

# Diferentes formas de desplegar Kubernetes

# Índice

- 01**   [Introducción](#)
- 02**   [Distribuciones de Kubernetes](#)
- 03**   [Kubernetes Administrado](#)
- 04**   [Kubernetes Cloud](#)
- 05**   [Kubernetes Empresariales](#)

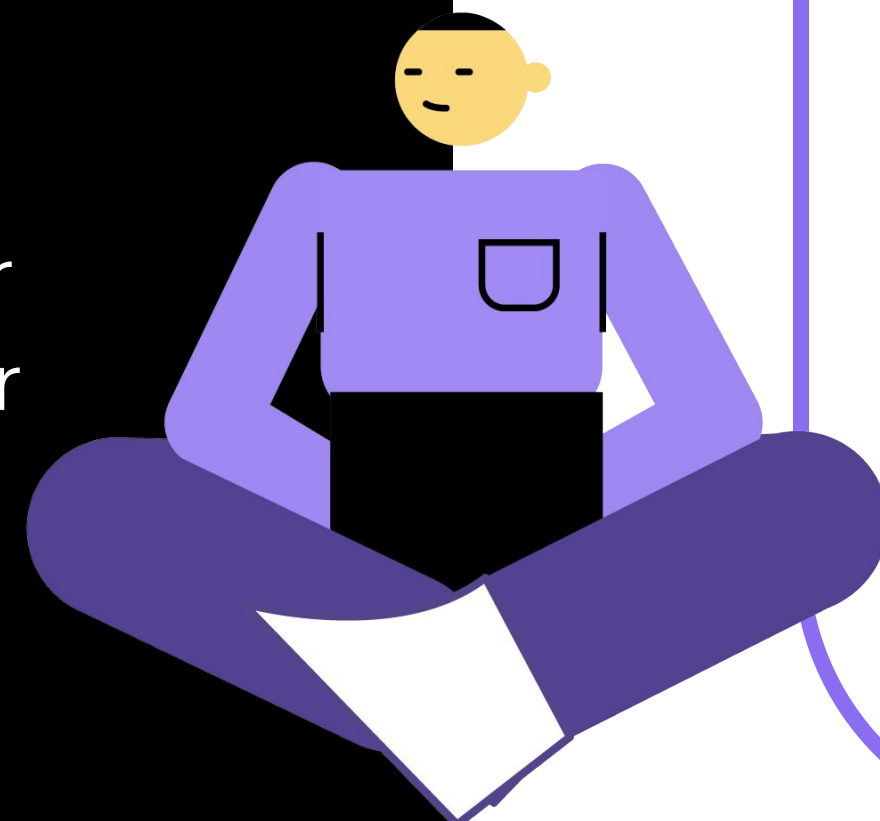


01

# Introducción



Kubernetes se ha convertido en la herramienta más utilizada cuando se necesita una orquestación de contenedores a escala. Es un proyecto muy bien visto, está bien soportado y evoluciona rápidamente. Por otro lado, se conoce que es extenso, complejo y difícil de configurar. No solo eso, sino que gran parte del trabajo pesado se deja al usuario final. El mejor enfoque, por lo tanto, no es agarrar los bits e intentar hacerlo solo, sino buscar una solución de contenedor completa que incluya Kubernetes como un componente soportado y mantenido.



02

# Distribuciones de Kubernetes

# ¿Por qué existen distribuciones de Kubernetes?

A medida que la adopción de Kubernetes se extiende, muchas empresas están descubriendo que, si bien el uso de contenedores y Kubernetes ofrecen ventajas en cuanto a desarrollo, velocidad, agilidad y gestión de costes, también crean un nuevo conjunto de desafíos que hay que superar.

Si bien hay puntos en común en los tipos de desafíos que las empresas encuentran con Kubernetes, no hay una verdadera manera de gestionarlos que sirva para todos, sino más bien una serie de enfoques potenciales. Cada uno con sus propias ventajas e inconvenientes.

Diferentes tipos de distribuciones de Kubernetes han surgido para abordar este problema.

