paloalto TECHDOCS

在云端和本地部署 CN 系列防火墙

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

在 GKE 上部署 CN 系列防火墙

在何处可以使用？	需要提供什么？
• CN-Series部署	• CN-Series 10.1.x or above Container Images • 运行 PAN-OS 10.1.x 或更高版本的 Panorama • 使用 Helm 部署 CN 系列的 Helm 3.6 or above version client

在利用 CN 系列保护 Kubernetes 环境的安全中查看 CN 系列构建块和该工作流程的高级概述后，您可以开始在 GKE 平台上部署 CN 系列防火墙，以保护同一集群中容器之间，以及容器和其他工作负载类型（例如虚拟机和裸机服务器）之间的流量。

您需要标准 *Kubernetes* 工具（例如 *kubectl* 或 *Helm*）部署和管理 *Kubernetes* 集群、应用程序和防火墙服务。

有关更多信息，请参阅使用 Helm 图表和模板部署 CN 系列防火墙。*Panorama* 并非旨在成为 *Kubernetes* 集群部署和管理的 *Orchestrator*。托管 *Kubernetes* 提供商已提供用于集群管理的模板。您还可以使用社区支持的模板部署具有 Helm 和 Terraform 的 *CN* 系列。

- 在 GKE 上将 CN 系列防火墙部署为 Kubernetes 服务
- 在 GKE 上将 CN 系列防火墙部署为 DaemonSet

从*CN* 系列即 *DaemonSet* 部署迁移到 *CN* 系列即服务之前（反之亦然），您必须删除并重新应用 `plugin-serviceaccount.yaml`。有关更多信息，请参阅为集群身份验证创建服务帐户。

- 在 *GKE* 上将 *CN* 系列部署为 *DaemonSet* 时，不能存在 `pan-plugin-cluster-mode-secret`。
- 在 *GKE* 上将 *CN* 系列部署为 *Kubernetes* 服务时，必须存在 `pan-plugin-cluster-mode-secret`。

在 GKE 上将 CN 系列防火墙部署为 Kubernetes 服务

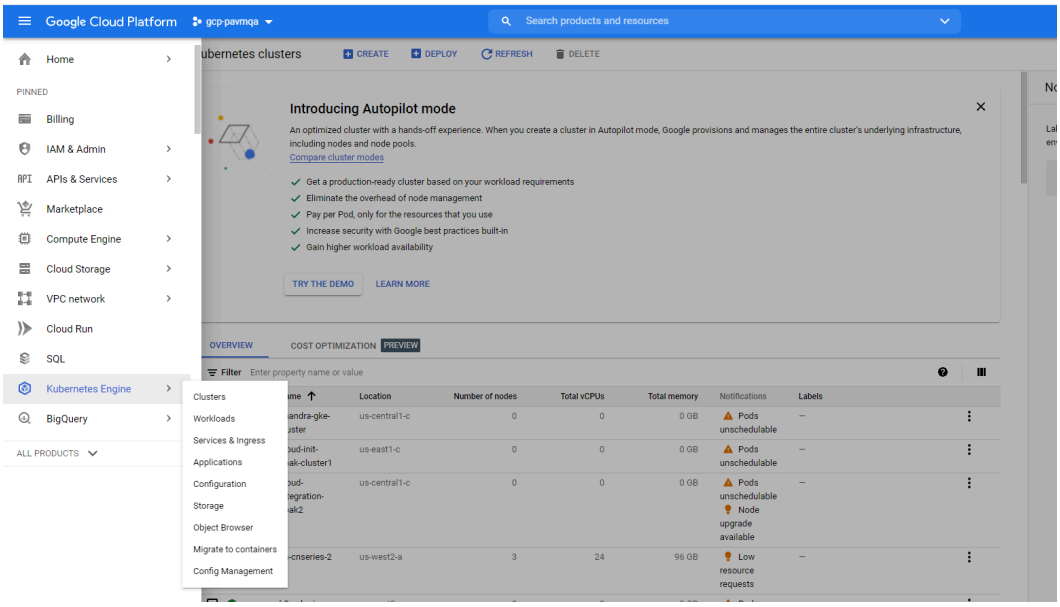
在何处可以使用？	需要提供什么？
• CN-Series部署	• CN-Series 10.1.x or above Container Images • 运行 PAN-OS 10.1.x 或更高版本的 Panorama • 使用 Helm 部署 CN 系列的 Helm 3.6 or above version client

完成以下过程以在 GKE 平台上将 CN 系列防火墙部署为 Kubernetes 服务。

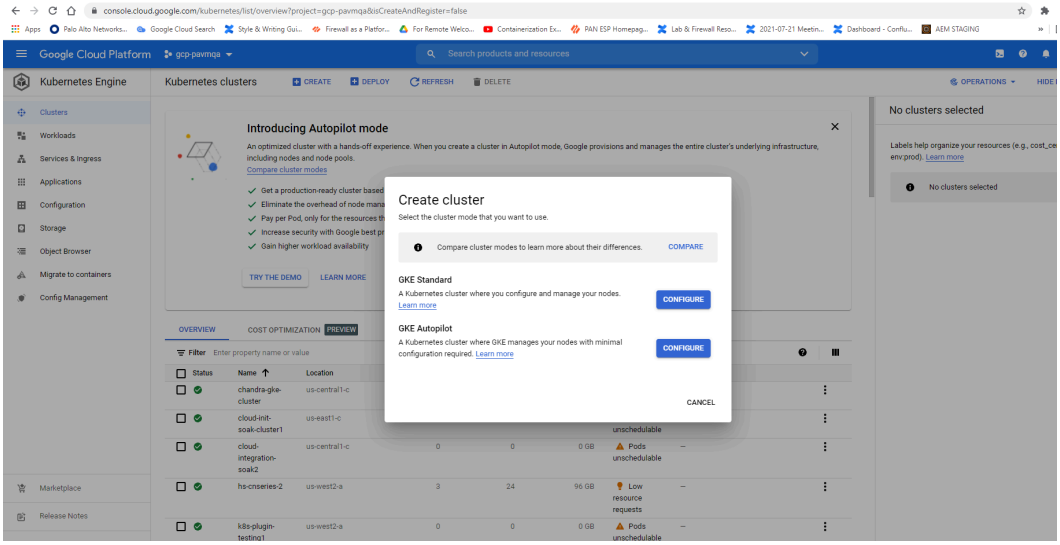
STEP 1 | 设置 Kubernetes 集群。

要在 GKE 中创建集群，请执行以下操作：

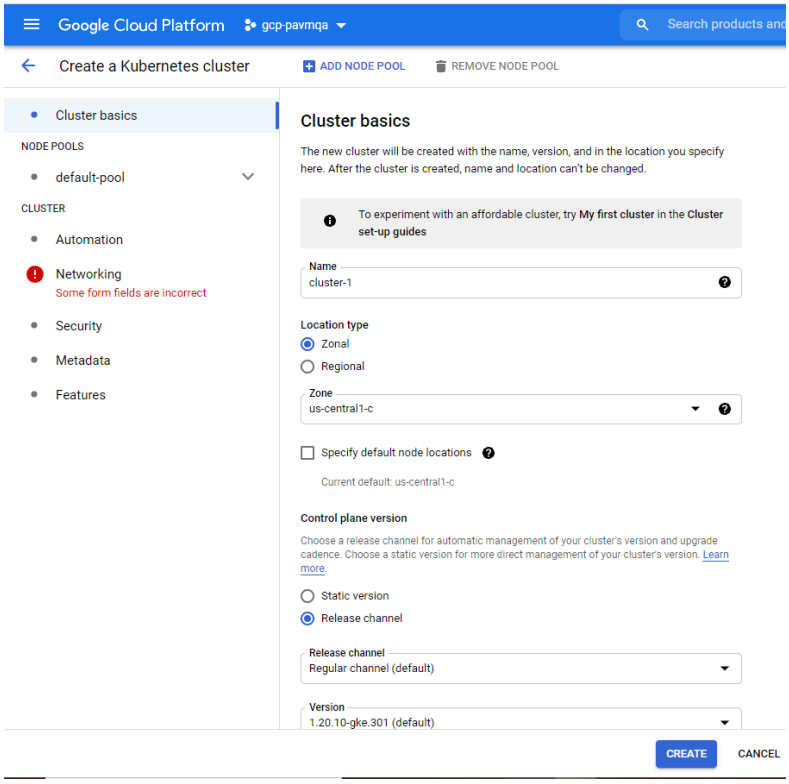
1. 单击导航菜单，转到 **Kubernetes Engine**，然后选择集群。



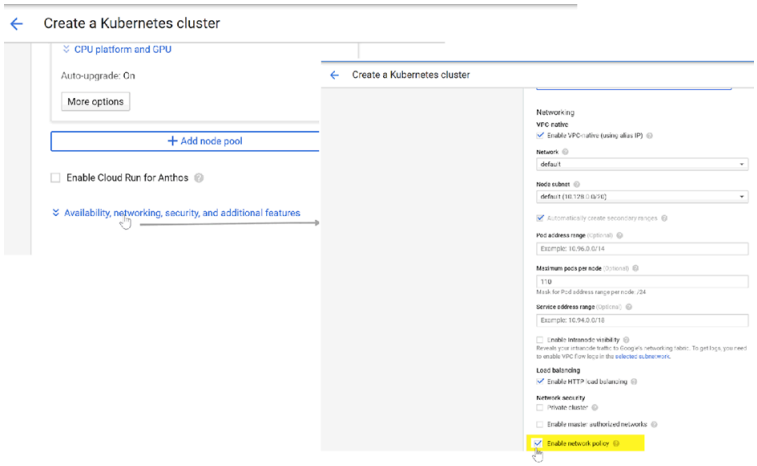
2. 单击 **Create**（创建）。
3. 选择 **GKE** 标准作为要使用的集群模式，然后单击配置。



4. 输入集群基本信息，包括名称、版本、位置、节点子网，然后单击创建。



如果集群在 *GKE* 上，请确保启用 *Kubernetes* 网络策略 *API* 以允许集群管理员指定允许相互通信的 *Pod*。同样，*CN-NGFW* 和 *CN-MGMT Pod* 进行通信也需要此 *API*。



1. 验证集群是否有足够的资源。默认 GKE 节点池规范不适用于 CN 系列防火墙。您必须确保集群具有 CN 系列先决条件资源，以便支持防火墙：

   kubectl get nodes

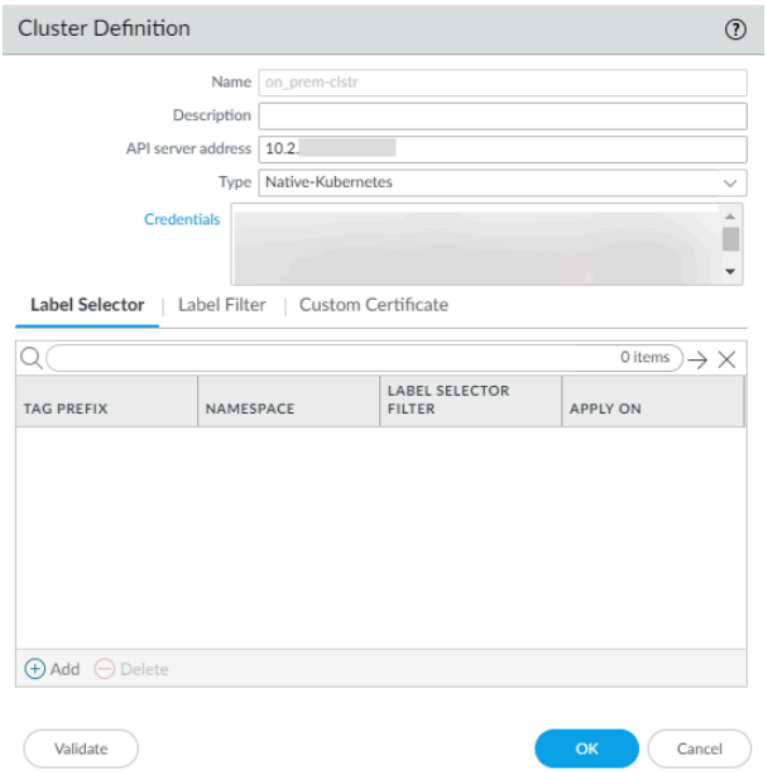
   kubectl describe node <node-name>

查看命令输出中容量标题下的信息，以了解指定节点上可用的 CPU 和内存。

CPU、内存和磁盘存储分配将取决于您的需求。参阅 CN 系列性能和可扩展性。

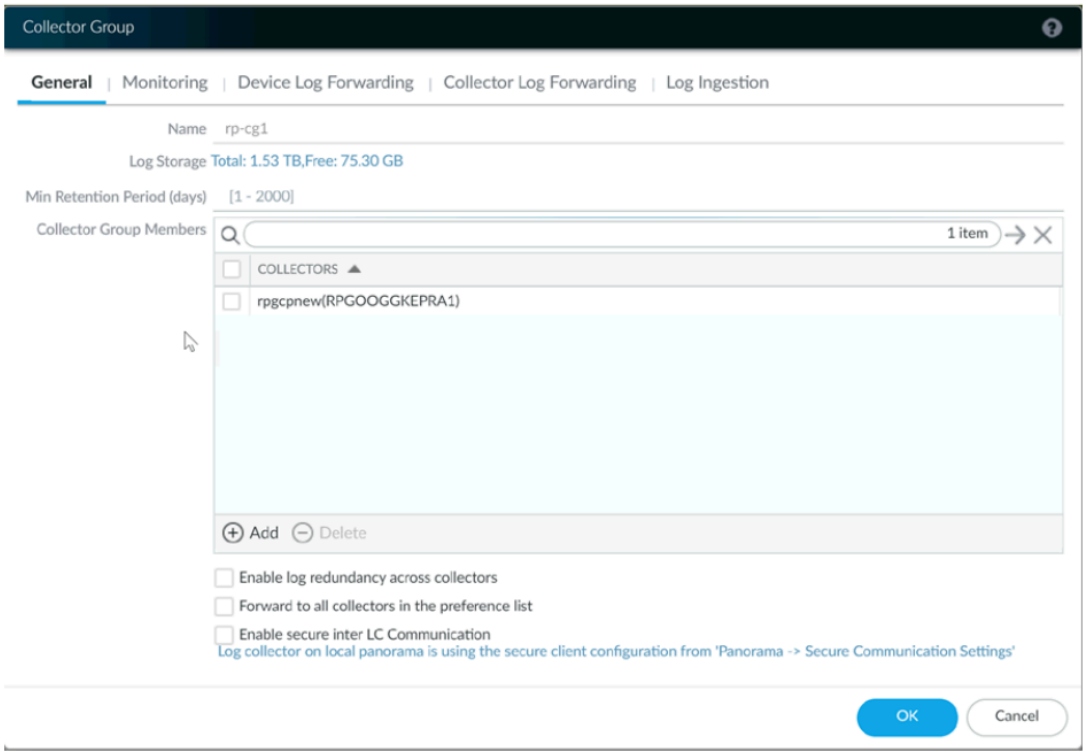
确保有以下信息：

- 收集端点 IP 地址，以便在 Panorama 上设置 API 服务器。



  Panorama 使用此 IP 地址连接到 Kubernetes 集群。

- 从 Panorama 收集模板堆栈名称、设备组名称、Panorama IP 地址和可选的日志收集器组名称。



有关详细信息，请参阅创建父设备组和模板堆栈。

- 收集授权代码以及自动注册 PIN ID 和值。
- 将映像下载到的容器映像存储库的位置。

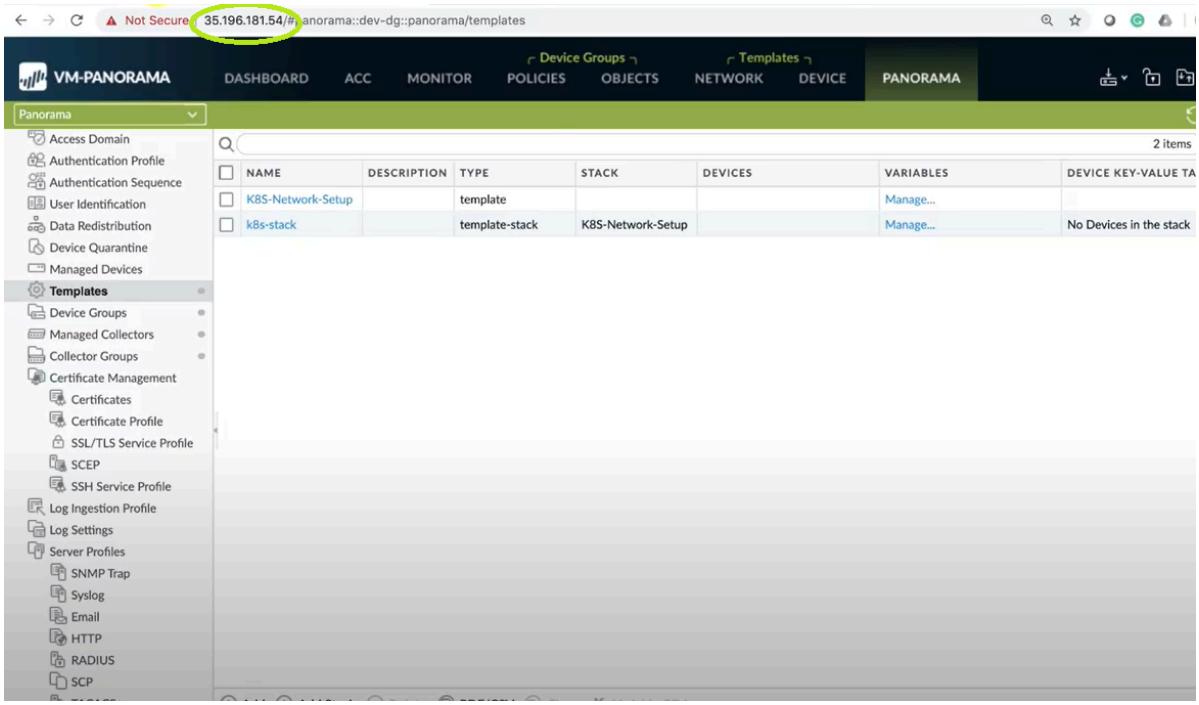
STEP 2 | （可选）如果您在 Kubernetes 插件中为 Panorama 配置了自定义证书，则必须通过执行以下命令来创建证书密钥。不要更改 ca.crt 中的文件名。pan-cn-mgmt.yaml 和 pan-cn-ngfw.yaml 中自定义证书的数量是可选的。

```
kubectl -n kube-system create secret generic custom-ca --from-file=ca.crt
```

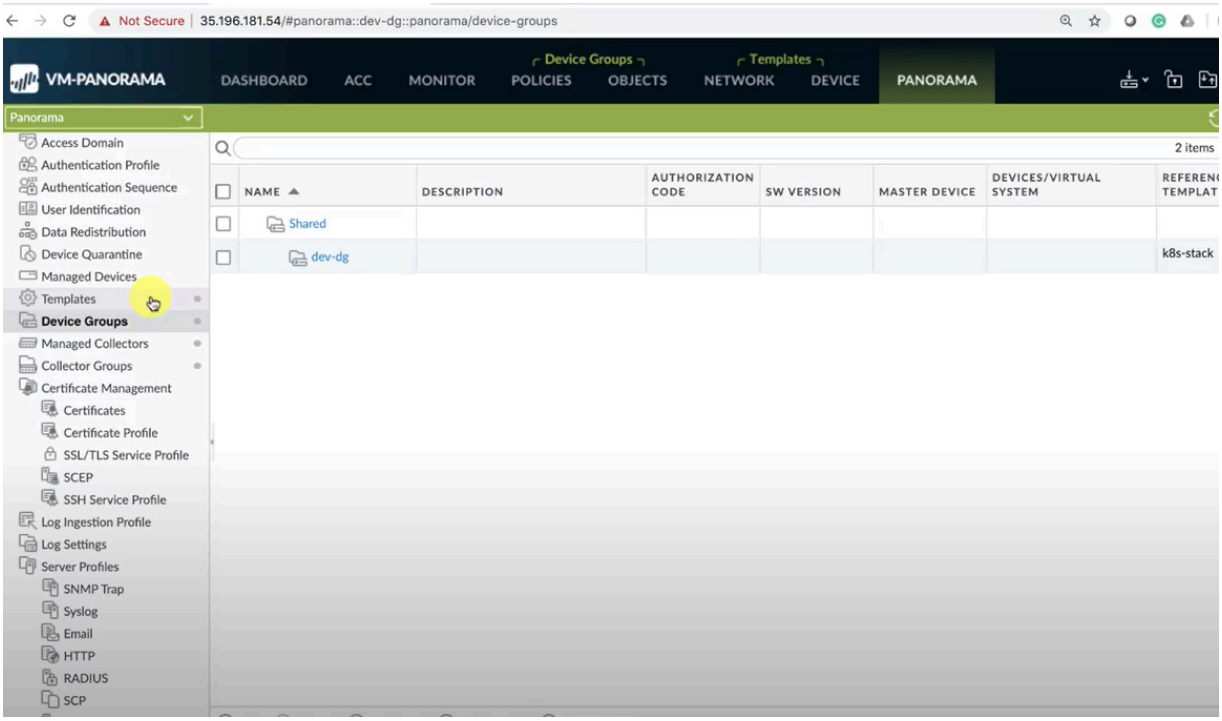
STEP 3 | 编辑 YAML 文件以提供部署 CN 系列防火墙所需的详细信息。

```
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:
  PAN_DEVICE_GROUP: "dev-dg"
  PAN_TEMPLATE: "k8s-stack"
#Non-mandatory parameters
  PAN_PANORAMA_CGNAME: "rp-cg1"
  #PAN_CERTIFICATE: ""
  #PAN_CERTKEYFILE: ""
  #PAN_CERTPASSPHRASE: ""
```

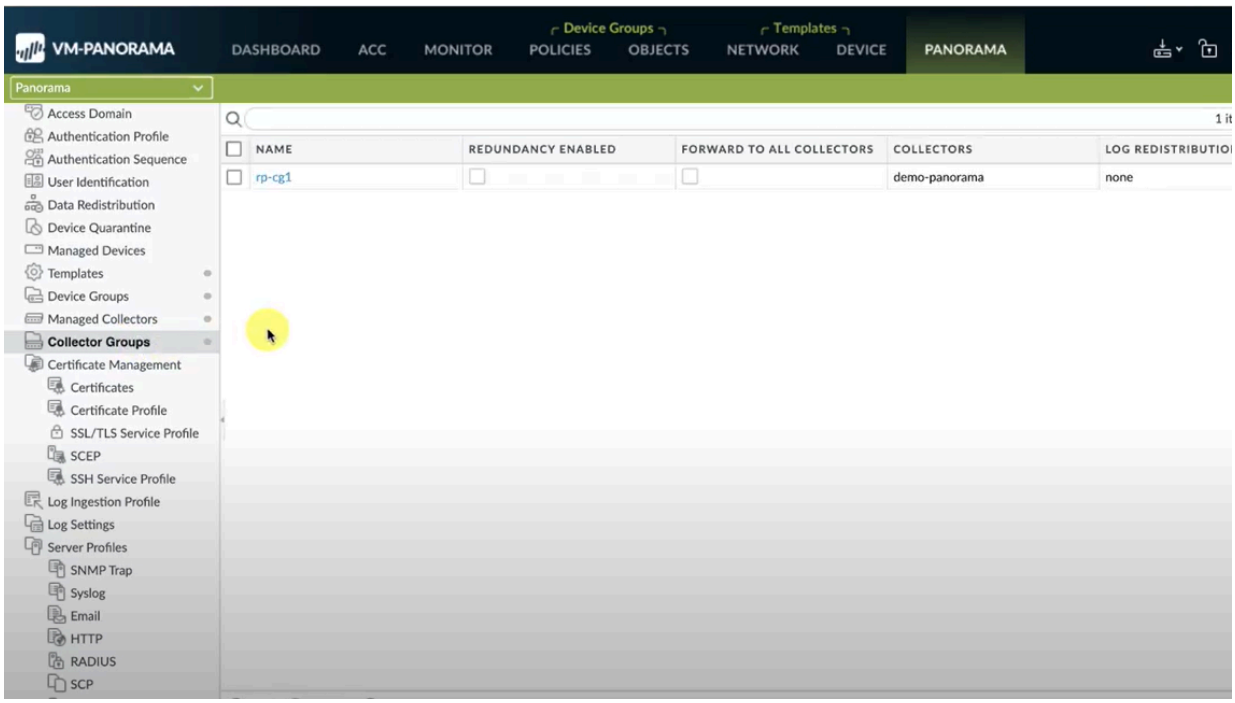
您必须确保 YAML 文件中 `PAN_PANORAMA_IP` 参数的值与实际 Panorama IP 地址匹配，如下图所示：



您必须确保 YAML 文件上的 `PAN_DEVICE_GROUP` 和 `PAN_TEMPLATE` 的参数值与在 Panorama 上创建的设备组和模板堆栈的名称匹配，如下图所示：



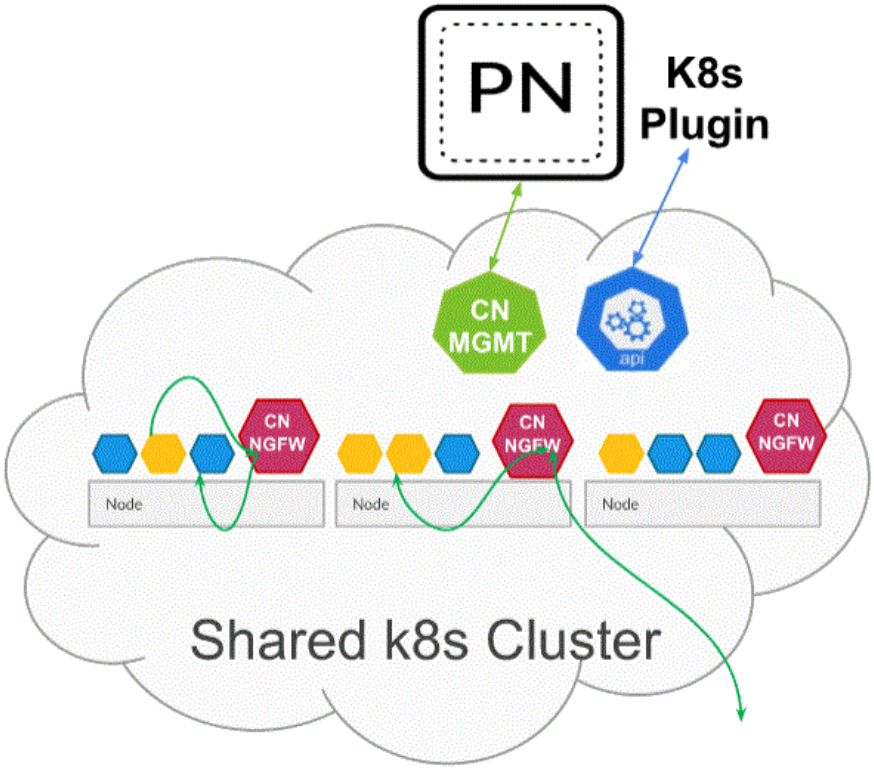
您必须确保 PAN_PANORAMA_CG_NAME 的参数值与创建的日志收集器名称相同。



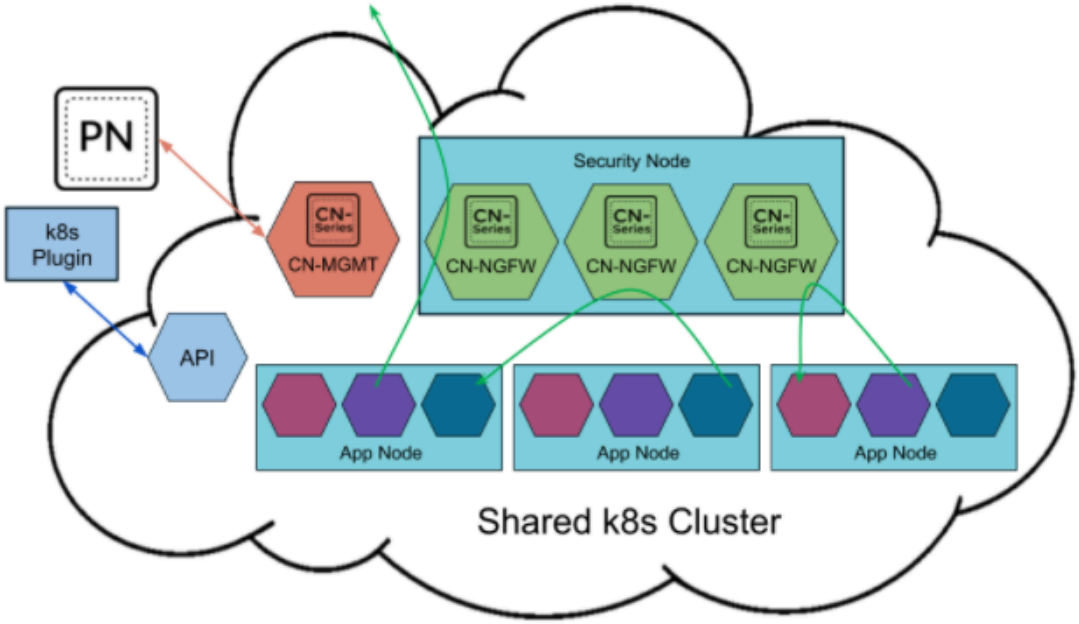
有关详细信息，请参阅 CN 系列部署 YAML 文件中的可编辑参数。

STEP 4 | 如果在 Kubernetes 环境中使用自动缩放，请参阅启用水平 Pod 自动缩放。

STEP 5 | 部署 CN-NGFW 服务。执行以下步骤：



当部署为 Kubernetes 服务时，CN-NGFW 的实例可以部署在安全节点上，而应用程序 Pod 流量将被重定向到可用的 CN-NGFW 实例以进行检查和实施。



