

Kubernetes For Storm

Noviembre 2020

Unidad 2: Introducción a Kubernetes

El proyecto kubernetes

- Gestiona el despliegue de aplicaciones sobre contenedores, automatizando el despliegue, con énfasis en la escalabilidad y controlando todo el ciclo de vida:
 - Despliega aplicaciones rápidamente
 - Escala las aplicaciones al vuelo
 - Integra cambios sin interrupciones
 - Permite limitar los recursos a utilizar
- Es un proyecto centrado en la puesta en PRODUCCIÓN de contenedores y por tanto indicada su gestión para administradores de sistemas y personal de equipos de operaciones.
- Afecta a los desarrolladores, ya que las aplicaciones deben adaptarse para poder desplegarse en kubernetes

El proyecto kubernetes

- Para desplegarse en nube: pública, privada o híbrida. Puede instalarse también en servidores físicos, pero no es su entorno natural
- Extensible: Módulos, plugins, adaptable, ...
- Autoreparable



- Proyecto comenzado por Google en 2014, ahora gestionado por Cloud Native Computing Foundation
- Permite gestionar un clúster de nodos en los que desplegar aplicaciones sobre contenedores
- kubernetes es una palabra del griego antiguo que significa “timonel”
- Habitualmente se escribe k8s
- Es muy importante el ecosistema que se está desarrollando alrededor

El proyecto kubernetes

- Primera versión: 7 junio 2014
- Última versión: 26 agosto 2020 (1.19). Poco a poco ralentizándose la publicación de nuevas versiones
- Desarrollado en Go
- Licencia Apache 2.0
- <https://github.com/kubernetes/kubernetes>
- Kubernetes está inspirado en Borg, software de Google encargado de la gestión de los servicios de Google
- Casi 3000 personas han contribuido
- Más de 94000 commits en 6 años

El ecosistema

- Una de los aspectos más interesantes de k8s es la cantidad de desarrollo que tiene alrededor, un ecosistema en ebullición con desarrollo de múltiples aplicaciones de las que pocas sobreviven suficiente tiempo

<https://landscape.cncf.io/>

Arquitectura básica

- ~~minion~~ node worker. Componente de un clúster de k8s
- Puede ser una máquina virtual o física
- Tiene los servicios para ejecutar contenedores
- Es gestionado por los componentes ~~master~~ controller
- Cada nodo tiene:
 - Direcciones: Hostname, IP externa, IP interna
 - Condición: Campo que describe el estado (listo, sin red, sin disco, ...)
 - Capacidad: Describe los recursos disponibles
 - Info: Información general (kernel, sw instalado, versiones)
- Nodo controlador es el que contiene componentes para controlar otros nodos.
- Aunque es poco habitual, es posible tener nodos con Windows Server para ejecutar aplicaciones en contenedores Windows [1]

Componentes del nodo controlador

- **kube-apiserver** Gestiona la API de k8s
- **etcd** Almacén clave-valor que guarda la configuración del clúster
- **kube-scheduler** Selecciona el nodo donde ejecutar los pods
- **kube-controller-manager** Ejecuta los controladores
- **docker/rkt/containerd/...** Ejecuta los contenedores del master
- **cloud-controller-manager** Ejecuta los controladores que interactúan con el proveedor de nube:
 - nodos
 - enrutamiento
 - balanceadores
 - volúmenes

Complementos (addons)