Sin embargo, navegar por estas soluciones para elegir la correcta a menudo es un desafío, ya que no existe una verdadera “talla única”. Cada ruta que se toma para adoptar Kubernetes tiene sus pros y sus contras. Esto se vuelve aún más complicado cuando se considera que lo que podría ser un factor decisivo para una organización, podría no ser un problema para otra, dependiendo del perfil específico de cada empresa.

Entonces, antes de sumergirnos en los desafíos y las soluciones de cada tipo de Kubernetes, exploremos algunas de las consideraciones clave para las empresas, que influyen en qué distribución es la más adecuada para sus necesidades.

# Vanilla Kubernetes

Es el proyecto de código abierto. Es un término acuñado para describir una distribución base de Kubernetes. Solo dispone de los componentes principales para poder ejecutar Kubernetes, pero no mucho más. Incluye los nodos masters, que ejecutan etcd, API server, controller y schedule. También incluye los nodos workers, y cada nodo ejecuta kubelet, kube-proxy y un runtime de contenedores.

Si bien esta instalación básica les dará a los usuarios un control total sobre su instancia de Kubernetes, es verdad que también va a requerir más tiempo para su puesta en marcha. Los usuarios, además de disponibilizar una instancia de Kubernetes, también deberán agregar un balanceador de carga, sistema de escalado automático, monitoreo, gestión de alertas, etc.



El término “Vanilla” se suele utilizar para referenciar a los programas o sistemas operativos que no han sufrido cambios, adaptaciones o actualizaciones con respecto a su versión original.

# Distribución de Kubernetes

## Administrado

- Son aquellas distribuciones que simplifican la instalación y puesta en marcha de Kubernetes. Sus funcionalidades pueden ser extendidas mediante desarrollos propios de la distribución.

## Cloud

- Son las distribuciones de Kubernetes que ofrecen los proveedores de la nube. Estas distribuciones se integran y/o hacen uso de los servicios propios de la nube.

## Empresarial

- Estas distribuciones se asemejan más a un **PaaS**, ya que además del soporte empresarial, brindan distintas funcionalidades, como mejoras de seguridad, procesos de CI/CD integrados, monitoreo, API gateways, etc.



03

# Kubernetes Administrado

# Rancher Labs

Es la compañía de software detrás de Rancher y varios productos más. Crearon la consola de Rancher para administrar los clústeres de Kubernetes y crearon algunas distribuciones de Kubernetes propias, como RKE y K3.

- **RKE (Rancher Kubernetes Engine):** Es una distribución de Kubernetes con certificación CNCF que se ejecuta completamente dentro de los contenedores de Docker. Esto facilita la preparación de un host y la instalación porque lo único que debemos hacer es ejecutar una versión compatible de Docker.



- **K3s:** es una distribución ligera certificada de Kubernetes. Es un único binario de menos de 40 MB y contiene todo lo necesario para ejecutar un clúster de Kubernetes. Incluye etcd, NGINX y todos los componentes de Kubernetes. Está diseñado para ejecutar ubicaciones remotas con recursos limitados o dentro de dispositivos IoT.
- **Rancher:** es un software para administrar clústeres de Kubernetes. Incluye no solo la gestión de clústeres existentes, sino que también la posibilidad de crear nuevos clústeres. Además, facilita el aprovisionamiento y la administración de clústeres de Kubernetes. Se pueden importar clústeres existentes, personalizados o administrados como EKS y GKE o bien definir e implementar nuestros propios clústeres con RKE o K3.

# Minikube

Minikube es una herramienta que facilita la ejecución local de Kubernetes. Ejecuta un clúster de Kubernetes de un solo nodo dentro de una máquina virtual (VM) en nuestro ordenador para que podamos probar Kubernetes o desarrollar con él día a día. Es una herramienta realmente fácil de usar.

