1 /mnt/pan-local2 /mnt/pan-local3 /
mnt/pan-local4 /mnt/pan-local5 /mnt/pan-local6
```

2. Modifiez pan-cn-pv-local.yaml.

Faites correspondre le nom d'hôte sous `nodeaffinity`, et vérifiez que vous avez modifié les répertoires que vous avez créés ci-dessus dans `spec.local.path` puis déployez le fichier pour créer une nouvelle storage class pan-local-storage et des PV locaux.

2. Vérifiez que vous avez modifié les fichiers YAML pan-cn-mgmt et pan-cn-mgmt-configmap.

Exemple de pan-cn-mgmt-configmap de l'EKS.

```
 apiVersion: v1 kind: ConfigMap metadata: name: pan-mgmt-
config namespace: kube-system data: PAN_SERVICE_NAME:
 pan-mgmt-svc PAN_MGMT_SECRET: pan-mgmt-secret # Panorama
 settings PAN_PANORAMA_IP: "<panorama-IP>" PAN_DEVICE_GROUP:
 "<panorama-device-group>" PAN_TEMPLATE_STACK: "<panorama-
template-stack>" PAN_CGNAME: "<panorama-collector-
group>" # ctnr mode: "k8s-service", "k8s-ilbservice"
 PAN_CTNR_MODE_TYPE: "k8s-service" #Non-mandatory parameters #
 Recommended to have same name as the cluster name provided in
 Panorama Kubernetes plugin - helps with easier identification
 of pods if managing multiple clusters with same Panorama
 #CLUSTER_NAME: "<Cluster name>" #PAN_PANORAMA_IP2: "" #
 Comment out to use CERTs otherwise PSK for IPSec between pan-
mgmt and pan-ngfw #IPSEC_CERT_BYPASS: "" # No values needed
 # Override auto-detect of jumbo-frame mode and force enable
 system-wide #PAN_JUMBO_FRAME_ENABLED: "true" # Start MGMT
 pod with GTP enabled. For complete functionality, need GTP #
 enable at Panorama as well. #PAN_GTP_ENABLED: "true" # Enable
 high feature capacities. These need high memory for MGMT pod
 and # higher/matching memory than specified below for NGFW
 pod. #PAN_NGFW_MEMORY="6Gi" #PAN_NGFW_MEMORY="40Gi" # For
 enabling faster datapath - AF_XDP, default is AF_PACKETV2.
 This requires kernel support. #PAN_DATA_MODE: "next-gen" #HPA
 params #PAN_CLOUD: "EKS" #PAN_NAMESPACE_EKS: "EKSNamespace"
 #PUSH_INTERVAL: "15" #time interval to publish metrics to AWS
 cloudwatch
```

Exemple de pan-cn-mgmt.yaml

```
initContainers: - name: pan-mgmt-init image: <your-private-
registry-image-path>
```

```
 containers: - name: pan-mgmt image: <your-private-registry-
image-path> terminationMessagePolicy: FallbackToLogsOnError
```


3. Utilisez Kubectl pour exécuter les fichiers yaml.

   ```
   kubectl apply -f pan-cn-mgmt-configmap.yaml
   kubectl apply -f pan-cn-mgmt-slot-crd.yaml
   kubectl apply -f pan-cn-mgmt-slot-cr.yaml
   kubectl apply -f pan-cn-mgmt-secret.yaml
   kubectl apply -f pan-cn-mgmt.yaml
   ```

   Vous devez exécuter pan-mgmt-serviceaccount.yaml, uniquement si vous n'aviez pas déjà terminé la création de compte de service pour l'authentification de cluster.

4. Vérifiez que les pods CN-MGMT sont activés.

   Cela prend environ 5-6 minutes.

   Utilisez **`kubectl get pods -l app=pan-mgmt -n kube-system`**

STEP 8 | Déployez les pods CN-NGFW.

1. Vérifiez que vous avez modifié les fichiers YAML comme indiqué dans PAN-CN-NGFW-CONFIGMAP et PAN-CN-NGFW.

   `containers: - name: pan-ngfw-container image: <your-private-registry-image-path>`

2. Utilisez l'application Kubectl pour exécuter le fichier pan-cn-ngfw-configmap.yaml.

   `kubectl apply -f pan-cn-ngfw-configmap.yaml`

3. Utilisez l'application Kubectl pour exécuter le pan-cn-ngfw.yaml.

   `kubectl apply -f pan-cn-ngfw.yaml`

4. Vérifiez que les pods CN-NGFW sont en cours d'exécution.

   `kubectl get pods -n kube-system -l app=pan-ngfw -o wide`

STEP 9 | Activer la mise à l'échelle automatique horizontale du pod sur CN-Series.

STEP 10 | Vérifiez que vous pouvez voir CN-MGMT, le CN-NGFW et le PAN-CNI sur le cluster Kubernetes.

```
kubectl -n kube-system get pods
```


STEP 11 | Annotez l'application yaml ou l'espace de noms afin que le trafic de leurs nouveaux pods soit redirigé vers le pare-feu.

Vous devez ajouter l'annotation suivante pour rediriger le trafic vers le CN-NGFW pour inspection :

```
annotations: paloaltonetworks.com/firewall: pan-fw
```

Par exemple, pour tous les nouveaux pods dans l'espace de noms « par défaut :

```
kubectl annotate namespace default paloaltonetworks.com/
firewall=pan-fw
```

Sur certaines plateformes, les pods de l'application peuvent démarrer lorsque le pan-cni n'est pas actif dans la chaîne de plug-ins CNI. Pour éviter de tels scénarios, vous devez spécifier les volumes comme indiqué ici dans le pod d'application YAML.

```
volumes: - name: pan-cni-ready hostPath: path: /var/log/
pan-appinfo/pan-cni-ready type: Répertoire
```

STEP 12 | Déployez votre application dans le cluster.

Déploiement du pare-feu CN-Series en tant que DaemonSet dans AWS EKS

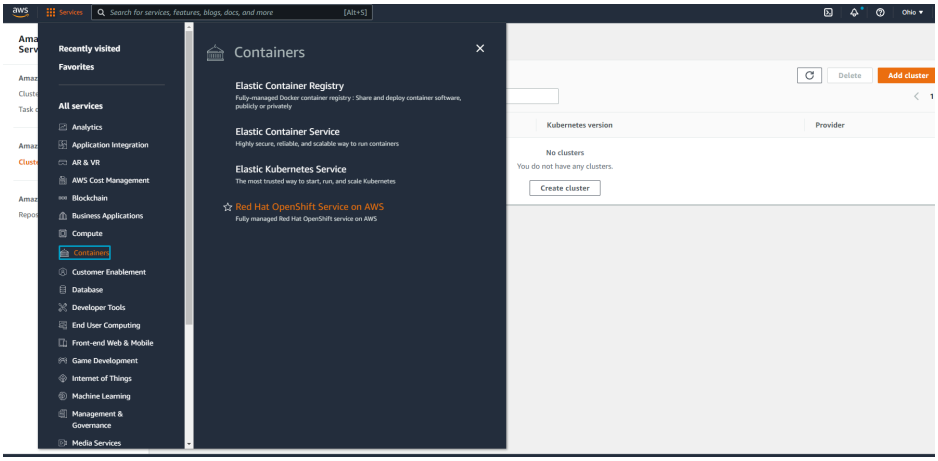
Où puis-je l'utiliser ?	De quoi ai-je besoin ?
• CN-Series déploiement	• CN-Series 10.1.x or above Container Images • Panorama sous PAN-OS 10.1.x ou une version supérieure • Helm 3.6 or above version client pour le déploiement CN-Series à l'aide de Helm

Procédez comme suit pour déployer le pare-feu CN-Series en tant que dameonset dans AWS EKS :

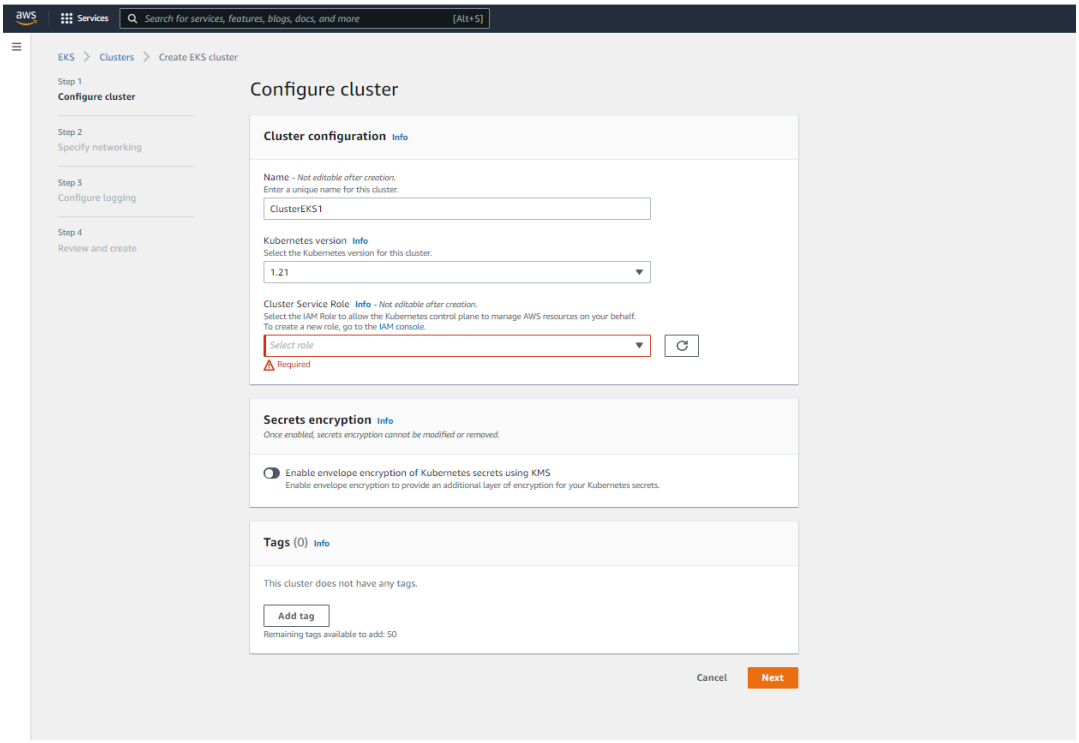
STEP 1 | Configurez votre cluster Kubernetes.

Pour créer un cluster dans AWS EKS, procédez comme suit :

1. Cliquez sur le menu de navigation **Services**, accédez à **Containers (Conteneurs)->Elastic Kubernetes Service (Service Elastic Kubernetes)**.



2. Cliquez sur **Create Cluster (Créer un cluster)**.
3. Renseignez les détails requis, puis cliquez sur **Create (Créer)**.



Vérifiez que le cluster dispose des ressources adéquates. Assurez-vous que ce cluster dispose des conditions préalables de CN-Series pour prendre en charge le pare-feu.

```
kubectl get nodes
```

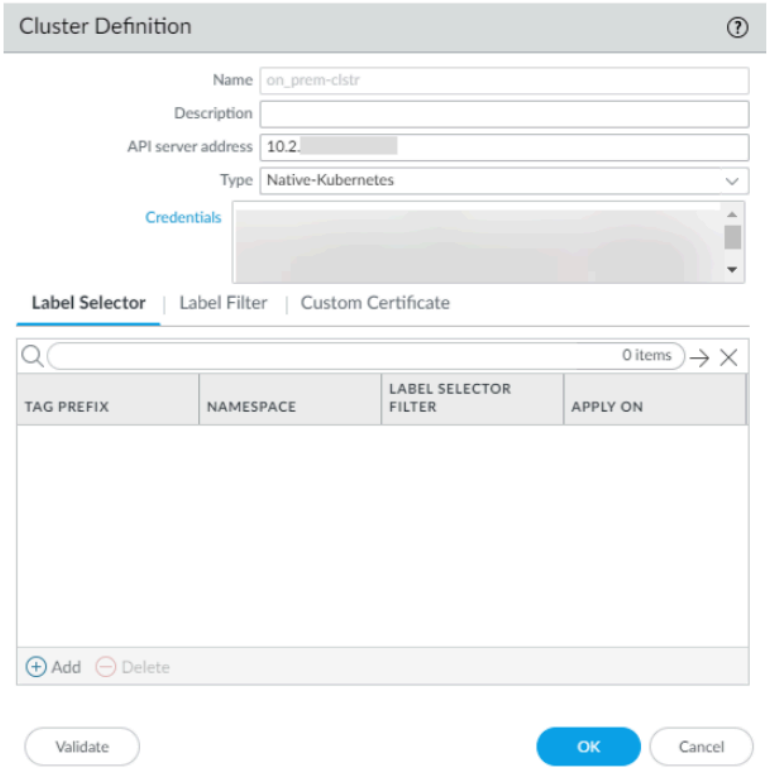
```
kubectl describe node <node-name>
```


Affichez les informations sous l'en-tête Capacity (Capacité) dans la sortie de la commande pour voir le processeur et la mémoire disponibles sur le nœud spécifié.

L'allocation du processeur, de la mémoire et du stockage sur disque dépendra de vos besoins. Voir Performances et mise à l'échelle de CN-Series.

Assurez-vous d'avoir les informations suivantes :

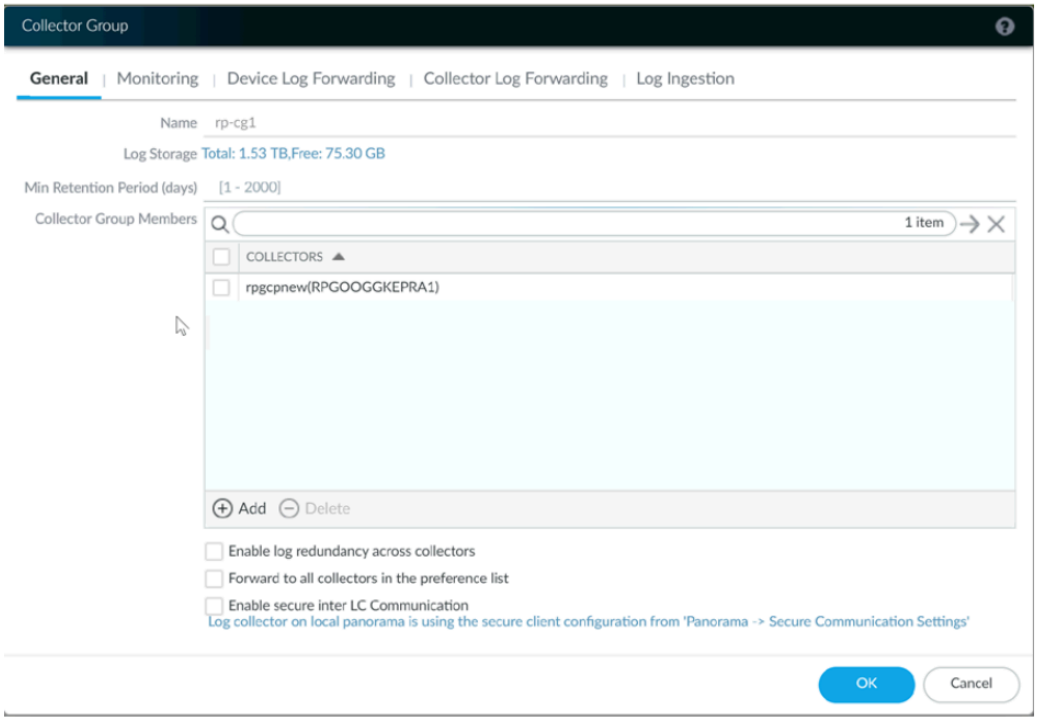
- Collectez l'adresse IP du terminal pour configurer le serveur API sur Panorama.



  Panorama utilise cette adresse IP pour se connecter à votre cluster Kubernetes.

  Pour plus d'informations, consultez Configurer le plug-in Kubernetes pour la surveillance des clusters.

- Collectez le nom de la pile de modèles, le nom du groupe d'appareils, l'adresse IP Panorama et éventuellement le nom du groupe du collecteur de journaux à partir de Panorama.



Pour plus d'informations, consultez Créer un groupe d'appareils parents et une pile de modèles.

- Collectez le code d'autorisation et l'ID et la valeur du code PIN d'enregistrement automatique.
- L'emplacement du conteneur d'images dans lequel vous avez téléchargé les images.

STEP 2 | (facultatif) Si vous avez configuré un certificat personnalisé dans le plug-in Kubernetes pour Panorama, vous devez créer le secret de certificat en exécutant la commande suivante. Ne modifiez pas le nom de fichier de ca.crt. Le volume des certificats personnalisés dans pan-cn-mgmt.yaml et pan-cn-ngfw.yaml est facultatif.