1. 使用 pan-cni-serviceaccount.yaml 文件验证您是否已创建服务帐户。

   参阅创建用于集群身份验证的服务帐户。
2. 使用 Kubectl 运行 pan-cni-configmap.yaml 文件。

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

3. 使用 kubectl 运行 pan-cn-ngfw-svc.yaml。

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

> 该 *yaml* 必须在 *pan-cni.yaml* 之前部署。

4. 使用 Kubectl 运行 pan-cni.yaml 文件。

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

5. 验证是否已修改 pan-cni-configmap 和 pan-cni YAML 文件。
6. 运行以下命令并验证输出是否与以下示例相似。

STEP 6 | 部署 CN-MGMT StatefulSet。

默认情况下，部署管理平面作为提供容错功能的 StatefulSet。最多可以将 30 个防火墙 CN-NGFW Pod 连接到 CN-MGMT StatefulSet。

1. （仅对于静态配置的 PV 为必需）为 CN-MGMT StatefulSet 部署持久卷 (PV)。
   1. 创建目录以匹配 pan-cn-pv-local.yaml 文件中定义的本地卷名称。

      在至少 2 个工作节点上需要六 (6) 个目录。登录将在其中部署 CN-MGMT StatefulSet 的每个工作节点，以创建目录。例如，要创建名为 /mnt/pan-local1 到 /mnt/pan-local6 的目录，请使用以下命令：

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

   2. 修改 pan-cn-pv-local.yaml。

      在 `nodeaffinity` 下匹配主机名，并验证是否已修改在 `spec.local.path` 中创建的上述目录，然后部署文件以创建新的存储类 pan-local-storage 和本地 PV。
2. 确认您已修改 pan-cn-mgmt-configmap 和 pan-cn-mgmt YAML 文件

   EKS 中的 pan-cn-mgmt-configmap 示例。

```
 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
```

pan-cn-mgmt.yaml 文件示例

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

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

3. 使用 Kubectl 运行 yaml 文件。

   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

   您必须运行 pan-mgmt-serviceaccount.yaml，前提是您之前未完成为集群身份验证创建服务帐户。

4. 通过运行以下命令验证 CN-MGMT Pod 是否已启动：

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

   大约需要 5 至 6 分钟的时间。

STEP 7 | 部署 CN-NGFW Pod。

1. 验证是否已按 PAN-CN-NGFW-CONFIGMAP 和 PAN-CN-NGFW 中的详细说明修改 YAML 文件。

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

2. 使用 Kubectl apply 运行 pan-cn-ngfw-configmap.yaml 文件。

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

3. 使用 Kubectl apply 运行 pan-cn-ngfw.yaml 文件。

   kubectl apply -f pan-cn-ngfw.yaml

4. 确认已部署 CN-NGFW Pod。

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

STEP 8 | 通过执行以下步骤启用水平 Pod 自动缩放：

1. 在 CN 系列集群中部署自定义指标堆栈驱动程序适配器。集群名称必须通过 K8s 机密提供。
2. 从 Palo Alto Networks GitHub 存储库下载 GKE 特定的 HPA yaml 文件。
3. 如果 CN-MGMT 部署在自定义命名空间中，请使用自定义命名空间更新 pan-cn-adapater.yaml。默认命名空间为 **kube-system**。
4. 更新特定于 GKE 的 pan-cn-mgmt-configmap.yaml 中的 HPA 参数。

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

```
#HPA_NAME: "<name>" #unique name to identify hpa resource per
 namespace or per tenant
```

```
#PUSH_INTERVAL:"15" #time interval to publish metrics to
 stackdriver
```

5. 使用上面 pan-cn-mgmt-configmap.yaml 文件中更新的 HPA_NAME（替换名称）修改 **pan-cn-hpa-dp.yaml** 和 **pan-cn-hpa-mp.yaml**，并根据要触发的 HPA 更新指标。
   1. 输入副本的最小和最大数量。
   2. （可选）更改缩小和放大频率值以使其适合您的部署。如果不更改这些值，则会使用默认值。
   3. （可选）更改要用于扩展的每个指标的阈值。如果不更改这些值，则会使用默认值。
   4. 保存更改。
6. 部署 HPA yaml 文件。必须按照下面所述的顺序部署这些文件。
   1. 使用 Kubectl 运行 pan-cn-adapter.yaml

      ```
      kubectl apply -f pan-cn-adapter.yaml
      ```
   2. 使用 Kubectl 运行 pan-cn-crole.yaml

      ```
      kubectl apply -f pan-cn-crole.yaml
      ```
   3. 使用 Kubectl 运行 pan-cn-hpa-dp.yaml

      ```
      kubectl apply -f pan-cn-hpa-dp.yaml
      ```
   4. 使用 Kubectl 运行 pan-cn-hpa-mp.yaml

      ```
      kubectl apply -f pan-cn-hpa-mp.yaml
      ```
7. 验证部署。
   - 使用 kubectl 验证自定义指标命名空间中的自定义指标适配器 Pod 是否存在。

     ```
     kubectl get pods -n custom-metrics
     ```
   - 使用 kubectl 检查 HPA 资源。

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

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

有关更多信息，请参阅在 CN 系列上启用水平 Pod 自动缩放。

STEP 9 | 验证您是否可以在 Kubernetes 集群上看到 CN-MGMT、CN-NGFW 和 PAN-CNI。

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

STEP 10 | 为应用程序 yaml 文件或命名空间添加注释，以便将来自其新 Pod 的流量重定向到防火墙。

您需要添加以下注解，以将流量重定向到 CN-NGFW 来进行检查：

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

例如，对于“default”命名空间中的所有新 Pod：

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

在某些平台上，当在 *CNI* 插件链中未激活 *pan-cni* 时，可以启动应用程序 *Pod*。为避免此类情况，您必须按如下在应用程序 *Pod YAML* 中指定卷。

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

STEP 11 | 在集群中部署应用程序。

在 GKE 上将 CN 系列防火墙部署为 DaemonSet

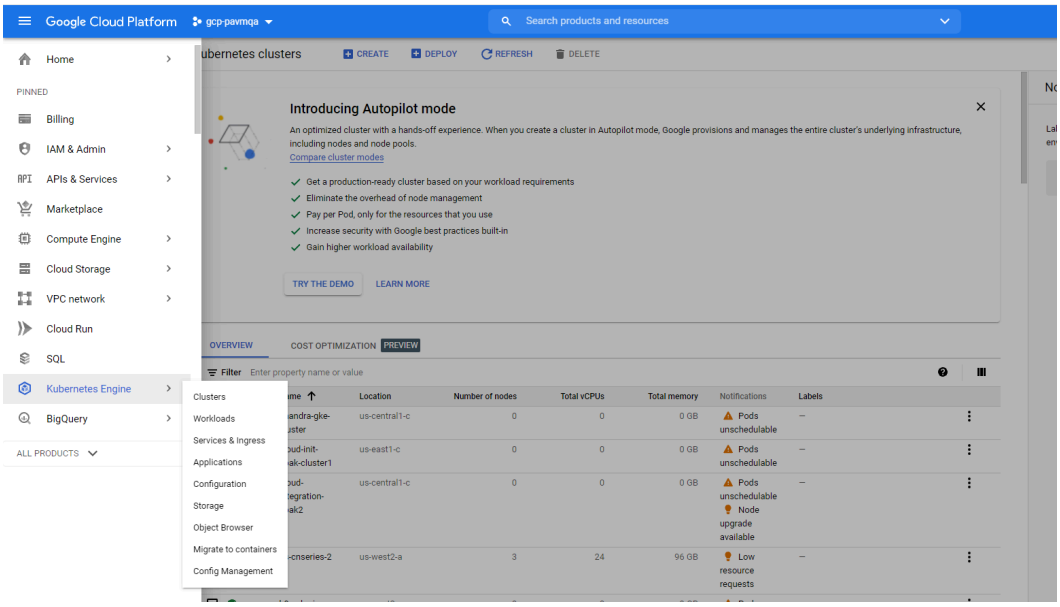
在何处可以使用？	需要提供什么？
• CN-Series部署	• CN-Series 10.1.x or above Container Images • 运行 PAN-OS 10.1.x 或更高版本的 Panorama • 使用 Helm 部署 CN 系列的 Helm 3.6 or above version client

完成以下过程以在 GKE 平台上将 CN 系列防火墙部署为 Daemonset：

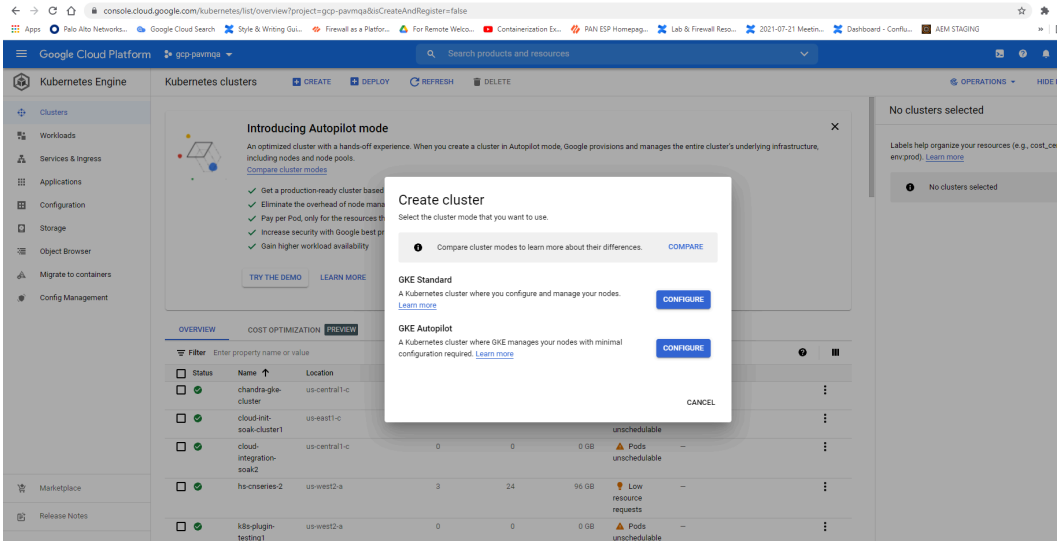
STEP 1 | 设置 Kubernetes 集群。

要在 GKE 中创建集群，请执行以下操作：

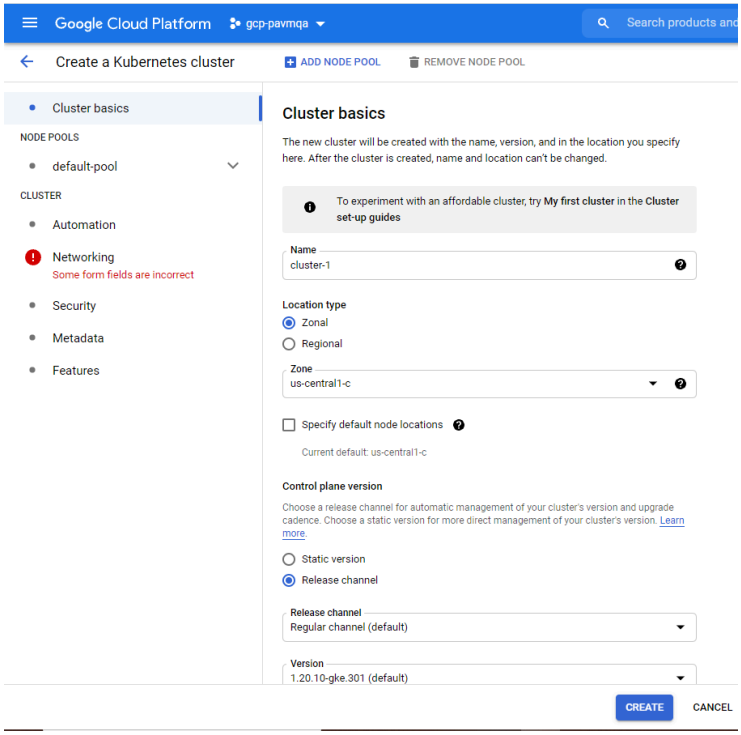
1. 单击导航菜单，转到 **Kubernetes Engine**，然后选择集群。



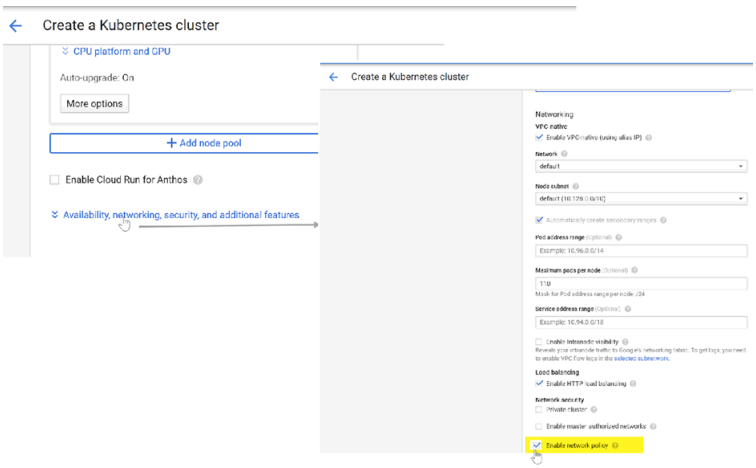
2. 单击 **Create**（创建）。
3. 选择 **GKE** 标准作为要使用的集群模式，然后单击配置。



4. 输入集群基本信息，包括名称、版本、位置、节点子网，然后单击创建。



如果集群在 *GKE* 上，请确保启用 *Kubernetes* 网络策略 *API* 以允许集群管理员指定允许相互通信的 *Pod*。同样，*CN-NGFW* 和 *CN-MGMT Pod* 进行通信也需要此 *API*。



验证集群是否有足够的资源。确保该集群具有 CN 系列系统要求，以便支持防火墙。

kubectl get nodes

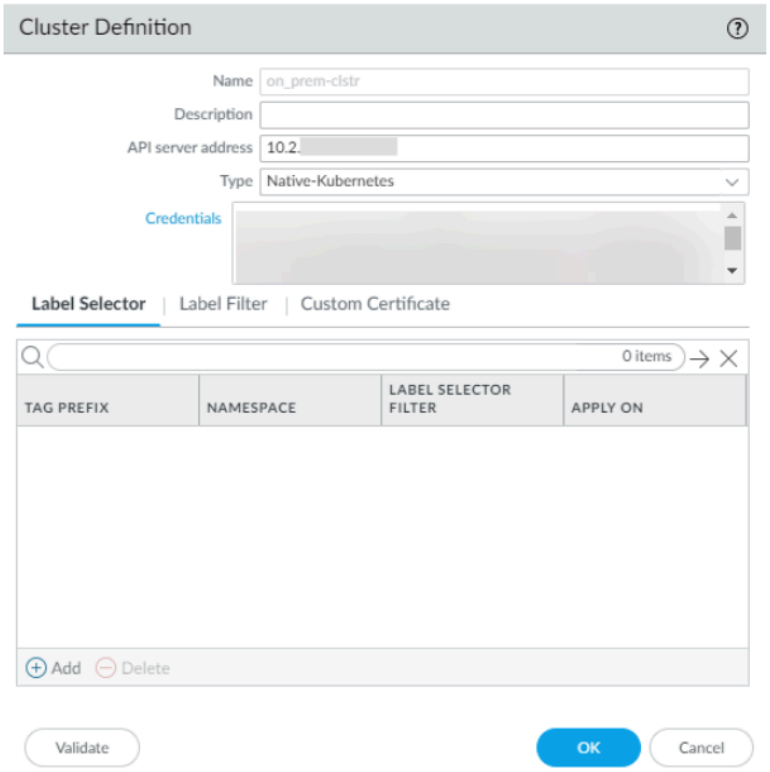
kubectl describe node <node-name>

查看命令输出中容量标题下的信息，以了解指定节点上可用的 CPU 和内存。

CPU、内存和磁盘存储分配将取决于您的需求。请参阅 CN 系列的性能和可扩展性。

确保有以下信息：

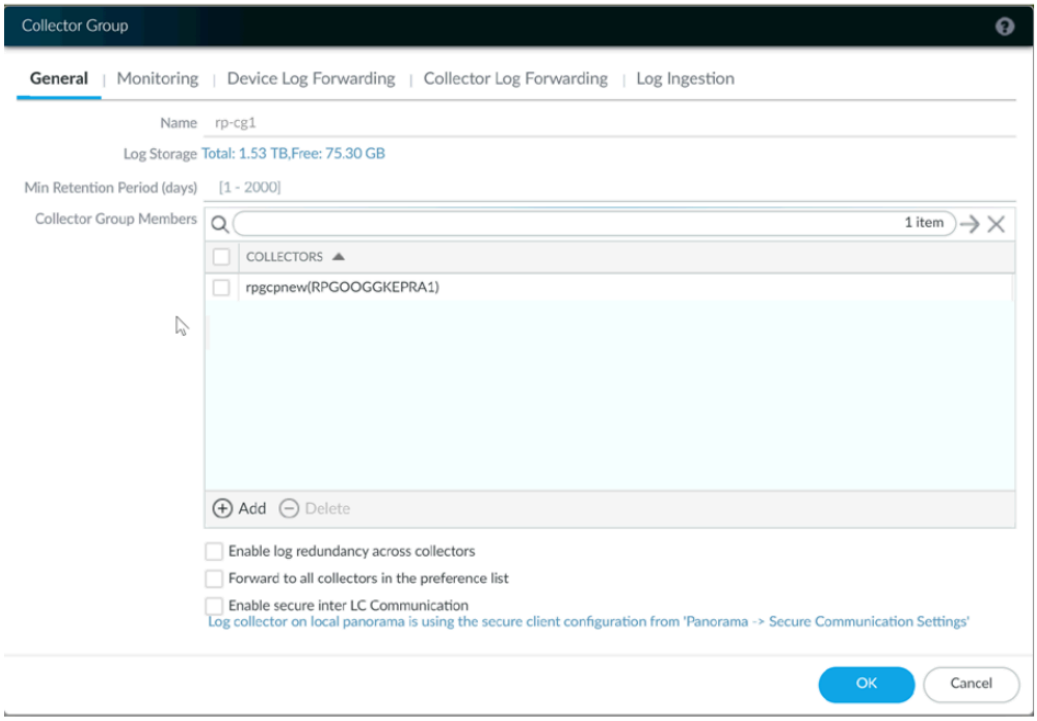
- 收集端点 IP 地址，用于在 Panorama 上设置 API 服务器。



Panorama 使用此 IP 地址连接到 Kubernetes 集群。

有关更多信息，请参阅 设置 Kubernetes 插件以监控集群。

- 从 Panorama 收集模板堆栈名称、设备组名称、Panorama IP 地址和可选的日志收集器组名称。



有关详细信息，请参阅创建父设备组和模板堆栈。

- 收集授权代码以及自动注册 PIN ID 和值。
- 将映像下载到的容器映像存储库的位置。

STEP 2 | （可选）如果您在 Kubernetes 插件中为 Panorama 配置了自定义证书，则必须通过执行以下命令来创建证书密钥。不要更改 ca.crt 中的文件名。pan-cn-mgmt.yaml 和 pan-cn-ngfw.yaml 中自定义证书的数量是可选的。

```
kubectl -n kube-system create secret generic custom-ca --from-file=ca.crt
```

STEP 3 | 编辑 YAML 文件以提供部署 CN 系列防火墙所需的详细信息。

您需要替换 YAML 文件中的映像路径，以包括私有 Google 容器注册表的路径并提供所需的参数。有关详细信息，请参阅 CN 系列部署 yaml 文件中的可编辑参数。

STEP 4 | 部署 CNI DaemonSet。

CNI 容器作为 DaemonSet 部署（每个节点一个 Pod），并且在 CN-NGFW Pod 上为节点上部署的每个应用程序创建两个接口。使用 kubectl 命令运行 pan-cni YAML 文件时，该容器将成为每个节点上服务链的一部分。

1. CN 系列防火墙需要三个具有最低权限的服务帐户，这些帐户授权防火墙与 Kubernetes 集群资源进行通信。您要 CN 系列集群身份验证创建服务帐户，并验证是否已使用 pan-cni-serviceaccount.yaml 创建服务帐户。
2. 使用 Kubectl 运行 pan-cni-configmap.yaml 文件。

   kubectl apply -f pan-cni-configmap.yaml
3. 使用 Kubectl 运行 pan-cni.yaml 文件。

   kubectl apply -f pan-cni.yaml
4. 验证是否已修改 pan-cni-configmap 和 pan-cni YAML 文件。
5. 运行以下命令并验证输出是否与以下示例相似。

STEP 5 | 部署 CN-MGMT StatefulSet。

默认情况下，部署管理平面作为提供容错功能的 StatefulSet。最多可以将 30 个防火墙 CN-NGFW Pod 连接到 CN-MGMT StatefulSet。

1. 验证是否已修改 pan-cn-mgmt-configmap 和 pan-cn-mgmt YAML 文件。

示例 **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

# Panorama settings

PAN_PANORAMA_IP: "x.y.z.a"

PAN_DEVICE_GROUP: "dg-1"

PAN_TEMPLATE_STACK: "temp-stack-1"

PAN_CGNAME:"CG-GKE"

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
```

pan-cn-mgmt.yaml 文件示例

```
initContainers:
```

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

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

```
containers: - name: pan-mgmt
```

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