## Características principales:

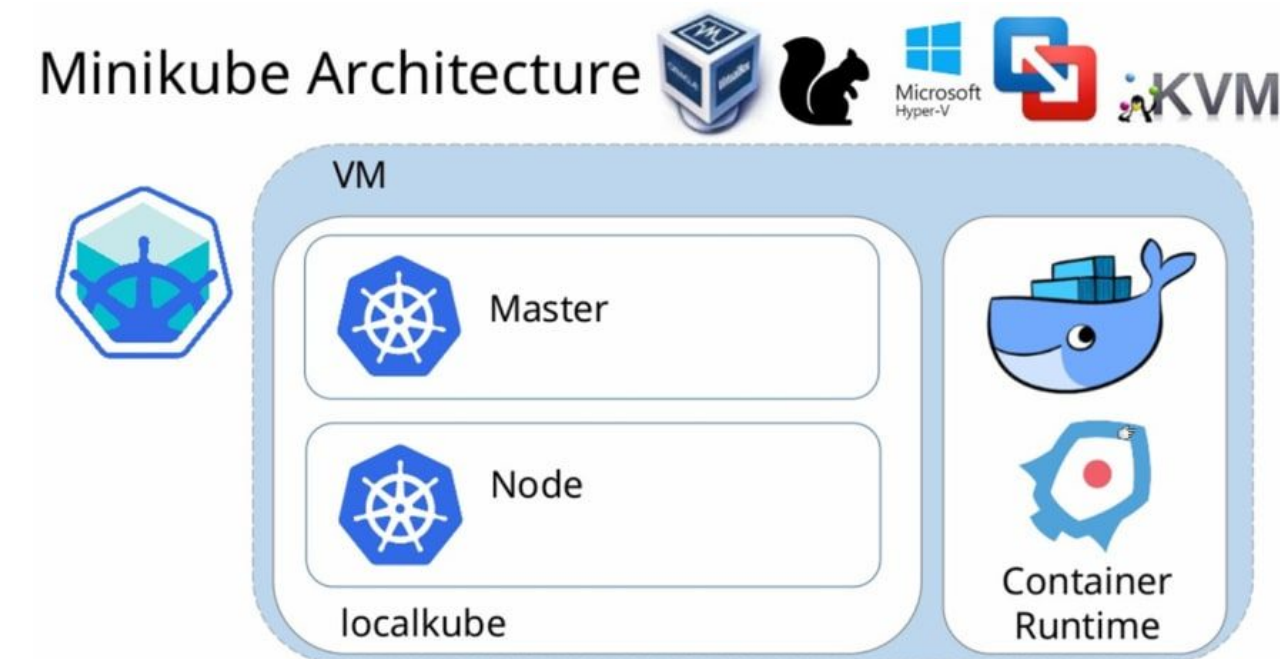
- DNS
- NodePorts
- ConfigMaps and Secrets
- Dashboards
- Container Runtime: Docker, CRI-O, and containerd
- Enabling CNI (Container Network Interface)
- Ingress



## Contras:

- No tiene soporte multinodo.
- Se ejecuta dentro de una máquina virtual.

Minikube tiene un "Dashboard" donde se puede ver todo lo que está pasando dentro del clúster y que ayuda a tener una visión más completa.