```
kubectl -n kube-system crée un secret générique custom-ca --from-file=ca.crt
```

STEP 3 | Modifiez les fichiers YAML afin de fournir les détails nécessaires au déploiement des pare-feu CN-Series.

Vous devez remplacer le chemin d'accès de l'image dans les fichiers YAML pour inclure le chemin d'accès à votre répertoire privé Google Container et fournir les paramètres requis. Pour plus d'informations, consultez Paramètres modifiables dans les fichiers yaml de déploiement CN-Series.

STEP 4 | Déployez le DaemonSet CNI.

Le conteneur CNI est déployé comme un DaemonSet (un pod par nœud) et il crée deux interfaces sur le pod CN-NGFW pour chaque application déployée sur le nœud. Lorsque vous utilisez les commandes kubectl pour exécuter les fichiers YAML pan-cni, il devient une partie de la chaîne de service sur chaque nœud.

1. Le pare-feu CN-Series nécessite trois comptes de service avec les autorisations minimales qui l'autorisent à communiquer avec les ressources de votre cluster Kubernetes. Vous devez créer

Création d’un compte de service pour l’authentification du cluster et vérifier que vous avez créé le compte de service à l’aide du fichier pan-cni-serviceaccount.yaml.

2. Utilisez Kubectl pour exécuter le fichier pan-cni-configmap.yaml.

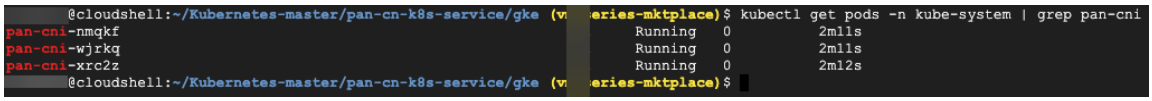
   kubectl apply -f pan-cni-configmap.yaml

3. Utilisez Kubectl pour exécuter le fichier pan-cni.yaml.

   kubectl apply -f pan-cni.yaml

4. Vérifiez que vous avez modifié les fichiers YAML pan-cni et pan-cni-configmap.

5. Exécutez la commande suivante et vérifiez que votre sortie est similaire à l’exemple suivant.



STEP 5 | Mettez à jour la classe de stockage. Pour prendre en charge CN-Series déployé sur AWS Outpost, vous devez utiliser le pilote de stockage aws-ebs-csi-driver, qui garantit qu’Outpost extrait les volumes d’Outpost lors de la création dynamique de volume persistant (PV).

1. Appliquez le yaml suivant.

   kubectl apply -k "github.com/kubernetes-sigs/aws-ebs-csi-driver/deploy/kubernetes/overlays/stable/?ref=release-0.10"

2. Vérifiez que le contrôleur ebs-sc est en cours d’exécution.

   kubectl -n kube-system get pods

3. Mettez à jour pan-cn-storage-class.yaml pour qu’il corresponde à l’exemple ci-dessous.

```
apiVersion: v1 kind: StorageClass apiVersion: storage.k8s.io/
v1 metadata: name: ebs-sc provisioner: ebs.csi.aws.com
 volumeBindingMode: WaitForFirstConsumer parameters: type: gp2
```

4. Ajoutez **storageClassName: ebs-sc** à pan-cn-mgmt.yaml aux emplacements indiqués ci-dessous.

```
volumeClaimTemplates: - metadata: name: panlogs spec:
 #storageClassName: pan-cn-storage-class //For better disk
 iops performance for logging accessModes: [ "ReadWriteOnce" ]
 storageClassName: ebs-sc // resources: requests: storage:
 20Gi # change this to 200Gi while using storageClassName
 for better disk iops - metadata: name: varlogpan spec:
 #storageClassName: pan-cn-storage-class //For better disk
 iops performance for dp logs accessModes: [ "ReadWriteOnce" ]
 storageClassName: ebs-sc resources: requests: storage: 20Gi #
 change this to 200Gi while using storageClassName for better
 disk iops - metadata: name: varcores spec: accessModes:
 [ "ReadWriteOnce" ] storageClassName: ebs-sc resources:
 requests: storage: 2Gi - metadata: name: panplugincfg spec:
 accessModes: [ "ReadWriteOnce" ] storageClassName: ebs-sc
 resources: requests: storage: 1Gi - metadata: name: panconfig
 spec: accessModes: [ "ReadWriteOnce" ] storageClassName:
 ebs-sc resources: requests: storage: 8Gi - metadata:
```



```
 name: panplugins spec: accessModes: [ "ReadWriteOnce" ]
 storageClassName: ebs-sc resources: requests: storage: 200Mi
```

STEP 6 | Déployez le StatefulSet CN-MGMT.

Par défaut, le plan de gestion est déployé comme un StatefulSet qui garantie la tolérance aux pannes. Jusqu'à 30 pods pare-feu CN-NGFW peuvent se connecter à un StatefulSet CN-MGMT.

1. (Requis uniquement pour les PV provisionnés statiquement) Déployez les volumes persistants (PV) pour le StatefulSet CN-MGMT.

   1. Créez les répertoires qui correspondent aux noms des volumes locaux définis dans le fichier pan-cn-pv-local.yaml.

   Vous avez besoin de six (6) répertoires sur au moins 2 nœuds esclaves. Connectez-vous à chaque nœud esclave sur lequel le StatefulSet CN-MGMT sera déployé pour créer les répertoires. Par exemple, pour créer des répertoires nommés /mnt/pan-local1 vers /mnt/pan-local6, utilisez la commande :

   ```
   mkdir -p /mnt/pan-local1 /mnt/pan-local2 /mnt/pan-local3 /
   mnt/pan-local4 /mnt/pan-local5 /mnt/pan-local6
   ```

   2. Modifiez pan-cn-pv-local.yaml.

   Faites correspondre le nom d'hôte sous `nodeaffinity`, et vérifiez que vous avez modifié les répertoires que vous avez créés ci-dessus dans `spec.local.path` puis déployez le fichier pour créer une nouvelle storage class pan-local-storage et des PV locaux.

2. Vérifiez que vous avez modifié les fichiers YAML pan-cn-mgmt et pan-cn-mgmt-configmap.

   Exemple de pan-cn-mgmt-configmap de l'EKS.

```
 Session Contents Restored apiVersion: v1 kind: ConfigMap
 metadata: name: pan-mgmt-config namespace: kube-system
 data: PAN_SERVICE_NAME: pan-mgmt-svc PAN_MGMT_SECRET: pan-
mgmt-secret # Panorama settings PAN_PANORAMA_IP: "x.y.z.a"
 PAN_DEVICE_GROUP: "dg-1" PAN_TEMPLATE_STACK: "temp-stack-1"
 PAN_CGNAME: "CG-EKS" # Intended License Bundle type - "CN-
X-BASIC", "CN-X-BND1", "CN-X-BND2" # based on the authcode
 applied on the Panorama K8S plugin" PAN_BUNDLE_TYPE: "CN-X-
BND2" #Non-mandatory parameters # Recommended to have same
 name as the cluster name provided in Panorama Kubernetes
 plugin - helps with easier identification of pods if managing
 multiple clusters with same Panorama #CLUSTER_NAME: "Cluster-
name" #PAN_PANORAMA_IP2: "passive-secondary-ip" # Comment
 out to use CERTs otherwise bypass encrypted connection to
 etcd in pan-mgmt. # Not using CERTs for etcd due to EKS bug
 ETCD_CERT_BYPASS: "" # No value needed # Comment out to use
```



```
 CERTs otherwise PSK for IPSec between pan-mgmt and pan-ngfw #
 IPSEC_CERT_BYPASS: "" # No values needed
```

Exemple de pan-cn-mgmt.yaml

```
initContainers: - name: pan-mgmt-init image: <your-private-
registry-image-path>
```

```
 containers: - name: pan-mgmt image: <your-private-registry-
image-path> terminationMessagePolicy: FallbackToLogsOnError
```

3. Utilisez Kubectl pour exécuter les fichiers yaml.

   kubectl apply -f pan-cn-mgmt-configmap.yaml

   kubectl apply -f pan-cn-mgmt-slot-crd.yaml

   kubectl apply -f pan-cn-mgmt-slot-cr.yaml

   kubectl apply -f pan-cn-mgmt-secret.yaml

   kubectl apply -f pan-cn-mgmt.yaml

   Vous devez exécuter pan-mgmt-serviceaccount.yaml, uniquement si vous n'aviez pas déjà terminé la création de comptes de service pour l'authentification de cluster avec le pare-feu CN-Series.

4. Vérifiez que les pods CN-MGMT sont activés.

   Cela prend environ 5-6 minutes.

   Utilisez **kubectl get pods -l app=pan-mgmt -n kube-system**

   ```
   NAME READY STATUS RESTARTS AGEpan-mgmt-sts-0 1/1 Running 0
   27hpan-mgmt-sts-1 1/1 Running 0 27h
   ```


STEP 7 | Déployez les pods CN-NGFW.

Par défaut, le pod CN-NGFW du plan de données du pare-feu est déployé comme un DaemonSet. Une instance du pod CN-NFGW peut sécuriser le trafic pour un maximum de 30 pods d'application sur un nœud.

1. Vérifiez que vous avez modifié les fichiers YAML comme indiqué dans PAN-CN-NGFW-CONFIGMAP et PAN-CN-NGFW.

   containers: - name: pan-ngfw-container image: <your-private-registry-image-path>

2. Utilisez l'application Kubectl pour exécuter le fichier pan-cn-ngfw-configmap.yaml.

   kubectl apply -f pan-cn-ngfw-configmap.yaml

3. Utilisez l'application Kubectl pour exécuter le pan-cn-ngfw.yaml.

   kubectl apply -f pan-cn-ngfw.yaml

4. Vérifiez que tous les pods CN-NGFW sont en cours d'exécution (un par nœud dans votre cluster)

   Il s'agit d'un exemple de résultat provenant d'un cluster de 4 nœuds sur site.

   kubectl get pods -n kube-system -l app=pan-ngfw -o wide

   ```
   NAME READY STATUS RESTARTS AGE IP NODE NOMINATED NODE READINESS GATES

   pan-ngfw-ds-8g5xb 1/1 Running 0 27h 10.233.71.113 rk-k8-node-1 <none> <none>

   pan-ngfw-ds-qsrm6 1/1 Running 0 27h 10.233.115.189 rk-k8-vm-worker-1 <none> <none>

   pan-ngfw-ds-vqk7z 1/1 Running 0 27h 10.233.118.208 rk-k8-vm-worker-3 <none> <none>

   pan-ngfw-ds-zncqg 1/1 Running 0 27h 10.233.91.210 rk-k8-vm-worker-2 <none> <none>
   ```


STEP 8 | Vérifiez que vous pouvez voir CN-MGMT, le CN-NGFW et le PAN-CNI sur le cluster Kubernetes.

```
kubectl -n kube-system get pods
0 27hpan-cni-5fhbg 1/1 En cours d'exécution
0 27hpan-cni-9j4rs 1/1 En cours d'exécution
0 27hpan-cni-ddwb4 1/1 En cours d'exécution
0 27hpan-cni-fwfrk 1/1 En cours d'exécution
0 27hpan-cni-h57lm 1/1 En cours d'exécution
0 27hpan-cni-h57lm 1/1 En cours d'exécution
0 27hpan-cni-j62rk 1/1 En cours d'exécution
0 27hpan-cni-lmxdz 1/1 En cours d'exécution
0 27hpan-mgmt-sts-0 1/1 En cours d'exécution
0 27hpan-mgmt-sts-1 1/1 En cours d'exécution
0 27hpan-ngfw-ds-8g5xb 1/1 En cours d'exécution
27hpan-ngfw-ds-qsrm6 1/1 En cours d'exécution
0 27hpan-ngfw-ds-vqk7z 1/1 En cours d'exécution
0 27hpan-ngfw-ds-zncqg 1/1 En cours d'exécution
```

STEP 9 | Annotez l'application yaml ou l'espace de noms afin que le trafic de leurs nouveaux pods soit redirigé vers le pare-feu.

Vous devez ajouter l'annotation suivante pour rediriger le trafic vers le CN-NGFW pour inspection :

```
annotations: paloaltonetworks.com/firewall: pan-fw
```

Par exemple, pour tous les nouveaux pods dans l'espace de noms « par défaut :

```
kubectl annotate namespace default paloaltonetworks.com/
firewall=pan-fw
```

Sur certaines plateformes, les pods de l'application peuvent démarrer lorsque le pan-cni n'est pas actif dans la chaîne de plug-ins CNI. Pour éviter de tels scénarios, vous devez spécifier les volumes comme indiqué ici dans le pod d'application YAML.

```
volumes: - name: pan-cni-ready hostPath: path: /var/log/
pan-appinfo/pan-cni-ready type: Répertoire
```

STEP 10 | Déployez votre application dans le cluster.

Déployer le pare-feu CN-Series à partir d'AWS Marketplace

Où puis-je utiliser ceci ?	De quoi ai-je besoin ?
• Déploiement CN-Series	• CN-Series 10.1.x or above Container Images • Panorama sous PAN-OS 10.1.x ou une version supérieure • Helm 3.6 or above version client pour le déploiement CN-Series avec Helm

Vous pouvez concéder sous licence votre pare-feu CN-Series en tant que service Kubernetes déployé dans AWS EKS via AWS Marketplace. La CN-Series peut être concédée sous licence pour un mois, un an, deux ans ou trois ans et déployée dans EKS 1.19 et versions ultérieures ou Redhat Openshift 4.7 et versions ultérieures.

Ce produit est en avant-première.

L'utilisation de cette licence nécessite que vous mettiez à jour la politique IAM attachée à votre nœud de travail Kubernetes.

Si vous utilisez une licence PAYG achetée via AWS Marketplace pour votre déploiement CN-Series, n'ajoutez pas de code d'autorisation au plug-in Panorama pour Kubernetes.

STEP 1 | Remplissez les conditions préalables suivantes.

1. Créez votre cluster EKS ou Redhat OpenShift.
2. Déployez Panorama et installez le plug-in Kubernetes.

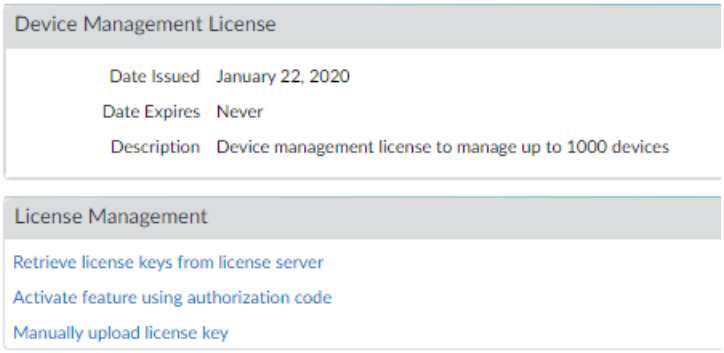
   Ignorez ces étapes si vous disposez déjà d'une instance Panorama sous licence déployée dans AWS.

   1. Installez Panorama sur une instance Amazon EC2.
   2. Installez le plug-in Kubernetes pour le CN-Series.
   3. Une fois Panorama installé, veuillez envoyer un e-mail à l'équipe CN-Series à **cn-series-aws-marketplace@paloaltonetworks.com** pour demander une licence pour votre Panorama. Veuillez inclure votre nom complet, l'adresse e-mail de votre entreprise, le nom de votre entreprise, votre numéro de bon de commande, votre nom de compte AWS et votre identifiant de compte AWS.

STEP 2 | Appliquez votre numéro de série et votre licence à Panorama.

1. Connectez-vous à l'interface Web Panorama.
2. Sélectionnez **Panorama** > **Setup (Configuration)** > **Management (Gestion)** et cliquez sur l'icône de modification .
3. Saisissez le **Serial Number (Numéro de série)** de Panorama (fourni dans l'e-mail de confirmation de commande), puis cliquez sur **OK**.
4. Sélectionnez **Panorama** > **Licenses (Licences).**
5. Cliquez sur **Activate feature using authorization code (Activer la fonction à l'aide du code d'autorisation**.
6. Saisissez le code d'autorisation de licence de gestion du pare-feu et cliquez sur **OK** pour activer la licence.
7. Vérifiez que la licence de gestion du pare-feu est activée.

   La section Device Management License (Licence de gestion du périphérique) apparaît et affiche la date d'émission de la licence, la date d'expiration de la licence et une description de la licence de gestion du pare-feu.



STEP 3 | Mettez à jour vos politiques IAM et associez la politique à votre nœud de travail Kubernetes.

1. Connectez-vous à AWS Management Console et ouvrez la console IAM.
1. Sélectionnez **Policies (Politiques)**.
2. Dans la liste des politiques, sélectionnez **AWSLicenseManagerConsumptionPolicy** et **AWSMarketplaceMeteringRegisterUsage**.
3. Sélectionnez **Actions** puis choisissez **Attach (Associer)**.
4. Sélectionnez l'identité de votre nœud de travail à laquelle associer la politique. Après avoir sélectionné l'identité, cliquez sur **Attach policy (Associer la politique)**.

STEP 4 | Téléchargez le **`plugin-serviceaccount.yaml`** et appliquez le yaml avant de déployer les graphiques Helm.

