

09:10

Entendiendo la arquitectura kubernetes

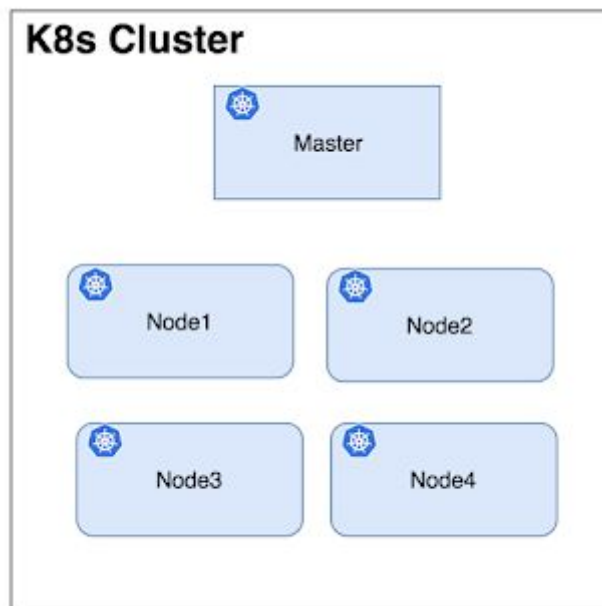
 Ajeet Khan  AWS , nube , DevOps , cómo Kubernetes obras , K8S , cúmulo K8S , Kops , kubecti , kubernetes , Kubernetes arquitectura  1 comentario

Cómo funciona Kubernetes | Interno de Kubernetes

Resumen de #Kubernetes:

- Kubernetes es una plataforma de gestión de contenedores.
- Creado por Google
- Escrito en Go / GoLang
- También conocido como K8s