```
terminationMessagePolicy:FallbackToLogsOnError
```

2. 使用 Kubectl 运行 yaml 文件。

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

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

   kubectl apply -f pan-cn-mgmt.yaml

   如果您之前未完成为集群身份验证创建服务帐户，则必须运行 pan-mgmt-serviceaccount.yaml。

3. 验证 CN-MGMT Pod 是否启动。

   大约需要 5 至 6 分钟的时间。

   使用 **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 6 | 部署 CN-NGFW Pod。

默认情况下，将防火墙数据平面 CN-NGFW Pod 作为 DaemonSet 部署。CN-NFGW Pod 的实例可保护节点上最多 30 个应用程序 Pod 的流量。

1. 验证是否已按 PAN-CN-NGFW-CONFIGMAP 和 PAN-CN-NGFW 中的详细说明修改 YAML 文件。

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

2. 使用 Kubectl apply 运行 pan-cn-ngfw-configmap.yaml 文件。

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

3. 使用 Kubectl apply 运行 pan-cn-ngfw.yaml 文件。

   kubectl apply -f pan-cn-ngfw.yaml

4. 验证所有 CN-NGFW Pod 是否正在运行（集群中每个节点一个）。

   以下是 4 个节点本地集群的输出示例。

   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 | 验证您是否可以在 Kubernetes 集群上看到 CN-MGMT、CN-NGFW 和 PAN-CNI。

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

STEP 8 | 为应用程序 yaml 文件或命名空间添加注释，以便将来自其新 Pod 的流量重定向到防火墙。

您需要添加以下注解，以将流量重定向到 CN-NGFW 来进行检查：

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

例如，对于“default”命名空间中的所有新 Pod：

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

在某些平台上，当在 *CNI* 插件链中未激活 *pan-cni* 时，可以启动应用程序 *Pod*。为避免此类情况，您必须按如下在应用程序 *Pod YAML* 中指定卷。

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

STEP 9 | 在集群中部署应用程序。

在 OKE 上部署 CN 系列防火墙

在何处可以使用？	需要提供什么？
• CN-Series部署	• CN-Series 10.1.x or above Container Images • 运行 PAN-OS 10.1.x 或更高版本的 Panorama • 使用 Helm 部署 CN 系列的 Helm 3.6 or above version client

Oracle Kubernetes Engine (OKE) 是一种 OCI 服务，可支持部署 Kubernetes 集群。现在，您可以在 OKE 集群上将 CN 系列防火墙作为 DaemonSet 部署，并将 Kubernetes 作为服务部署。

在利用 CN 系列保护 Kubernetes 环境的安全中查看 CN 系列构建块和该工作流程的高级概述后，您可以开始在 OKE 平台上部署 CN 系列防火墙，以保护同一集群中容器之间，以及容器和其他工作负载类型（例如虚拟机和裸机服务器）之间的流量。

您需要标准 *Kubernetes* 工具（例如 *kubectl* 或 *Helm*）部署和管理 *Kubernetes* 集群、应用程序和防火墙服务。

有关更多信息，请参阅使用 Helm 图表和模板部署 CN 系列防火墙。*Panorama* 并非旨在成为 *Kubernetes* 集群部署和管理的 *Orchestrator*。托管 *Kubernetes* 提供商已提供用于集群管理的模板。您还可以使用社区支持的模板部署具有 Helm 和 Terraform 的 *CN* 系列。

- 在 OKE 上将 CN 系列防火墙部署为 Kubernetes 服务
- 在 OKE 上将 CN 系列防火墙部署为 DaemonSet

从*CN* 系列即 *DaemonSet* 部署迁移到 *CN* 系列即服务之前（反之亦然），您必须删除并重新应用 `plugin-serviceaccount.yaml`。有关更多信息，请参阅为集群身份验证创建服务帐户。

- 在 *OKE* 上将 *CN* 系列部署为 *DaemonSet* 时，不能存在 `pan-plugin-cluster-mode-secret`。
- 在 *OKE* 上将 *CN* 系列部署为 *Kubernetes* 服务时，必须存在 `pan-plugin-cluster-mode-secret`。

在 OKE 上将 CN 系列防火墙部署为 Kubernetes 服务

在何处可以使用？	需要提供什么？
• CN-Series部署	• CN-Series 10.1.x or above Container Images • 运行 PAN-OS 10.1.x 或更高版本的 Panorama • 使用 Helm 部署 CN 系列的 Helm 3.6 or above version client

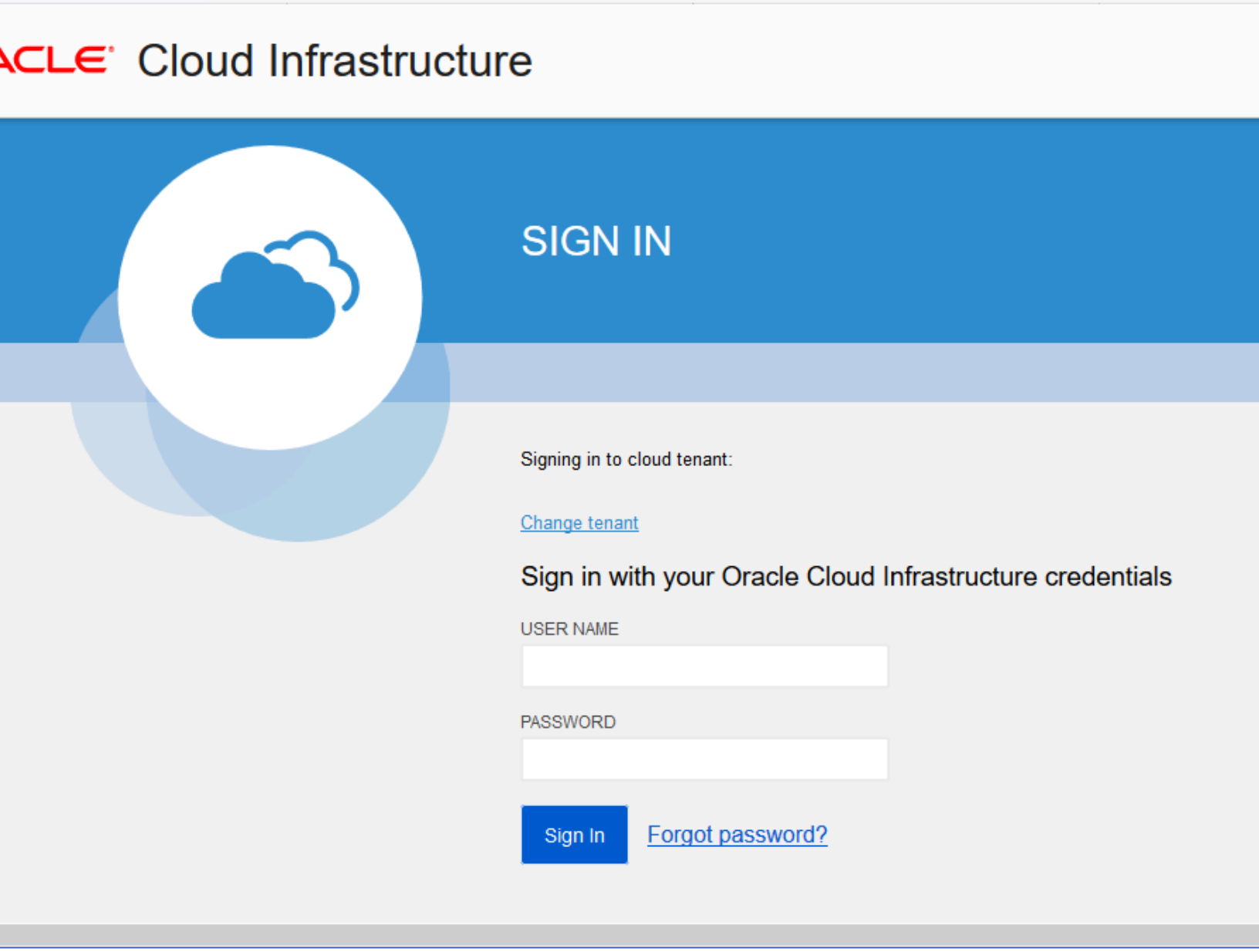
完成以下过程以在 OKE 平台上将 CN 系列防火墙部署为 Kubernetes 服务。

Oracle Linux 8.5 操作系统是在 *OKE* 上部署 *CN* 系列防火墙的唯一合格环境。

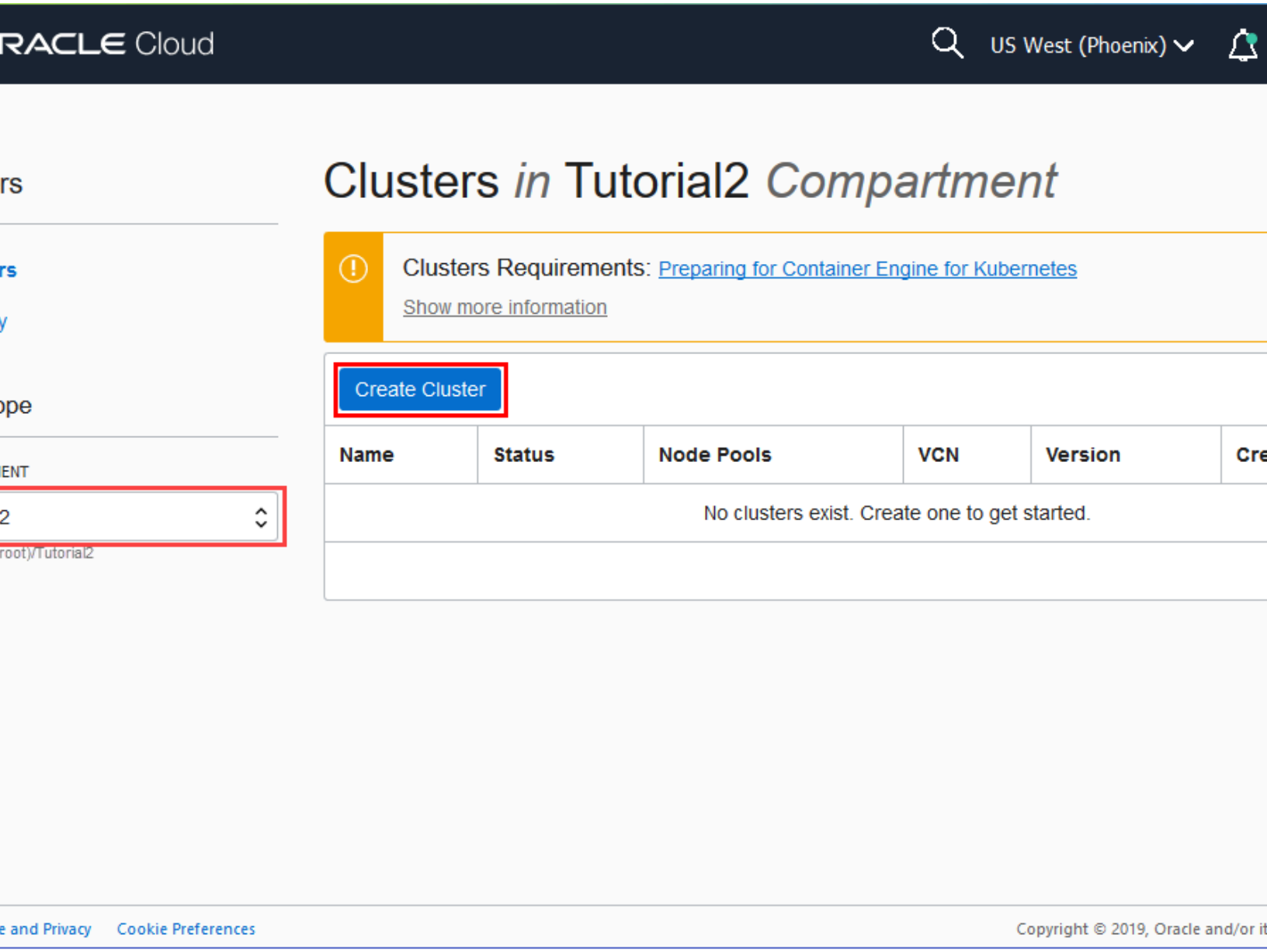
STEP 1 | 设置 Kubernetes 集群。

要在 OKE 中创建集群，请执行以下操作：

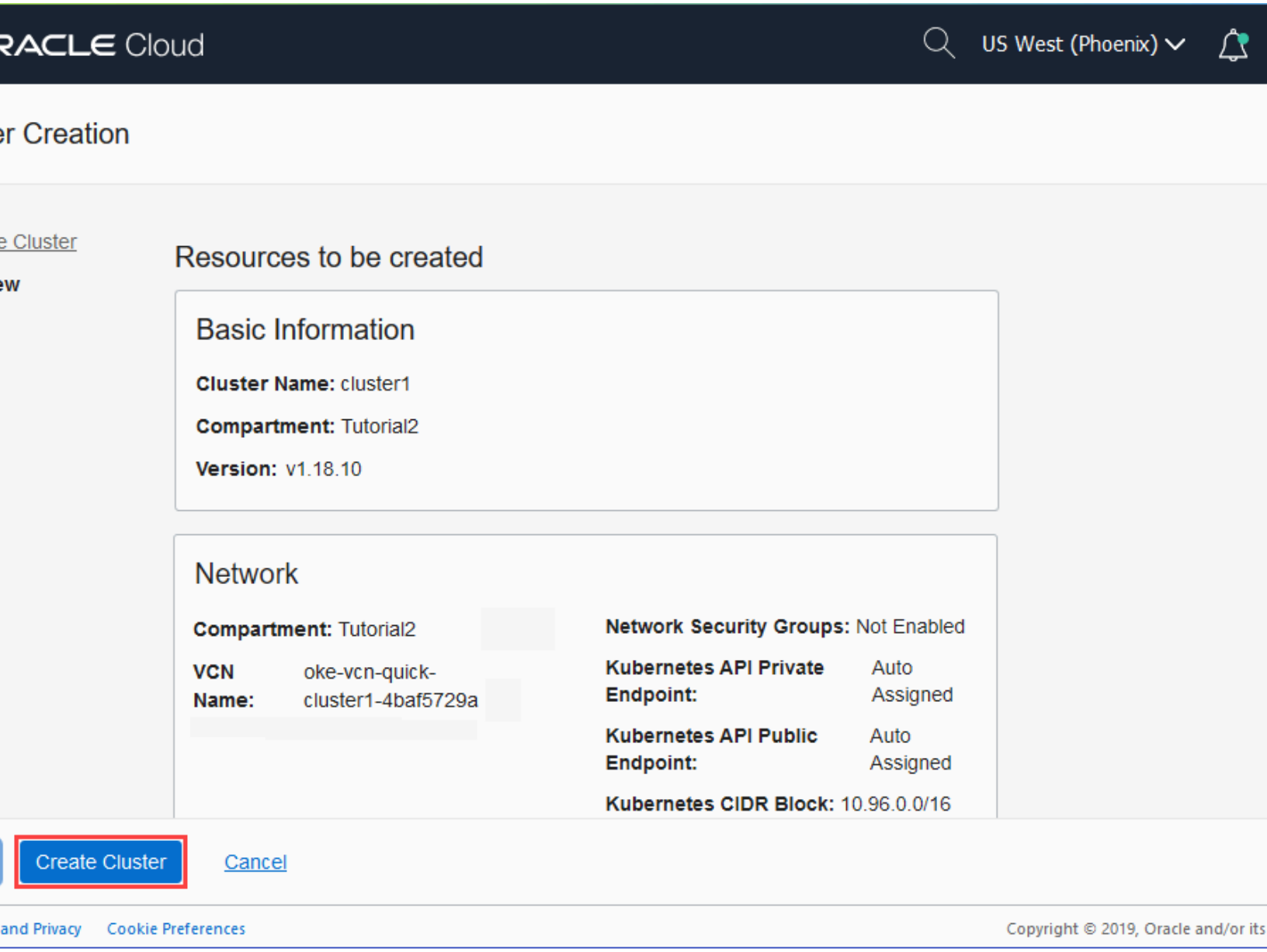
1. 登录到“Oracle 云基础架构”。



2. 单击导航菜单，转到在解决方案和平台下，然后单击开发人员服务。
3. 单击 **Kubernetes** 集群。
4. 选择一个区段并单击创建集群。



5. 在“创建集群”对话框中，单击快速创建，然后单击启动工作流程。
6. 在创建集群页面上，输入集群名称和其他详细信息。
7. 单击下一步以检查为新集群输入的详细信息。
8. 在“检查”页面上，单击创建集群。



1. 您必须确保集群具有 CN 系列先决条件资源，以便支持防火墙：

   kubectl get nodes

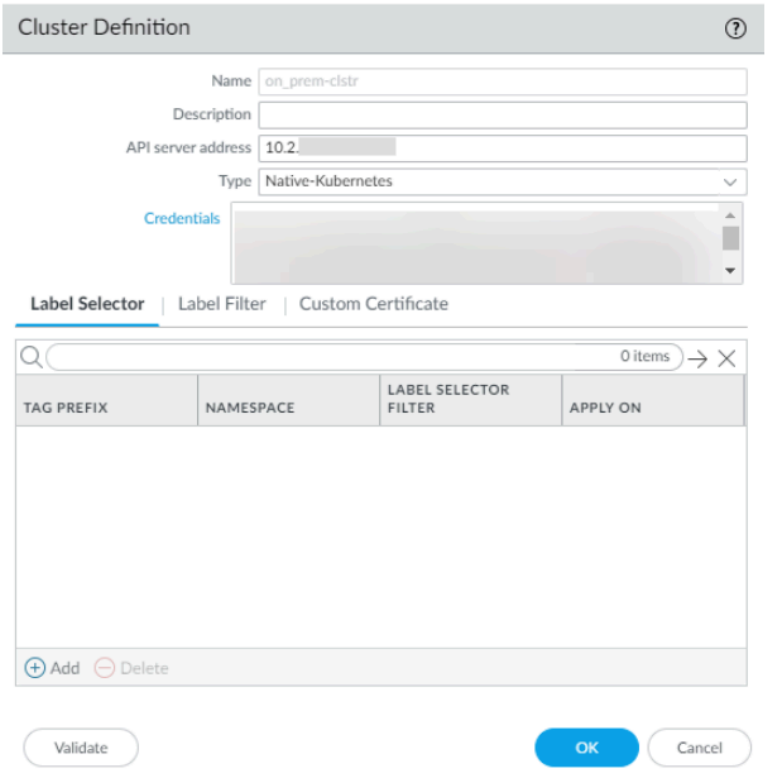
   kubectl describe node <node-name>

   查看命令输出中容量标题下的信息，以了解指定节点上可用的 CPU 和内存。

   CPU、内存和磁盘存储分配将取决于您的需求。参阅 CN 系列性能和可扩展性。

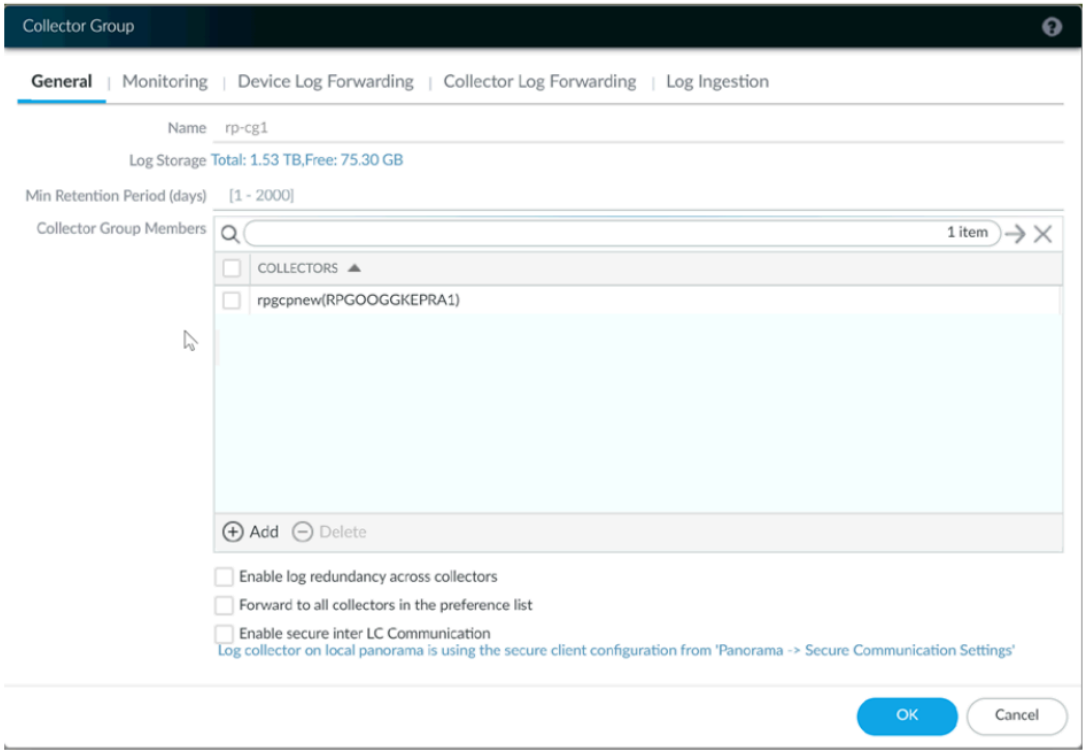
   确保有以下信息：

   - 收集端点 IP 地址，以便在 Panorama 上设置 API 服务器。



Panorama 使用此 IP 地址连接到 Kubernetes 集群。

- 从 Panorama 收集模板堆栈名称、设备组名称、Panorama IP 地址和可选的日志收集器组名称。



有关详细信息，请参阅创建父设备组和模板堆栈。

- 收集授权代码以及自动注册 PIN ID 和值。
- 准备好下载映像的容器映像存储库位置。

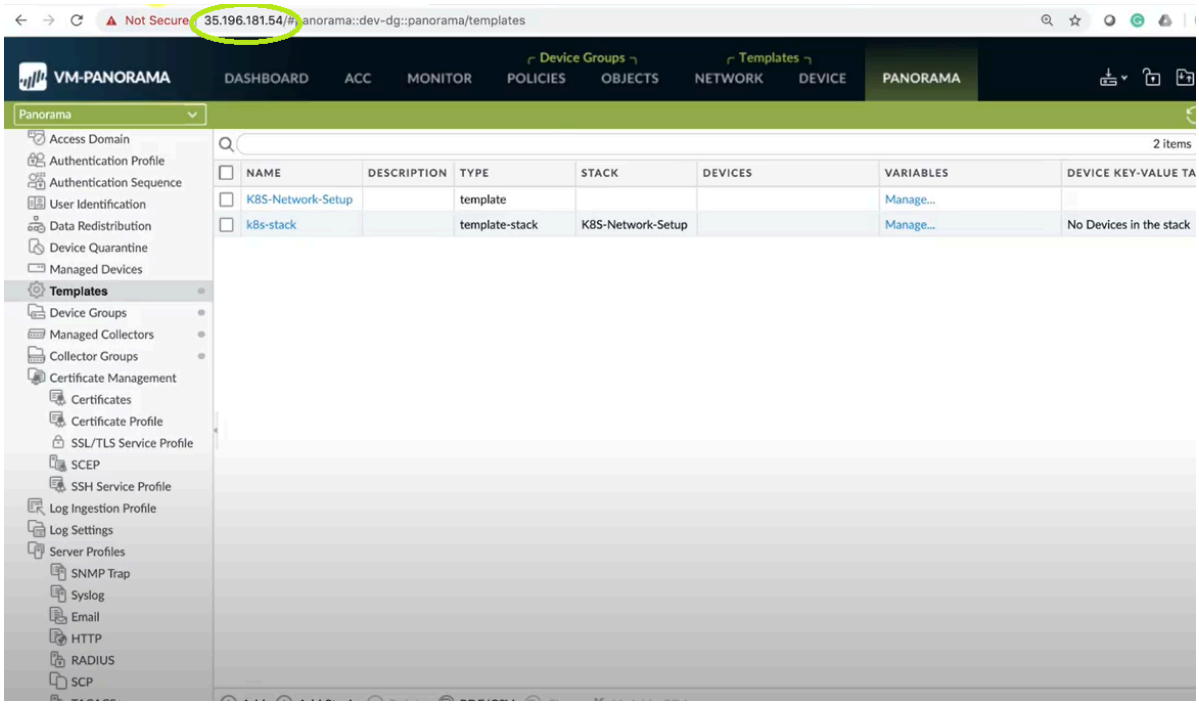
STEP 2 | （可选）如果您在 Kubernetes 插件中为 Panorama 配置了自定义证书，则必须通过执行以下命令来创建证书密钥。不要更改 ca.crt 中的文件名。pan-cn-mgmt-dynamic-pv.yaml 和 pan-cn-ngfw.yaml 中自定义证书的数量是可选的。

```
kubectl -n kube-system create secret generic custom-ca --from-file=ca.crt
```

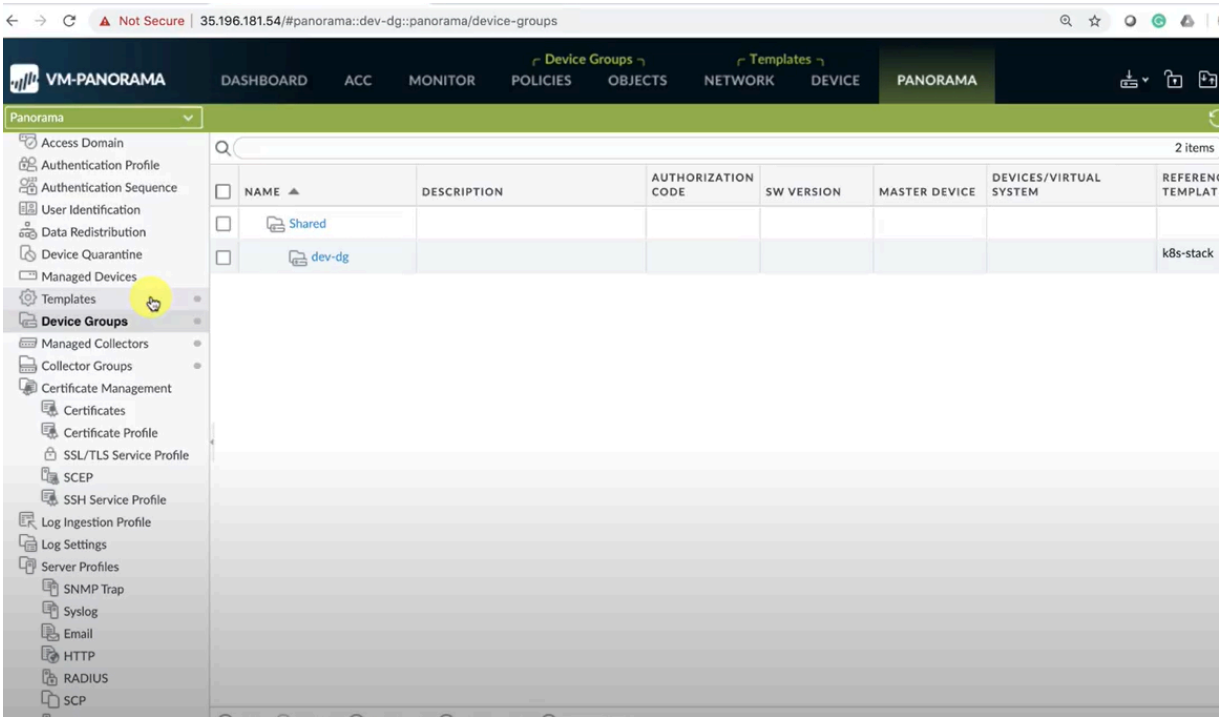
STEP 3 | 编辑 YAML 文件以提供部署 CN 系列防火墙所需的详细信息。

```
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:
  PAN_DEVICE_GROUP: "dev-dg"
  PAN_TEMPLATE: "k8s-stack"
#Non-mandatory parameters
  PAN_PANORAMA_CGNAME: "rp-cg1"
  #PAN_CERTIFICATE: ""
  #PAN_CERTKEYFILE: ""
  #PAN_CERTPASSPHRASE: ""
```

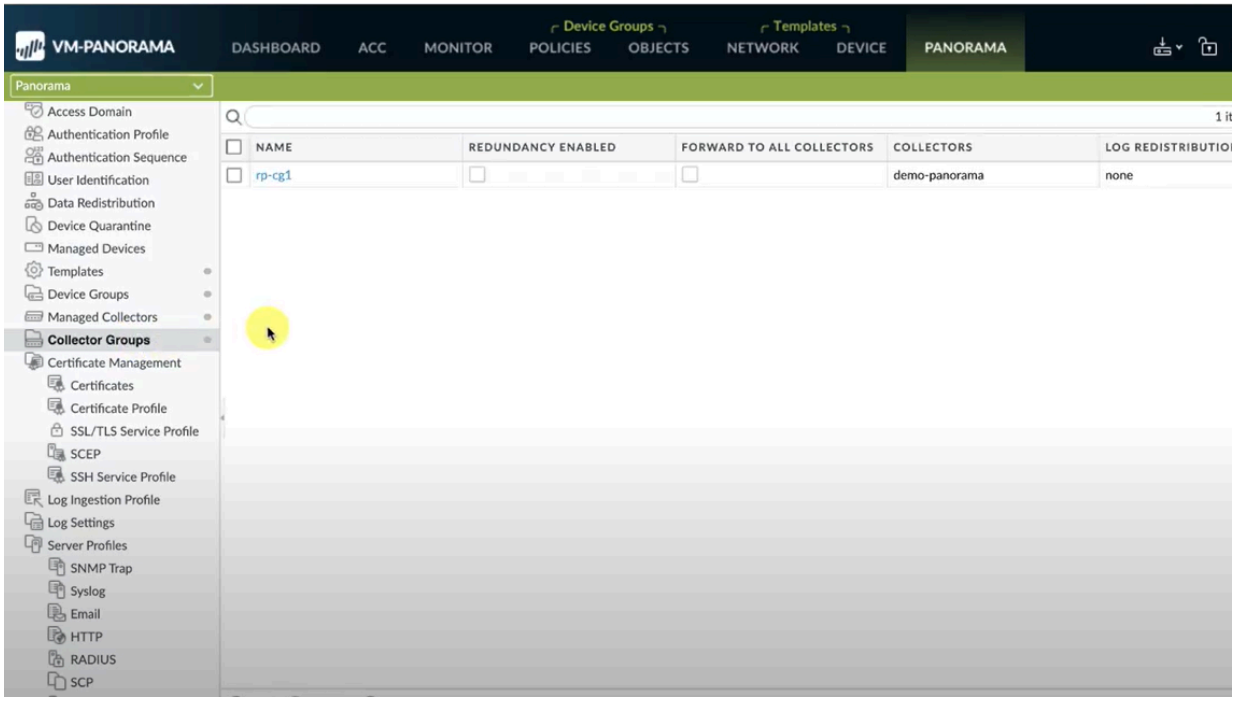
您必须确保 YAML 文件中 `PAN_PANORAMA_IP` 参数的值与实际 Panorama IP 地址匹配，如下图所示：



您必须确保 YAML 文件上的 `PAN_DEVICE_GROUP` 和 `PAN_TEMPLATE` 的参数值与在 Panorama 上创建的设备组和模板堆栈的名称匹配，如下图所示：

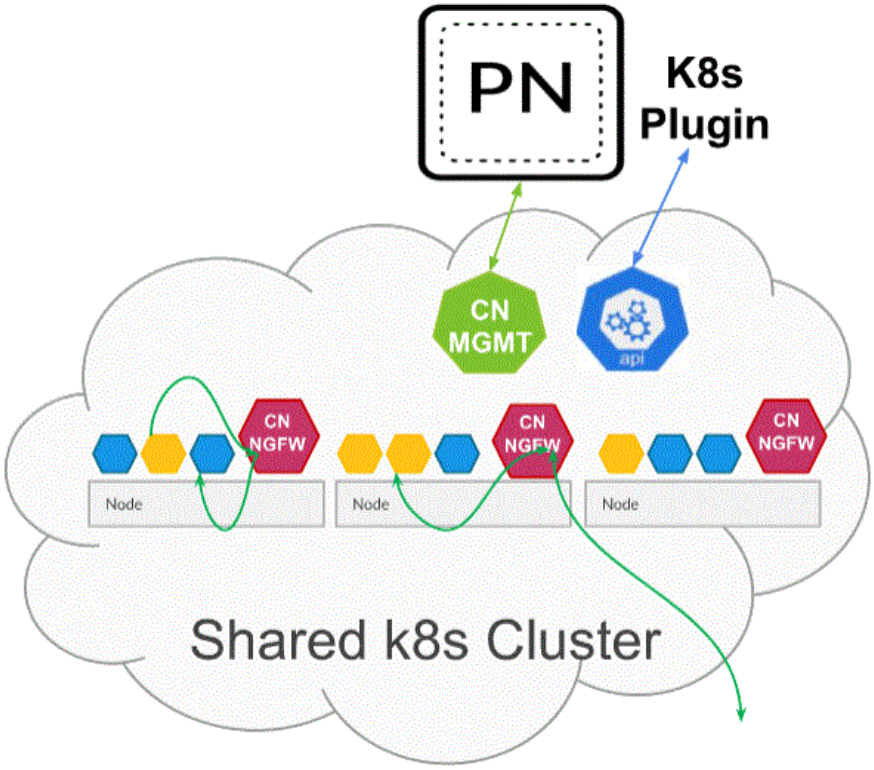


您必须确保 PAN_PANORAMA_CG_NAME 的参数值与创建的日志收集器名称相同。



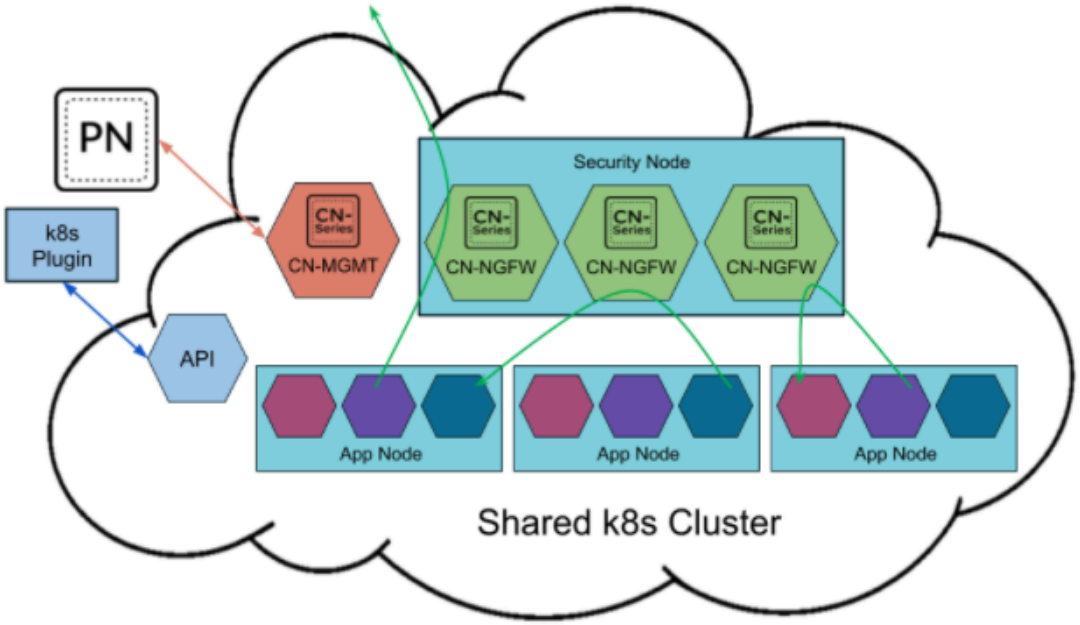
有关详细信息，请参阅 CN 系列部署 YAML 文件中的可编辑参数。

STEP 4 | 部署 CN-NGFW 服务。执行以下步骤：



当部署为 Kubernetes 服务时，CN-NGFW 的实例可以部署在安全节点上，而应用程序 Pod 流量将被重定向到可用的 CN-NGFW 实例以进行检查和实施。

在 *OKE* 上将 *CN* 系列防火墙部署为 *Kubernetes* 服务时，可以使用 pan-cn-k8s-service 本机文件夹中的 *yaml* 文件。



1. 使用 pan-cni-serviceaccount.yaml 文件验证您是否已创建服务帐户。

   参阅创建用于集群身份验证的服务帐户。

2. 使用 Kubectl 运行 pan-cni-configmap.yaml 文件。

   kubectl apply -f pan-cni-configmap.yaml

3. 使用 kubectl 运行 pan-cn-ngfw-svc.yaml。

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

   该 *yaml* 必须在 *pan-cni.yaml* 之前部署。

4. 使用 Kubectl 运行 pan-cni.yaml 文件。

   kubectl apply -f pan-cni.yaml

5. 验证是否已修改 pan-cni-configmap 和 pan-cni YAML 文件。
6. 运行以下命令并验证输出是否与以下示例相似。

STEP 5 | 部署 CN-MGMT StatefulSet。

默认情况下，部署管理平面作为提供容错功能的 StatefulSet。最多可以将 30 个防火墙 CN-NGFW Pod 连接到 CN-MGMT StatefulSet。

1. 验证是否已修改 pan-cn-mgmt-configmap 和 pan-cn-mgmt YAML 文件。

   OKE 中的 pan-cn-mgmt-configmap 示例。