```
kubectl apply -f plugin-serviceaccount.yaml
```

STEP 5 | Accédez à AWS Marketplace et recherchez la **liste CN-Series pour AWS Marketplace** .

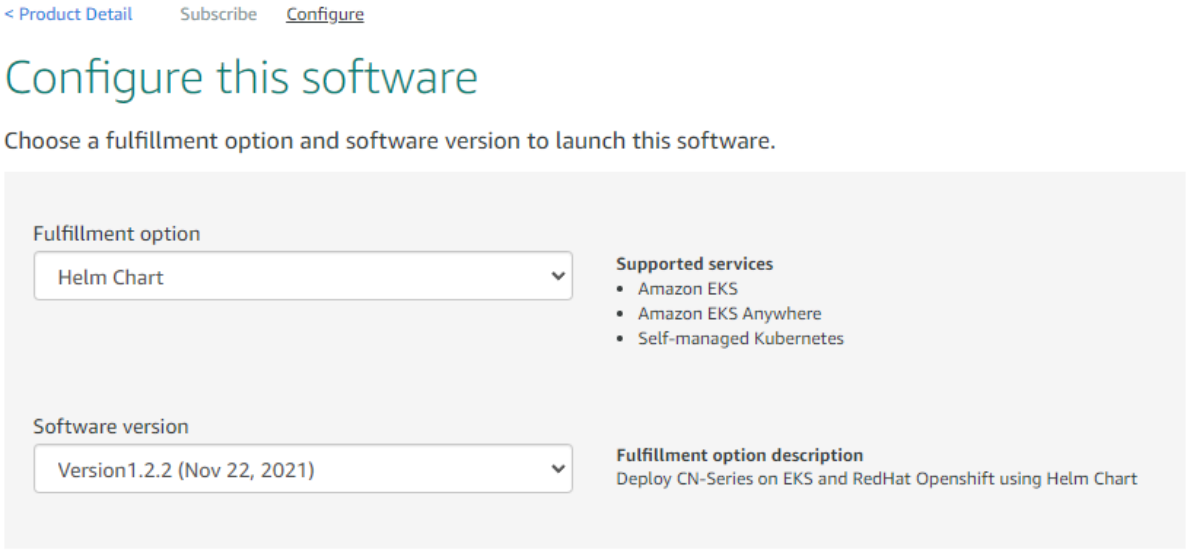
STEP 6 | Cliquez sur **Continue to Subscribe (Continuer pour vous abonner)**.

STEP 7 | Saisissez le nombre de licences que vous souhaitez acheter. Chaque droit de licence équivaut à un vCPU utilisé par votre déploiement CN-Series.

Reportez-vous aux Exigences du système CN-Series et aux Performances et mise à l'échelle CN-Series pour obtenir des conseils sur le nombre de vCPU requis pour répondre aux besoins de votre déploiement.

STEP 8 | Cliquez sur **Continue to Configuration (Continuer la configuration)**. Cela ajoute les licences à votre compte AWS.

1. Sélectionnez **Helm Chart (Graphique Helm)** comme **Fulfillment option (Option d'exécution)**.
2. Sélectionnez la dernière version pour **Software version (Version du logiciel)**.



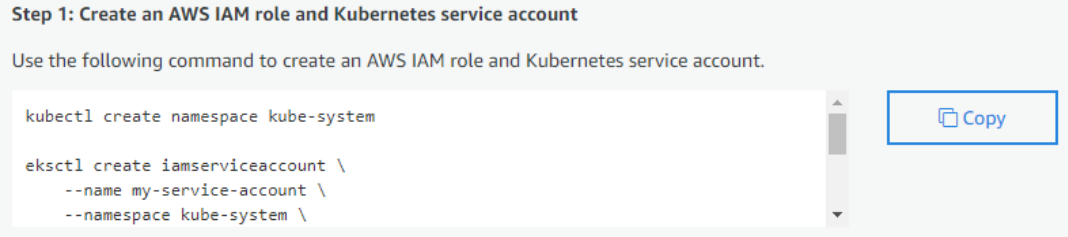
STEP 9 | Cliquez sur **Continue to Launch (Continuer pour lancer)**.

1. Sélectionnez votre **Launch target (Cible de lancement)** : **Kubernetes géré par Amazon** ou **Kubernetes autogéré**. Le mode autogéré est déployé dans Redhat OpenShift.
2. Suivez les **Launch instruction (Instructions de lancement)** affichées dans la liste AWS Marketplace. Les instructions diffèrent en fonction de votre cible de lancement.

   - **Kubernetes géré par Amazon**

     1. Copiez les commandes de **l'étape 1** des **Launch instructions (instructions de lancement)**.
     2. Mettez à jour les commandes copiées pour ajouter le nom de votre cluster.

        `--cluster <ENTER_YOUR_CLUSTER_NAME_HERE>`
     3. Exécutez la commande copiée sur votre cluster EKS.

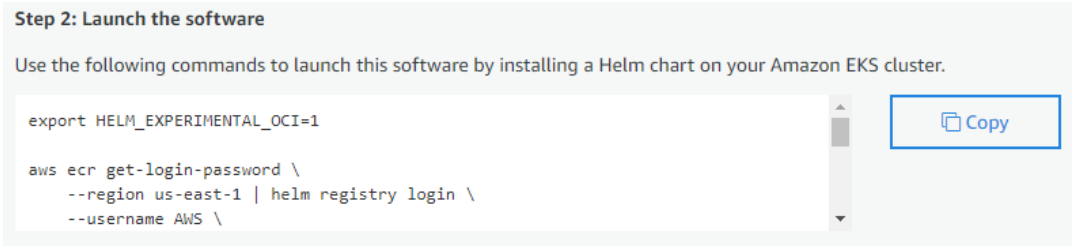


     4. Copiez les commandes du graphique Helm à partir de **l'étape 2** des **Launch instructions (Instructions de lancement)**.
     5. Mettez à jour les informations d'installation de Helm pour inclure votre adresse IP Panorama, votre clé d'authentification Panorama, le nom du groupe d'appareils, le nom de la pile de modèles et le nom du groupe de collecte. Définissez **`cluster.deployTo`** sur **`eks`**.