04

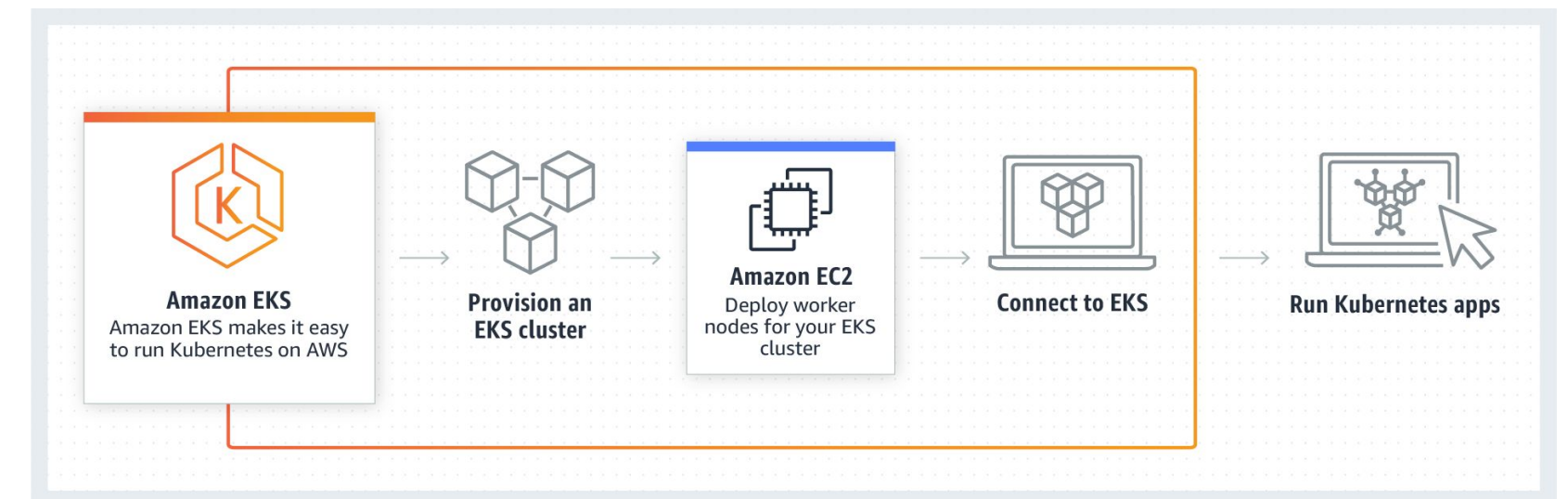
# Kubernetes Cloud

# Amazon EKS

- **Amazon Elastic Kubernetes Service (EKS)** permite ejecutar, iniciar y escalar aplicaciones de Kubernetes en la nube de AWS, así como en las instalaciones. Poco a poco, Amazon EKS está reemplazando a ECS, que es el orquestador propietario de AWS.
- **EKS** automatiza importantes tareas de administración de Kubernetes, como el aprovisionamiento de nodos, parches y actualizaciones. También incluye cifrado y seguridad incorporada, integración con **CloudWatch** para registro, actualización automática, IAM para permisos de acceso y CloudTrail para auditoría. AWS contribuyó a la base de K8s de código abierto para maximizar la funcionalidad para sus usuarios.



- AWS incluso introdujo una nueva distribución K8s de código abierto, conocida como **EKS Distro**, y una nueva opción de implementación para Amazon EKS, conocida como **Amazon EKS Anywhere**. Este nos permite crear y también operar clústeres de Kubernetes en nuestra infraestructura, incluidas máquinas virtuales y bare metal.