```
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"
```

示例 pan-cn-mgmt-dynamic-pv.yaml

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

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

2. 使用 Kubectl 运行 yaml 文件。

   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

   您必须运行 pan-mgmt-serviceaccount.yaml，前提是您之前未完成为集群身份验证创建服务帐户。

3. 通过运行以下命令验证 CN-MGMT Pod 是否已启动：

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

   大约需要 5 至 6 分钟的时间。

STEP 6 | 部署 CN-NGFW Pod。

1. 验证是否已按 PAN-CN-NGFW-CONFIGMAP 和 PAN-CN-NGFW 中的详细说明修改 YAML 文件。

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

2. 使用 Kubectl apply 运行 pan-cn-ngfw-configmap.yaml 文件。

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

3. 使用 Kubectl apply 运行 pan-cn-ngfw.yaml 文件。

   kubectl apply -f pan-cn-ngfw.yaml

4. 确认已部署 CN-NGFW Pod。

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

STEP 7 | 验证您是否可以在 Kubernetes 集群上看到 CN-MGMT、CN-NGFW 和 PAN-CNI。

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

STEP 8 | 为应用程序 yaml 文件或命名空间添加注释，以便将来自其新 Pod 的流量重定向到防火墙。

您需要添加以下注解，以将流量重定向到 CN-NGFW 来进行检查：

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

例如，对于“default”命名空间中的所有新 Pod：

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

在某些平台上，当在 *CNI* 插件链中未激活 *pan-cni* 时，可以启动应用程序 *Pod*。为避免此类情况，您必须按如下在应用程序 *Pod YAML* 中指定卷。

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

STEP 9 | 在集群中部署应用程序。

在 OKE 上将 CN 系列防火墙部署为 DaemonSet

在何处可以使用？	需要提供什么？
• CN-Series部署	• CN-Series 10.2.x or above Container Images • 运行 PAN-OS 10.2.x 或更高版本的 Panorama • 使用 Helm 部署 CN 系列的 Helm 3.6 or above version client

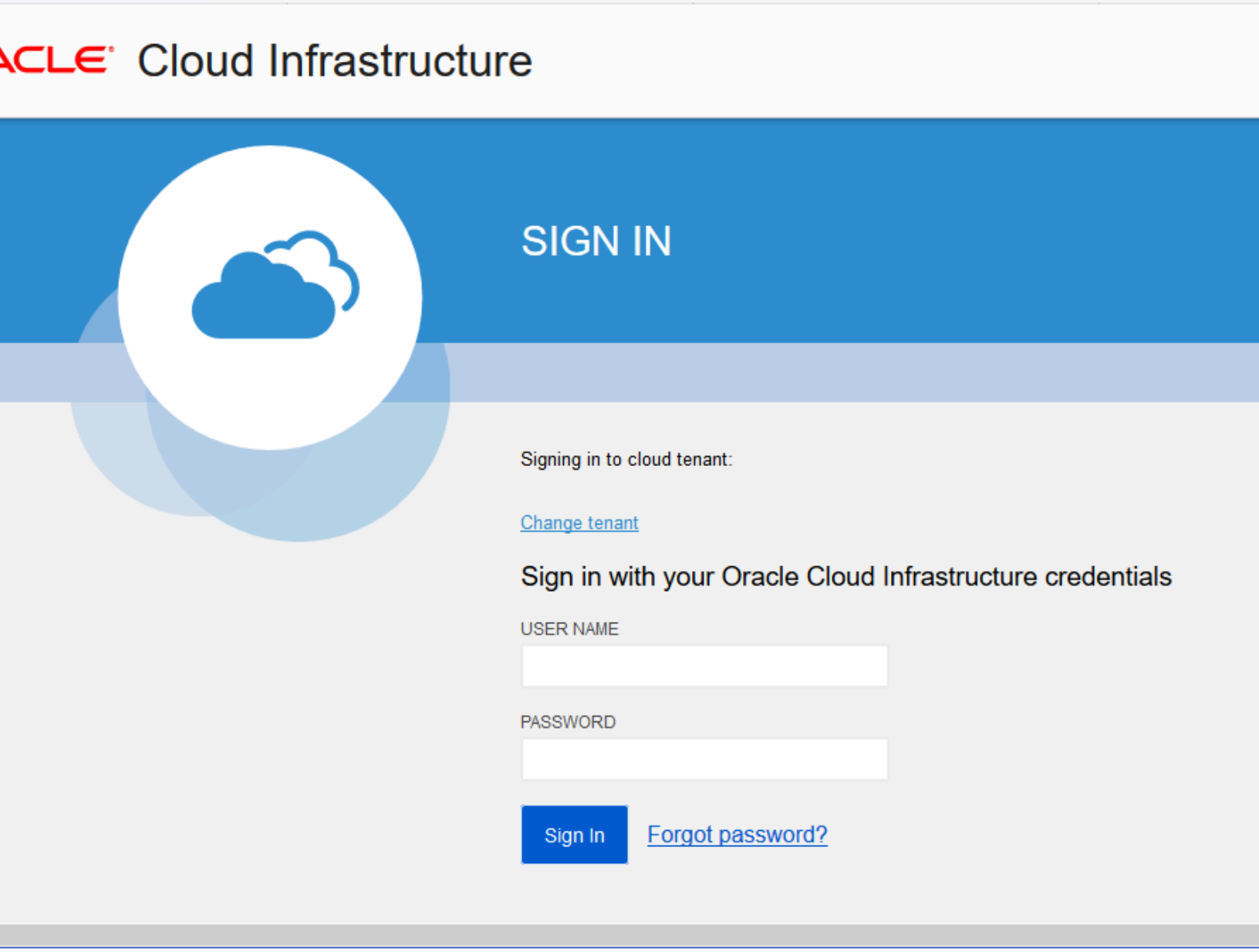
完成以下过程以在 OKE 平台上将 CN 系列防火墙部署为 Daemonset：

Oracle Linux 8.5 操作系统是在 *OKE* 上部署 *CN* 系列防火墙的唯一合格环境。

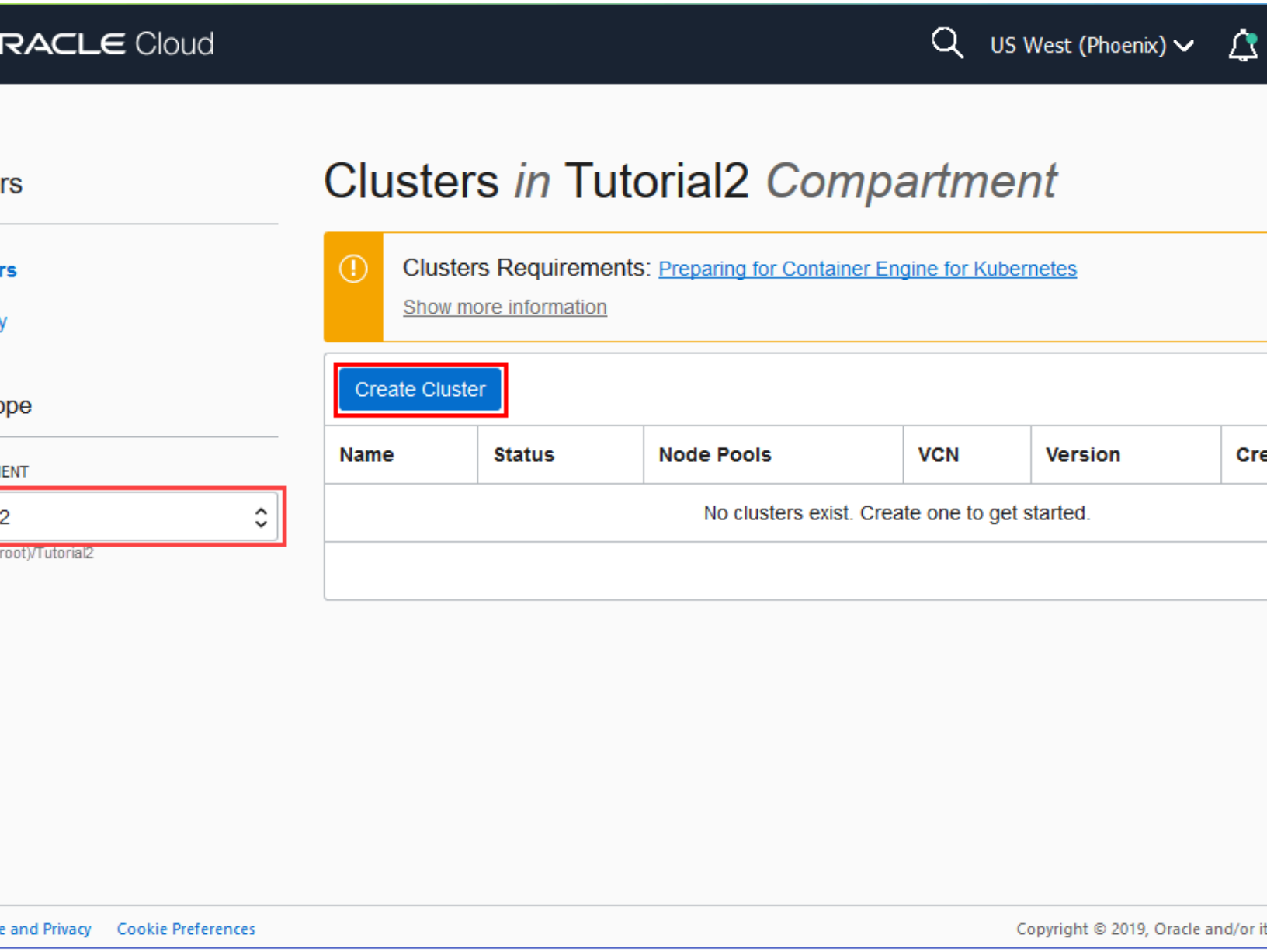
STEP 1 | 设置 Kubernetes 集群。

要在 OKE 中创建集群，请执行以下操作：

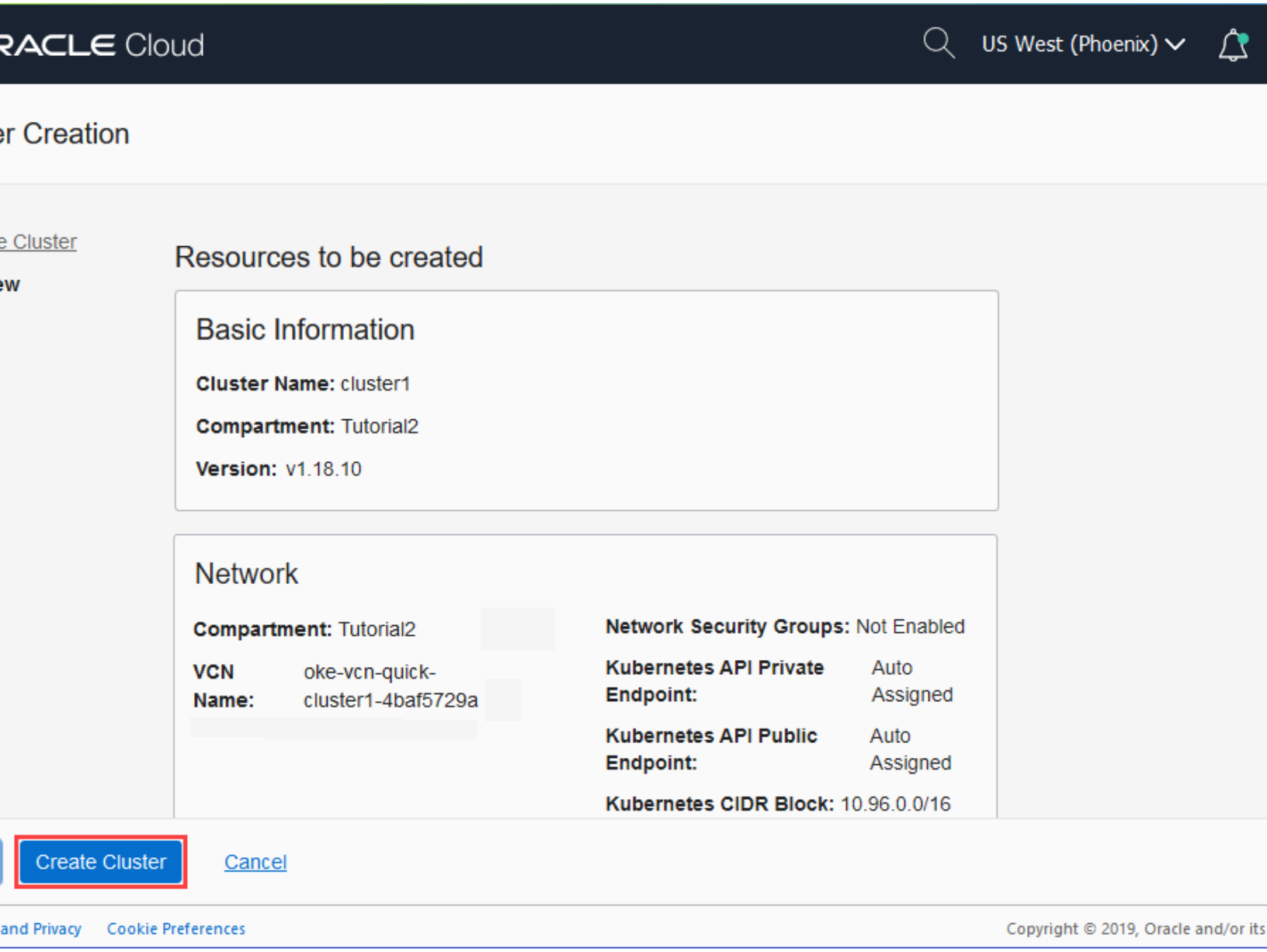
1. 登录到 Oracle 云基础设施。



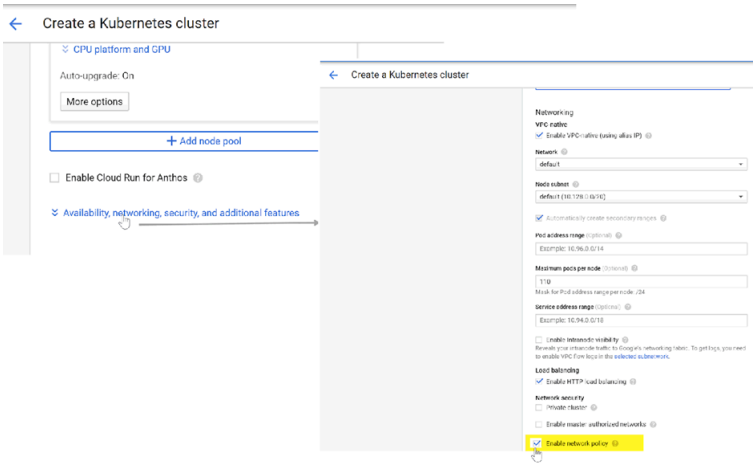
2. 单击导航菜单，转到在解决方案和平台下，然后单击开发人员服务。
3. 单击 **Kubernetes** 集群。
4. 选择一个区段并单击创建集群。



5. 在“创建集群”对话框中，单击快速创建，然后单击启动工作流程。
6. 在创建集群页面上，输入集群名称和其他详细信息。
7. 单击下一步以检查为新集群输入的详细信息。
8. 在“检查”页面上，单击创建集群。



如果集群在 *OKE* 上，请确保启用 *Kubernetes* 网络策略 *API* 以允许集群管理员指定允许相互通信的 *Pod*。同样，*CN-NGFW* 和 *CN-MGMT Pod* 进行通信也需要此 *API*。



验证集群是否有足够的资源。确保集群具有 CN 系列前提条件资源，以便支持防火墙。

kubectl get nodes

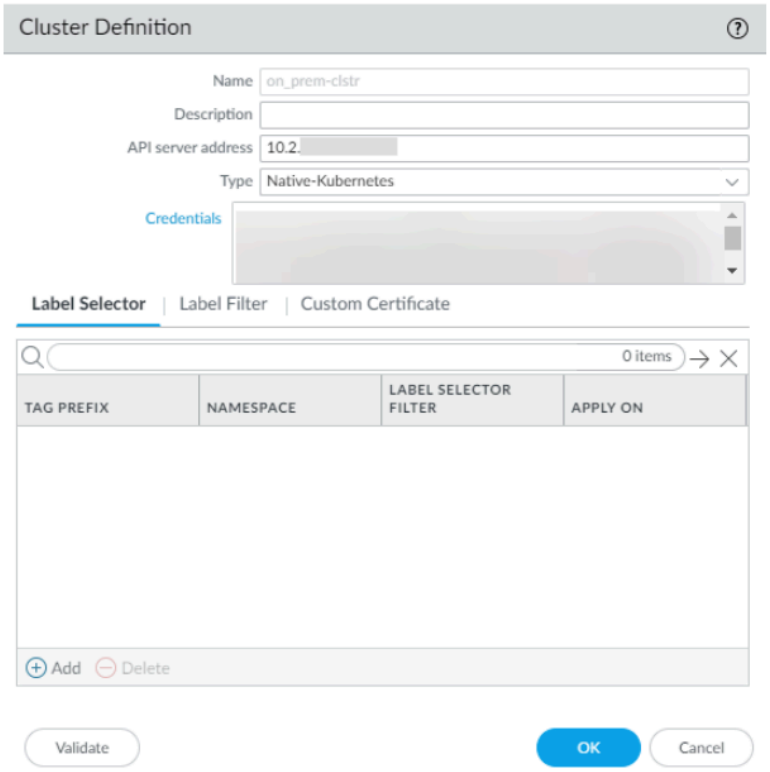
kubectl describe node <node-name>

查看命令输出中容量标题下的信息，以了解指定节点上可用的 CPU 和内存。

CPU、内存和磁盘存储分配将取决于您的需求。请参阅 CN 系列的性能和可扩展性。

确保有以下信息：

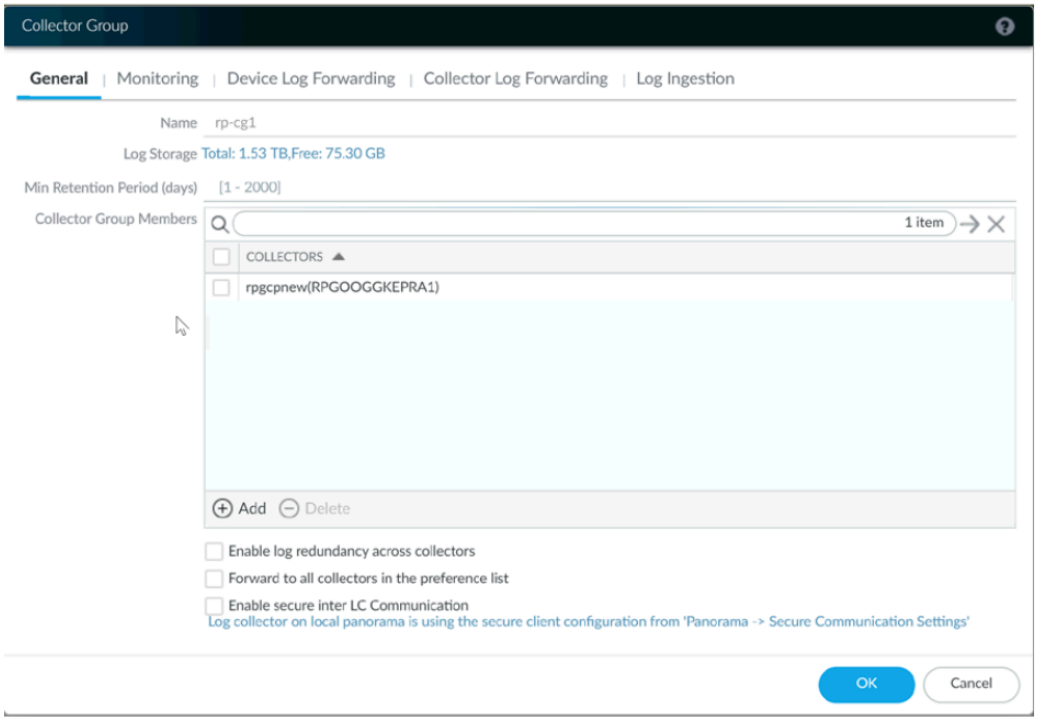
- 收集端点 IP 地址，用于在 Panorama 上设置 API 服务器。



Panorama 使用此 IP 地址连接到 Kubernetes 集群。

有关更多信息，请参阅 设置 Kubernetes 插件以监控集群。

- 从 Panorama 收集模板堆栈名称、设备组名称、Panorama IP 地址和可选的日志收集器组名称。



有关详细信息，请参阅创建父设备组和模板堆栈。

- 收集授权代码以及自动注册 PIN ID 和值。
- 将映像下载到的容器映像存储库的位置。

STEP 2 | （可选）如果您在 Kubernetes 插件中为 Panorama 配置了自定义证书，则必须通过执行以下命令来创建证书密钥。不要更改 ca.crt 中的文件名。pan-cn-mgmt-dynamic-pv.yaml 和 pan-cn-ngfw.yaml 中自定义证书的数量是可选的。

```
kubectl -n kube-system create secret generic custom-ca --from-file=ca.crt
```

STEP 3 | 编辑 YAML 文件以提供部署 CN 系列防火墙所需的详细信息。

您需要替换 YAML 文件中的映像路径，以包括私有 Google 容器注册表的路径并提供所需的参数。有关详细信息，请参阅 CN-Series 部署 yaml 文件中的可编辑参数。

STEP 4 | 部署 CNI DaemonSet。

CNI 容器作为 DaemonSet 部署（每个节点一个 Pod），并且在 CN-NGFW Pod 上为节点上部署的每个应用程序创建两个接口。使用 kubectl 命令运行 pan-cni YAML 文件时，该容器将成为每个节点上服务链的一部分。

在 *OKE* 上将 *CN* 系列防火墙部署为 *Daemonset* 时，您可以使用 pan-cn-k8s-daemonset 本机文件夹中的 *yaml* 文件。

1. CN 系列防火墙需要三个具有最低权限的服务帐户，这些帐户授权防火墙与 Kubernetes 集群资源进行通信。您要使用 CN-Series 创建用于集群身份验证的服务帐户，并验证您是否已使用 pan-cni-serviceaccount.yaml 创建服务帐户。
2. 使用 Kubectl 运行 pan-cni-configmap.yaml 文件。

   kubectl apply -f pan-cni-configmap.yaml
3. 使用 Kubectl 运行 pan-cni.yaml 文件。

   kubectl apply -f pan-cni.yaml
4. 验证是否已修改 pan-cni-configmap 和 pan-cni YAML 文件。
5. 运行以下命令并验证输出是否与以下示例相似。

STEP 5 | 部署 CN-MGMT StatefulSet。

默认情况下，部署管理平面作为提供容错功能的 StatefulSet。最多可以将 30 个防火墙 CN-NGFW Pod 连接到 CN-MGMT StatefulSet。

1. 验证是否已修改 pan-cn-mgmt-configmap 和 pan-cn-mgmt YAML 文件。

   示例 **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"
```

示例 **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. 使用 Kubectl 运行 yaml 文件。

   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

   如果您之前未完成为集群身份验证创建服务帐户，则必须运行 pan-mgmt-serviceaccount.yaml。

3. 验证 CN-MGMT Pod 是否启动。

   大约需要 5 至 6 分钟的时间。

   使用 **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 6 | 部署 CN-NGFW Pod。

默认情况下，防火墙数据平面 CN-NGFW Pod 部署为 DaemonSet。CN-NFGW Pod 的实例可保护节点上最多 30 个应用程序 Pod 的流量。

1. 验证是否已按 PAN-CN-NGFW-CONFIGMAP 和 PAN-CN-NGFW 中的详细说明修改 YAML 文件。

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

2. 使用 Kubectl apply 运行 pan-cn-ngfw-configmap.yaml 文件。

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

3. 使用 Kubectl apply 运行 pan-cn-ngfw.yaml 文件。

   kubectl apply -f pan-cn-ngfw.yaml

4. 验证所有 CN-NGFW Pod 是否正在运行（集群中每个节点一个）。

   以下是 4 个节点本地集群的输出示例。

   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 | 验证您是否可以在 Kubernetes 集群上看到 CN-MGMT、CN-NGFW 和 PAN-CNI。

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

STEP 8 | 为应用程序 yaml 文件或命名空间添加注释，以便将来自其新 Pod 的流量重定向到防火墙。

您需要添加以下注解，以将流量重定向到 CN-NGFW 来进行检查：

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

例如，对于“default”命名空间中的所有新 Pod：

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

在某些平台上，当在 *CNI* 插件链中未激活 *pan-cni* 时，可以启动应用程序 *Pod*。为避免此类情况，您必须按如下在应用程序 *Pod YAML* 中指定卷。

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

STEP 9 | 在集群中部署应用程序。

在 EKS 上部署 CN 系列防火墙

在何处可以使用？	需要提供什么？
• CN-Series部署	• CN-Series 10.1.x or above Container Images • 运行 PAN-OS 10.1.x 或更高版本的 Panorama • 使用 Helm 部署 CN 系列的 Helm 3.6 or above version client

在利用 CN 系列保护 Kubernetes 环境的安全中查看 CN 系列构建块和该工作流程的高级概述后，您可以开始在 AWS EKS 平台上部署 CN 系列防火墙，以保护同一集群中容器之间，以及容器和其他工作负载类型（例如虚拟机和裸机服务器）之间的流量。

您需要标准 *Kubernetes* 工具（例如 *kubectl* 或 *Helm*）部署和管理 *Kubernetes* 集群、应用程序和防火墙服务。

有关更多信息，请参阅使用 Helm 图表和模板部署 CN 系列防火墙。*Panorama* 并非旨在成为 *Kubernetes* 集群部署和管理的 *Orchestrator*。托管 *Kubernetes* 提供商已提供用于集群管理的模板。您还可以使用社区支持的模板部署具有 Helm 和 Terraform 的 *CN* 系列。

- 在 AWS EKS 上将 CN 系列防火墙部署为 Kubernetes 服务
- 将 CN 系列防火墙部署为 AWS EKS 上的 Daemonset
- 从 AWS Marketplace 部署 CN 系列

从*CN* 系列即 *DaemonSet* 部署迁移到 *CN* 系列即服务之前（反之亦然），您必须删除并重新应用 `plugin-serviceaccount.yaml`。有关更多信息，请参阅为集群身份验证创建服务帐户。

- 在 *EKS* 上将 *CN* 系列部署为 *DaemonSet* 时，不能存在 `pan-plugin-cluster-mode-secret`。
- 在 *EKS* 上将 *CN* 系列部署为 *Kubernetes* 服务时，必须存在 `pan-plugin-cluster-mode-secret`。

在 AWS EKS 上将 CN 系列防火墙部署为 Kubernetes 服务

在何处可以使用？	需要提供什么？
• CN-Series部署	• CN-Series 10.1.x or above Container Images • 运行 PAN-OS 10.1.x 或更高版本的 Panorama • 使用 Helm 部署 CN 系列的 Helm 3.6 or above version client

完成以下过程以将 CN 系列防火墙部署为 Kubernetes 服务。

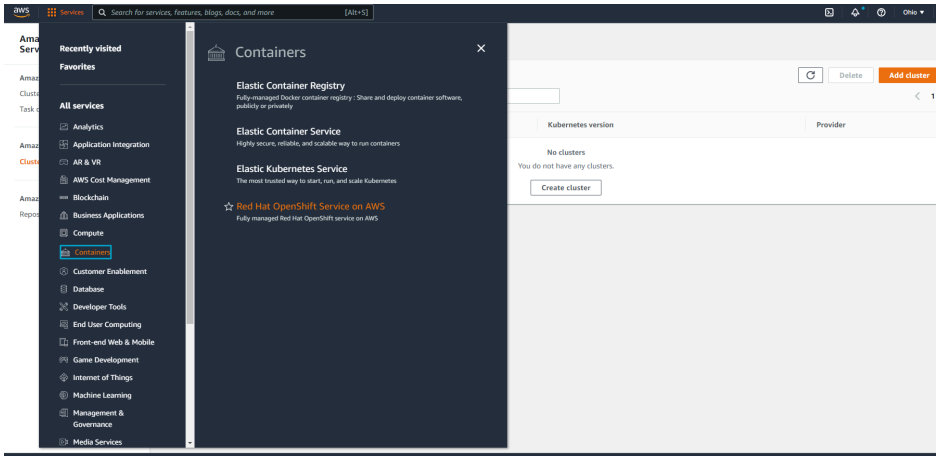
在开始之前，请确保 CN 系列 YAML 文件版本与 PAN OS 版本兼容。

- PAN-OS 10.1.2 或更高版本需要 YAML 2.0.2
- PAN-OS 10.1.0 和 10.1.1 需要 YAML 2.0.0 或 2.0.1