```
helm install cn-series-helm \ --namespace kube-system ./
awsmp-chart/* \ --set serviceAccount.create=false
 \ --set serviceAccount.name=my-service-
account \ --set cluster.deployTo=eks \ --set
 panorama.ip=Panorama-IP \ --set panorama.ip2=Panorama-
IP2 \ --set panorama.authKey=000xxxxxxx
 \ --set panorama.deviceGroup=Panorama-DG
 \ --set panorama.template=Panorama-TS \
```



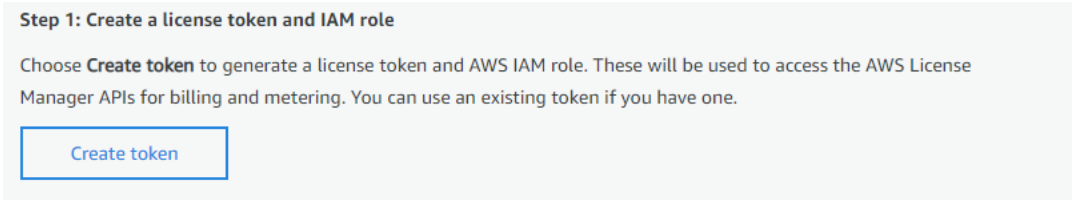
```
--set panorama.cgName=Panorama-CG \ --set
imagePullSecrets=awsmp-image-pull-secret
```



6. Exécutez la commande d'installation helm sur votre cluster EKS après avoir mis à jour les valeurs répertoriées ci-dessus.

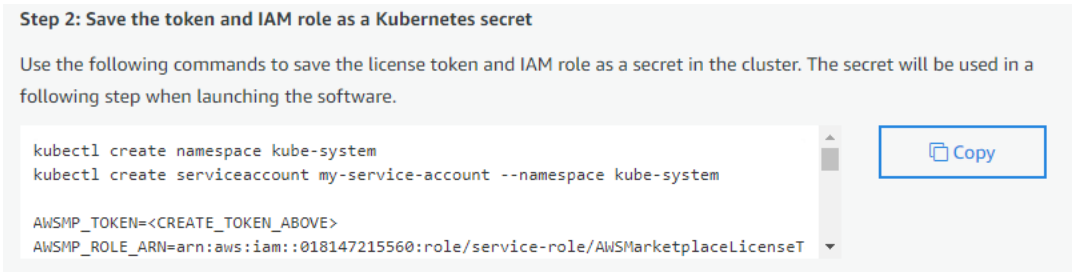
- **Kubernetes autogéré**

1. Effectuez l'étape 1 des instructions de lancement pour créer un jeton de licence et un rôle IAM.



2. Copiez les commandes de **l'étape 2** des **Launch instructions (Instructions de lancement)**.
3. Mettez à jour les commandes copiées pour ajouter la valeur du jeton.

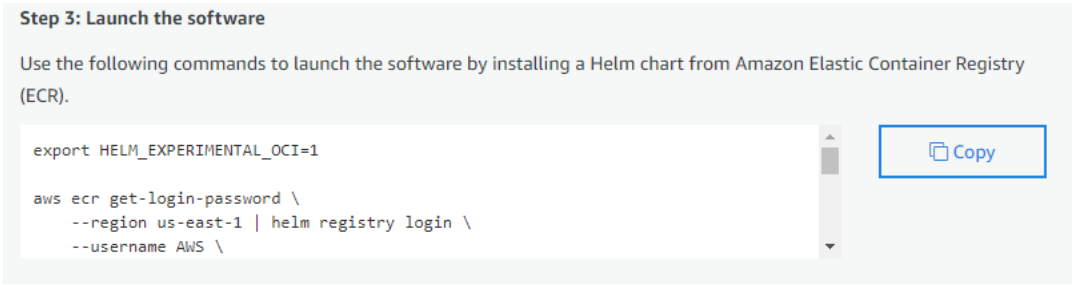
   AWSMP_TOKEN=<CREATE_TOKEN_ABOVE>
4. Exécutez la commande copiée sur votre cluster OpenShift.



5. Copiez les commandes du graphique Helm à partir de **l'étape 3** des **Launch instructions (Instructions de lancement)**.
6. Mettez à jour les informations d'installation de Helm pour inclure votre adresse IP Panorama, votre clé d'authentification Panorama, le nom du groupe

d’appareils, le nom de la pile de modèles et le nom du groupe de collecte. Définissez **`cluster.deployTo`** sur **`openshift`**.

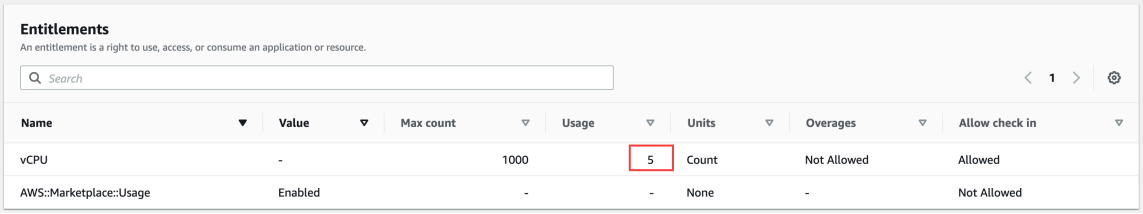
```
helm install cn-series-helm \ --namespace kube-system ./
awsmp-chart/* \ --set serviceAccount.create=false
 \ --set serviceAccount.name=my-service-account
 \ --set cluster.deployTo=eks|openshift \ --set
 panorama.ip=Panorama-IP \ --set panorama.ip2=Panorama-
IP2 \ --set panorama.authKey=000xxxxxxx
 \ --set panorama.deviceGroup=Panorama-DG
 \ --set panorama.template=Panorama-TS \
 --set panorama.cgName=Panorama-CG \ --set
 imagePullSecrets=awsmp-image-pull-secret
```



7. Exécutez la commande d’installation helm sur votre cluster OpenShift après avoir mis à jour les valeurs répertoriées ci-dessus.

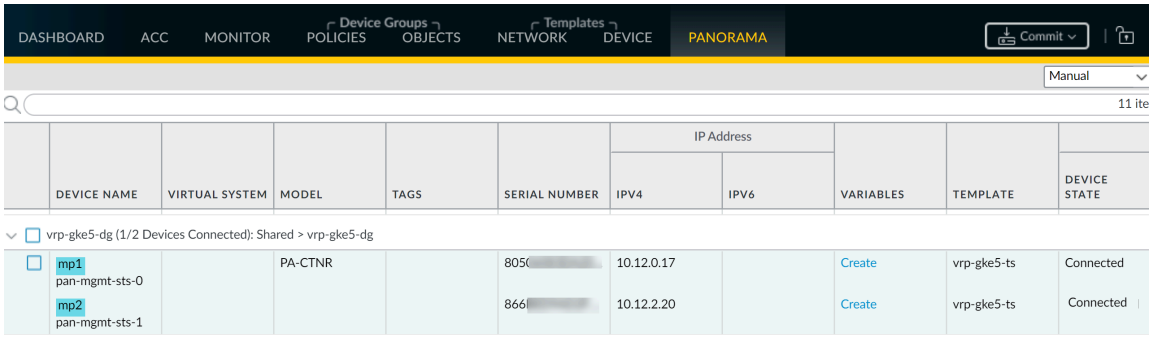
STEP 10 | Vérifiez que la licence a été ajoutée avec succès à votre compte.

1. Accédez à AWS License Manager.
2. Sélectionnez **Granted Licenses (Licences accordées)** et recherchez la liste CN-Series pour AWS Marketplace.
3. Sous **Entitlements (Droits)**, vous pouvez voir le nombre total de licences et le nombre de licences consommées.

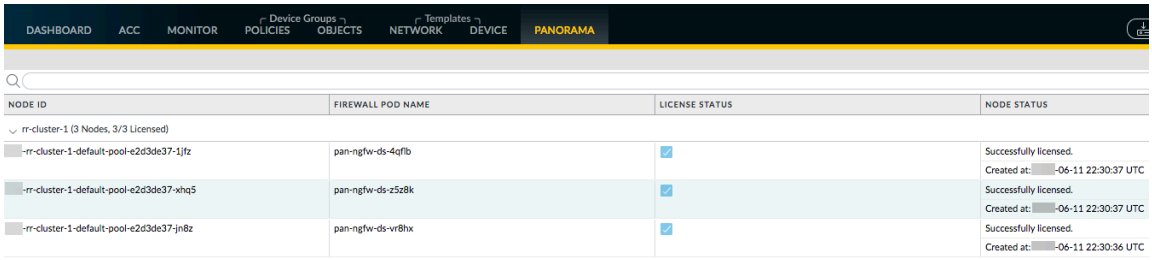


STEP 11 | Vérifiez que les pare-feu CN-Series apparaissent dans Panorama.

1. Connectez-vous à Panorama.
2. Pour afficher les pods CN-MGMT, sélectionnez **Panorama** > **Managed Devices (Appareils gérés)** > **Summary (Résumé)**.



3. Pour vérifier que les pods CN-NGFW sont sous licence, sélectionnez **Panorama** > **Plugins (Plug-ins)** > **Kubernetes** > **License Usage (Utilisation de licence)** et vérifiez que chaque pod a reçu un jeton de licence.



Déployer le pare-feu CN-Series en tant que service Kubernetes sur AliCloud (ACK)

Où puis-je utiliser ceci ?	De quoi ai-je besoin ?
• Déploiement CN-Series	• CN-Series 10.1.x or above Container Images • Panorama sous PAN-OS 10.1.x ou PAN-OS 10.2.x

Après avoir examiné les blocs de construction principaux CN-Series et la présentation générale du flux de travail dans Sécuriser les charges de travail Kubernetes avec CN-Series , vous pouvez commencer à déployer le pare-feu CN-Series sur la plate-forme AliCloud ACK pour sécuriser le trafic entre les conteneurs au sein du même cluster, ainsi qu'entre les conteneurs et d'autres types de charges de travail tels que les machines virtuelles et les serveurs bare-metal.

Vous devez vous assurer que vous appliquez le fichier `plugin-serviceaccount.yaml`. Pour plus d'informations, consultez Créer des comptes de service pour l'authentification des clusters.

- *Lorsque vous déployez le pare-feu CN-Series en tant que service Kubernetes sur ACK, le `pan-plugin-cluster-mode-secret` doit être présent.*

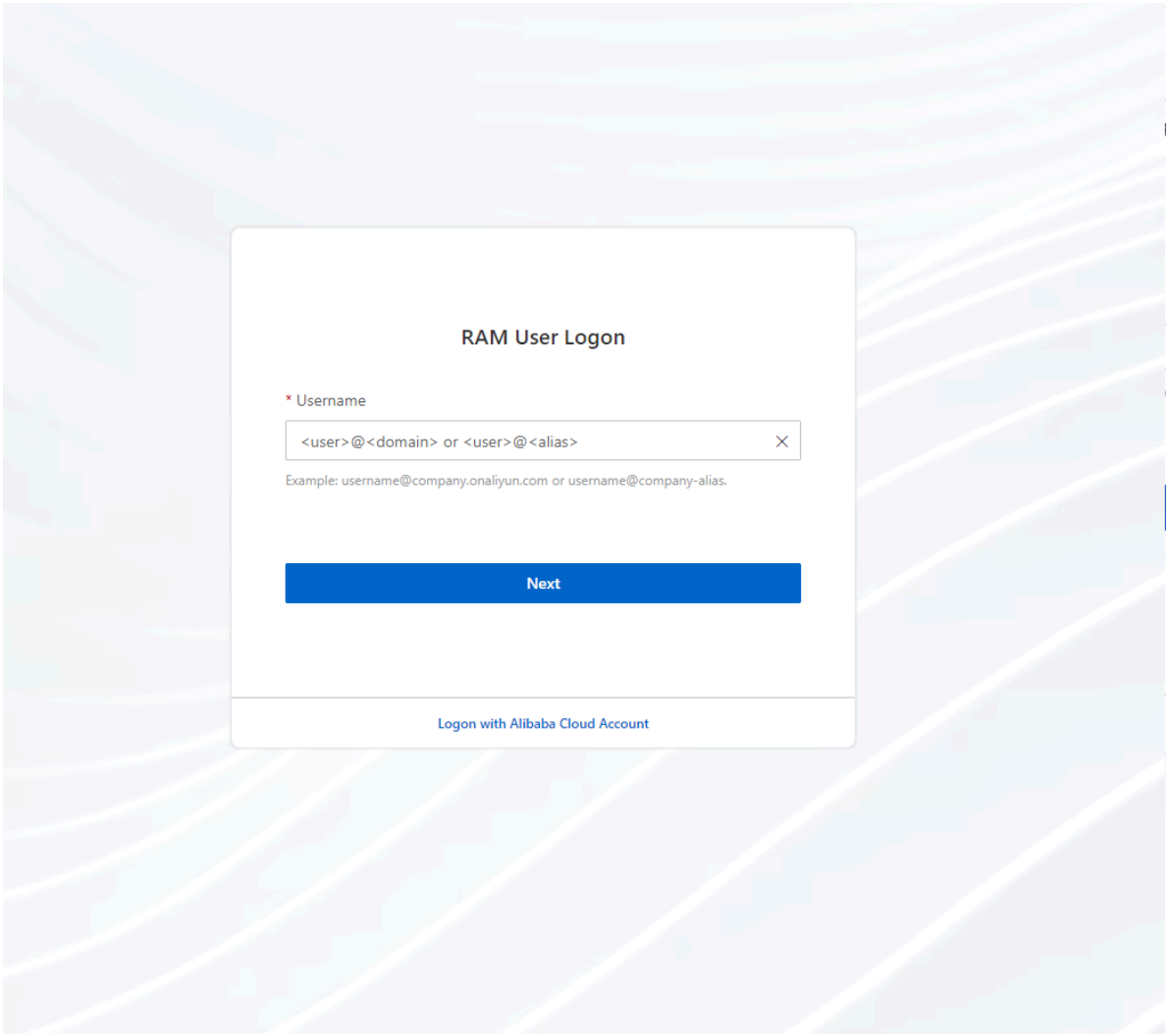
Avant de commencer, assurez-vous que la version du fichier YAML CN-Series est compatible avec la version PAN-OS. Pour plus d'informations, consultez YAML CN-Series.

Effectuez la procédure suivante pour déployer le pare-feu CN-Series en tant que service Kubernetes sur la plateforme ACK :

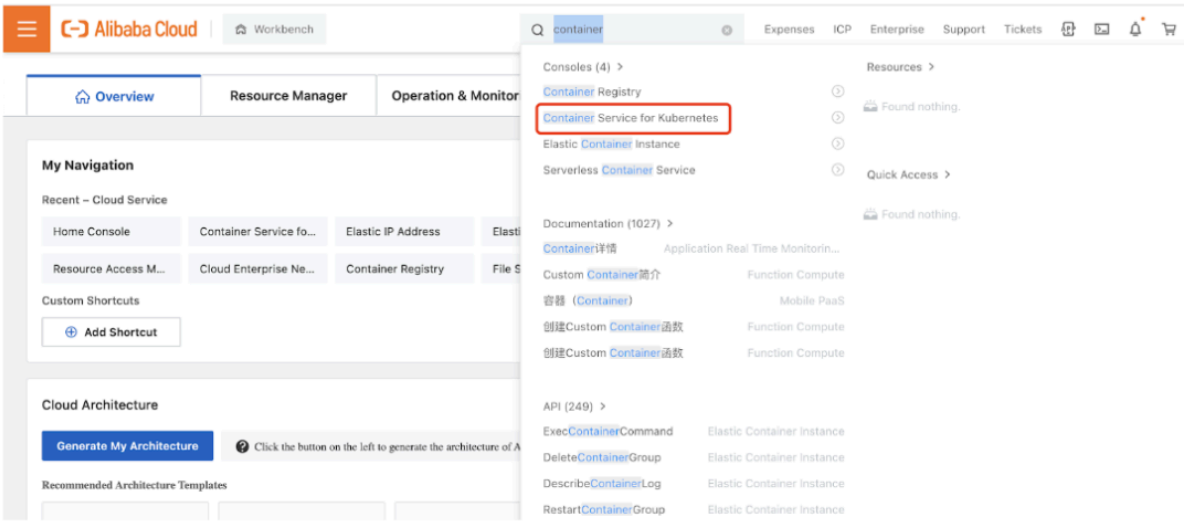
STEP 1 | Configurez votre cluster Kubernetes.

Pour créer un cluster dans ACK, procédez comme suit :

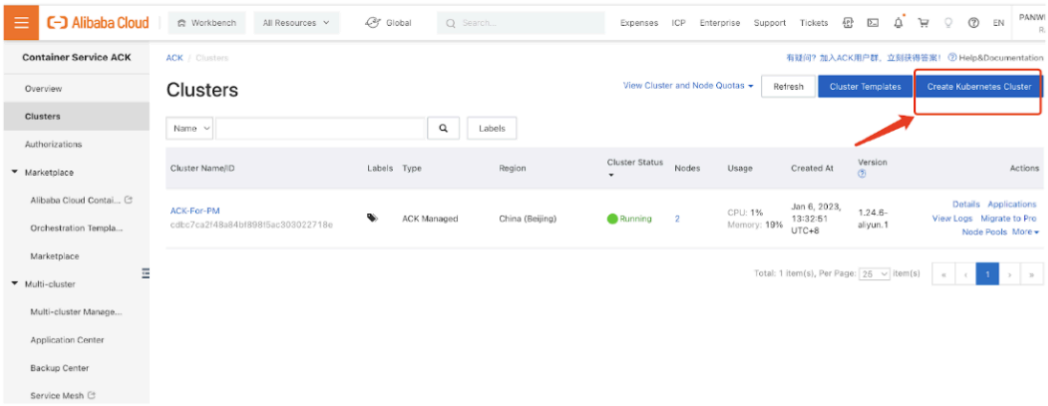
1. Connectez-vous à la connexion utilisateur RAM à l’aide de vos informations de connexion RAM.



2. Dans la barre de navigation supérieure, sélectionnez la région dans laquelle vous souhaitez créer un cluster et sélectionnez un groupe de ressources en fonction des besoins de votre entreprise.
   - La région d'un cluster ne peut pas être modifiée une fois le cluster créé.
   - Tous les groupes de ressources de votre compte sont affichés par défaut.
3. Recherchez **Container Service pour Kubernetes** dans le menu de la barre de recherche.



4. Cliquez sur **Créer un cluster Kubernetes**.

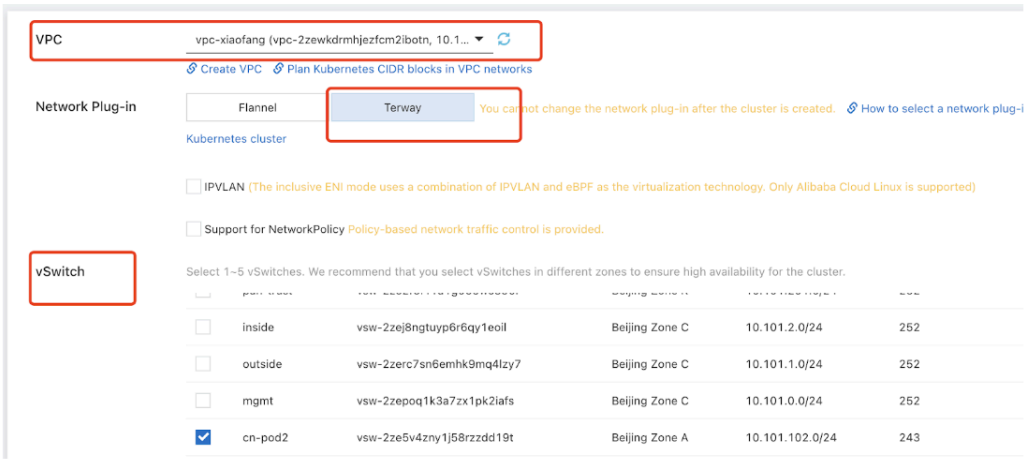


5. Pour créer un cluster, vous devez configurer les paramètres logiciels, les paramètres matériels et les paramètres de base en suivant le guide de l'assistant. Pour plus d'informations sur la configuration

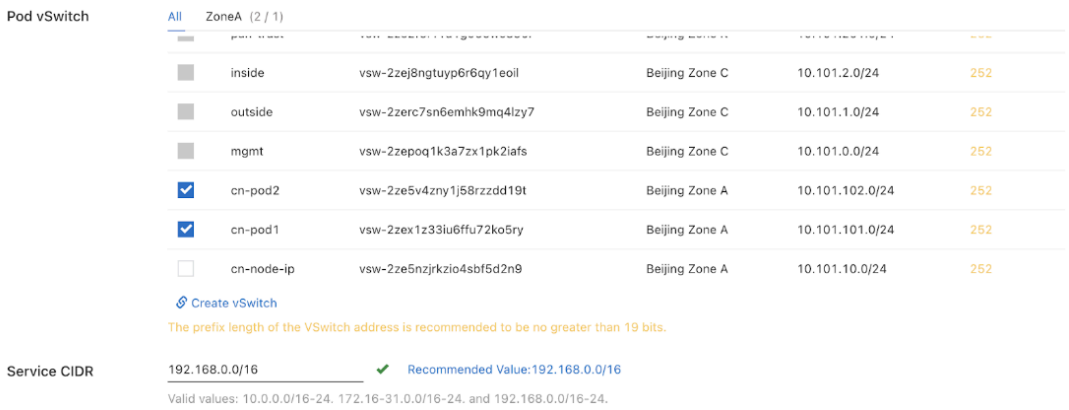
de ces paramètres requis, consultez Créer un cluster sur ACK. Les étapes suivantes représentent un exemple de création de cluster sur la plateforme ACK :

CN-Series sur le cloud Alibaba ACK prend en charge uniquement le plug-in réseau Terway.

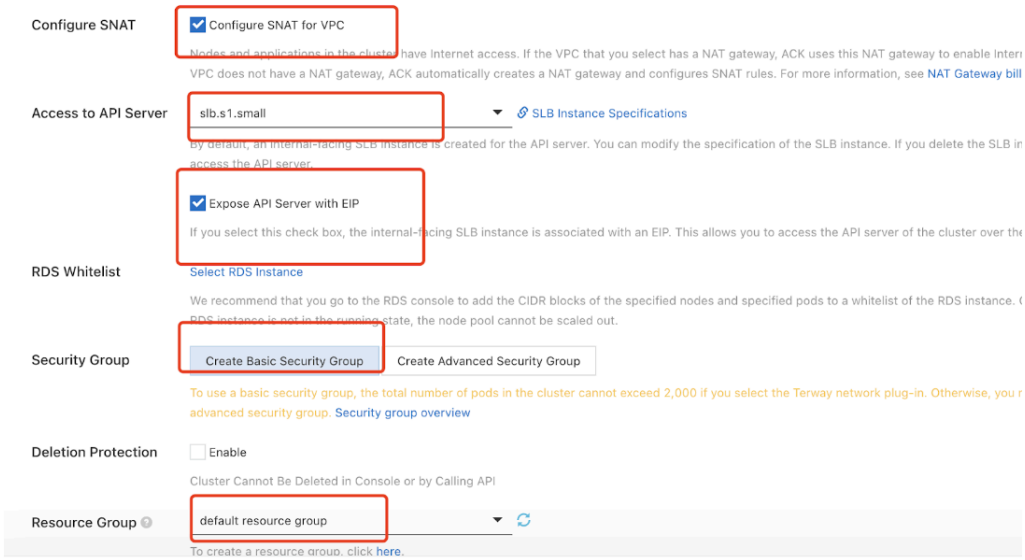
- Sélectionnez **VPC, plug-in réseau** et **vSwitch**.



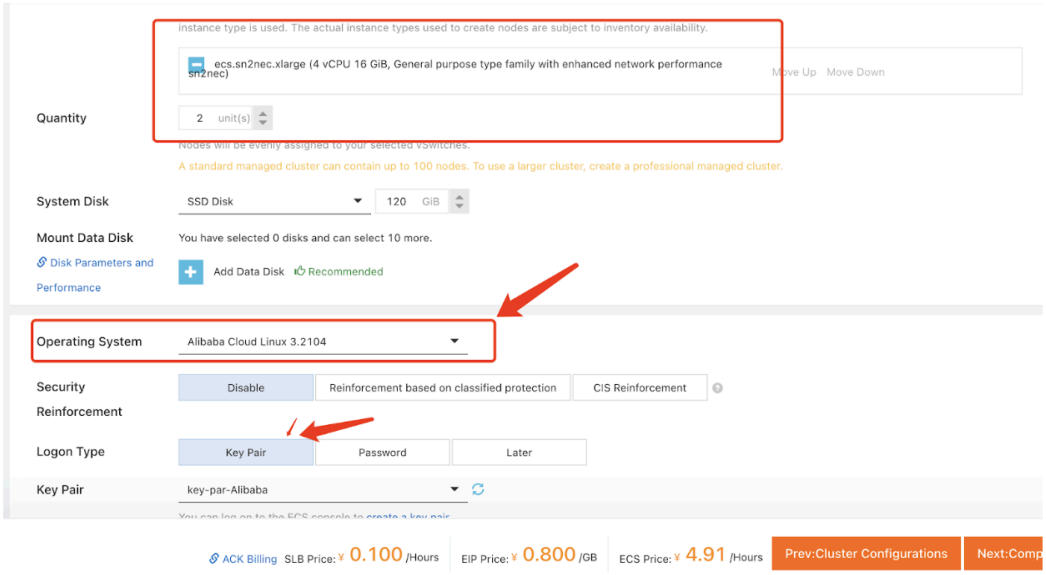
- Sélectionnez **POD contre Switch**.



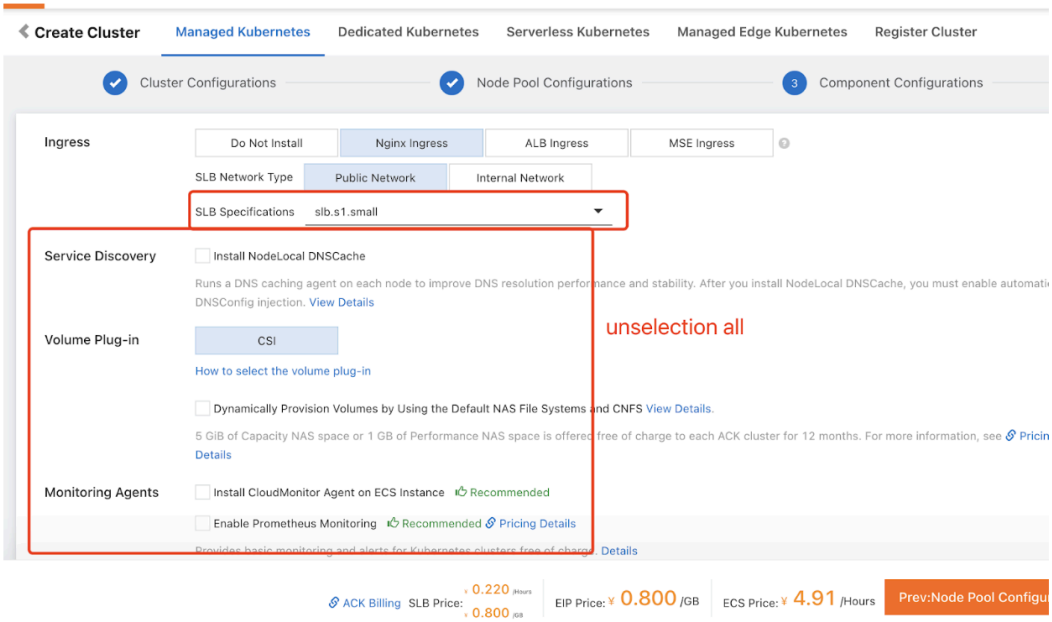
- Sélectionnez **Configurer SNAT, Accès au serveur API, Groupes de sécurité**et **Groupe de ressources**.



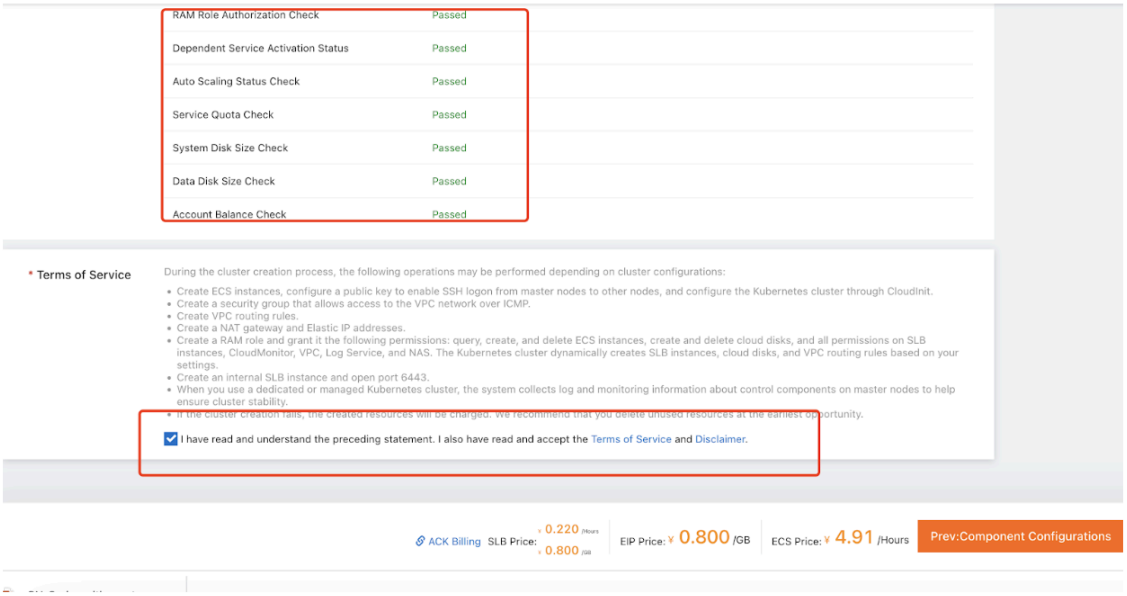
- Sélectionnez **Quantité, Système d'exploitation** et **Type de connexion** pour les configurations de pool de nœuds.



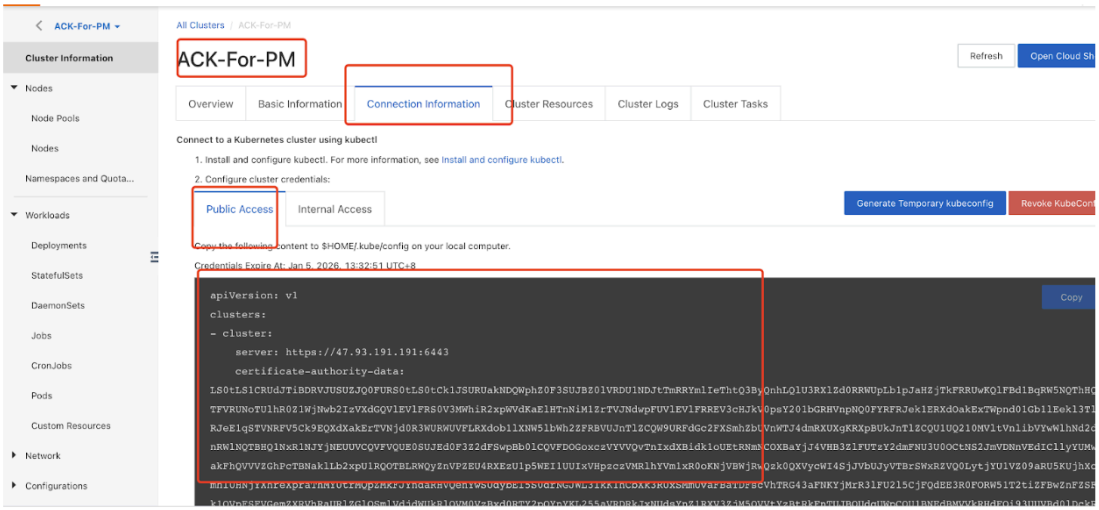
- Accédez à **l'onglet Réseau public**, décochez les cases**Découverte des services, Plug-in de volume** et **Agents de surveillance**.



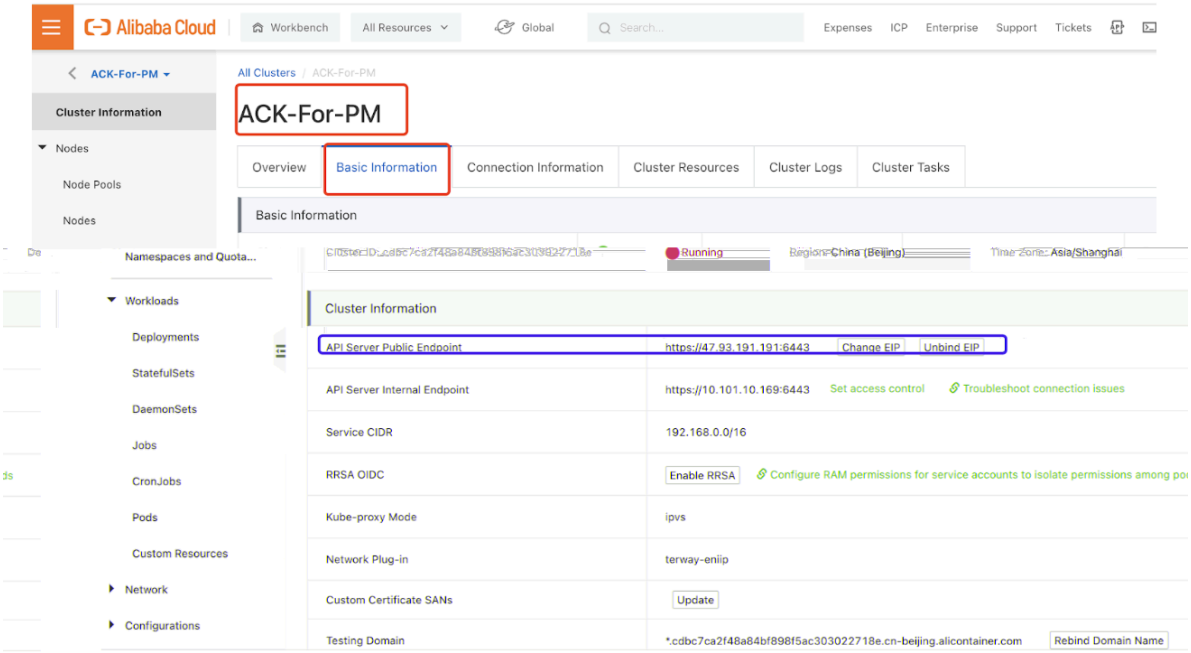
6. Cochez la case **Conditions d'utilisation**.

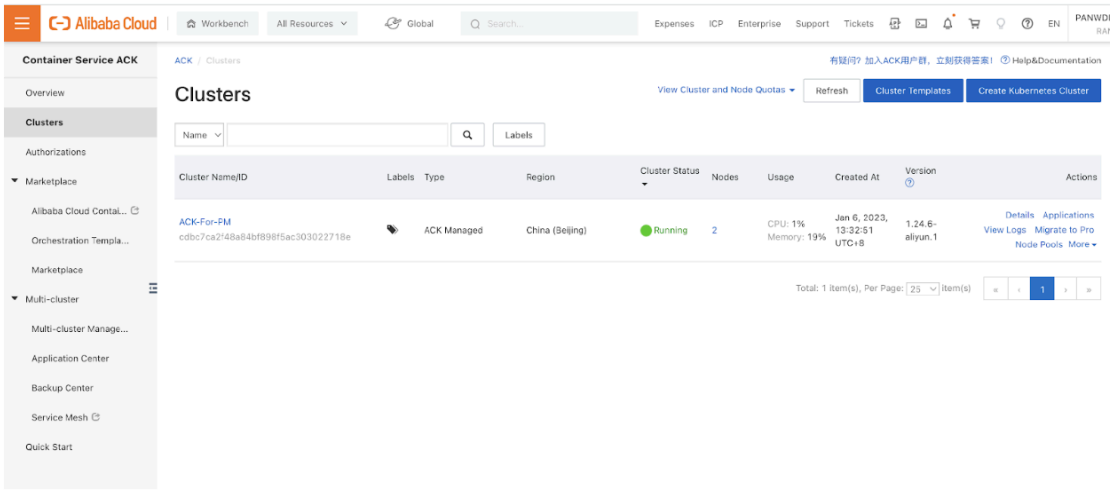


7. Cliquez sur **Create Cluster (Créer un cluster)**.
8. Vérifiez la clé du serveur API pour vous connecter au cluster ACK et copiez le contenu suivant dans `$HOME/.kube/config` sur votre ordinateur local.



9. Obtenez l’adresse du terminal public du serveur API du cluster ACK.





Vérifiez que le cluster dispose des ressources adéquates. La spécification par défaut du pool de nœuds GKE n'est pas adaptée au pare-feu CN-Series. Vous devez vous assurer que ce cluster dispose des conditions préalables de CN-Series pour prendre en charge le pare-feu :