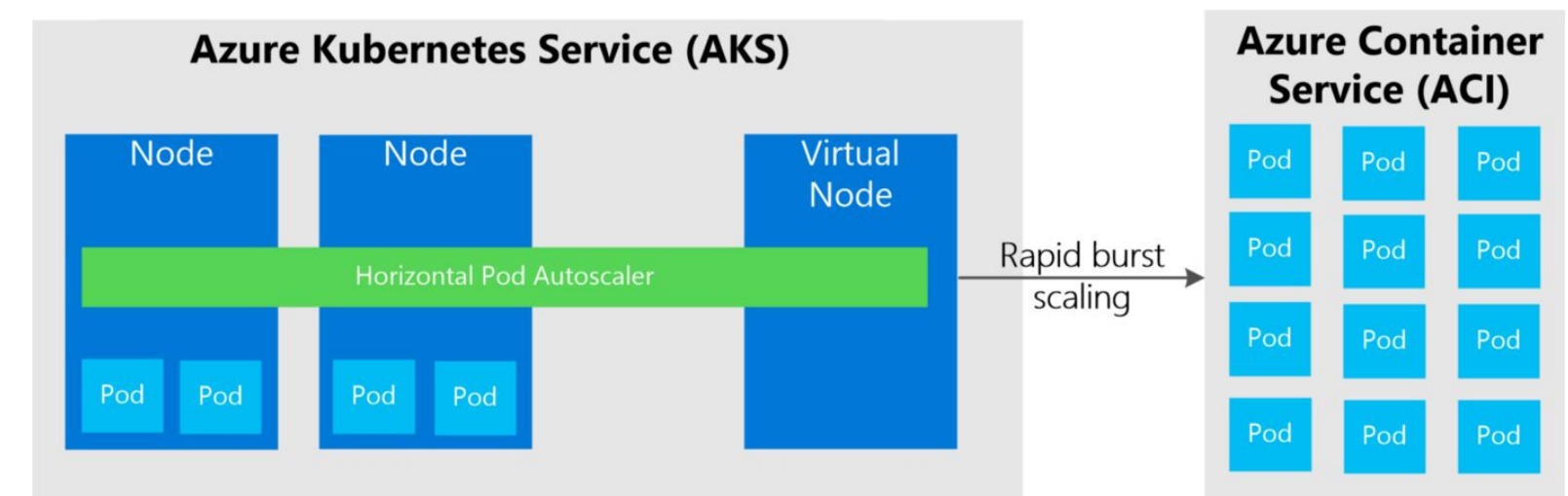


# AKS

- **Microsoft Azure** ofreció originalmente una solución llamada ACS, que admitía Apache Mesos, K8s y Docker Swarm. En octubre de 2017, presentó **Azure Kubernetes Service (AKS)** como un servicio de alojamiento de Kubernetes dedicado, lo que provocó que las otras opciones fueran deprecadas.
- **AKS** ofrece administración de los clústeres K8s de forma gratuita. Está certificado por CNCF como compatible con K8s, lo que significa que no hay modificaciones personalizadas.
- Microsoft invirtió muchos recursos en K8s en general y en **AKS** en particular. **AKS** incluye una sólida integración con ActiveDirectory, que ofrece la supervisión, gestión de registros y los servicios de almacenamiento de Azure. También brinda gestión de los contenedores, redes y nodos.



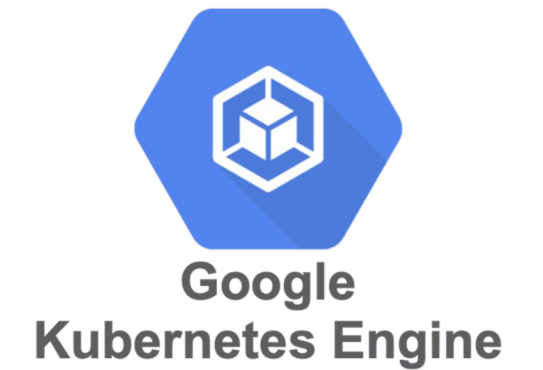
- Una de las características más interesantes de **AKS** es su uso de Azure Container Instances (ACI) como nodos virtuales. Esto permite usar todo el ecosistema de Kubernetes, ejecutar parte del workload en las propias VMs del clúster, pero tener otra parte del workload, la altamente variable, ejecutándose en ACI que se crean y destruyen bajo demanda.