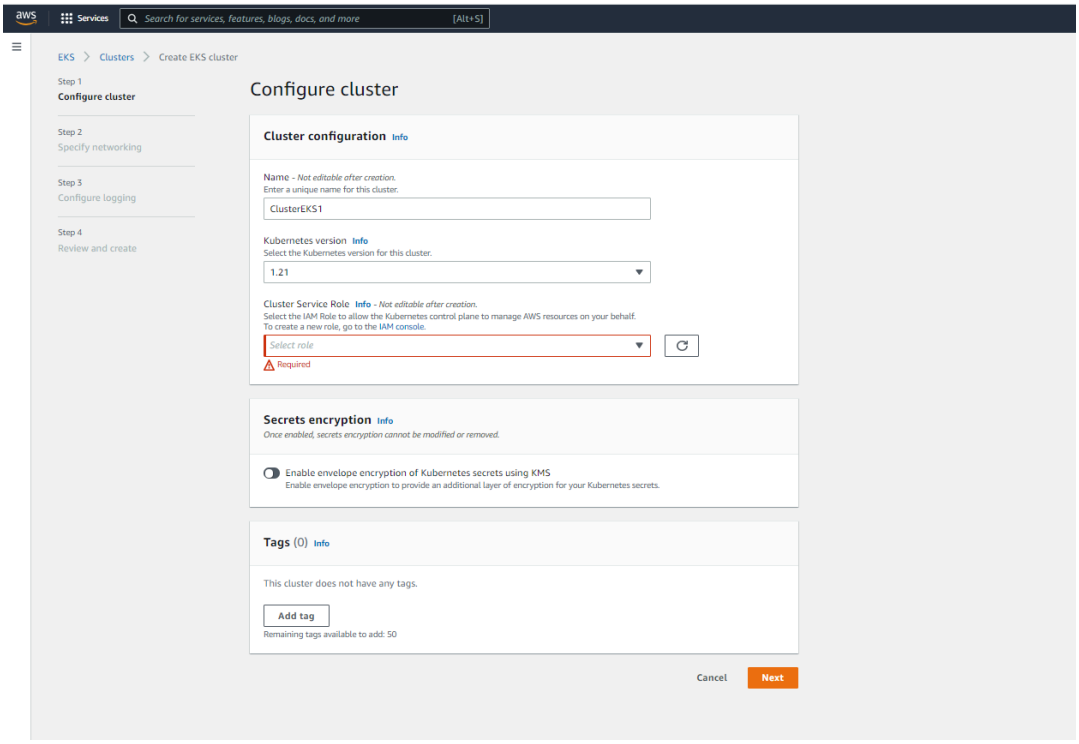
STEP 1 | 设置 Kubernetes 集群。

要在 AWS EKS 中创建集群，请执行以下操作：

1. 单击 **Services**（服务）导航菜单，转到 **Containers**（容器）->**Elastic Kubernetes Service**（**Elastic Kubernetes** 服务）。



2. 单击 **Create Cluster**（创建集群）。
3. 填写所需的详细信息，然后单击 **Create**（创建）。



1. 验证集群是否有足够的资源。确保该群集具有 CN 系列先决条件资源，以便支持防火墙。

   kubectl get nodes

   kubectl describe node <node-name>

   查看命令输出中容量标题下的信息，以了解指定节点上可用的 CPU 和内存。

CPU、内存和磁盘存储分配将取决于您的需求。参阅 CN 系列性能和可扩展性。

确保有以下信息：

- 收集端点 IP 地址，用于在 Panorama 上设置 API 服务器。Panorama 使用此 IP 地址连接到 Kubernetes 集群。
- 从 Panorama 收集模板堆栈名称、设备组名称、Panorama IP 地址和可选的日志收集器组名称。
- 收集授权代码以及自动注册 PIN ID 和值。
- 将映像下载到的容器映像存储库的位置。

STEP 2 | （可选）如果您在 Kubernetes 插件中为 Panorama 配置了自定义证书，则必须通过执行以下命令来创建证书密钥。不要更改 ca.crt 中的文件名。pan-cn-mgmt.yaml 和 pan-cn-ngfw.yaml 中自定义证书的数量是可选的。

```
kubectl -n kube-system create secret generic custom-ca --from-file=ca.crt
```

STEP 3 | 编辑 YAML 文件以提供部署 CN 系列防火墙所需的详细信息。

您需要替换 YAML 文件中的映像路径，以包括私有注册表的路径并提供所需的参数。有关详细信息，请参阅 CN 系列部署 yaml 文件中的可编辑参数。

STEP 4 | 更新存储类。要支持部署在 AWS Outpost 上的 CN 系列，您必须使用存储驱动程序 aws-ebs-csi-driver，以确保 Outpost 在动态持久性卷 (PV) 创建期间从 Outpost 拉取卷。

1. 应用以下 yaml。

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

2. 验证 ebs-sc 控制器是否正在运行。

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

3. 更新 pan-cn-storage-class.yaml 以匹配以下示例。

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

4. 将 **storageClassName: ebs-sc** 添加到 pan-cn-mgmt.yaml 的如下所示位置。

   ```
   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 | 如果您在 Kubernetes 环境中使用自动缩放，请执行以下操作：

1. 在 CN 系列即服务集群中部署适用于 Kubernetes 的 Amazon CloudWatch 指标适配器。您必须允许 CloudWatch 完全访问与 Kubernetes Pod 和集群关联的两个 IAM 角色。要将自定义指标发布到 CloudWatch，Worker 节点的角色必须具有 AWS 托管策略 **CloudWatch AgentServerPolicy**，HPA 才能检索这些指标。
2. 从 Palo Alto Networks GitHub 存储库下载 EKS 特定的 HPA yaml 文件。
3. 如果 CN-MGMT 部署在自定义命名空间中，请使用自定义命名空间更新 pan-cn-adapater.yaml。默认命名空间为 **kube-system**。

4. 修改 **pan-cn-hpa-dp.yaml** 和 **pan-cn-hpa-mp.yaml**。
   1. 输入副本的最小和最大数量。
   2. （可选）更改缩小和放大频率值以使其适合您的部署。如果不更改这些值，则会使用默认值。
   3. 为要用于缩放的每个指标复制以下部分。

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

   4. 更改要使用的指标的名称，然后将 **AveragValue** 设置为上表中所述的阈值。如果不更改这些值，则会使用默认值。
   5. 保存更改。

      有关更多信息，请参阅水平 Pod 自动缩放。
5. 部署 HPA yaml 文件。必须按照下面所述的顺序部署这些文件。
   1. 使用 Kubectl 运行 pan-cn-adapter.yaml

      ```
      kubectl apply -f pan-cn-adapter.yaml
      ```
   2. 使用 Kubectl 运行 pan-cn-externalmetrics.yaml

      ```
      kubectl apply -f pan-cn-externalmetrics.yaml
      ```
   3. 使用 Kubectl 运行 pan-cn-hpa-dp.yaml

      ```
      kubectl apply -f pan-cn-hpa-dp.yaml
      ```
   4. 使用 Kubectl 运行 pan-cn-hpa-mp.yaml

      ```
      kubectl apply -f pan-cn-hpa-mp.yaml
      ```
6. 验证部署。

   使用 kubectl 验证自定义指标命名空间中的自定义指标适配器 Pod 是否存在。

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

   使用 kubectl 检查 HPA 资源。

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

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

STEP 6 | 部署 CN-NGFW 服务。

1. 使用 pan-cni-serviceaccount.yaml 文件验证您是否已创建服务帐户。

   参阅创建用于集群身份验证的服务帐户。

2. 使用 Kubectl 运行 pan-cni-configmap.yaml 文件。

   kubectl apply -f pan-cni-configmap.yaml

3. 使用 kubectl 运行 pan-cn-ngfw-svc.yaml。

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

   该 *yaml* 必须在 *pan-cni.yaml* 之前部署。

4. 使用 Kubectl 运行 pan-cni.yaml 文件。

   kubectl apply -f pan-cni.yaml

5. 验证是否已修改 pan-cni-configmap 和 pan-cni YAML 文件。
6. 运行以下命令并验证输出是否与以下示例相似。

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

```
        @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 7 | 部署 CN-MGMT StatefulSet。

默认情况下，部署管理平面作为提供容错功能的 StatefulSet。最多可以将 30 个防火墙 CN-NGFW Pod 连接到 CN-MGMT StatefulSet。

1. （仅对于静态配置的 PV 为必需）为 CN-MGMT StatefulSet 部署持久卷 (PV)。

   1. 创建目录以匹配 pan-cn-pv-local.yaml 文件中定义的本地卷名称。

      在至少 2 个工作节点上需要六 (6) 个目录。登录将在其中部署 CN-MGMT StatefulSet 的每个工作节点，以创建目录。例如，要创建名为 /mnt/pan-local1 到 /mnt/pan-local6 的目录，请使用以下命令：

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

   2. 修改 pan-cn-pv-local.yaml。

      在 `nodeaffinity` 下匹配主机名，并验证是否已修改在 `spec.local.path` 中创建的上述目录，然后部署文件以创建新的存储类 pan-local-storage 和本地 PV。

2. 验证是否已修改 pan-cn-mgmt-configmap 和 pan-cn-mgmt YAML 文件。

   EKS 中的 pan-cn-mgmt-configmap 示例。

```
 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
```

pan-cn-mgmt.yaml 文件示例

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

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

3. 使用 Kubectl 运行 yaml 文件。

   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

   您必须运行 pan-mgmt-serviceaccount.yaml，前提是您之前未完成为集群身份验证创建服务帐户。

4. 验证 CN-MGMT Pod 是否启动。

   大约需要 5 至 6 分钟的时间。

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

STEP 8 | 部署 CN-NGFW Pod。

1. 验证是否已按 PAN-CN-NGFW-CONFIGMAP 和 PAN-CN-NGFW 中的详细说明修改 YAML 文件。

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

2. 使用 Kubectl apply 运行 pan-cn-ngfw-configmap.yaml 文件。

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

3. 使用 Kubectl apply 运行 pan-cn-ngfw.yaml 文件。

   kubectl apply -f pan-cn-ngfw.yaml

4. 确认已部署 CN-NGFW Pod。

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

STEP 9 | 在 CN 系列上启用水平 Pod 自动缩放。

STEP 10 | 验证您是否可以在 Kubernetes 集群上看到 CN-MGMT、CN-NGFW 和 PAN-CNI。

kubectl -n kube-system get pods

STEP 11 | 为应用程序 yaml 文件或命名空间添加注释，以便将来自其新 Pod 的流量重定向到防火墙。

您需要添加以下注解，以将流量重定向到 CN-NGFW 来进行检查：

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

例如，对于“default”命名空间中的所有新 Pod：

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

在某些平台上，当在 *CNI* 插件链中未激活 *pan-cni* 时，可以启动应用程序 *Pod*。为避免此类情况，您必须按如下在应用程序 *Pod YAML* 中指定卷。

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

STEP 12 | 在集群中部署应用程序。

将 CN 系列防火墙部署为 AWS EKS 上的 Daemonset

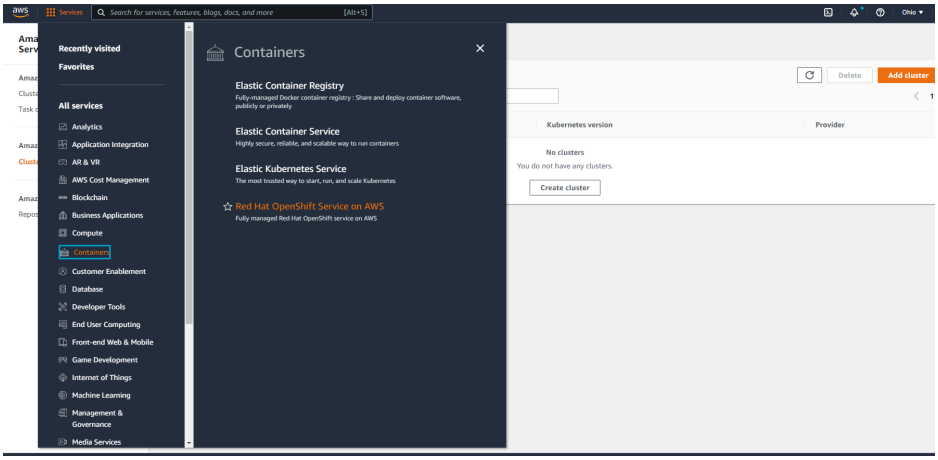
在何处可以使用？	我需要什么？
• CN-Series 部署	• CN-Series 10.1.x or above Container Images • 运行 PAN-OS 10.1.x 或更高版本的 Panorama • 适用于使用 Helm 部署 CN 系列的 Helm 3.6 or above version client

完成以下步骤可将 CN 系列部署为 AWS EKS 上的 Dameonset：

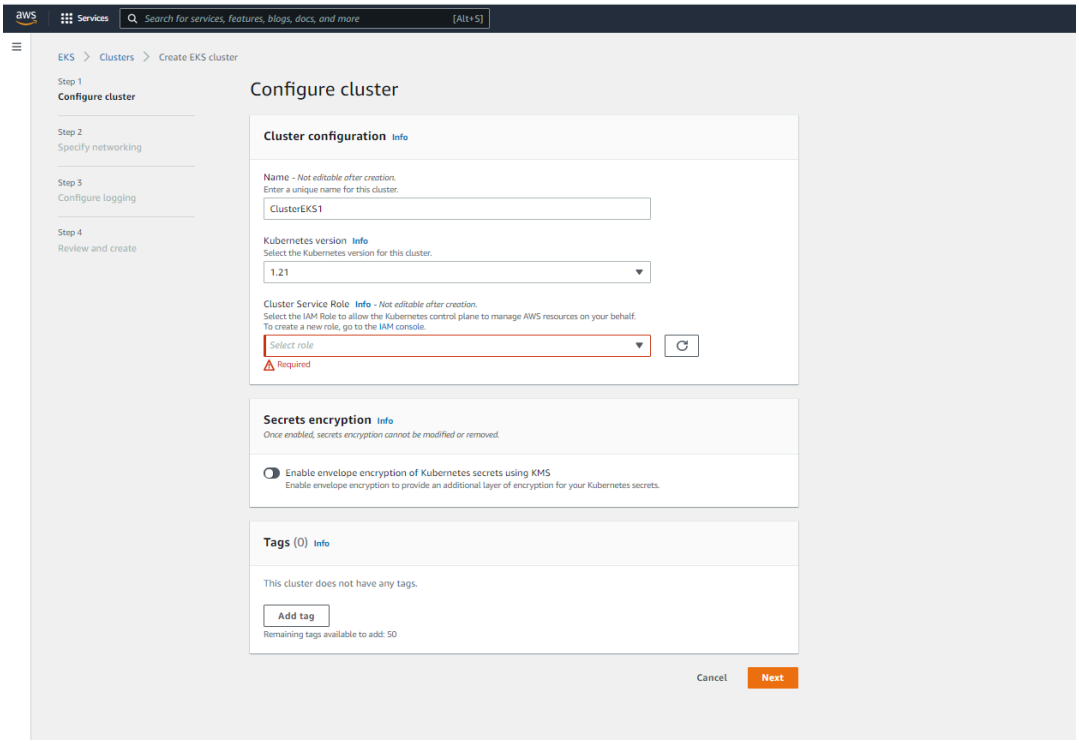
STEP 1 | 设置 Kubernetes 集群。

要在 AWS EKS 中创建集群，请执行以下操作：

1. 单击 **Services**（服务）导航菜单，转到 **Containers**（容器）->**Elastic Kubernetes Service**（**Elastic Kubernetes** 服务）。



2. 单击 **Create Cluster**（创建集群）。
3. 填写所需的详细信息，然后单击 **Create**（创建）。



验证集群是否有足够的资源。确保群集具有支持防火墙的 CN 系列先决条件资源。

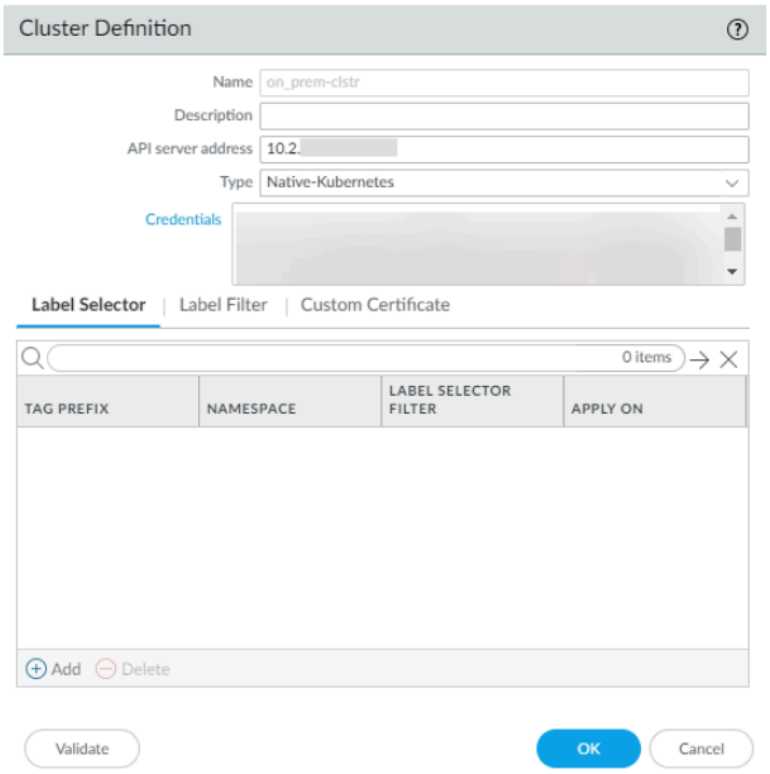
```
kubectl get nodes
kubectl describe node <node-name>
```

查看命令输出中容量标题下的信息，以了解指定节点上可用的 CPU 和内存。

CPU、内存和磁盘存储分配将取决于您的需求。参阅 CN 系列性能和可扩展性。

确保有以下信息：

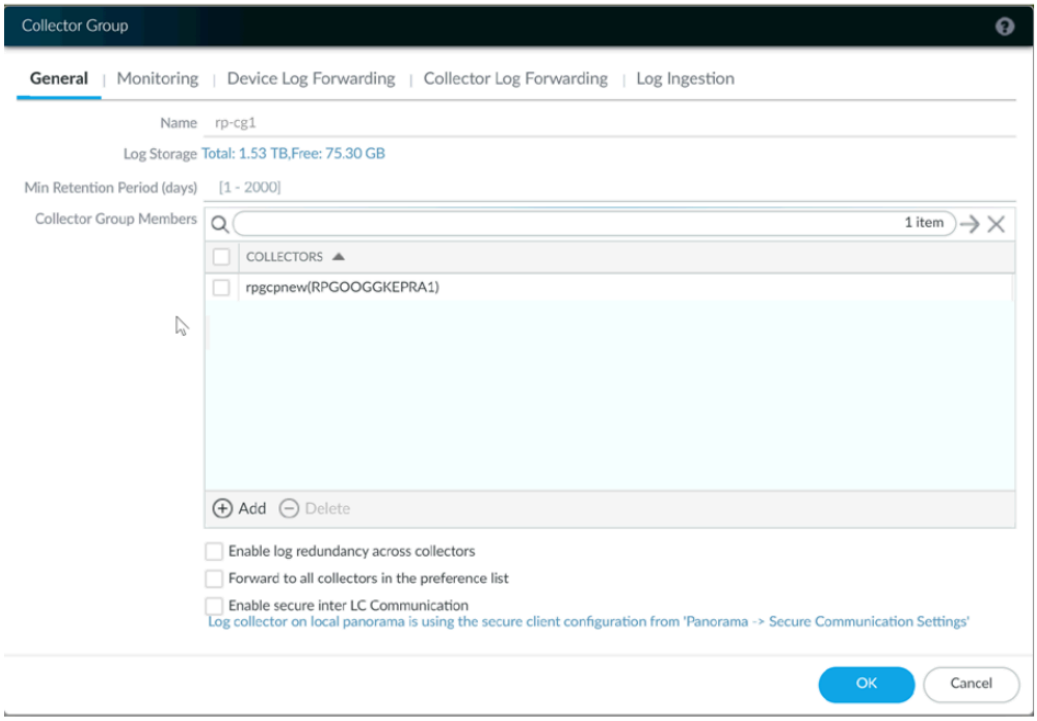
- 收集端点 IP 地址，用于在 Panorama 上设置 API 服务器。



  Panorama 使用此 IP 地址连接到 Kubernetes 集群。

  有关更多信息，请参阅 设置 Kubernetes 插件以监控集群。

- 从 Panorama 收集模板堆栈名称、设备组名称、Panorama IP 地址和可选的日志收集器组名称。



有关详细信息，请参阅创建父设备组和模板堆栈。

- 收集授权代码以及自动注册 PIN ID 和值。
- 将映像下载到的容器映像存储库的位置。

STEP 2 | （可选）如果您在 Kubernetes 插件中为 Panorama 配置了自定义证书，则必须通过执行以下命令来创建证书密钥。不要更改 ca.crt 中的文件名。pan-cn-mgmt.yaml 和 pan-cn-ngfw.yaml 中自定义证书的数量是可选的。

```
kubectl -n kube-system create secret generic custom-ca --from-file=ca.crt
```

STEP 3 | 编辑 YAML 文件以提供部署 CN 系列防火墙所需的详细信息。

您需要替换 YAML 文件中的映像路径，以包括私有 Google 容器注册表的路径并提供所需的参数。有关详细信息，请参阅 CN 系列部署 YAML 文件中的可编辑参数。

STEP 4 | 部署 CNI DaemonSet。

CNI 容器作为 DaemonSet 部署（每个节点一个 Pod），并且在 CN-NGFW Pod 上为节点上部署的每个应用程序创建两个接口。使用 kubectl 命令运行 pan-cni YAML 文件时，该容器将成为每个节点上服务链的一部分。

1. CN 系列防火墙需要三个具有最低权限的服务帐户，这些帐户授权防火墙与 Kubernetes 集群资源进行通信。您要群集身份验证创建服务帐户，并验证是否已使用 pan-cni-serviceaccount.yaml 创建服务帐户。
2. 使用 Kubectl 运行 pan-cni-configmap.yaml 文件。

   kubectl apply -f pan-cni-configmap.yaml
3. 使用 Kubectl 运行 pan-cni.yaml 文件。

   kubectl apply -f pan-cni.yaml
4. 验证是否已修改 pan-cni-configmap 和 pan-cni YAML 文件。
5. 运行以下命令并验证输出是否与以下示例相似。

STEP 5 | 更新存储类。要支持部署在 AWS Outpost 上的 CN 系列，您必须使用存储驱动程序 aws-ebs-csi-driver，以确保 Outpost 在动态持久性卷 (PV) 创建期间从 Outpost 拉取卷。

1. 应用以下 yaml。

   kubectl apply -k "github.com/kubernetes-sigs/aws-ebs-csi-driver/deploy/kubernetes/overlays/stable/?ref=release-0.10"
2. 验证 ebs-sc 控制器是否正在运行。

   kubectl -n kube-system get pods
3. 更新 pan-cn-storage-class.yaml 以匹配以下示例。

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

4. 将 **storageClassName: ebs-sc** 添加到 pan-cn-mgmt.yaml 的如下所示位置。

```
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 | 部署 CN-MGMT StatefulSet。

默认情况下，部署管理平面作为提供容错功能的 StatefulSet。最多可以将 30 个防火墙 CN-NGFW Pod 连接到 CN-MGMT StatefulSet。

1. （仅对于静态配置的 PV 为必需）为 CN-MGMT StatefulSet 部署持久卷 (PV)。

   1. 创建目录以匹配 pan-cn-pv-local.yaml 文件中定义的本地卷名称。

      在至少 2 个工作节点上需要六 (6) 个目录。登录将在其中部署 CN-MGMT StatefulSet 的每个工作节点，以创建目录。例如，要创建名为 /mnt/pan-local1 到 /mnt/pan-local6 的目录，请使用以下命令：

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

   2. 修改 pan-cn-pv-local.yaml。

      在 `nodeaffinity` 下匹配主机名，并验证是否已修改在 `spec.local.path` 中创建的上述目录，然后部署文件以创建新的存储类 pan-local-storage 和本地 PV。

2. 验证是否已修改 pan-cn-mgmt-configmap 和 pan-cn-mgmt YAML 文件。

   EKS 中的 pan-cn-mgmt-configmap 示例。