```
kubectl get nodes
```

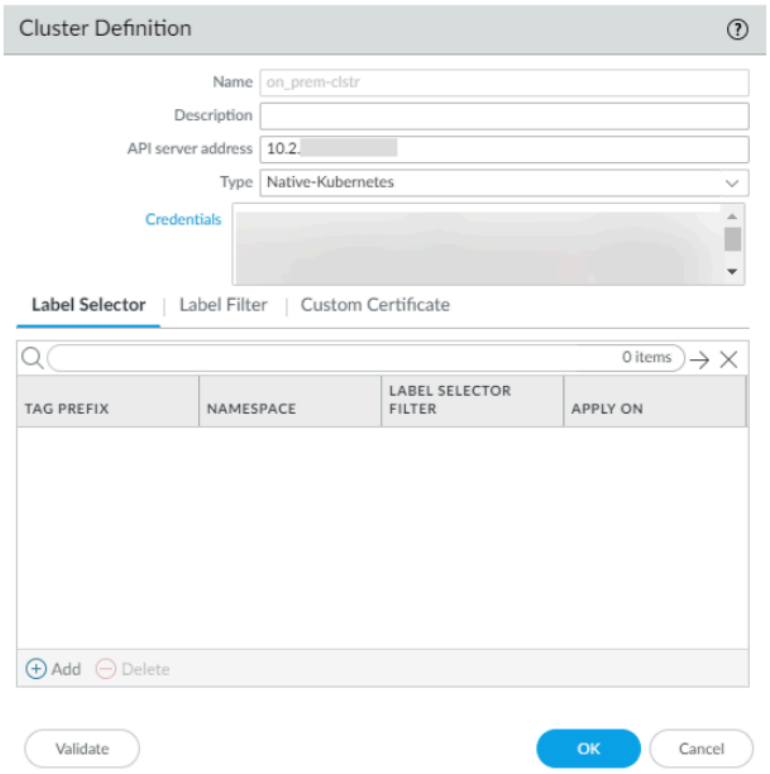
```
kubectl describe node <node-name>
```

Affichez les informations sous l'en-tête Capacity (Capacité) dans la sortie de la commande pour voir le processeur et la mémoire disponibles sur le nœud spécifié.

L’allocation du processeur, de la mémoire et du stockage sur disque dépendra de vos besoins. Voir Performances et évolutivité de CN-Series.

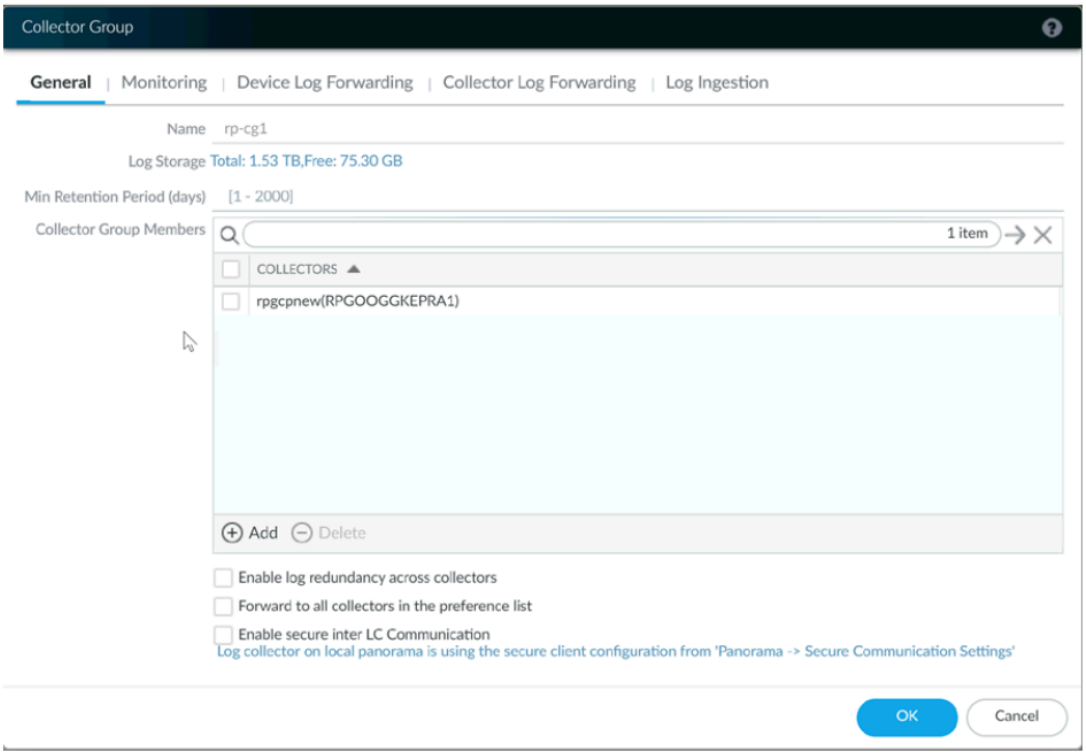
Vous devez vous assurer que vous disposez des informations suivantes :

- Collectez l’adresse IP du terminal pour configurer le serveur API sur Panorama.



  Panorama utilise cette adresse IP pour se connecter à votre cluster Kubernetes.

- Collectez le nom de la pile de modèles, le nom du groupe d’appareils, l’adresse IP Panorama et éventuellement le nom du groupe de collecteurs de journaux à partir de Panorama.



Pour plus d'informations, consultez Créer un groupe d'appareils parents et une pile de modèles.

- Collectez la clé d'authentification VM et l'ID et la valeur du code PIN d'enregistrement automatique.
- L'emplacement du conteneur d'images dans lequel vous avez téléchargé les images.

STEP 2 | (facultatif) Si vous avez configuré un certificat personnalisé dans le plug-in Kubernetes pour Panorama, vous devez créer le secret de certificat en exécutant la commande suivante. Ne modifiez pas le nom de fichier de ca.crt. Le volume des certificats personnalisés dans pan-cn-mgmt.yaml et pan-cn-ngfw.yaml est facultatif.