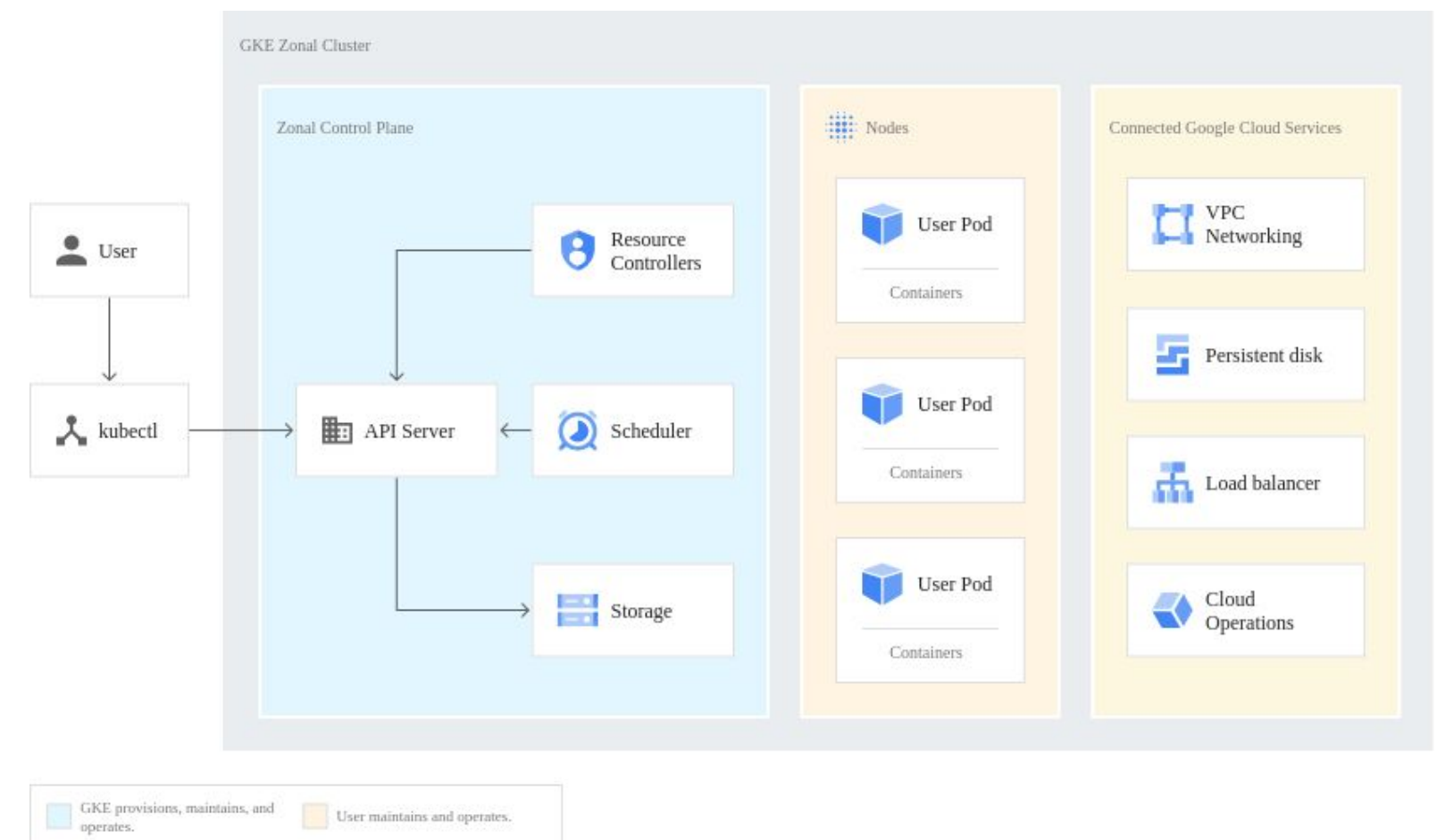


# Google Kubernetes Engine

- **Google Kubernetes Engine** es el primer servicio de Kubernetes administrado basado en la nube disponible en el mercado. **GKE** es un entorno administrado para escalar, implementar y administrar aplicaciones en contenedores en una infraestructura segura de Google.
- Como K8s fue creado por los propios ingenieros de Google para su organización de contenedores interna, se cree que **GKE** es una de las plataformas de Kubernetes más avanzadas disponibles en la actualidad. La plataforma está diseñada para su uso en Google Cloud e incluso se puede implementar en entornos híbridos, así como en las instalaciones.



- Además de facilitar la creación de clústeres, **GKE** también proporciona funciones avanzadas de administración de clústeres, como escalado automático, balanceo de carga, reparación automática, actualizaciones automáticas, registro y supervisión, etc.



05

# Kubernetes Empresariales



# Red Hat

Es una empresa multinacional estadounidense de software que provee software de código abierto, principalmente a empresas.

Es conocida en gran medida por su sistema operativo empresarial **Red Hat Enterprise Linux**.

- Esta plataforma basada en Kubernetes fue desarrollada por **Red Hat**. **OpenShift** permite orquestar contenedores y redes, así como gestionar o asignar recursos escalables y distribuidos. Este software puede desplegarse en múltiples entornos en la nube, entre los que se encuentran tanto las nubes privadas con su propia infraestructura informática como las infraestructuras en la nube de **Amazon AWS** y **Microsoft Azure**



- La gran ventaja de **OpenShift** yace en que proporciona a los equipos de desarrolladores un entorno de trabajo consistente y robusto. Independientemente del entorno que se use (nube pública, privada o híbrida), los desarrolladores tienen a su disposición los mismos enfoques para crear, probar y ejecutar aplicaciones.
- Otro punto fuerte de **OpenShift** son las herramientas de monitoreo y de registro integradas que posee para gestionar el monitoreo y las notificaciones. Además, tiene acceso al **Red Hat Container Catalog**, el cual contiene imágenes de contenedor probadas y certificadas por **Red Hat**.
- OpenShift fue construido para proporcionar abstracción y automatización para todos los componentes propios de un **PaaS**.

¡Muchas gracias!