**Primer desplegament amb Kubernetes**

**Objectius**

* Comprendre els conceptes bàsics de Kubernetes: Pod, Deployment, Service.
* Aprendre a desplegar una aplicació simple amb kubectl.
* Visualitzar l'estat dels recursos desplegats.
* Accedir al servei des del navegador i verificar el funcionament.

**Fase 0: Preparació de l’entorn**

*En aquesta primera fase instal·larem les diferents aplicacions que farem servir. A part de Docker, necessitem també* ***kubectl*** *i* ***Minikube.***

***kubectl*** *és l’eina de línia de comandes oficial de Kubernetes.
Ens permet interactuar amb el clúster: crear, modificar, consultar i eliminar recursos com* ***pods, serveis, deployments****, entre d’altres.*

***Minikube*** *és una eina que ens permet crear i executar un clúster Kubernetes localment en un ordinador o màquina virtual.*

*Simula un entorn real de Kubernetes en una sola màquina, i per tant és ideal per fer* ***proves, pràctiques i formació*** *sense necessitat d’un entorn al núvol ni infraestructura complexa.*

1. Instal·lació de Docker (si no el teniu ja)

sudo apt update

sudo apt install -y docker.io

Afegir l’usuari al grup docker:

sudo usermod -aG docker $USER

newgrp docker

Verificar que Docker funciona:

docker version

2. Instal·lació de kubectl

curl -LO "https://dl.k8s.io/release/v1.29.3/bin/linux/amd64/kubectl"

chmod +x kubectl

sudo mv kubectl /usr/local/bin/

kubectl version --client

3. Instal·lació de Minikube

curl -LO https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64

chmod +x minikube-linux-amd64

sudo mv minikube-linux-amd64 /usr/local/bin/minikube

minikube version

4. Iniciar Minikube amb Docker com a driver

minikube start --driver=docker

Verificar que el clúster està actiu:

kubectl get nodes

kubectl cluster-info

**Fase 1: Desplegament d’una aplicació amb Kubernetes**

1. Crear un Deployment amb nginx

kubectl create deployment web-nginx --image=nginx

*Crea un* ***Deployment*** *anomenat web-nginx a partir de la imatge nginx.*

*Kubernetes s’encarrega de crear un* ***Pod*** *(la unitat bàsica d’execució) amb un contenidor que corre nginx.*

2. Comprovar que s’ha creat correctament

kubectl get deployments

kubectl get pods

*Aquí haurem d’esperar una mica a que es creï el contenidor, esperarem a que estigui actiu:*



Fase 2: Exposar l’aplicació com a servei

1. Exposar el Deployment com a NodePort

kubectl expose deployment web-nginx --port=80 --type=NodePort

*Aquesta comanda crea un recurs de tipus* ***Service*** *per al Deployment web-nginx.
Un* ***Service*** *en Kubernetes actua com una porta d’entrada cap a una aplicació desplegada dins el clúster, permetent-ne l’accés.*

***Explicació dels paràmetres***

* *expose deployment web-nginx
Indica que volem exposar l’aplicació desplegada amb el nom web-nginx.*
* *--port=80
Estem indicant que l’aplicació escolta al port 80 (el port estàndard de nginx).*
* *--type=NodePort
Aquest paràmetre crea un tipus de servei que permet accedir a l’aplicació des de fora del clúster, mitjançant un port que s’obrirà al node local.*

**Com comprovar el servei creat**

kubectl get services

Apareixerà una línia com:

****

**Accedir a l’aplicació des del navegador**

Pots fer-ho amb la següent comanda:

kubectl port-forward service/web-nginx 8080:80

*Aquesta comanda crea una connexió temporal entre el port 8080 del teu ordinador i el port 80 del servei web-nginx que corre dins del clúster Kubernetes creat amb Minikube.*

*Tot i que el clúster també s’està executant localment, aquest entorn està aïllat en contenidors. Aquesta comanda ens permet accedir-hi fàcilment des del navegador del sistema amfitrió, com si l’aplicació estigués escoltant directament al port 8080 de l’ordinador.*

Obrir el navegador i visitar:

http://localhost:8080

Això et permetrà veure la pàgina de benvinguda de nginx desplegada dins del teu clúster Kubernetes local.