```
kubectl -n kube-system crée un secret générique custom-ca --from-file=ca.crt
```

STEP 3 | Modifiez les fichiers YAML afin de fournir les détails nécessaires au déploiement des pare-feu CN-Series.

```
apiVersion: v1 kind: ConfigMap metadata: name: pan-mgmt-config namespace: kube-system data: PAN_SERVICE_NAME: pan-mgmt-svc PAN_MGMT_SECRET: pan-mgmt-secret # Panorama settings
 PAN_PANORAMA_IP: "<panorama-IP>" PAN_DEVICE_GROUP: "<panorama-device-group>" PAN_TEMPLATE_STACK: "<panorama-template-stack>"
 PAN_CGNAME: "<panorama-collector-group>" PAN_CTNR_MODE_TYPE: "k8s-service"
```

```
apiVersion: v1 kind: Secret metadata: name: pan-mgmt-secret
 namespace: kube-system type: Opaque stringData: # Panorama Auth
```

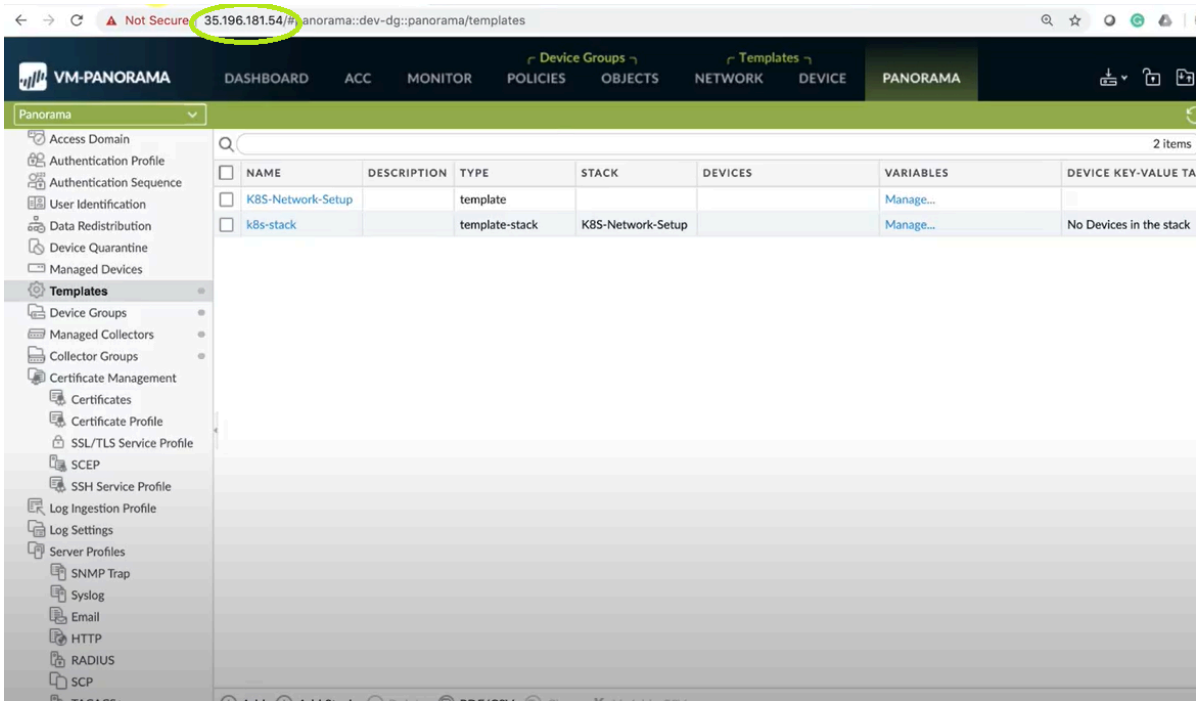


```
  Key PAN_PANORAMA_AUTH_KEY: "<panorama-auth-key>" # Thermite
```



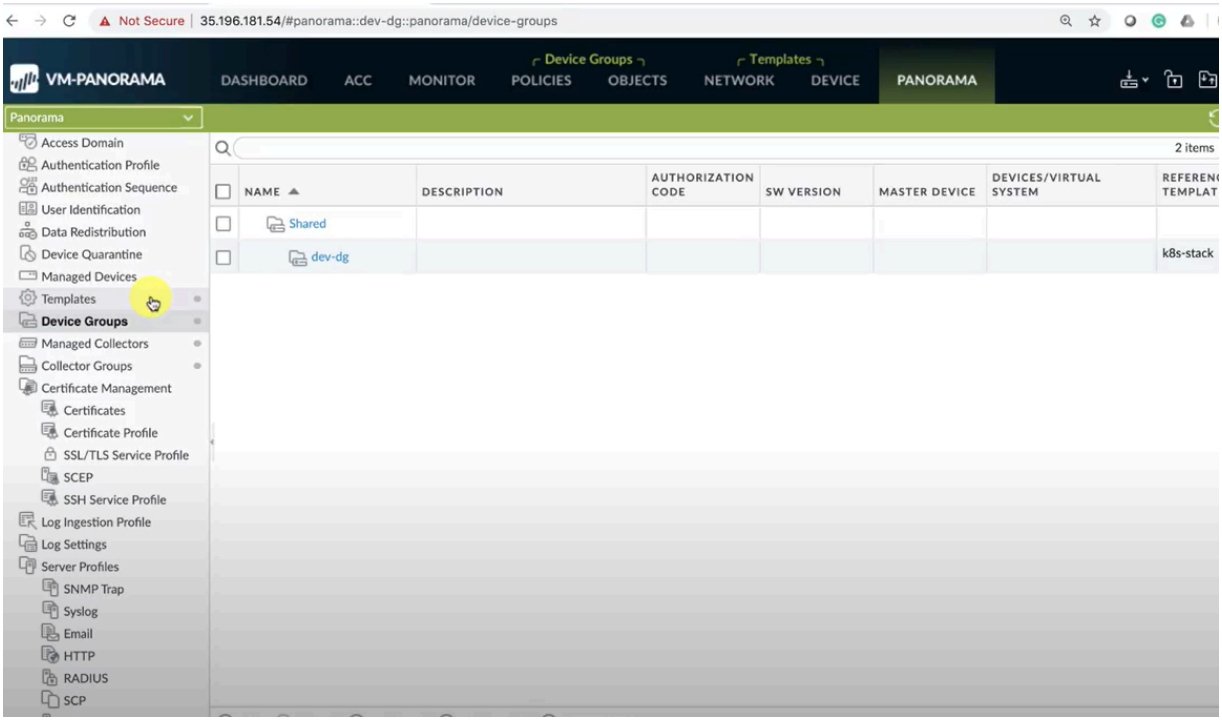
```
Certificate retrieval CN-SERIES-AUTO-REGISTRATION-PIN-ID: "<PIN
Id>" CN-SERIES-AUTO-REGISTRATION-PIN-VALUE: "<PIN-Value>"
```

Vous devez vous assurer que la valeur du paramètre `PAN_PANORAMA_IP` sur votre fichier YAML correspond à votre adresse IP Panorama réelle, comme indiqué dans le diagramme ci-dessous :

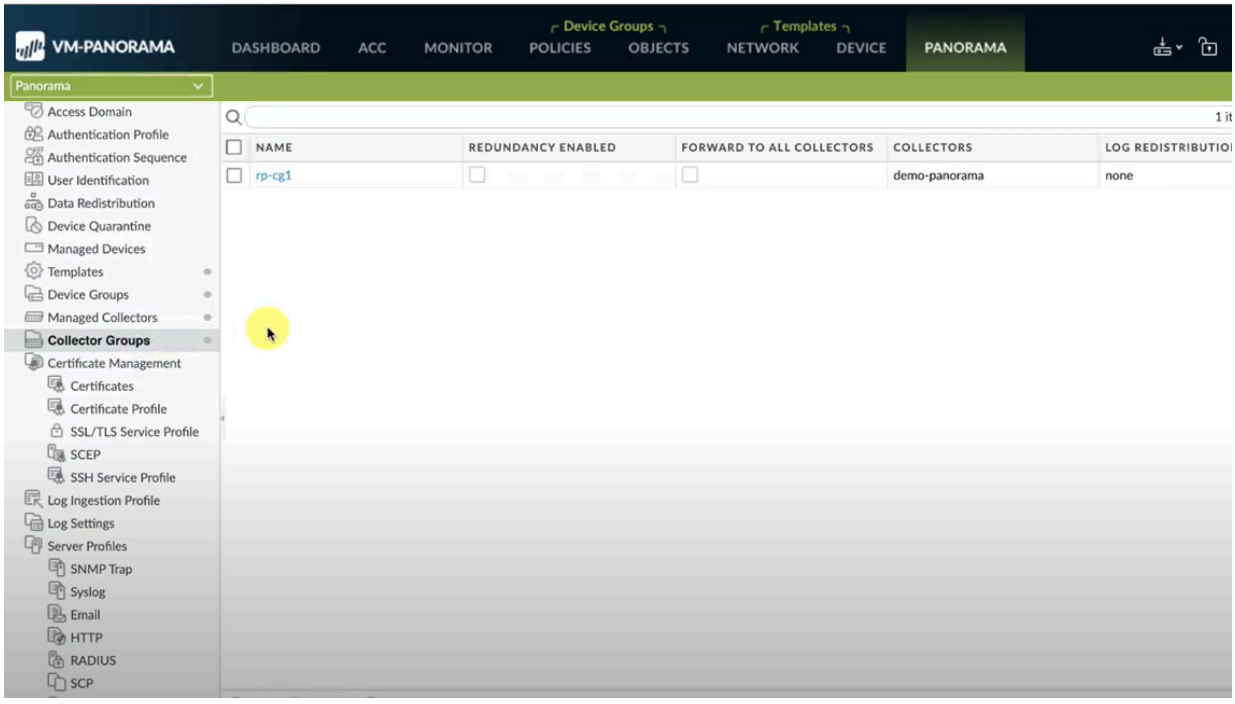


La dernière version des fichiers YAML est disponible sur le référentiel de Palo Alto Networks Kubernetes Security - CN Series. Vous pouvez sélectionner les dernières branches ou balises dans le menu déroulant ***Commutateurde branches/balises****.*

Vous devez vous assurer que la valeur des paramètres de `PAN_DEVICE_GROUP` et `PAN_TEMPLATE` sur votre fichier YAML correspond au nom du groupe d'appareils et de la pile de modèles que vous avez créés sur Panorama, comme indiqué dans le diagramme ci-dessous :



Vous devez vous assurer que la valeur du paramètre `PAN_PANORAMA_CG_NAME` est identique au nom du collecteur de journaux que vous avez créé.



Pour plus d'informations, consultez Paramètres modifiables dans les fichiers yaml CN-Series.

STEP 4 | Déployez le service CN-NGFW. Effectuez les étapes suivantes :

Lorsqu'elles sont déployées en tant que service Kubernetes, les instances du pod CN-NGFW peuvent être déployées sur des nœuds de sécurité et le trafic du pod d'application est redirigé vers une instance CN-NGFW disponible pour inspection et application.

1. Vérifiez que vous avez créé le compte de service à l'aide du fichier pan-cni-serviceaccount.yaml.

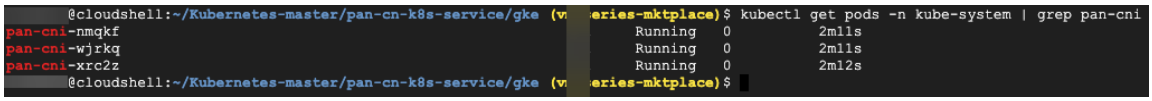
   Consultez Création de comptes de service pour l'authentification des clusters.
2. Utilisez Kubectl pour exécuter le fichier pan-cni-configmap.yaml.

   `kubectl apply -f pan-cni-configmap.yaml`
3. Utilisez kubectl pour exécuter le fichier pan-cn-ngfw-svc.yaml.

   `kubectl apply -f pan-cn-ngfw-svc.yaml`

   Ce fichier yaml doit être déployé avant pan-cni.yaml.
4. Utilisez Kubectl pour exécuter le fichier pan-cni.yaml.

   `kubectl apply -f pan-cni.yaml`
5. Vérifiez que vous avez modifié les fichiers YAML pan-cni et pan-cni-configmap.
6. Exécutez la commande suivante et vérifiez que votre sortie est similaire à l'exemple suivant.



Alicloud ACK prend en charge uniquement la mise à l'échelle automatique basée sur des métriques standard.

STEP 5 | Déployez le StatefulSet CN-MGMT.

Par défaut, le plan de gestion est déployé comme un StatefulSet qui garantie la tolérance aux pannes. Jusqu'à 30 pods pare-feu CN-NGFW peuvent se connecter à un StatefulSet CN-MGMT.

1. (Requis uniquement pour les PV provisionnés statiquement) Déployez les volumes persistants (PV) pour le StatefulSet CN-MGMT.

   1. Créez les répertoires qui correspondent aux noms des volumes locaux définis dans le fichier pan-cn-pv-local.yaml.

   Vous avez besoin de six (6) répertoires sur au moins 2 nœuds esclaves. Connectez-vous à chaque nœud esclave sur lequel le StatefulSet CN-MGMT sera déployé pour créer les

répertoires. Par exemple, pour créer des répertoires nommés `/mnt/pan-local1` à `/mnt/pan-local6`, utilisez la commande suivante :

```
mkdir -p /mnt/pan-local1 /mnt/pan-local2 /mnt/pan-local3 /
mnt/pan-local4 /mnt/pan-local5 /mnt/pan-local6
```

2. Modifiez pan-cn-pv-local.yaml.

   Faites correspondre le nom d'hôte sous `nodeaffinity`, et vérifiez que vous avez modifié les répertoires que vous avez créés ci-dessus dans `spec.local.path` puis déployez le fichier pour créer une nouvelle storage class pan-local-storage et des PV locaux.

   *Dans le fichier `pan-cn-mgmt.yaml`, vous devez ajouter le nom de la classe de stockage comme `alicloud-disk-available` lors de la création de **volumeClaimTemplates**.*

   Par exemple :

   ```
   storageClassName: alicloud-disk-available
   ```

   La taille de stockage doit être d'au moins 20 Go pour tous les PV.

2. Vérifiez que vous avez modifié les fichiers YAML pan-cn-mgmt et pan-cn-mgmt-configmap.

   Exemple de pan-cn-mgmt.yaml

   ```
   initContainers: - name: pan-mgmt-init image: <your-private-
   registry-image-path>
   ```

   ```
    containers: - name: pan-mgmt image: <your-private-registry-
   image-path> terminationMessagePolicy: FallbackToLogsOnError
   ```

3. Utilisez Kubectl pour exécuter les fichiers yaml.

   `kubectl apply -f pan-cn-mgmt-configmap.yaml`

   `kubectl apply -f pan-cn-mgmt-slot-crd.yaml`

   `kubectl apply -f pan-cn-mgmt-slot-cr.yaml`

   `kubectl apply -f pan-cn-mgmt-secret.yaml`

   `kubectl apply -f pan-cn-mgmt.yaml`

   Vous devez exécuter pan-mgmt-serviceaccount.yaml, uniquement si vous n'aviez pas déjà terminé la création de compte de service pour l'authentification de cluster.

4. Vérifiez que les pods CN-MGMT sont opérationnels en exécutant la commande suivante :

   `kubectl get pods -l app=pan-mgmt -n kube-system`

   Cela prend environ 5-6 minutes.

STEP 6 | Déployez les pods CN-NGFW.

1. Vérifiez que vous avez modifié les fichiers YAML comme indiqué dans PAN-CN-NGFW-CONFIGMAP et PAN-CN-NGFW.

   containers: - name: pan-ngfw-container image: <your-private-registry-image-path>

2. Utilisez l'application Kubectl pour exécuter le fichier pan-cn-ngfw-configmap.yaml.

   kubectl apply -f pan-cn-ngfw-configmap.yaml

3. Utilisez l'application Kubectl pour exécuter le pan-cn-ngfw.yaml.

   kubectl apply -f pan-cn-ngfw.yaml

4. Vérifiez que les pods CN-NGFW sont en cours d'exécution.

   kubectl get pods -n kube-system -l app=pan-ngfw -o wide

STEP 7 | Vérifiez que vous pouvez voir CN-MGMT, le CN-NGFW et le PAN-CNI sur le cluster Kubernetes.

kubectl -n kube-system get pods

STEP 8 | Annotez l'application yaml ou l'espace de noms afin que le trafic de leurs nouveaux pods soit redirigé vers le pare-feu.

Vous devez ajouter l'annotation suivante pour rediriger le trafic vers le CN-NGFW pour inspection :