```
 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
```

pan-cn-mgmt.yaml 文件示例

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

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

3. 使用 Kubectl 运行 yaml 文件。

   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

   如果您之前未完成使用 CN 系列防火墙为集群身份验证创建服务帐户，则必须运行 pan-mgmt-serviceaccount.yaml。

4. 验证 CN-MGMT Pod 是否启动。

   大约需要 5 至 6 分钟的时间。

   使用 **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 | 部署 CN-NGFW Pod。

默认情况下，将防火墙数据平面 CN-NGFW Pod 作为 DaemonSet 部署。CN-NFGW Pod 的实例可保护节点上最多 30 个应用程序 Pod 的流量。

1. 验证是否已按 PAN-CN-NGFW-CONFIGMAP 和 PAN-CN-NGFW 中的详细说明修改 YAML 文件。

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

2. 使用 Kubectl apply 运行 pan-cn-ngfw-configmap.yaml 文件。

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

3. 使用 Kubectl apply 运行 pan-cn-ngfw.yaml 文件。

   kubectl apply -f pan-cn-ngfw.yaml

4. 验证所有 CN-NGFW Pod 是否正在运行（集群中每个节点一个）。

   以下是 4 个节点本地集群的输出示例。

   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 | 验证您是否可以在 Kubernetes 集群上看到 CN-MGMT、CN-NGFW 和 PAN-CNI。

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

STEP 9 | 为应用程序 yaml 文件或命名空间添加注释，以便将来自其新 Pod 的流量重定向到防火墙。

您需要添加以下注解，以将流量重定向到 CN-NGFW 来进行检查：

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

例如，对于“default”命名空间中的所有新 Pod：

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

在某些平台上，当在 *CNI* 插件链中未激活 *pan-cni* 时，可以启动应用程序 *Pod*。为避免此类情况，您必须按如下在应用程序 *Pod YAML* 中指定卷。

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

STEP 10 | 在集群中部署应用程序。

从 AWS Marketplace 部署 CN 系列

在何处可以使用？	需要提供什么？
• CN-Series部署	• CN-Series 10.1.x or above Container Images • 运行 PAN-OS 10.1.x 或更高版本的 Panorama • 使用 Helm 部署 CN 系列的 Helm 3.6 or above version client

您可以通过 AWS Marketplace 授予许可，将 CN 系列防火墙作为 Kubernetes 服务部署在 AWS EKS 上。CN 系列的许可期限可为一月、一年、两年或三年，并可部署在 EKS 1.19 及更高版本或 Redhat Openshift 4.7 及更高版本上。

此产品处于预览状态。

要使用此许可证，您需要更新附加到 Kubernetes 工作进程节点的 IAM 策略。

如果使用通过 *AWS Marketplace* 购买的 *PAYG* 许可证进行 *CN*系列部署，请不要向 *Kubernetes* 的 *Panorama* 插件添加授权代码。

STEP 1 | 完成以下先决条件。

1. 创建 EKS 或 Redhat OpenShift 集群。
2. 部署 Panorama 并安装 Kubernetes 插件。

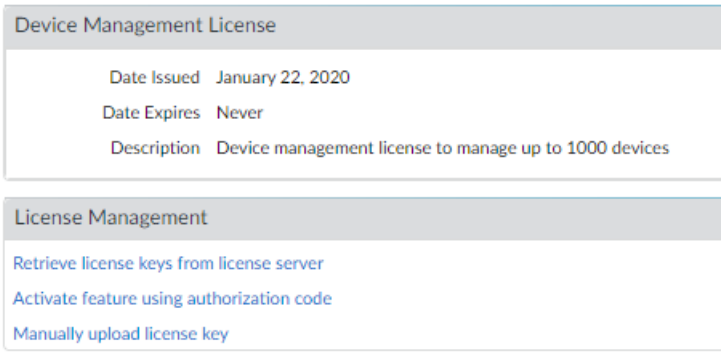
   如果已在 *AWS* 上部署了许可的 *Panorama* 实例，请跳过这些步骤。

   1. 在 Amazon EC2 实例上安装 Panorama。
   2. 安装适用于 CN 系列的 Kubernetes 插件。
   3. 安装 Panorama 后，请通过 **cn-series-aws-marketplace@paloaltonetworks.com** 向 CN 系列团队发送电子邮件，以请求 Panorama 许可证。请包括您的全名、公司电子邮件地址、公司名称、采购订单编号、AWS 帐户名称和 AWS 帐户 ID。

STEP 2 | 将序列号和许可证应用到 Panorama。

1. 登录到 Panorama Web 界面。
2. 选择 **Panorama** > **Setup**（设置） > **Management**（管理）并单击编辑 图标。
3. 输入 Panorama **Serial Number**（序列号）（订单执行电子邮件包含），然后单击 **OK**（确定）。
4. 选择 **Panorama** > **Licenses**（许可证）。
5. 单击 **Activate feature using authorization code**（使用授权代码激活功能）
6. 输入防火墙管理许可证授权码，然后单击 **OK**（确定）以激活许可证。
7. 确认防火墙管理许可证已激活。

   此时会出现“设备管理许可证”部分，其中显示内容包括许可证发放日期、许可证到期日期、以及对防火墙管理许可证的描述。



STEP 3 | 更新 IAM 策略并将该策略附加到 Kubernetes 工作进程节点。

1. 登录 AWS 管理控制台并打开 IAM 控制台。
1. 选择 **Policies**（策略）。
2. 从策略列表中，选择 **AWSLicenseManagerConsumptionPolicy** 和 **AWSMarketplaceMeteringRegisterUsage**。
3. 选取 **Actions**（操作），然后选择 **Attach**（附加）。
4. 选择要将策略附加到的工作进程节点身份。选择身份后，单击 **Attach policy**（附加策略）。

STEP 4 | 部署 Helm 图表之前，请下载 **`plugin-serviceaccount.yaml`** 并应用 yaml。

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

STEP 5 | 访问 AWS Marketplace 并找到适用于 **AWS Marketplace** 的 **CN** 系列列表。

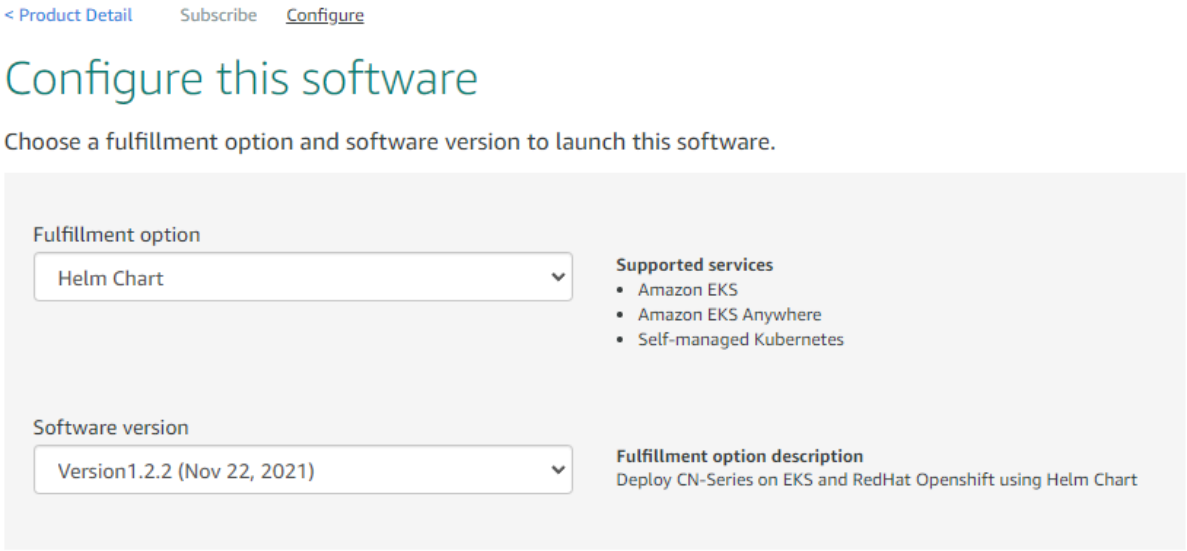
STEP 6 | 单击 **Continue to Subscribe**（继续订阅）。

STEP 7 | 输入要购买的许可证数量。每个许可证授权相当于 CN 系列部署使用的一个 vCPU。

有关满足部署需求的 vCPU 数量的指导，请参阅 CN 系列系统要求和 CN 系列性能和扩展。

STEP 8 | 单击 **Continue to Configuration**（继续配置）。这会将许可证添加到您的 AWS 帐户。

1. 选择 **Helm Chart**（**Helm** 图表）作为 **Fulfillment**（执行）选项。
2. 选择最新版本作为 **Software Version**（软件版本）。



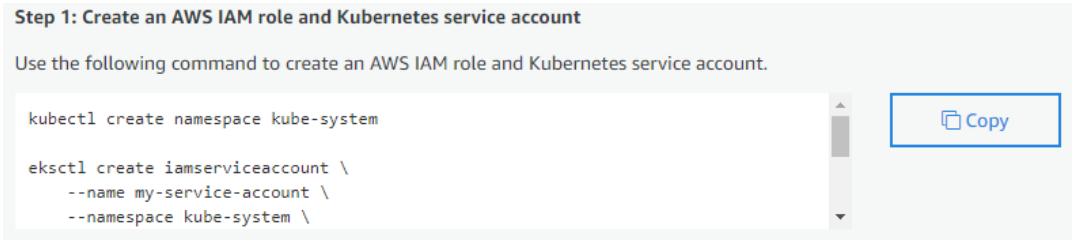
STEP 9 | 单击 **Continue to Launch**（继续发布）。

1. 选择 **Launch Target**（发布目标）— **Amazon-managed Kubernetes**（**Amazon** 托管 **Kubernetes**）或 **Self-managed Kubernetes**（自助托管 **Kubernetes**）。Redhat OpenShift 上部署了自助托管模式。
2. 按照 AWS Marketplace 列表中显示的 **Launch Instruction**（启动说明）进行操作。根据启动目标的不同，说明也会有所不同。

   - **Amazon** 托管 **Kubernetes**

     1. 从 **Launch Instructions**（发布说明）的 **Step 1**（步骤 **1**）中复制命令。
     2. 更新复制的命令以添加集群名称。

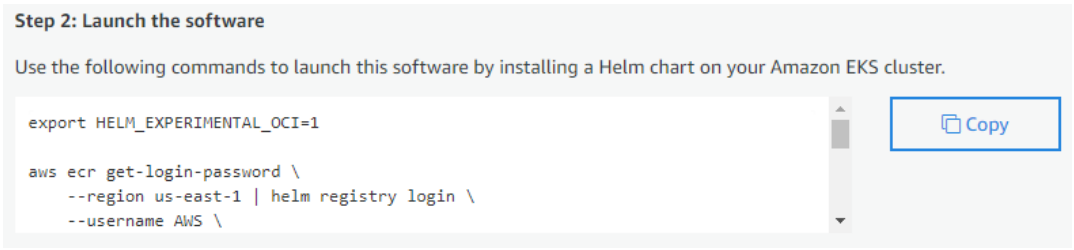
        `--cluster <ENTER_YOUR_CLUSTER_NAME_HERE>`
     3. 在 EKS 集群上执行复制的命令。

     4. 从 **Launch Instructions**（发布说明）的 **Step 2**（步骤 **2**）中复制 Helm 图表命令。
     5. 更新 Helm 安装信息以包含 Panorama IP、Panorama 身份验证密钥、设备组名称、模板堆栈名称和收集组名称。将 **`cluster.deployTo`** 设置为 **`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 \
        ```

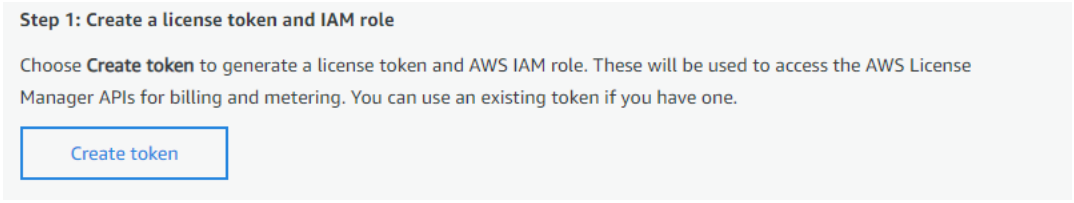
```
--set panorama.cgName=Panorama-CG \ --set
imagePullSecrets=awsmp-image-pull-secret
```



6. 更新上面列出的值后，在 EKS 集群上执行 helm install 命令。

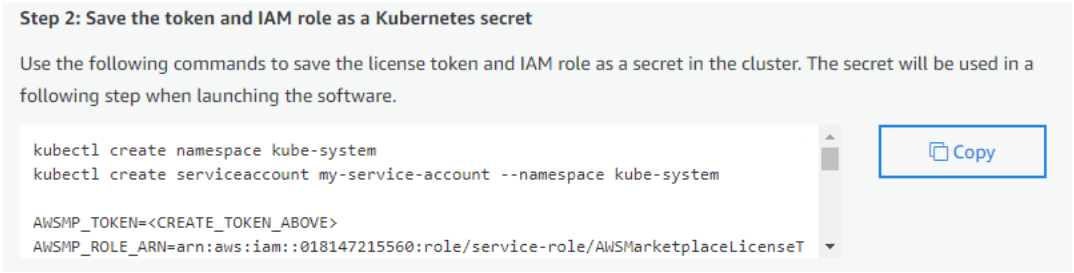
- 自助托管 **Kubernetes**

1. 完成发布说明中的步骤 1，以创建许可证令牌和 IAM 角色。



2. 从 **Launch Instructions**（发布说明）的 **Step 2**（步骤 **2**）中复制命令。
3. 更新复制的命令以添加令牌值。

   AWSMP_TOKEN=<CREATE_TOKEN_ABOVE>
4. 在 OpenShift 集群上执行复制的命令。



5. 从 **Launch Instructions**（发布说明）的 **Step 3**（步骤 **3**）中复制 Helm 图表命令。
6. 更新 Helm 安装信息以包含 Panorama IP、Panorama 身份验证密钥、设备组名称、模板堆栈名称和收集组名称。将 **`cluster.deployTo`** 设置为 **`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
```

Step 3: Launch the software

Use the following commands to launch the software by installing a Helm chart from Amazon Elastic Container Registry (ECR).

```
export HELM_EXPERIMENTAL_OCI=1

aws ecr get-login-password \
    --region us-east-1 | helm registry login \
    --username AWS \
```

Copy

7. 更新上面列出的值后，在 OpenShift 集群上执行 helm install 命令。

STEP 10 | 验证是否已将许可证成功添加到您的帐户。

1. 导航至 AWS 许可证管理器。
2. 选择 **Granted Licenses**（授予的许可证）并找到适用于 AWS Marketplace 的 CN 系列列表。
3. 在 **Entitlements**（权利）下，您可以看到许可证总数和已使用的许可证数量。

Entitlements

An entitlement is a right to use, access, or consume an application or resource.

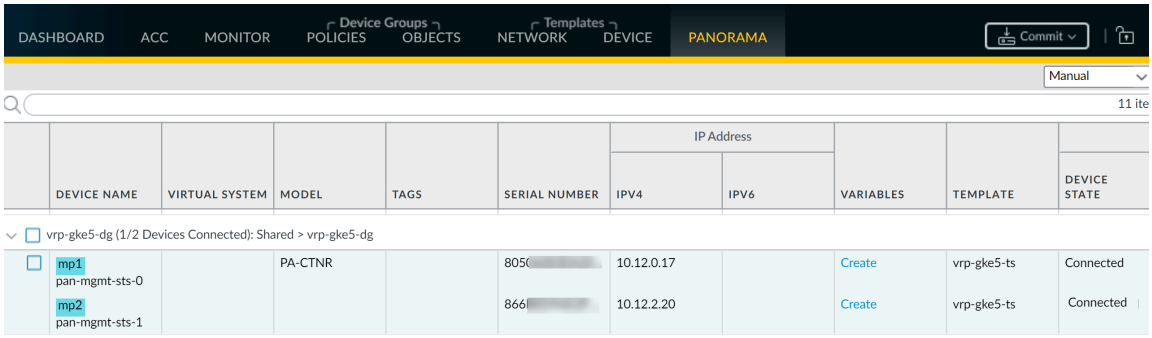
Search

1

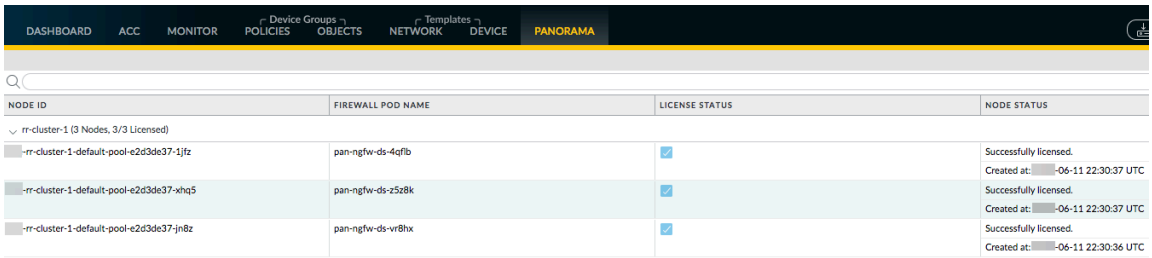
Name	Value	Max count	Usage	Units	Overages	Allow check in
vCPU	-	1000	5	Count	Not Allowed	Allowed
AWS::Marketplace::Usage	Enabled	-	-	None	-	Not Allowed

STEP 11 | 验证 CN 系列防火墙是否显示在 Panorama 中。

1. 登录到 Panorama。
2. 要查看 CN-MGMT Pod，请选择 **Panorama** > **Managed Devices**（托管设备） > **Summary**（摘要）。



3. 要验证 CN-NGFW Pod 是否已获得授权，请选择 **Panorama** > **Plugins**（插件） > **Kubernetes** > **License Usage**（许可证使用），并验证已为每个 Pod 分配一个许可证令牌。



将 CN 系列防火墙部署为阿里云上的 Kubernetes 服务 (ACK)

在何处可以使用？	需要提供什么？
• CN-Series部署	• CN-Series 10.1.x or above Container Images • 运行 PAN-OS 10.1.x 或 PAN-OS 10.2.x 版本的 Panorama

在使用 CN 系列保护 Kubernetes 工作负载的安全中查看 CN 系列核心构建块和该工作流程的高级概述后，您可以开始在 AliCloud ACK 平台上部署 CN 系列防火墙，以保护同一集群中容器之间，以及容器和其他工作负载类型（例如虚拟机和裸机服务器）之间的流量。

您必须确保应用 `plugin-serviceaccount.yaml` 文件。有关详细信息，请参阅创建用于集群身份验证的服务账户。

- 当您将 *CN* 系列防火墙部署为 *ACK* 上的 *Kubernetes* 服务时，必须存在 *`pan-plugin-cluster-mode-secret`*。

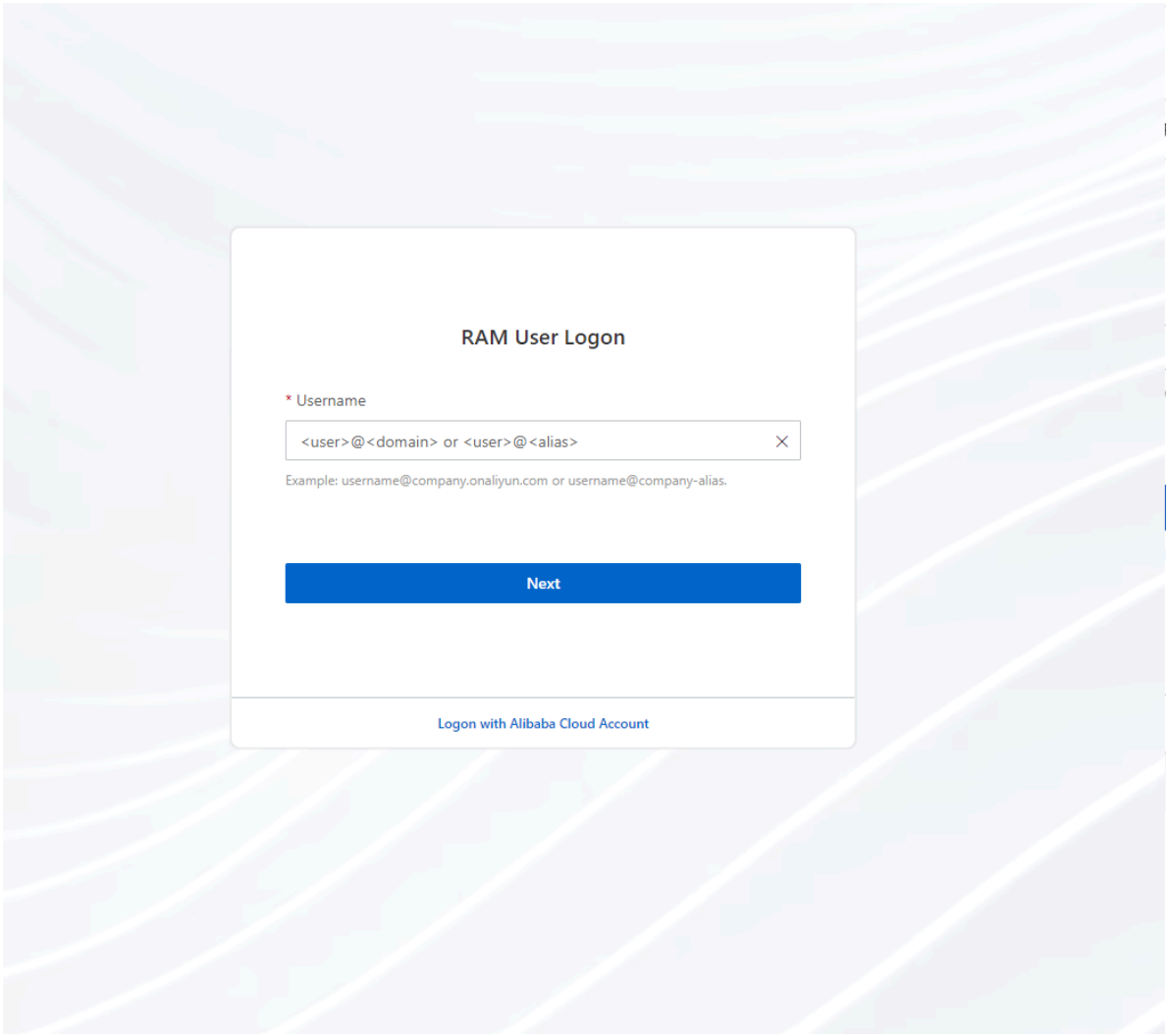
在开始之前，请确保 CN 系列 YAML 文件版本与 PAN OS 版本兼容。有关更多信息，请参阅 CN 系列 YAML。

完成以下过程以在 ACK 平台上将 CN 系列防火墙部署为 Kubernetes 服务。

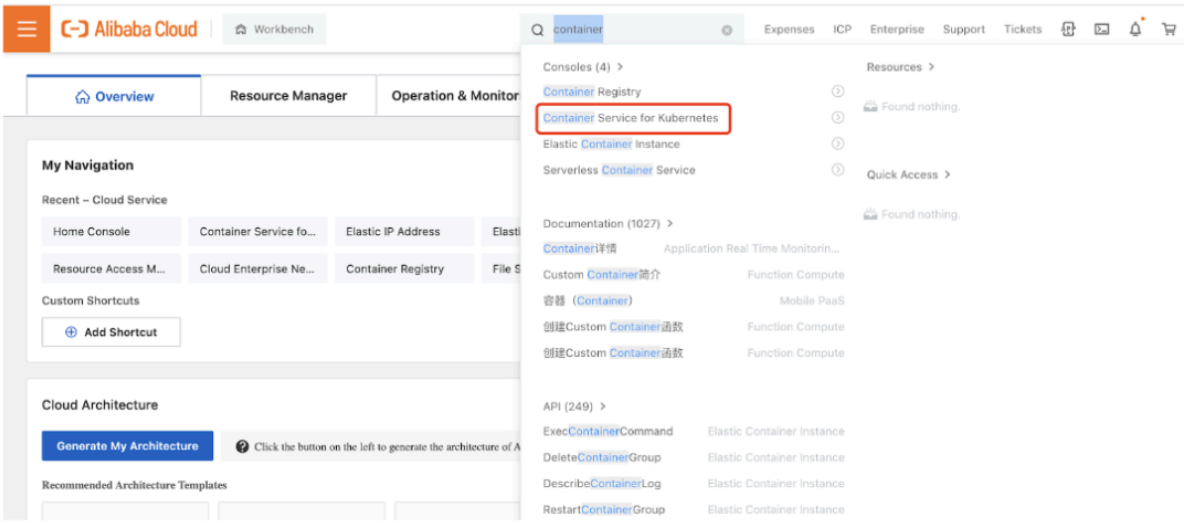
STEP 1 | 设置 Kubernetes 集群。

要在 ACK 中创建集群，请执行以下操作：

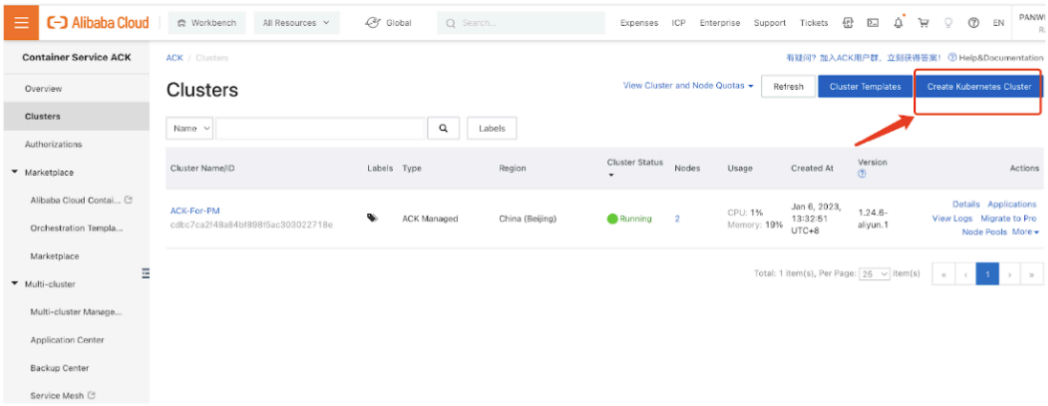
1. 使用您的 RAM 登录凭据登录 RAM 用户登录门户。



2. 在顶部导航栏中，选择需要创建集群的区域，并根据业务需求选择资源组。
   - 创建集群后无法更改集群的区域。
   - 在默认情况下，您帐户内的所有资源组都会显示。
3. 在搜索栏菜单上搜索 **Kubernetes** 的容器服务。



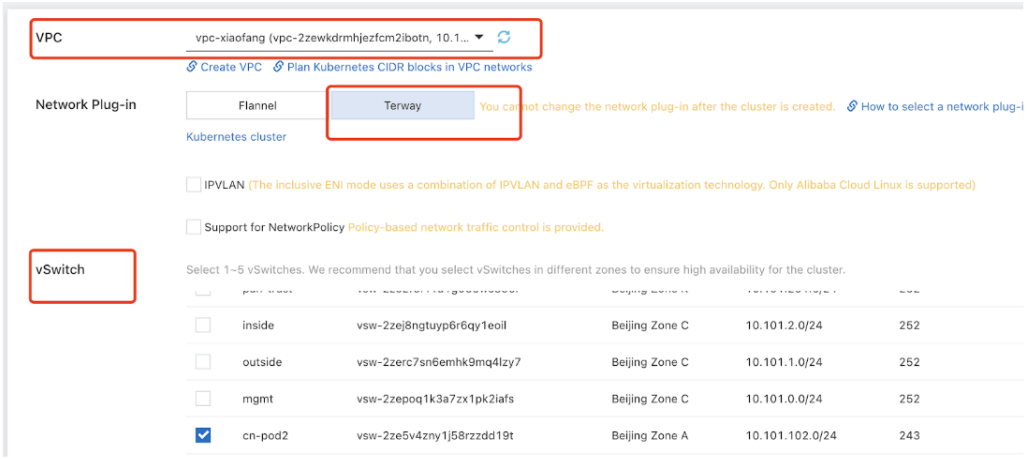
4. 单击创建 **Kubernetes** 集群。



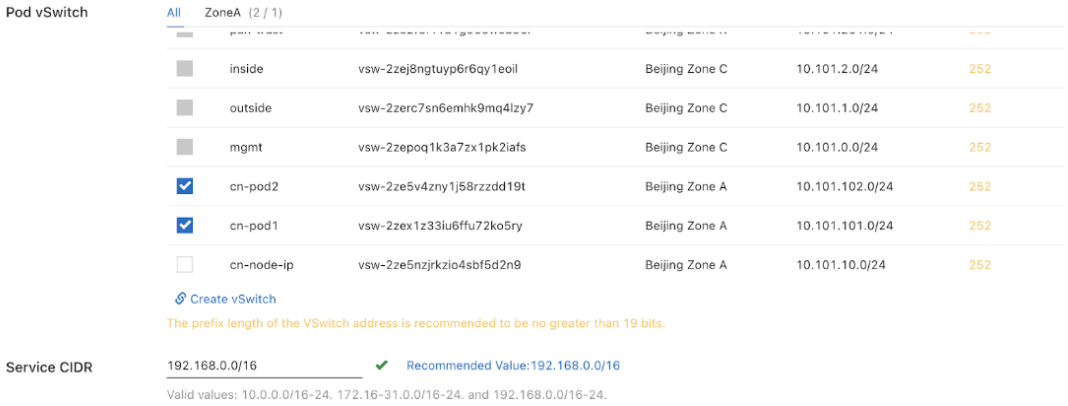
5. 要创建集群，您必须按照向导的提示配置软件参数、硬件参数和基本参数。有关配置这些必需参数的更多信息，请参阅在 ACK 上创建集群。以下步骤展示了在 ACK 平台上如何创建创建集群的示例：