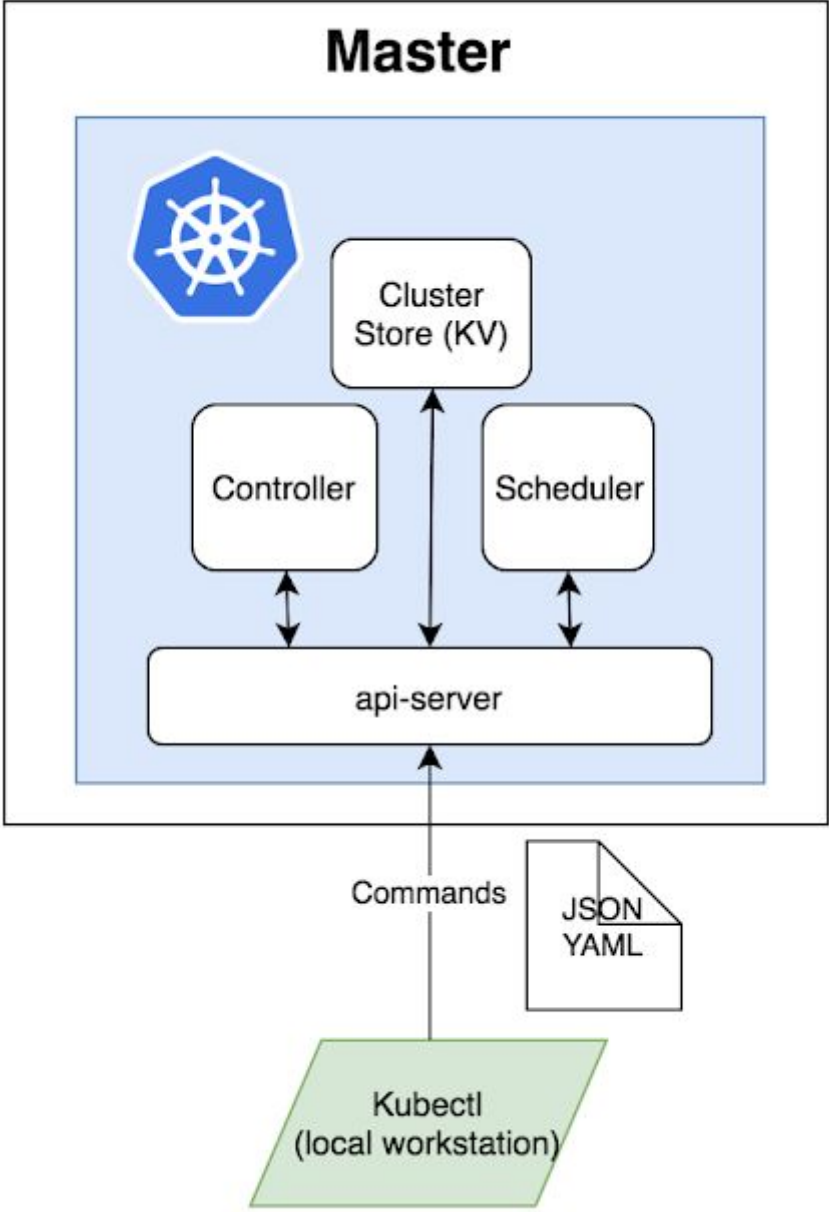
K8s Cluster



Nodo maestro

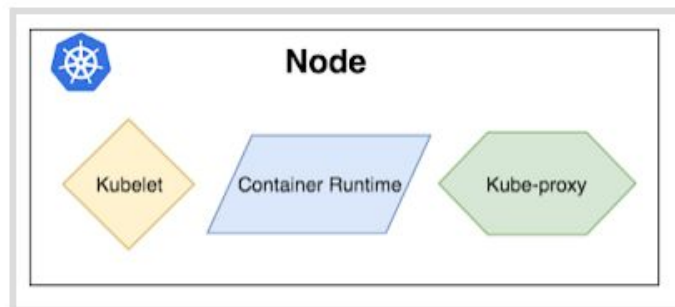
Maestro es el plano de control o el cerebro del grupo k8s. Un Maestro consta de pocos componentes:

- **api-server** - Expone la API REST para hablar con el clúster k8s, consume json, solo las conversaciones de api-server con Cluster Store.
- **Cluster Store (KV)** : administración de configuración y estado de clúster.
- **Programador** : mira el api-server en busca de nuevos pods y asigna un nodo para que funcione
- **Controlador** : un demonio que observa el estado del clúster para mantener el estado deseado. Los ejemplos son el controlador de replicación, el controlador de espacio de nombres, etc. Aparte de esto, realiza una recolección de basura de pods, nodos, eventos, etc.



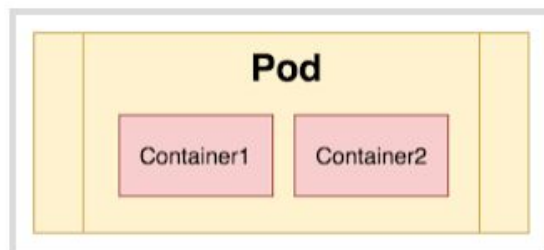
#Nodo

- **Kubelet** : el agente de k8s que registra nodos en un clúster, mira el api-server, crea instancias de pods, informa al api-server. Si el pod falla, informa al maestro y el maestro decide qué hacer. Expone el puerto 10255 en el nodo
- **Motor de contenedores** : hace la gestión de contenedores como tirar imágenes, iniciar / detener contenedores. Normalmente Docker se utiliza para el tiempo de ejecución de contenedores.
- **kube-proxy** : responsable de la red, proporciona IP única a los Pods, todo el contenedor en un pod comparte la misma IP, carga los saldos en todos los pods en un **servicio**



#Pods

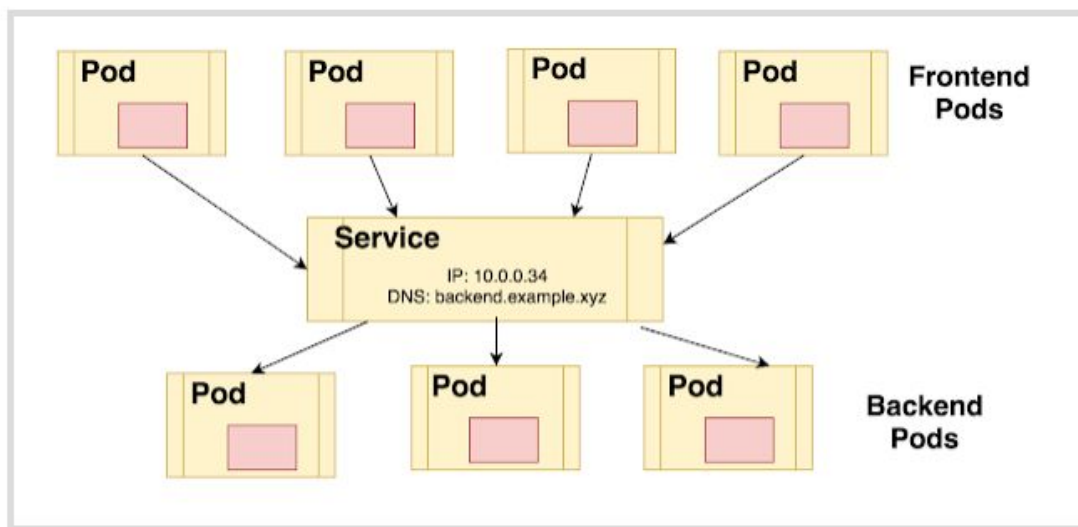
- Un entorno para ejecutar contenedores.
- Tiene pila de red, espacios de nombres de kernel y uno o más contenedores en ejecución
- El contenedor siempre corre dentro de una vaina
- Pod puede tener múltiples contenedores
- Es unidad de escalado en k8s.