```
annotations: paloaltonetworks.com/firewall: pan-fw
```

Par exemple, pour tous les nouveaux pods dans l'espace de noms « par défaut :

```
kubectl annotate namespace default paloaltonetworks.com/
firewall=pan-fw
```

STEP 9 | Déployez votre application dans le cluster.

Déployer CN-Series sur OpenShift

Où puis-je utiliser ceci ?	De quoi ai-je besoin ?
• Déploiement CN-Series sur un environnement OpenShift	• CN-Series 10.1.x or above Container Images • Panorama sous PAN-OS 10.1.x ou une version supérieure

Le fichier pan-cni sécurise le trafic sur l'interface par défaut « eth0 » du pod de l'application. Si vous avez des pods multi-hôtes, vous pouvez configurer le pod CN-NGFW afin de sécuriser des interfaces supplémentaires qui sont configurées avec une connexion basée sur un pont pour communiquer avec d'autres pods ou avec l'hôte. En fonction de l'annotation dans le YAML de l'application, vous pouvez configurer le pare-feu CN-Series pour inspecter le trafic de toutes les interfaces ou d'un nombre sélectionné d'interfaces attachées à chaque pod.

Le pan-cni ne crée pas de réseau et n'a donc pas besoin d'adresses IP comme les autres plug-ins du CNI.

PAN-OS 10.1.3 ou version ultérieure est requis pour déployer CN-Series en tant que service Kubernetes sur OpenShift. De plus, CN-Series en tant que service Kubernetes sur OpenShift ne sécurise que l'interface ***eth0****.*

STEP 1 | Déployez votre cluster.

Consultez la documentation du fournisseur de la plateforme cloud et vérifiez que les versions OpenShift et CNI sont prises en charge pour CN-Series. Consultez Obtenir le fichier image pour le pare-feu CN-Series et Paramètres modifiables dans les fichiers yaml CN-Series.

STEP 2 | Utilisez le flux de travail inclus dans Sécuriser les charges de travail Kubernetes avec CN-Series.

Vous devez créer les identifiants du service, et déployer les YAML du pare-feu.

Remarque : Si votre fichier d'identification de service fait plus de 10 Ko, vous devez le compresser puis effectuer un encodage base64 du fichier compressé avant de télécharger ou de coller le contenu du fichier dans l'API ou le CLI Panorama.

STEP 3 | Configurez le plug-in PAN-CNI pour qu'il fonctionne avec le plug-in CNI Multus.

Le CNI Multus sur OpenShift fonctionne comme un « méta-plug-in » qui appelle d'autres plug-ins CNI. Pour chaque application, vous devez :

1. Déployer le PAN-CNI NetworkAttachmentDefinition dans chaque espace de noms de pod

   `kubectl apply -f pan-cni-net-attach-def.yaml -n <target-namespace>`

2. Modifier le YAML de l'application.

   Après avoir déployé le pan-cni-net-attach-def.yaml, ajoutez l'annotation dans le yaml du pod de l'application :

   `paloaltonetworks.com/firewall: pan-fw`

   `k8s.v1.cni.cncf.io/networks: pan-cni`

   Si vous avez d'autres réseaux dans l'annotation ci-dessus, ajoutez **`pan-cni`** après les réseaux qui doivent être inspectés. Les réseaux qui suivent **`pan-cni`** ne sont pas redirigés et inspectés.

   Si votre pod possède plusieurs interfaces réseau, vous devez spécifier les noms des interfaces pour lesquelles vous souhaitez que le pod CN-NGFW inspecte le trafic, sous « interfaces » dans le fichier pan-cni-configmap.yaml.

   Par exemple :