   阿里云 *ACK* 上的 *CN* 系列仅支持 *Terway* 网络插件。

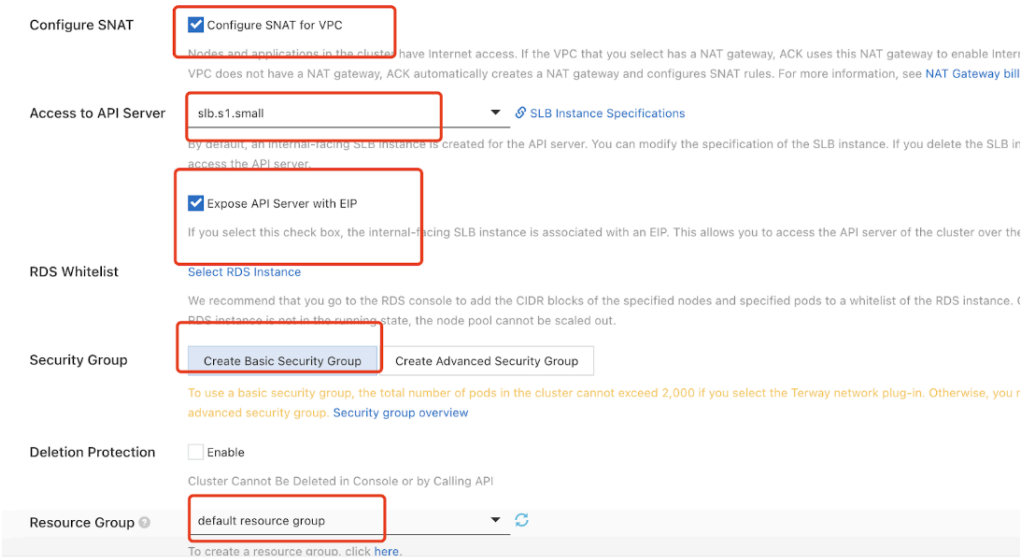
   - 选择 **VPC**、网络插件和 **vSwitch**。



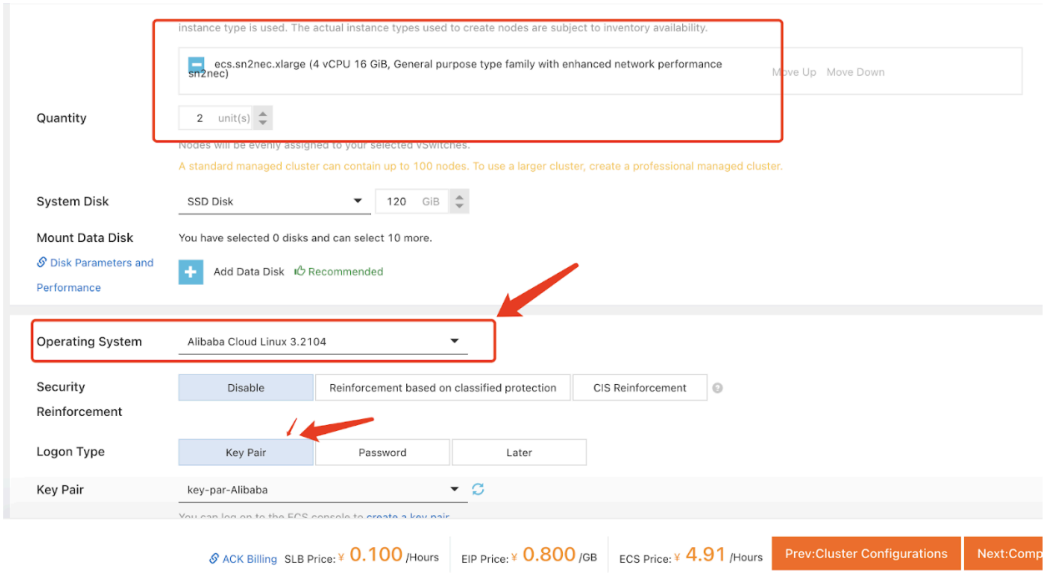
- 选择 **POD v Switch**。



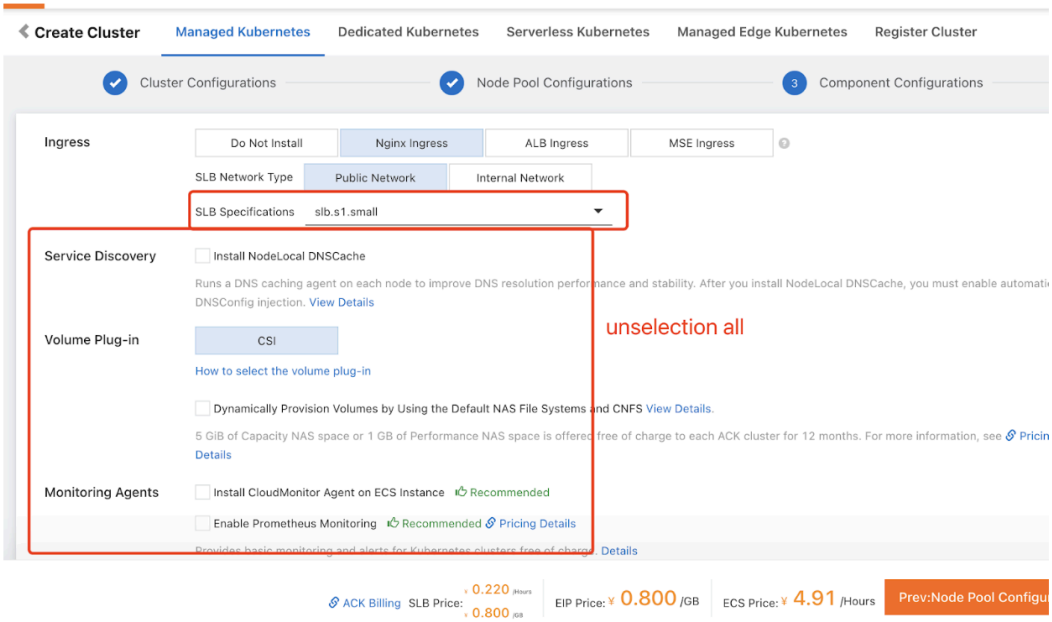
- 选择配置 **SNAT**、访问 **API** 服务器、安全组和资源组。



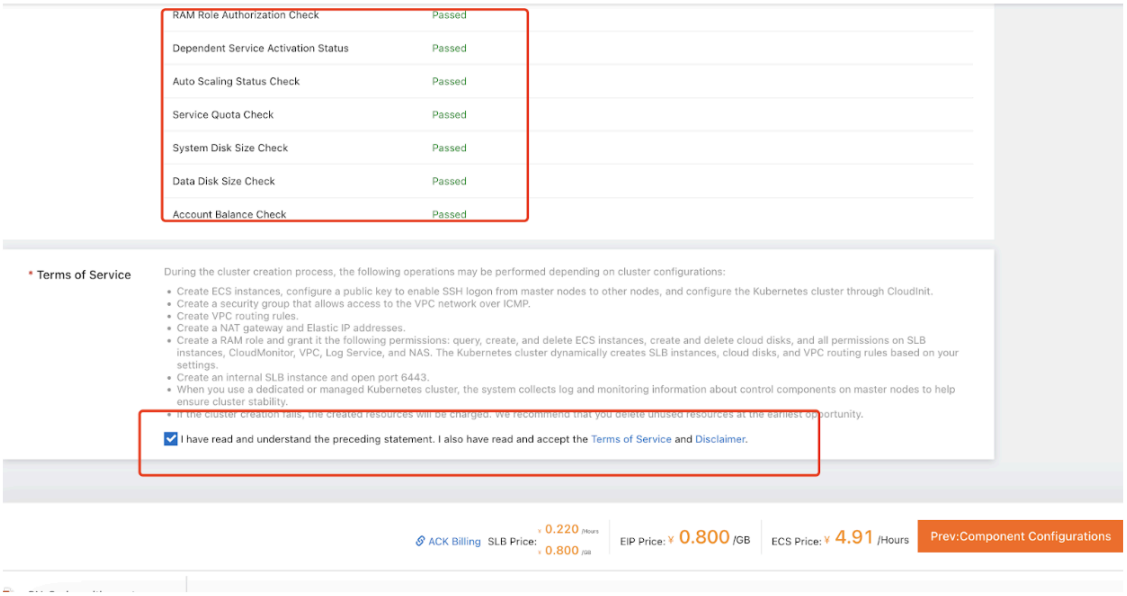
- 选择节点池配置的数量、操作系统和登录类型。



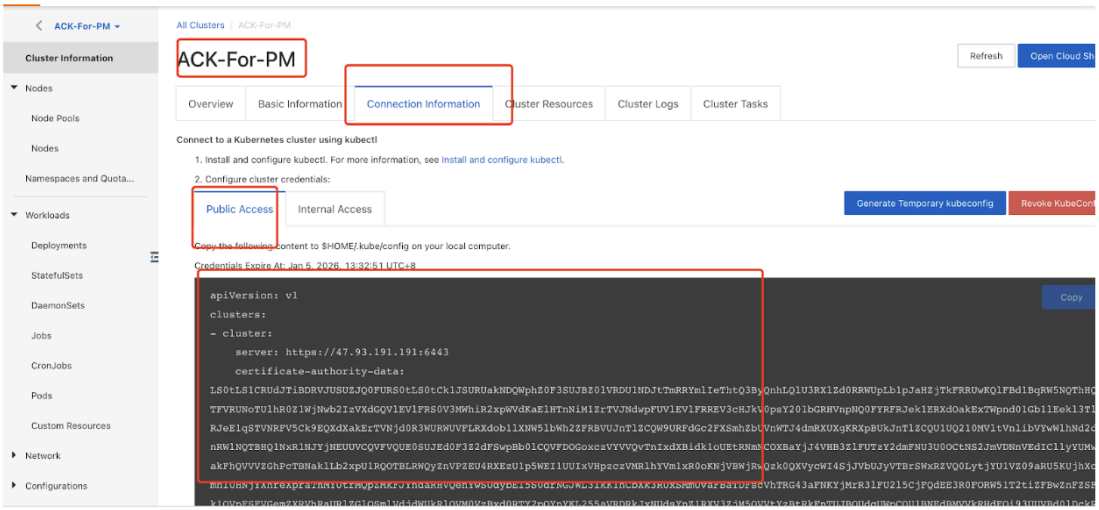
- 转到公共网络选项卡，取消选中服务发现、卷插件和监控代理复选框。



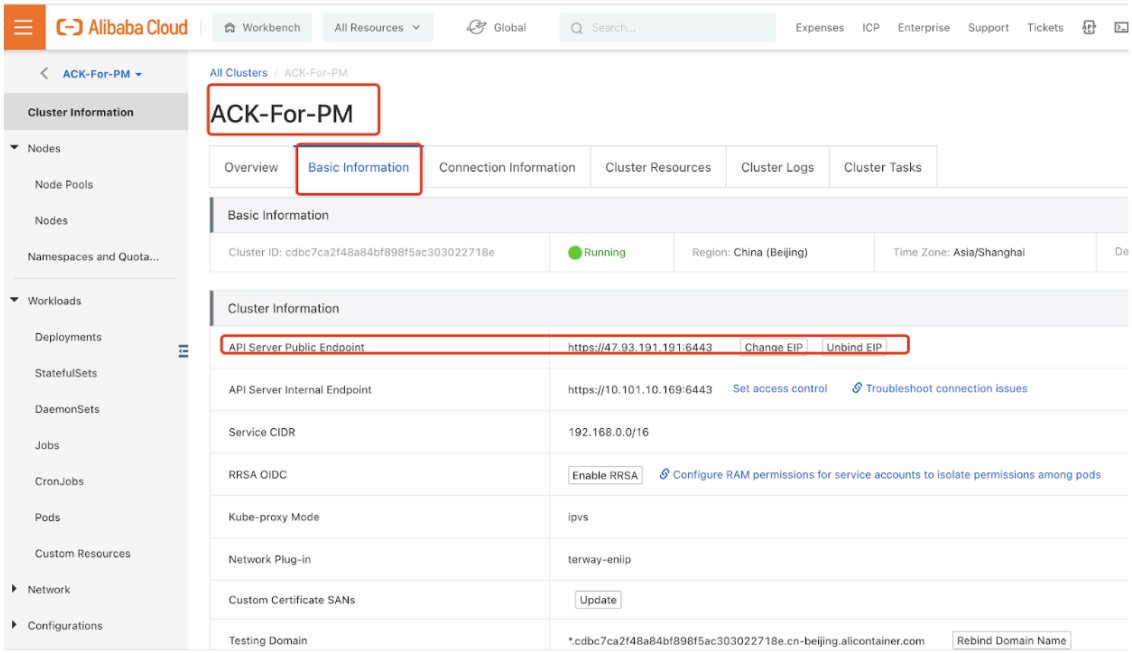
6. 选中服务条款复选框。

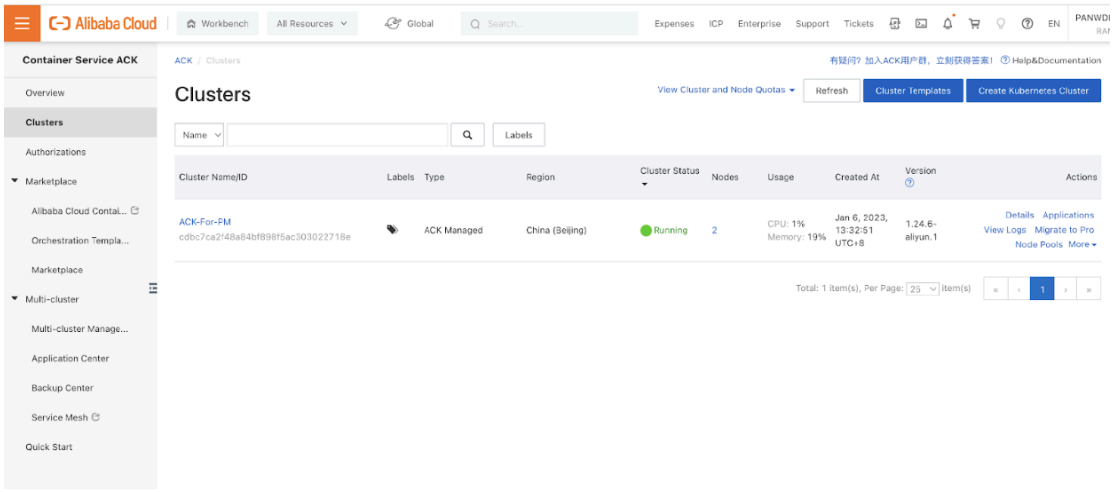


7. 单击 **Create Cluster**（创建集群）。
8. 检查登录 ACK 集群的 API 服务器密钥，并将以下内容复制到本地计算机上的 `$HOME/.kube/config` 中。



9. 获取 ACK 集群 API 服务器公共端点地址。





验证集群是否有足够的资源。默认 GKE 节点池规范不适用于 CN 系列防火墙。您必须确保集群具有 CN 系列先决条件资源，以便支持防火墙：

```
kubectl get nodes
```

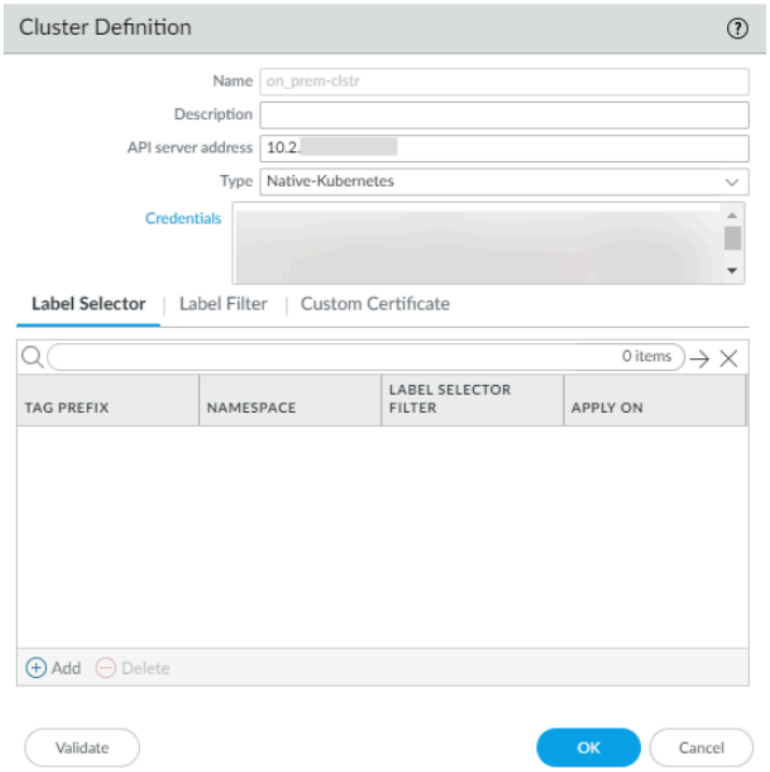
```
kubectl describe node <node-name>
```

查看命令输出中容量标题下的信息，以了解指定节点上可用的 CPU 和内存。

CPU、内存和磁盘存储分配将取决于您的需求。请参阅 CN 系列的性能和可扩展性。

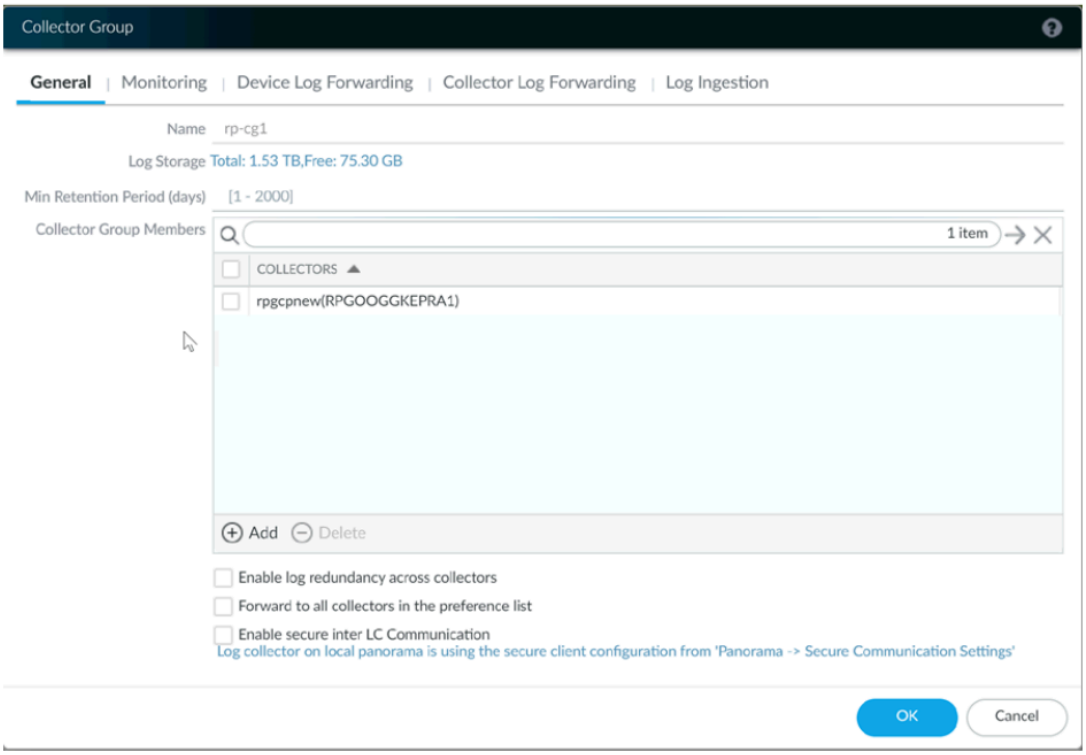
您必须确保具有以下信息：

- 收集端点 IP 地址，以便在 Panorama 上设置 API 服务器。



  Panorama 使用此 IP 地址连接到 Kubernetes 集群。

- 从 Panorama 收集模板堆栈名称、设备组名称、Panorama IP 地址和可选的日志收集器组名称。



有关详细信息，请参阅创建父设备组和模板堆栈。

- 收集 VM 身份验证密钥以及自动注册 PIN ID 和值。
- 将映像下载到的容器映像存储库的位置。

STEP 2 | （可选）如果您在 Kubernetes 插件中为 Panorama 配置了自定义证书，则必须通过执行以下命令来创建证书密钥。不要更改 ca.crt 中的文件名。pan-cn-mgmt.yaml 和 pan-cn-ngfw.yaml 中自定义证书的数量是可选的。

```
kubectl -n kube-system create secret generic custom-ca --from-file=ca.crt
```

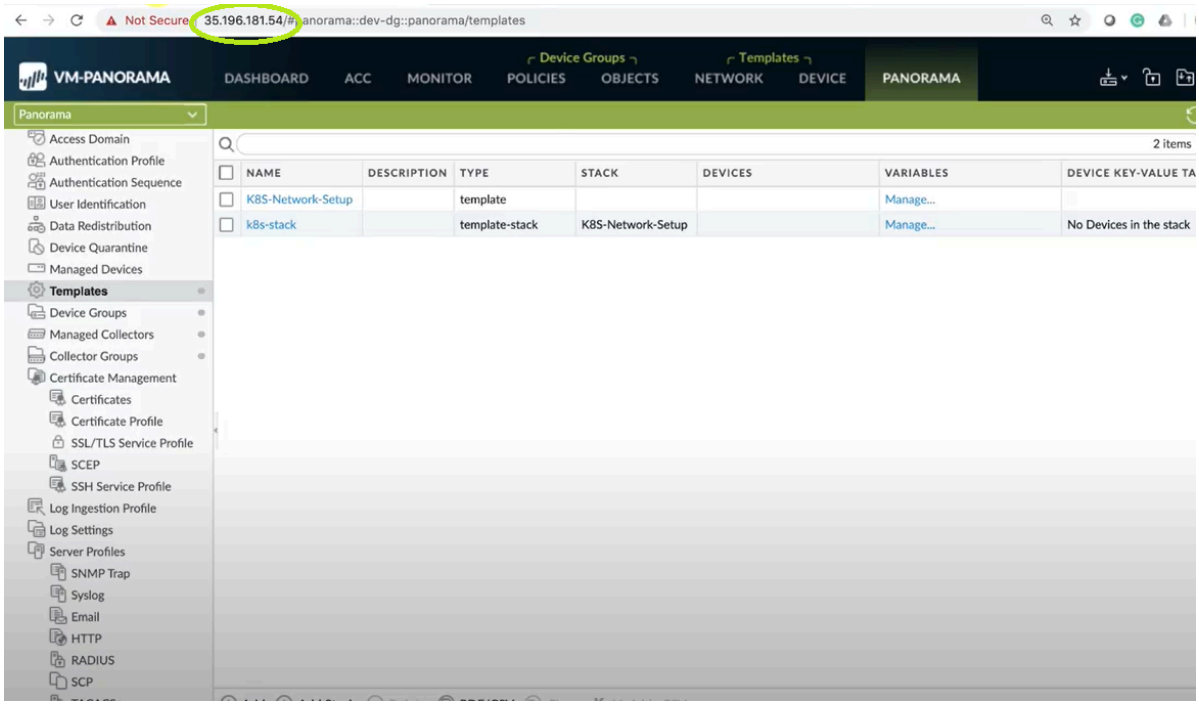
STEP 3 | 编辑 YAML 文件以提供部署 CN 系列防火墙所需的详细信息。

```
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
```

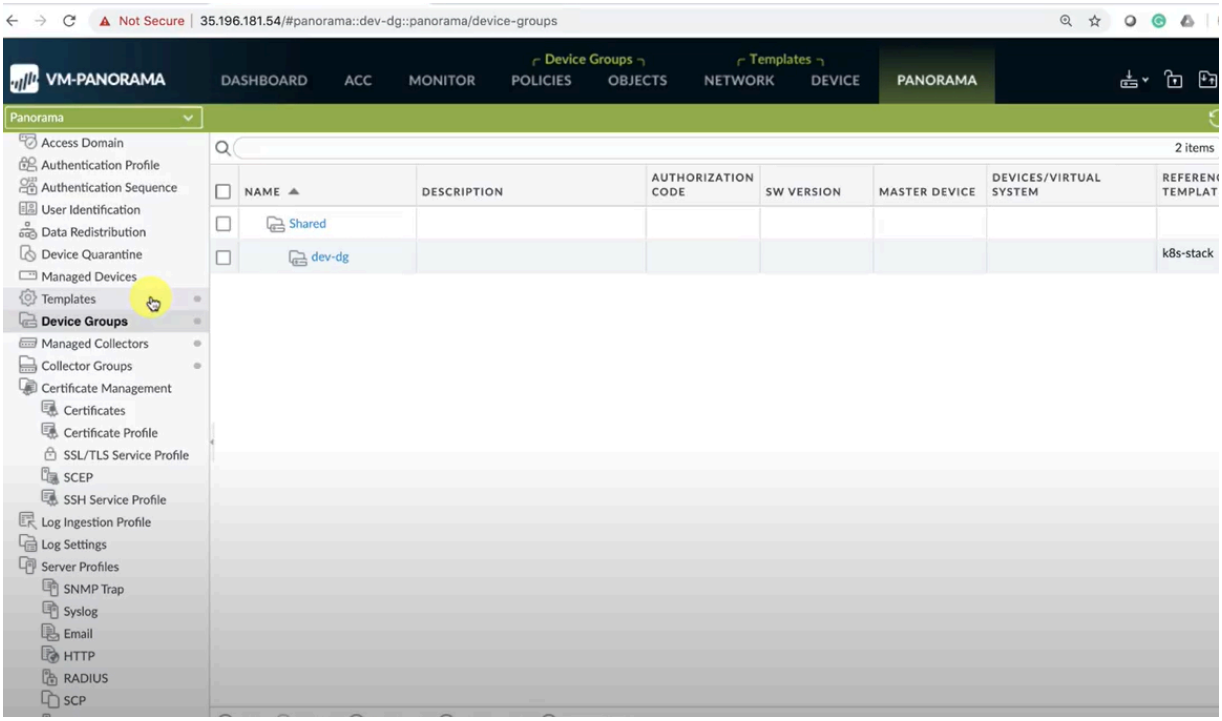
```
Certificate retrieval CN-SERIES-AUTO-REGISTRATION-PIN-ID: "<PIN
Id>" CN-SERIES-AUTO-REGISTRATION-PIN-VALUE: "<PIN-Value>"
```

您必须确保 YAML 文件中 PAN_PANORAMA_IP 参数的值与实际 Panorama IP 地址匹配，如下图所示：

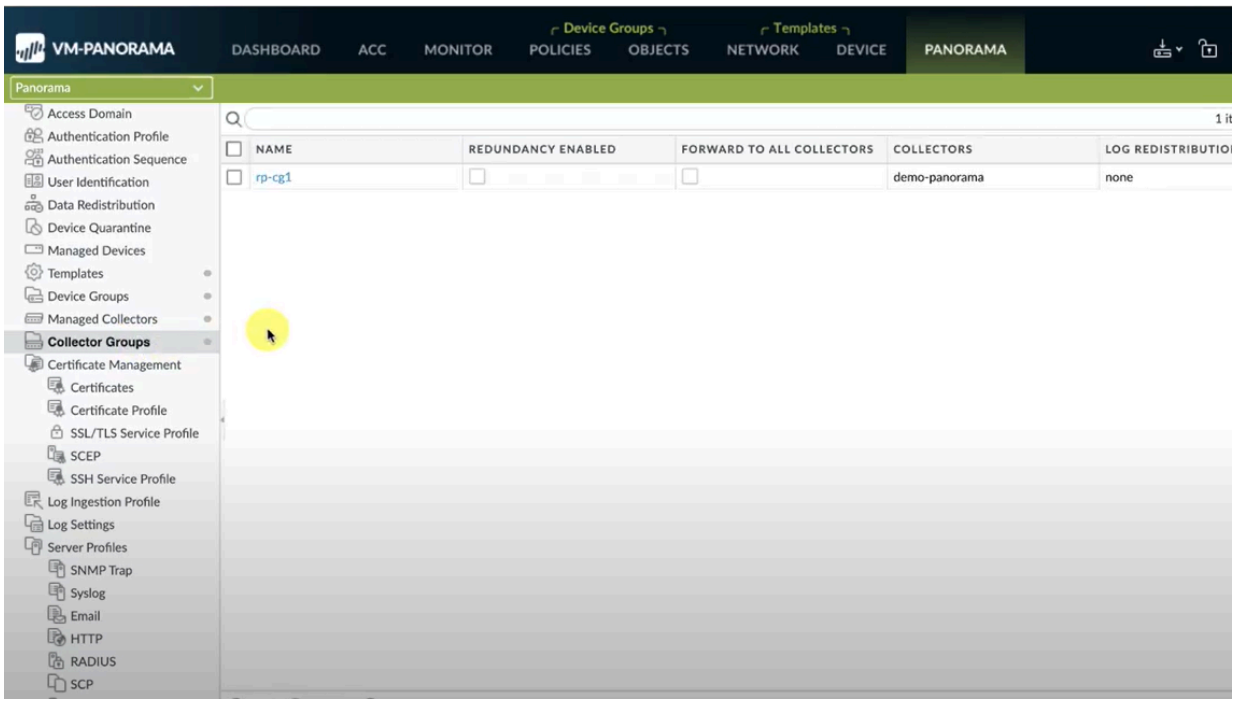


最新版本的 *YAML* 文件可在 Palo Alto Networks Kubernetes Security - CN 系列的存储库中找到。您可以从开关分支/标签下拉菜单中选择最新的分支或标签。

您必须确保 YAML 文件上的 PAN_DEVICE_GROUP 和 PAN_TEMPLATE 的参数值与在 Panorama 上创建的设备组和模板堆栈的名称匹配，如下图所示：



您必须确保 PAN_PANORAMA_CG_NAME 的参数值与创建的日志收集器名称相同。



有关详细信息，请参阅 CN 系列 yaml 文件的可编辑参数。

STEP 4 | 部署 CN-NGFW 服务。执行以下步骤：

当部署为 Kubernetes 服务时，CN-NGFW 的实例可以部署在安全节点上，而应用程序 Pod 流量将被重定向到可用的 CN-NGFW 实例以进行检查和实施。

1. 使用 pan-cni-serviceaccount.yaml 文件验证您是否已创建服务帐户。

   请参阅为集群身份验证创建服务帐户。

2. 使用 Kubectl 运行 pan-cni-configmap.yaml 文件。

   kubectl apply -f pan-cni-configmap.yaml

3. 使用 kubectl 运行 pan-cn-ngfw-svc.yaml。

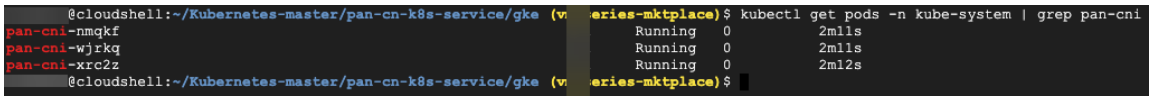
   kubectl apply -f pan-cn-ngfw-svc.yaml

   该 *yaml* 必须在 *pan-cni.yaml* 之前部署。

4. 使用 Kubectl 运行 pan-cni.yaml 文件。

   kubectl apply -f pan-cni.yaml

5. 验证是否已修改 pan-cni-configmap 和 pan-cni YAML 文件。
6. 运行以下命令并验证输出是否与以下示例相似。



阿里云 *ACK* 仅支持基于标准指标的自动扩缩。

STEP 5 | 部署 CN-MGMT StatefulSet。

默认情况下，部署管理平面作为提供容错功能的 StatefulSet。最多可以将 30 个防火墙 CN-NGFW Pod 连接到 CN-MGMT StatefulSet。

1. （仅对于静态配置的 PV 为必需）为 CN-MGMT StatefulSet 部署持久卷 (PV)。

   1. 创建目录以匹配 pan-cn-pv-local.yaml 文件中定义的本地卷名称。

      在至少 2 个工作节点上需要六 (6) 个目录。登录将在其中部署 CN-MGMT StatefulSet 的每个工作节点，以创建目录。例如，要在 `/mnt/pan-local6` 下创建名为 `/mnt/pan-local1` 的目录，请运行以下命令

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

   2. 修改 pan-cn-pv-local.yaml。

      在 `nodeaffinity` 下匹配主机名，并验证是否已修改在 `spec.local.path` 中创建的上述目录，然后部署文件以创建新的存储类 pan-local-storage 和本地 PV。

      在 *`pan-cn-mgmt.yaml`* 文件中创建 ***volumeClaimTemplates*** 时，您必须添加 *`alicloud-disk-available`* 存储类名称。