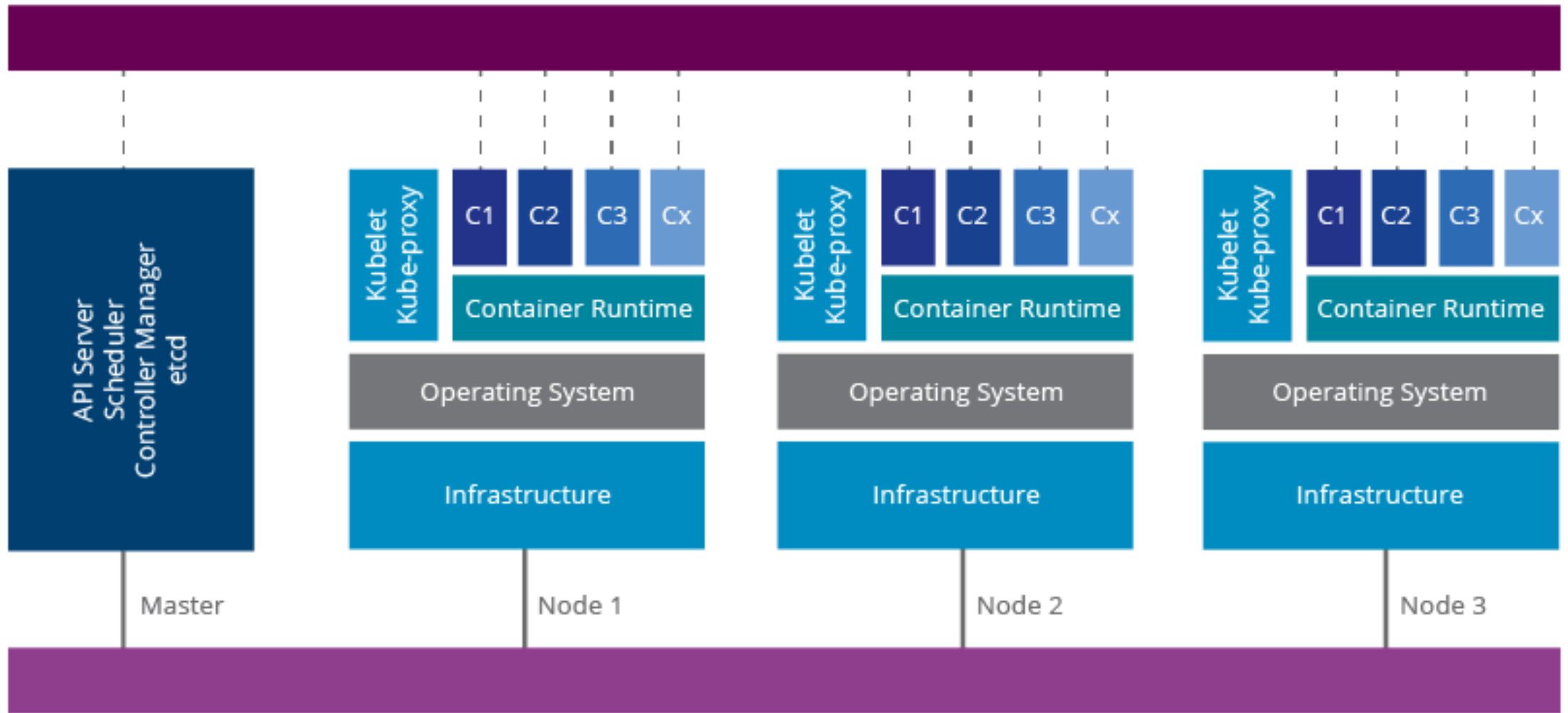
Son pods y servicios que proporcionan funcionalidad al clúster

- **Cluster DNS** Proporciona registros DNS para los servicios de k8s (compatible con el DNS de la organización)
- **Web UI** Interfaz web para el manejo de k8s
- **Container Resource Monitoring** Recoge métricas de forma centralizada. Múltiples opciones: prometheus, sysdig
- **Cluster-level Logging** Almacena los logs de los contenedores

Componentes de los worker

- **kubelet** Vigila los pods asignados a su nodo
- **kube-proxy** Permite la conexión a través de la red
- **docker/rkt/containerd/...** Ejecuta los contenedores
- **supervisord** Monitoriza y controla kubelet y docker
- **fluentd** Proporciona logs de contenedores a cluster-level logging

Overlay Network (Flannel/OpenVSwitch/Weave)



Physical Network

Kubernetes architecture

Alternativas de instalación simple

- Opciones de instalación y uso: <https://kubernetes.io/docs/setup/>
- ¿No se puede instalar desde paquetes de la distribución?

Proyectos propios de k8s	Ecosistema
<u>Minikube</u>	<u>Docker Desktop</u>
<u>Kind: Kubernetes IN Docker</u>	<u>Minishift</u>
<u>Kubeadm</u>	<u>MicroK8s</u>
	<u>k3s</u>

kubectl

- kubectl es la herramienta principal para gestionar los clusters de k8s
- Desarrollada en go
- Paquete debian: <https://packages.debian.org/search?keywords=kubernetes-client>
- Instalación directa de binarios:
<https://kubernetes.io/docs/tasks/tools/install-kubectl/#install-kubectl-binary-with-curl-on-linux>

Configuración de kubectl

- kubectl puede configurarse para gestionar distintos clústers mediante el uso de contextos
- kubectl config
- La configuración se guarda en ~/.kube/config
- Uso de kubectl
- Con “kubectl proxy” puede interactuarse con la API directamente utilizando las credenciales del usuario