```
        template: metadata: annotations: paloaltonetworks.com/
firewall: pan-fw k8s.v1.cni.cncf.io/networks: bridge-conf,
 macvlan-conf, sriov-conf, pan-cni
```

CN-Series prend désormais en charge le plug-in OVN-Kubernetes Container Network Interface (CNI) sur RedHat OpenShift version 4.13 et ultérieure, en mode de déploiement Kubernetes Service et en mode DaemonSet.

Déployer CN-Series sur le hub de l’opérateur OpenShift

Où puis-je utiliser ceci ?	De quoi ai-je besoin ?
• Déploiement CN-Series	• CN-Series 10.1.x or above Container Images • Panorama sous PAN-OS 10.2.x et version supérieure

Le pare-feu de conteneur CN-Series est maintenant disponible sur hub de l’opérateur de la plate-forme RedHat Openshift. Vous pouvez déployer, configurer et exploiter des pare-feu de conteneurs CN-Series directement depuis le hub de l’opérateur RedHat.

Prérequis pour CN-Series sur le hub de l’opérateur Openshift :

Voici les conditions préalables au déploiement du pare-feu CN-Series sur le hub de l'opérateur Openshift :

- Mettre sous licence le pare-feu CN-Series. La mise sous licence du pare-feu CN-Series est gérée par le plug-in Kubernetes sur Panorama. Générez votre code d’autorisation et gardez-le à portée de main lorsque vous êtes prêt à déployer le pare-feu CN-Series. Pour plus d'informations, voir Mettre sous licence le pare-feu CN-Series.
- Générez la clé d’authentification VM sur Panorama.
- Installez un certificat de périphérique sur le pare-feu CN-Series.
- Créez des comptes de service pour l’authentification des clusters.
- Déployer Panorama : vous devez utiliser Panorama pour configurer, déployer et gérer le déploiement de votre pare-feu CN-Series. Pour plus d’informations sur le déploiement et la configuration d’un appareil Panorama, voir Configurer Panorama.
- Installez le plug-in Kubernetes pour le pare-feu CN-Series.
- Le cluster OpenShift doit respecter les conditions préalables de CN-Series.
- Assurez-vous d'avoir accès au portail de service à la clientèle (CSP) de Palo Alto Networks et d'avoir des crédits Flex.
- Assurez-vous que vous êtes un client RedHat avec une licence OpenShift et un compte qui a les autorisations pour créer des ressources dans OpenShift.
- Assurez-vous que le cluster OpenShift respecte les conditions préalables de CN-Series.

Pour plus d'informations, voir Comment déployer facilement CN-Series sur le hub de l’opérateur RedHat Openshift.

Déployer CN-Series sur un hub de l’opérateur OpenShift :

Le fichier pan-cni sécurise le trafic sur l’interface par défaut **eth0** du pod de l’application. Si vous avez des pods multi-hôtes, vous pouvez configurer le pod CN-NGFW afin de sécuriser des interfaces supplémentaires qui sont configurées avec une connexion basée sur un pont pour communiquer avec d’autres pods ou avec l’hôte. En fonction de l’annotation dans le YAML de l’application, vous pouvez

configurer le pare-feu CN-Series pour inspecter le trafic de toutes les interfaces ou d’un nombre sélectionné d’interfaces attachées à chaque pod.

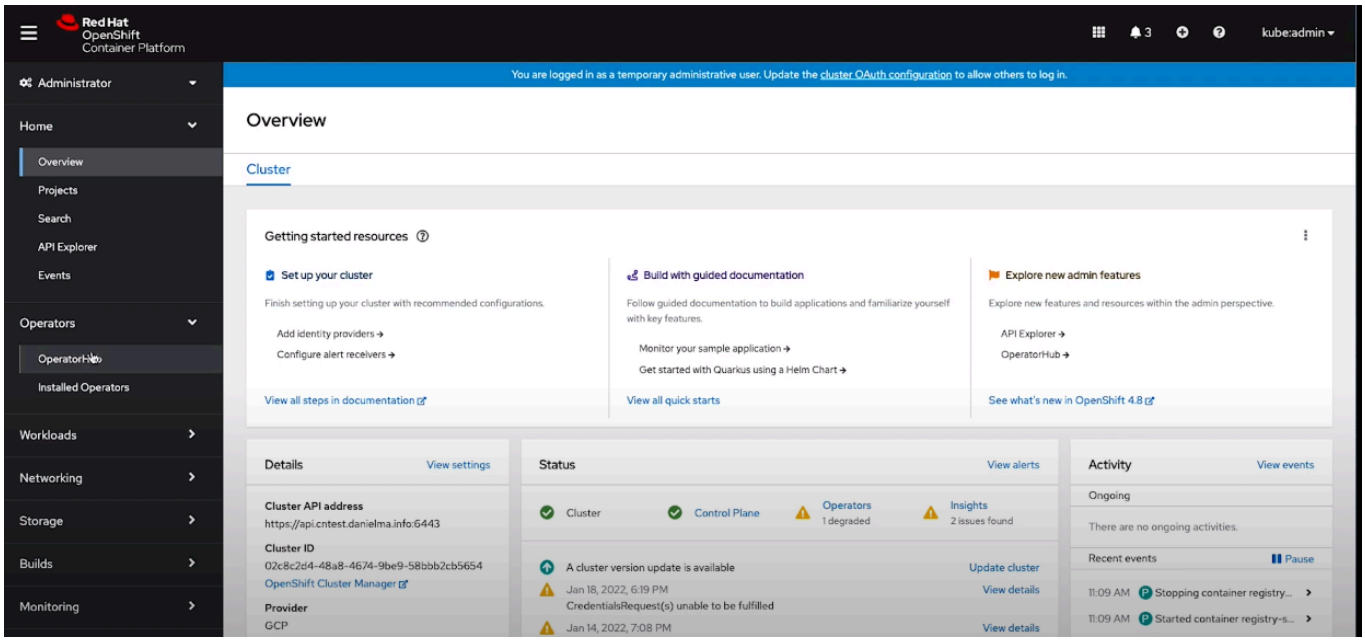
Le pan-cni ne crée pas de réseau et n’a donc pas besoin d’adresses IP comme les autres plug-ins du CNI.

Vous avez besoin de PAN-OS 10.2 ou version ultérieure pour déployer CN-Series sur le hub de l’opérateur OpenShift.

Voici les étapes à suivre pour déployer le pare-feu CN-Series sur votre hub de l’opérateur Redhat OpenShift :

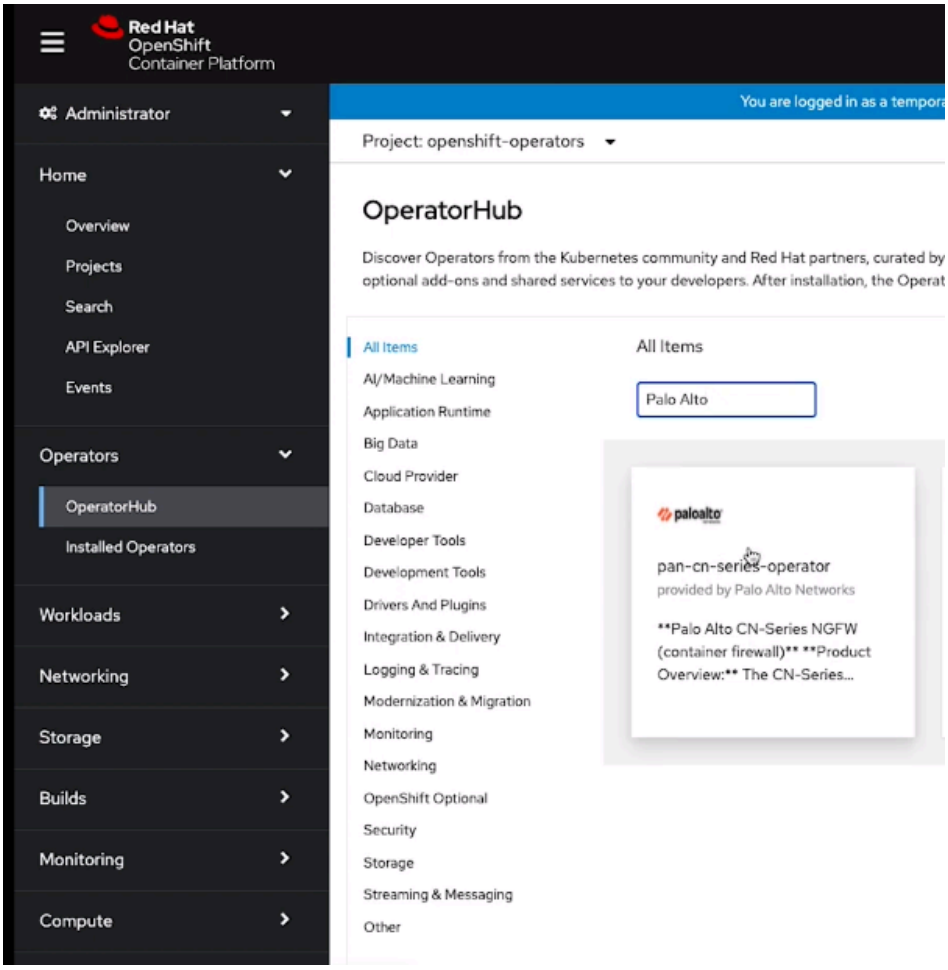
STEP 1 | Connectez-vous à la console du conteneur Redhat OpenShift.

STEP 2 | Accédez à **Opérateurs**, puis cliquez sur **OperatorHub**.



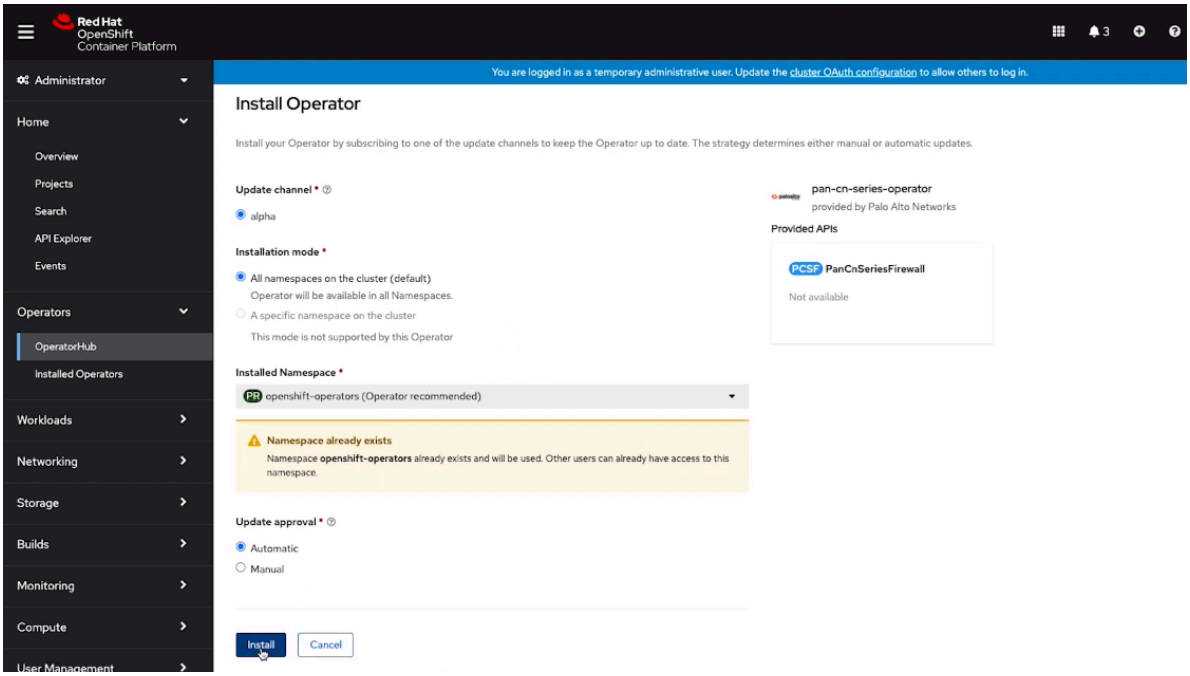
STEP 3 | Entrez **Palo Alto** dans la zone de recherche Opérateur.

STEP 4 | Cliquez sur **pan-cn-series-operator**.



La fenêtre d'installation s'ouvre lorsque vous cliquez sur la tuile **pan-cn-series-operator**.

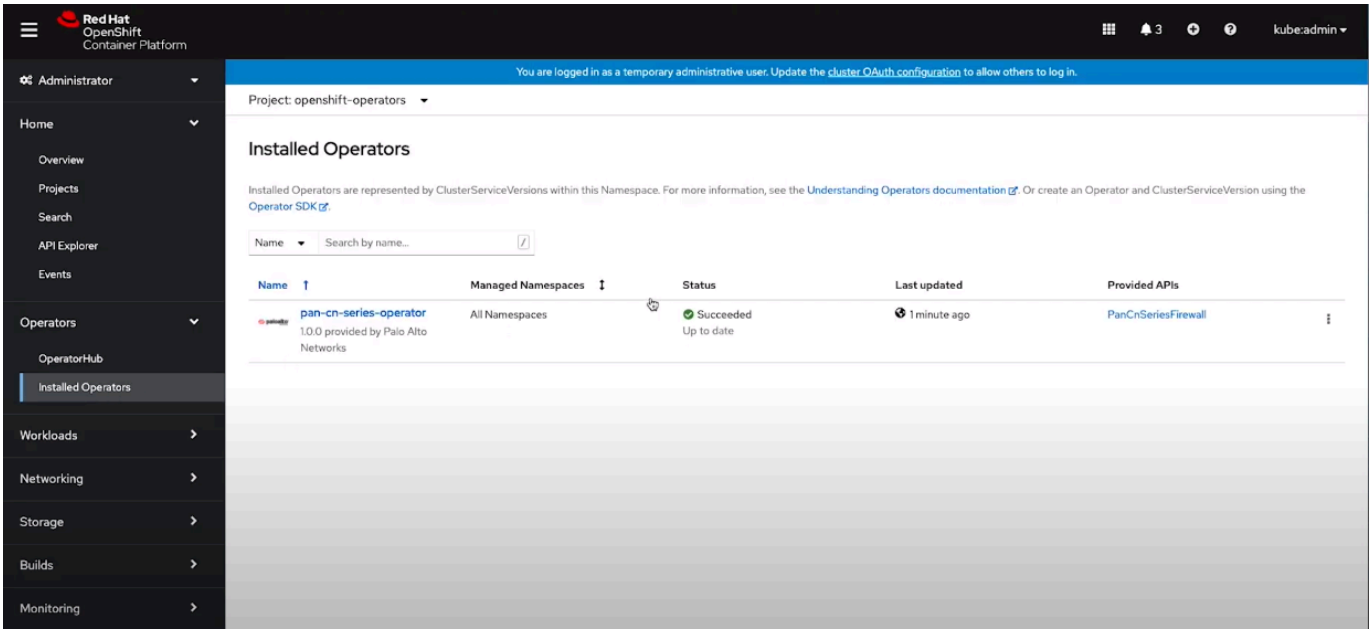
STEP 5 | Cliquez sur **Installer** pour installer l'opérateur pan-cn-series sur votre cluster OpenShift.



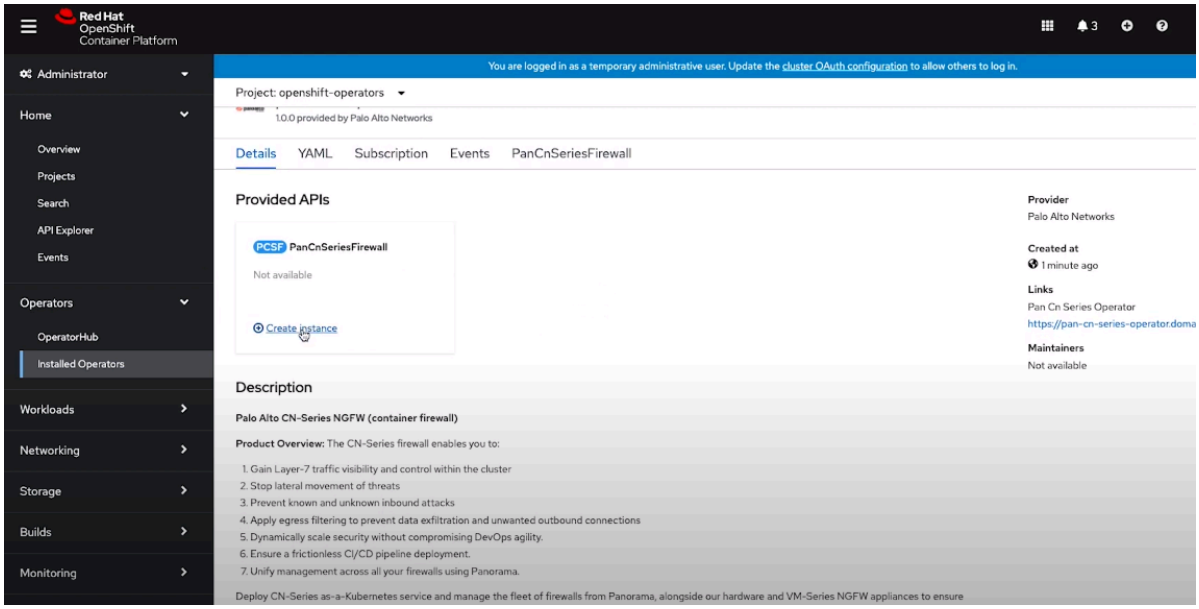
Terminez les étapes de pré-installation avant les étapes de déploiement suivantes indiquées ici.

Si votre fichier d'identification de service fait plus de 10 Ko, vous devez le compresser puis effectuer un encodage base64 du fichier compressé avant de télécharger ou de coller le contenu du fichier dans l'API ou le CLI Panorama.

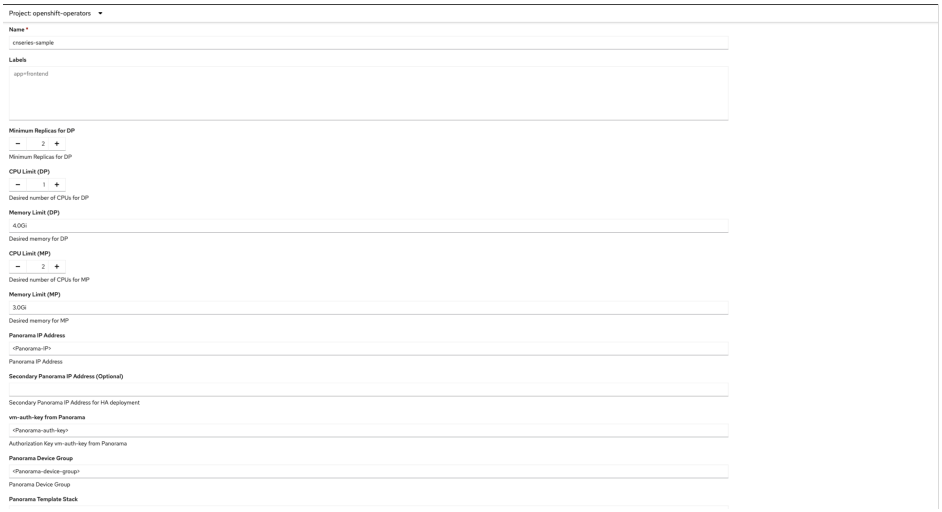
STEP 6 | Dans le menu de navigation, accédez à **Opérateurs installés**, puis cliquez sur **pan-cn-series-operator** que vous avez installé.



STEP 7 | Cliquez sur **Create Instance** (Créer une instance).



STEP 8 | Entrez un **Nom** d'opérande unique.



STEP 9 | Entrez les **Minimum Replicas for DP (Répliques minimales pour DP)**, **Memory Unit (Unité mémoire)** et **vCPU Limit (Limite vCPU)** pour les pods DP et MP. Pour en savoir plus sur les limites du vCPU, consultez Indicateurs de performance clés CN-Series.

STEP 10 | Entrez l'**adresse IP Panorama**.

STEP 11 | Facultatif Entrez l'**adresse IP Panorama secondaire** pour votre déploiement HA.

STEP 12 | Entrez la **clé d'authentification** Panorama CN-Series.

STEP 13 | Accédez au **groupe d’appareils Panorama**.

STEP 14 | Accédez à la **pile de modèles Panorama**.

STEP 15 | Entrez le **nom du groupe de collecteurs de journal Panorama**.

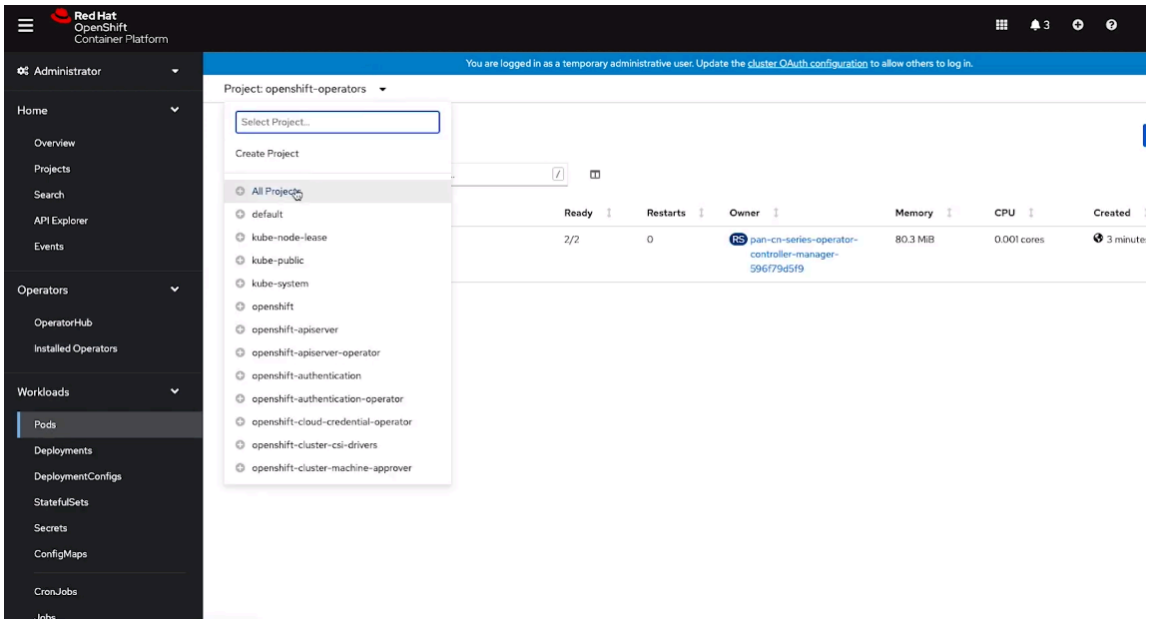
STEP 16 | Facultatif Entrez l'**identifiant PIN**, la **valeur PIN** et l'**URL alternative** du portail d'assistance à la clientèle (CSP).

STEP 17 | En fonction de votre version PAN-OS, liez les images appropriées pour DP, MP et CNI dans la console de registre de conteneur CN-Series.

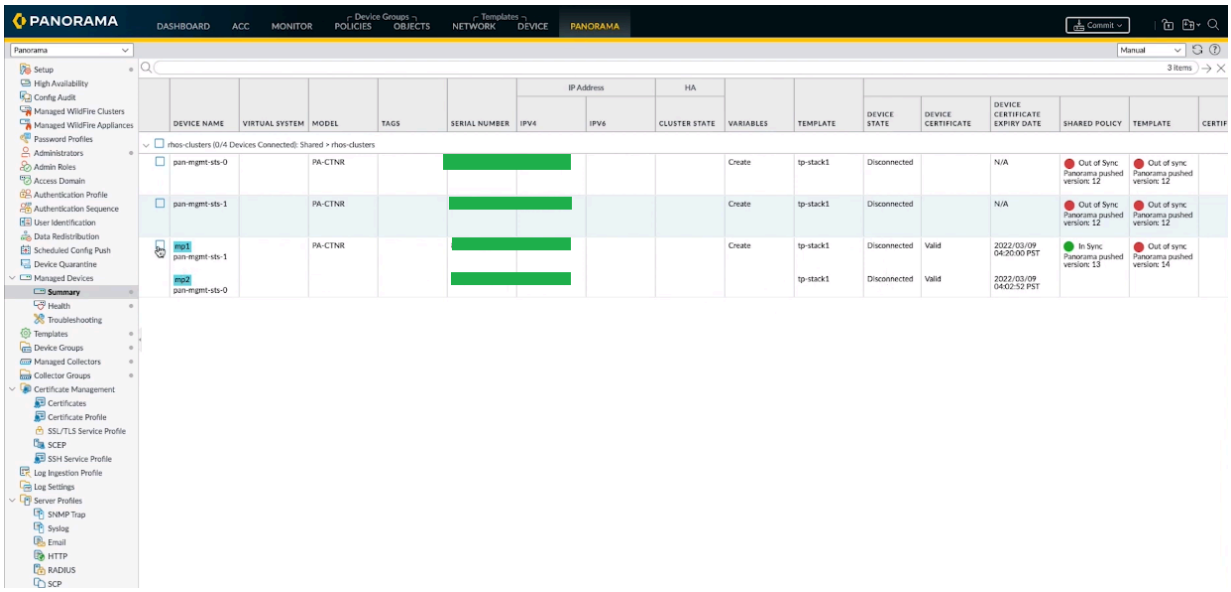
STEP 18 | Cliquez sur **Create (Créer)**.

STEP 19 | Dans le menu Navigation, accédez à **pods**.

STEP 20 | Sélectionnez les **OpenShift-operators** du projet, puis accédez à **kube-system** pour afficher le nom et l'état des pods CNI, de gestion et de plan de données que vous avez déployés dans le cadre de l'opérande.



Vous pouvez vérifier l'état de déploiement du pare-feu sur Panorama. L'**état du périphérique** passera à Connected (Connecté) moins de 5 minutes après le déploiement.



STEP 21 | Configurez le plug-in PALO ALTO NETWORKS-CNI pour qu'il fonctionne avec le plug-in Multus CNI.

Le CNI Multus sur OpenShift fonctionne comme un **méta-plug-in** qui appelle d'autres plug-ins CNI. Pour chaque application, vous devez :

1. Exécutez la commande suivante pour déployer le pan-cni-net-attach-def.yaml dans chaque espace de noms de pod :

   `kubectl apply -f pan-cni-net-attach-def.yaml -n <target-namespace>`

2. Modifier le YAML de l'application.

   Après avoir déployé le pan-cni-net-attach-def.yaml, ajoutez l'annotation suivante dans le yaml du pod de l'application :

   `paloaltonetworks.com/firewall: pan-fw`

   `k8s.v1.cni.cncf.io/networks: pan-cni`

   Si vous avez d'autres réseaux dans l'annotation ci-dessus, ajoutez **`pan-cni`** après les réseaux qui doivent être inspectés. Les réseaux qui suivent **`pan-cni`** ne sont pas redirigés et inspectés.

   *Si votre pod possède plusieurs interfaces réseau, vous devez spécifier les noms des interfaces pour lesquelles vous souhaitez que le pod CN-NGFW inspecte le trafic, sous la section **interfaces** dans le fichier `pan-cni-configmap.yaml`.*

   Par exemple :

```
        template: metadata: annotations: paloaltonetworks.com/
firewall: pan-fw k8s.v1.cni.cncf.io/networks: pan-cni
```