      例如：

      `storageClassName: alicloud-disk-available`

      所有 *PV* 的存储大小至少应为 *20G*。

2. 验证是否已修改 pan-cn-mgmt-configmap 和 pan-cn-mgmt YAML 文件。

   pan-cn-mgmt.yaml 文件示例

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

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

3. 使用 Kubectl 运行 yaml 文件。

   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

   如果您之前未完成为集群身份验证创建服务帐户，则必须运行 pan-mgmt-serviceaccount.yaml。

4. 通过运行以下命令验证 CN-MGMT Pod 是否已启动：

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

   大约需要 5 至 6 分钟的时间。

STEP 6 | 部署 CN-NGFW Pod。

1. 验证是否已按 PAN-CN-NGFW-CONFIGMAP 和 PAN-CN-NGFW 中的详细说明修改 YAML 文件。

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

2. 使用 Kubectl apply 运行 pan-cn-ngfw-configmap.yaml 文件。

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

3. 使用 Kubectl apply 运行 pan-cn-ngfw.yaml 文件。

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

4. 确认已部署 CN-NGFW Pod。

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

STEP 7 | 验证您是否可以在 Kubernetes 集群上看到 CN-MGMT、CN-NGFW 和 PAN-CNI。

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

STEP 8 | 为应用程序 yaml 文件或命名空间添加注释，以便将来自其新 Pod 的流量重定向到防火墙。

您需要添加以下注解，以将流量重定向到 CN-NGFW 来进行检查：

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

例如，对于“default”命名空间中的所有新 Pod：

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

STEP 9 | 在集群中部署应用程序。

在 OpenShift 上部署 CN 系列

在何处可以使用？	需要提供什么？
• OpenShift 环境上的 CN-Series部署	• CN-Series 10.1.x or above Container Images • 运行 PAN-OS 10.1.x 或更高版本的 Panorama

pan-cni 在应用程序 Pod 的默认“eth0”接口上保护流量。如果您拥有多宿主 Pod，则可以配置 CN-NGFW Pod 保护使用基于桥接的连接配置的其他接口，以与其他 Pod 或端口进行通信。根据应用程序 YAML 文件中的注释，您可以配置 CN 系列防火墙以检查来自所有接口或附加到每个 Pod 的选定数量的接口的流量。

pan-cni 不创建任何网络，因此不需要 IP 地址，如其他 CNI 插件。

需要 *PAN-OS 10.1.3* 或更高版本才能将 *CN* 系列部署为 *OpenShift* 上的 *Kubernetes* 服务。此外，作为 *OpenShift* 上的 *Kubernetes* 服务的 *CN* 系列仅保护接口 ***eth0***。

STEP 1 | 部署集群。

请参阅云平台供应商的文档，并验证 CN 系列是否支持 OpenShift 版本和 CNI。查看 CN 系列防火墙的映像文件和 CN 系列 yaml 文件中的可编辑参数。

STEP 2 | 按照使用 CN 系列保护 Kubernetes 工作负载的安全中包含的工作流程操作。

您必须创建服务凭据，并部署防火墙 YAML。

注意：如果服务凭据超过 *10KB*，则必须将文件 *Gzip*，并对压缩文件进行 *base64* 编码，然后将文件内容上传或粘贴到 *Panorama CLI* 或 *API*。

STEP 3 | 配置 PAN-CNI 插件以使用 Multus CNI 插件。

OpenShift 上的 Multus CNI 可用作调用其他 CNI 插件的“元插件”。对于每个应用程序，您必须：

1. 在每个 Pod 命名空间中部署 PAN-CNI 网络附件定义。

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

2. 修改应用程序 YAML 文件。

   部署 pan-cni-net-attach-def.yaml 文件后，在应用程序 Pod yaml 中添加注释：

   paloaltonetworks.com/firewall: pan-fw

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

   如果上述注释中还有其他网络，请在需要检查的网络后面添加 **pan-cni**。系统将不会重定向和检查后跟 **pan-cni** 的网络。

   如果 *Pod* 具有多个接口，则必须在 *pan-cni-configmap.yaml* 文件中的 *“interfaces”* 下指定 *CN-NGFW Pod* 要为其检查流量的接口名称。

   例如：

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

现在，在 *Kubernetes* 服务部署模式和 *DaemonSet* 模式下，*CN* 系列现在支持 *RedHat OpenShift 4.13* 及以上版本 的*OVN-Kubernetes* 容器网络接口 *(CNI)* 插件。

在 OpenShift Operator Hub 上部署 CN 系列

在何处可以使用？	需要提供什么？
• CN-Series部署	• CN-Series 10.1.x or above Container Images • 运行 PAN-OS 10.2.x 及以上版本的 Panorama

CN 系列容器防火墙现已在 RedHat Openshift 平台 Operator Hub 上提供。您可以直接从 RedHat Operator Hub 部署、配置和操作 CN 系列容器防火墙。

CN 系列在 Openshift Operator Hub 上的配置：

以下是在 Openshift Operator Hub 上部署 CN 系列防火墙的先决条件：

- 授予 CN 系列防火墙许可证Panorama 上的 Kubernetes 插件管理 CN 系列防火墙许可。生成您的身份验证代码，并在准备部署 CN 系列防火墙时将其放在手边。有关详细信息，请参阅 CN 系列防火墙许可。
- 在 Panorama 上生成 VM 身份验证密钥。
- 在 VM 系列防火墙上安装设备证书。
- 创建为集群身份验证创建服务帐户。
- 部署 Panorama — 您必须使用 Panorama 来配置、部署和管理 CN 系列防火墙部署。有关部署和设置 Panorama 设备的更多信息，请参阅设置 Panorama。
- 为 CN 系列防火墙安装 Kubernetes 插件。
- OpenShift 集群必须符合 CN 系列先决条件。
- 确保您可以访问 Palo Alto Networks 客户服务门户 (CSP)，并且有 Flex 积分。
- 确保您是拥有 OpenShift 许可证的 RedHat 客户，并且拥有在 OpenShift 中创建资源的权限。
- 确保 OpenShift 集群符合 CN 系列先决条件。

有关更多信息，请参阅如何在 RedHat Openshift Operator Hub 上轻松部署 CN 系列。

在 OpenShift Operator Hub 上部署 CN 系列：

pan-cni 在应用程序 Pod 的默认 **eth0** 接口上保护流量。如果您拥有多宿主 Pod，则可以配置 CN-NGFW Pod 保护使用基于桥接的连接配置的其他接口，以与其他 Pod 或端口进行通信。根据应用程序 YAML 文件中的注释，您可以配置 CN 系列防火墙以检查来自所有接口或附加到每个 Pod 的选定数量的接口的流量。

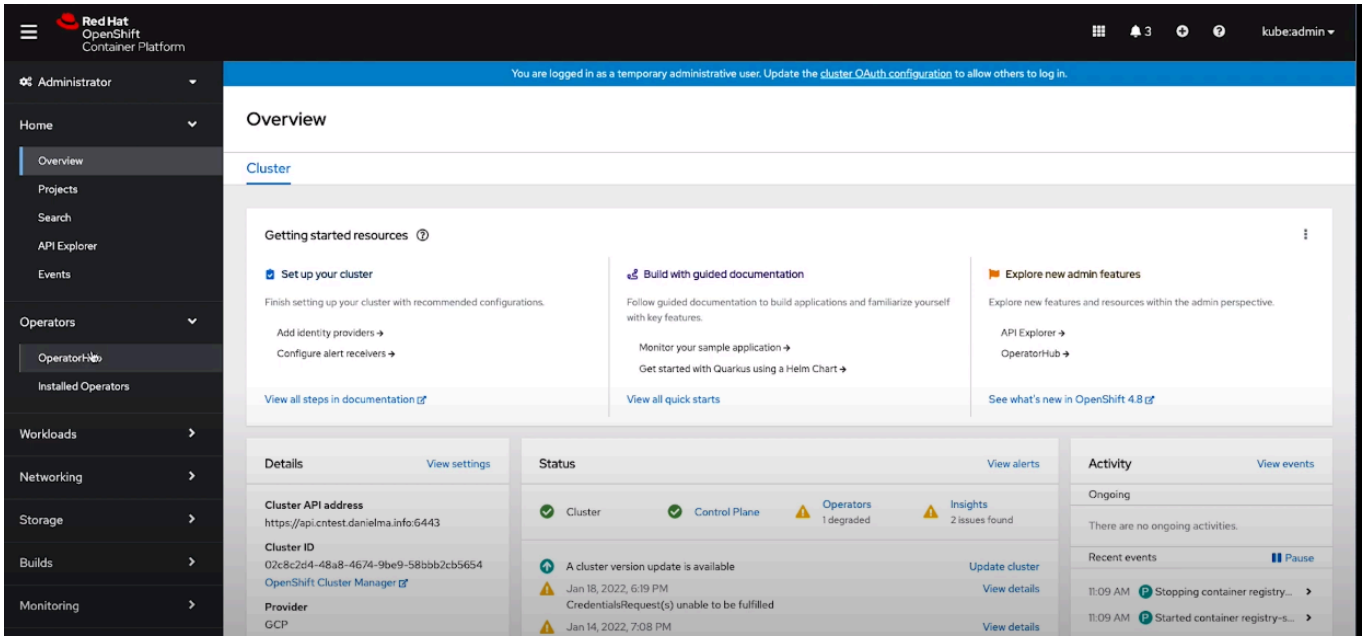
pan-cni 不创建网络，因此不像其他 CNI 插件那样需要 IP 地址。

您需要 *PAN-OS 10.2* 或更高版本才能在 *OpenShift Operator Hub* 上部署 *CN* 系列。

以下是在 Redhat OpenShift Operator Hub 上部署 CN 系列防火墙的步骤：

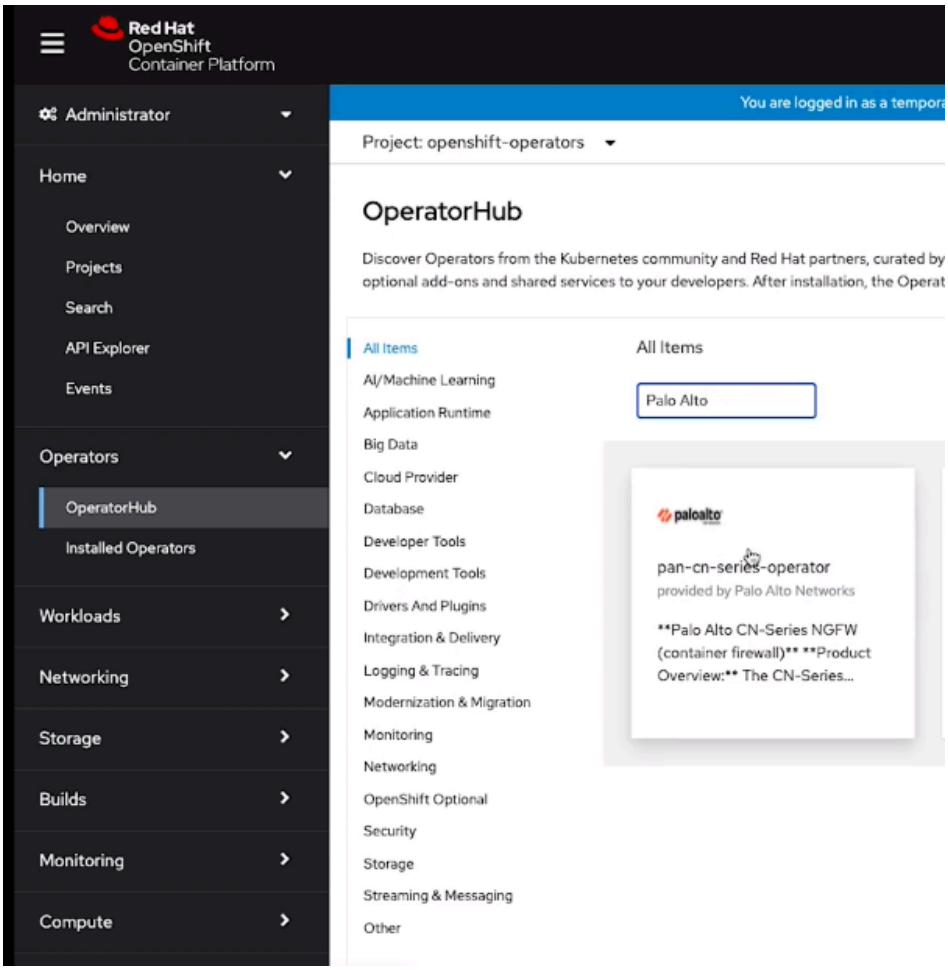
STEP 1 | 登录 Redhat OpenShift 容器控制台。

STEP 2 | 转到 **Operators**，然后单击 **OperatorHub**。



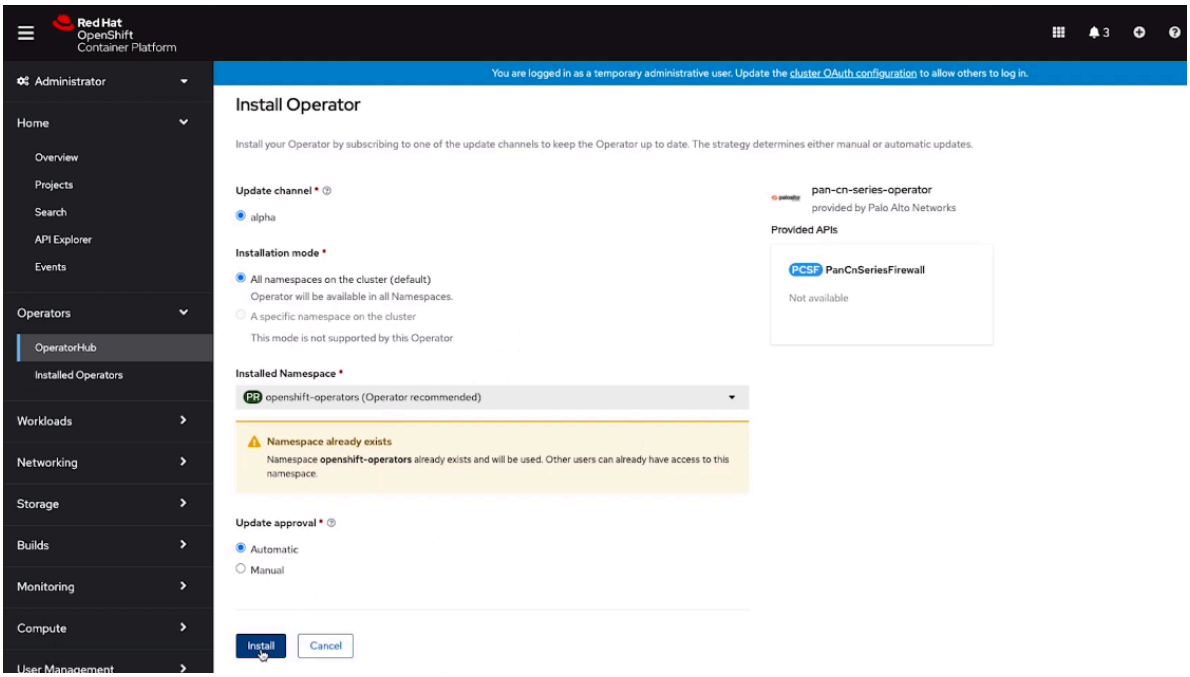
STEP 3 | 在 Operator 搜索框中输入 **Palo Alto**。

STEP 4 | 单击 **pan-cn-series-operator**。



单击 **pan-cn-series-operator** 磁贴时将打开安装窗口。

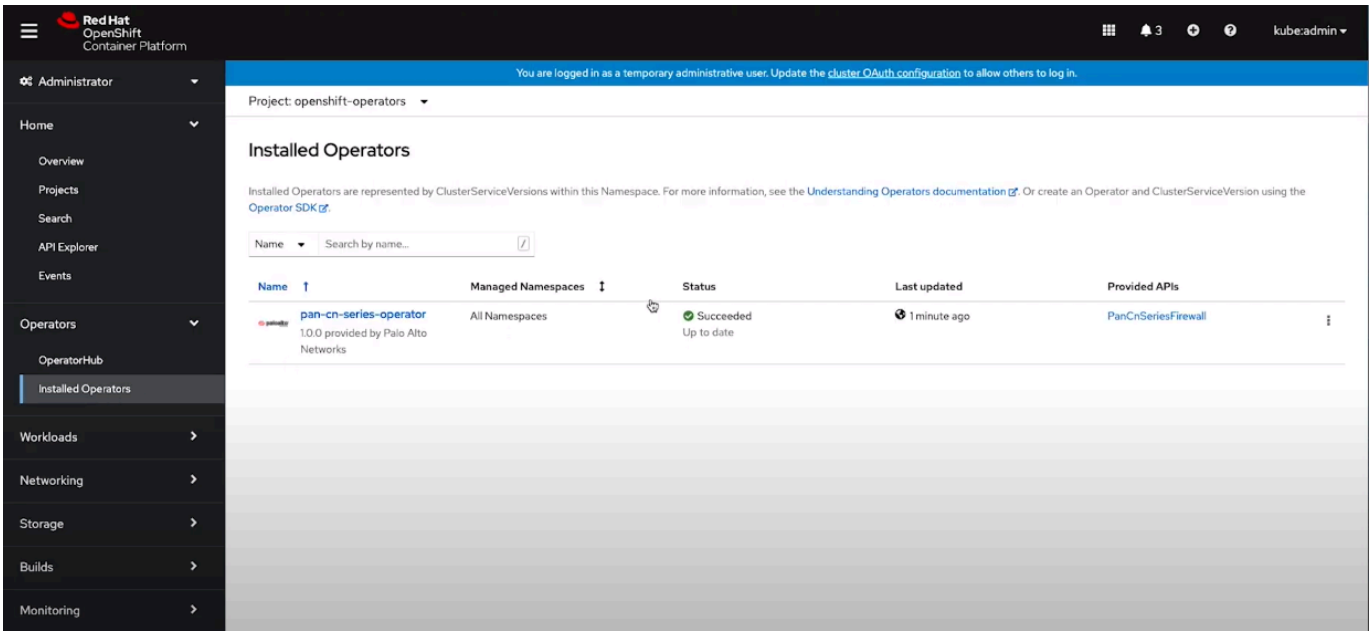
STEP 5 | 单击安装，以便在 OpenShift 集群上安装 pan-cn-series Operator。



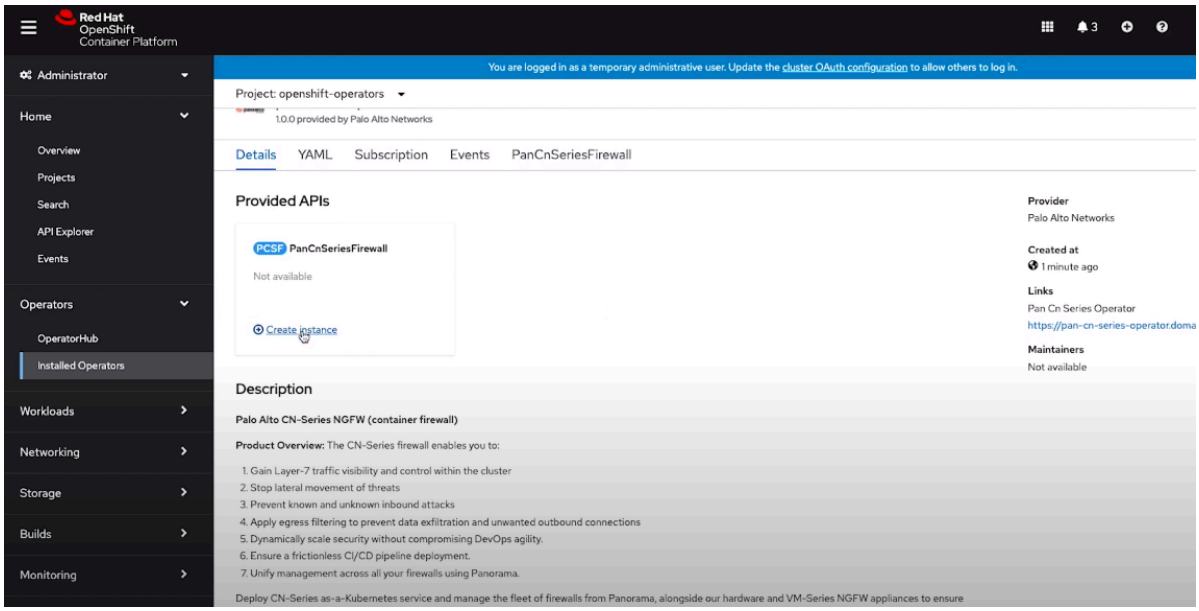
预安装步骤，然后执行此处提供的后续部署步骤。

如果服务凭据超过 *10KB*，则必须将文件 *Gzip*，并对压缩文件进行 *base64* 编码，然后将文件内容上传或粘贴到 *Panorama CLI* 或 *API*。

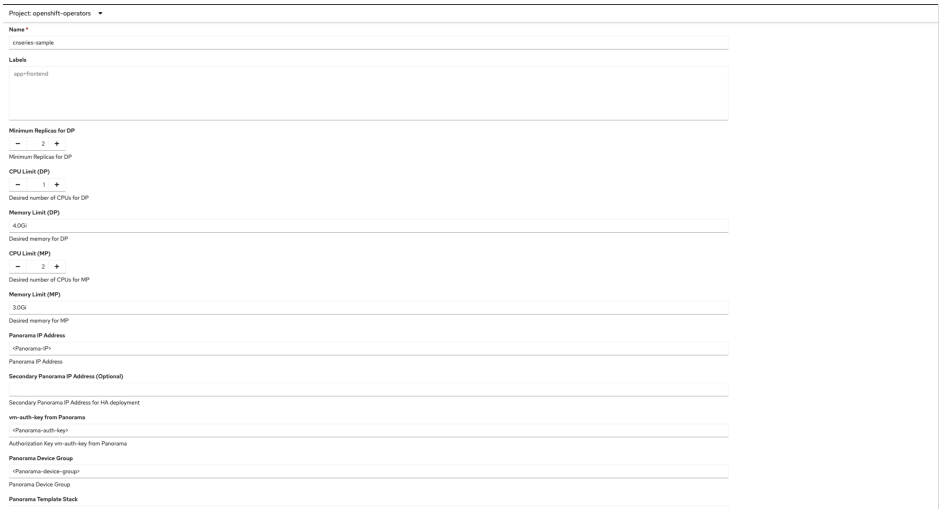
STEP 6 | 在导航菜单上，转到已安装的 **Operators**，然后单击已安装的 **pan-cn-series-operator**。



STEP 7 | 单击 **Create Instance**（创建实例）。



STEP 8 | 输入唯一的操作数名称。



STEP 9 | 输入 **DP** 的最小副本数、单元,以及 DP 和 MP Pod 的 **vCPU** 限制。有关 vCPU 限制的信息，请参阅 CN 系列关键性能指标。

STEP 10 | 输入 **Panorama IP** 地址。

STEP 11 | 可选输入 HA 部署的**Panorama** 辅助 **IP** 地址。

STEP 12 | 输入 CN 系列 Panorama 验证密钥。

STEP 13 | 输入 **Panorama** 设备组。

STEP 14 | 输入 **Panorama** 模板堆栈。

STEP 15 | 输入 **Panorama** 日志收集器组名称。

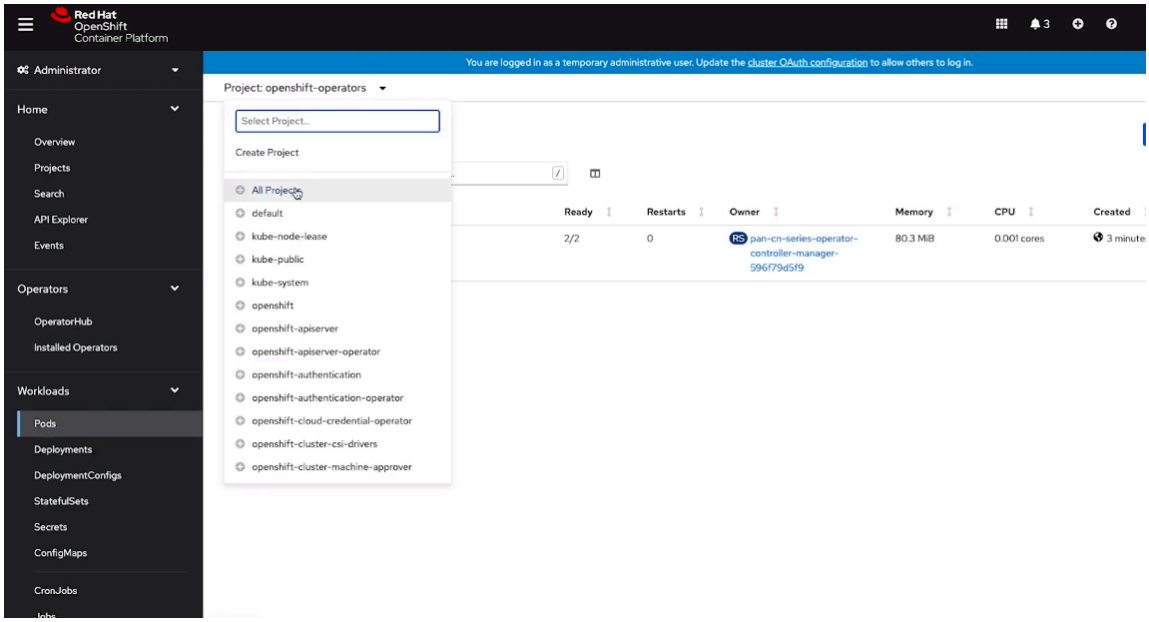
STEP 16 | 可选输入客户支持门户 (CSP) **PIN ID**、**PIN** 值和 **URL**。

STEP 17 | 根据您的 PAN-OS 版本，在 CN 系列容器注册表控制台中链接到 DP、MP 和 CNI 的相应映像。

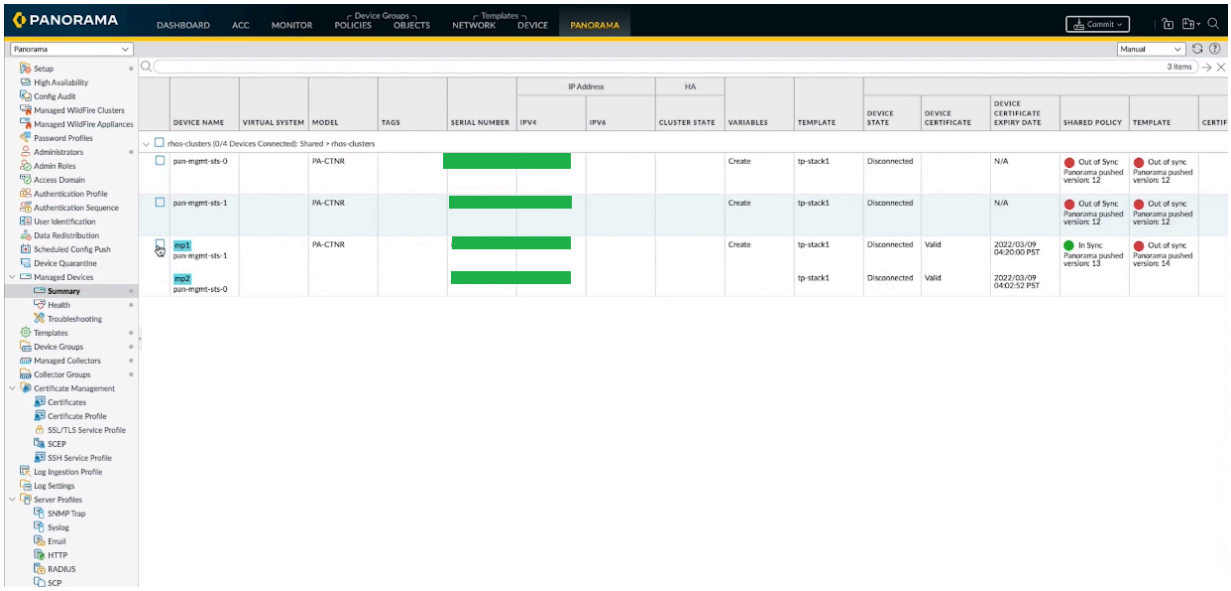
STEP 18 | 单击 **Create**（创建）。

STEP 19 | 在导航菜单上，转到 **Pod**。

STEP 20 | 选择项目 **OpenShift-operators**，然后转到 **kube-system**，查看作为操作数的一部分部署的 CNI、管理和数据平面 Pod 的名称和状态。



您可以在 Panorama 上查看防火墙部署状态。设备状态将在部署后 5 分钟内更改为“已连接”。



STEP 21 | 将 PALO ALTO NETWORKS-CNI 插件配置为与 Multus CNI 插件一起使用。

OpenShift 上的 Multus CNI 可用作调用其他 CNI 插件的 **meta-plugin**。对于每个应用程序，您必须：

1. 运行以下命令，在每个 Pod 命名空间内部署 pan-cni-net-attach-def.yaml：

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

2. 修改应用程序 YAML 文件。

   在部署 pan-cni-net-attach-def.yaml 之后，在 app pod yaml 中添加以下注释：

   paloaltonetworks.com/firewall: pan-fw

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

   如果上述注释中还有其他网络，请在需要检查的网络后面添加 **pan-cni**。系统将不会重定向和检查后跟 **pan-cni** 的网络。

   如果 *Pod* 具有多个接口，则必须在 *pan-cni-configmap.yaml* 文件中的 ***interfaces*** 下指定 *CN-NGFW Pod* 要为其检查流量的接口名称。

   例如：

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