#Servicios

Pods va y viene con diferentes IPs. Para distribuir la carga y actuar como una única fuente de interacción para todos los pods de una aplicación, el **servicio** cumple la función.

- Tiene un solo IP y DNS
- Creado con un archivo JSON manifiesto
- Todos los pods nuevos se agregan / registran en el servicio.
- Qué pod se debe asignar a qué servicios se deciden por **etiquetas**
- El **servicio** y los **pods** tienen **etiquetas** en función del servicio que identifica a sus **pods**
- Solo envía tráfico a vainas saludables.
- El **servicio** puede apuntar cosas fuera del clúster
- usa tcp por defecto (udp también es compatible)



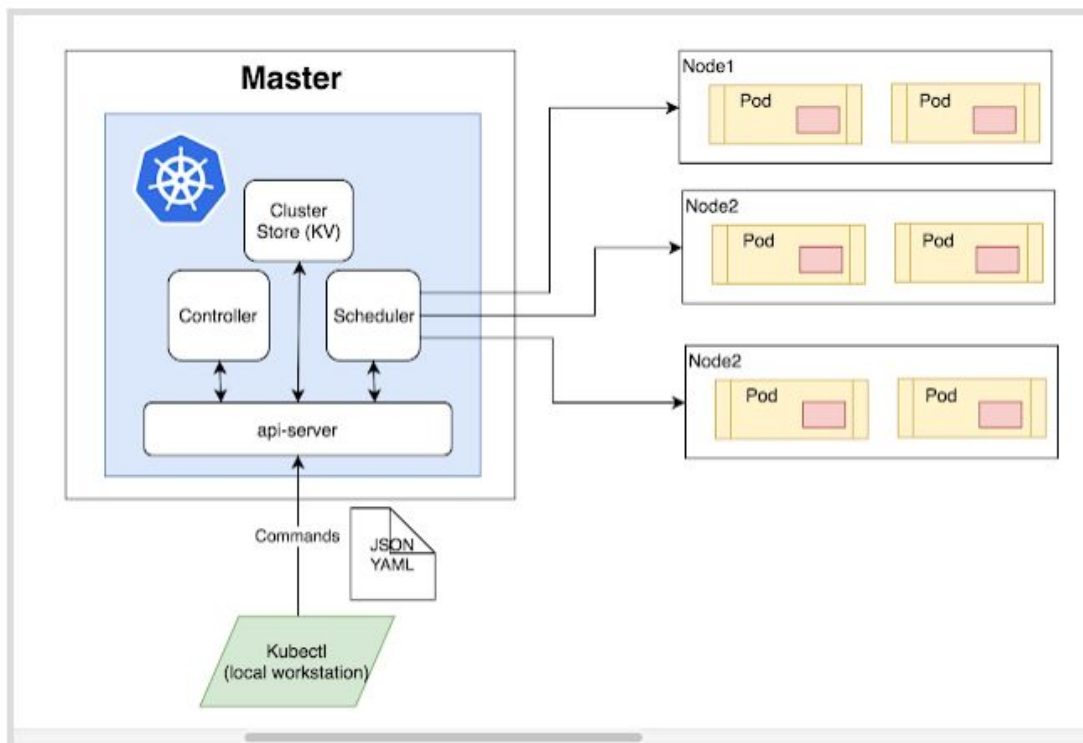
Despliegues

Es un objeto k8s cuya tarea es administrar pods idénticos ejecutándose y actualizándolos de forma controlada.

- Desplegado usando el manifiesto YAML / JSON
- Desplegado a través de api-server
- Proporcionar actualización de vainas
- Proporcionar rollbacks

```
apiVersion: extensions/v1beta1
kind: Deployment
metadata:
  name: xyz
spec:
  replicas: 4
```

Arquitectura detallada



Flujo general

- kubectl escribe en el servidor API
- El servidor API valida la solicitud y la conserva en el almacén de clústeres (etcd)
- El almacén de clúster (etcd) notifica al servidor API
- El servidor API invoca el programador
- El programador decide dónde ejecutar el pod y lo devuelve al servidor API
- API Server lo persiste a etcd
- etcd notifica de nuevo el servidor API.
- El servidor API invoca Kubelet en el nodo correspondiente
- Kubelet habla con el demonio Docker usando la API sobre el socket Docker para crear el contenedor
- Kubelet actualiza el estado del pod al servidor API
- El servidor API persiste el nuevo estado en etcd

Curso #Kubernetes

- Introducción a Kubernetes
- Configurar Kubernetes localmente para el entorno de desarrollo
- Configurar Kubernetes en AWS utilizando KOPS
- Configurar Kubernetes en la nube de Google
- Configuración del clúster Kubernetes utilizando kubeadm | listo para producción
- Cómo crear trabajos cron en Kubernetes