

[kubectl](#), [comandos](#), [kubernetes](#), [cluster](#), [pod](#), [nodo](#)

## Comandos de Kubectl

<https://kubernetes.io/docs/reference/kubectl/cheatsheet/>

- para conocer el estado del cluster kubernetes → kubectl status
- para saber la versión → kubectl version
- ver la configuración → kubectl config view
- kubectl run → para crear y ejecutar un pod
- kubectl get pods → Para ver los pods de nuestro cluster
- kubectl api-resources → Listado de todos los comandos que permite la API
- kubectl cluster-info → información del cluster
- kubectl get pods -n kube-system → pods del sistema
- kubectl get deploy,rs,service,pods → es igual a un kubectl get all
- kubectl api-resources → información sobre la API de kubernetes
- kubectl get service → lista de los servicios
- kubectl get deployments → lista de deployments
- kubectl get namespaces → lista de namespaces
- kubectl delete service <nombre\_servicio> → eliminar servicio <nombre\_servicio>
- kubectl delete deployment <nombre\_del\_deployment> → eliminar deployment <nombre\_del\_deployment>
- kubectl apply -f deployment.yaml → aplicar el contenido del fichero deployment.yaml
- kubeadm token list → listar los tokens
- exponer un deployment → kubectl expose deployment <mi-deployment> -port=80 -type=NodePort
- escalar a 3 replicas un deployment → kubectl scale -replicas=3 deployment <mi-deployment> -n <mi-namespace>
- crear un secret → kubectl create secret generic mysql-pass --from-literal=password=<mi-contraseña>

==== Comando para PODs

### Crear un POD

Hay dos formas de crear pods. De forma imperativa y de forma declarativa

#### Imperativa

Ejecutamos el comando para crearlo. Hasta la versión 1.17 se debe usar la forma siguiente

```
kubectl run --generator=run-pod/v1 <nombre pod> --image=<nombre imagen>
created
```



<https://v1-17.docs.kubernetes.io/docs/reference/kubectl/conventions/#generators>

Apartir de la versión 1.18 la parte de generator está en deshuso y ahora se utiliza

```
kubectl run <nombre pod> --image=<nombre de la imagen del repositorio Docker Hub >
```

Por ejemplo para crear un pod con una imagen de un nginx basado en alpine

```
kubectl run podtest --image=nginx:alpine
```

Para crear un pod temporal que se elimine al salir del shell

```
kubectl run --rm -ti podtest --image=nginx:alpine -- sh
```

## **Declarativa**

Definimos un archivo en formato yaml para después aplicarlo

```
kubecttl apply -f nombre_fichero.yaml
```

## **Listar los pods que tenemos**

```
kubectol get pods
```

Listar los pods de todos los namespaces

```
kubectl get pods --all-namespaces -o wide
```

Listar los pods de un determinado namespace

```
kubectl get pods -n <nombre_namespaces>
```

## **Ver una descripción del pod**

Nos permite ver los eventos de un pod y su descripción

```
kubectl describe pod <nombredelpod>
```

## **Etiquetar un pod**

```
kubectl label pods <nombredelpod> app=mietiqueta
```

## Recursos que consume un pod

```
kubectl top pod <nombre pod>
```

## Eliminar un pod

```
kubectl delete pod <nombredelpod>
```



en nombre del pod podemos poner uno o varios nombres separados por espacio

Si los pods lo hemos creado desde un manifiesto. Para borrarlos lo mejor es ejecutar

```
kubectl delete -f <nombremanifiesto>
```

## Entrar al shell de nuestro pod

```
kubectl exec -ti <nombredelpod> --sh
```

En el caso de que tengamos un pod con dos contenedores en su interior y queramos entrar al shell de primer contenedor

```
kubectl exec -ti <nombredelpod> -c <nombrecontenedor> --sh
```

## Ver los logs de un pod

```
kubectl logs <nombredelpod>
```

Si tenemos mas de un contenedor en el pod

```
kubectl logs <nombredelpod> -c <nombredelcontenedor>
```

si son pods del system , por ejemplo del driver kube-flannel

```
kubectl logs -p kube-flannel-ds-xcl6f -n kube-system
```

## Ejecutar un comando sobre un contenedor

```
kubectl exec <nombredelpod> -c <nombredelcontenedor> <comando>
```

### Copiar un fichero a un contenedor

```
kubectl cp /tmp/archivoprueba <nombredelpod>:/var/www/html -c  
<nombredelcontenedor>
```

### Acceder a un puerto dentro de un pod

```
kubectl port-forward --address 0.0.0.0 pod/<pod-name> <host-port>:<pod-port>
```

## Trabajar con Cluster

- Para obtener información sobre el cluster → **kubectl cluster-info**
- Información sobre los nodos → **kubectl get nodes**
- descripción de un nodo → **kubectl describe node <nombre del nodo>**
- ver el consumo de un nodo → **kubectl top node <nombre nodo>**
- Poner un nodo fuera de servicio (para actualizar por ejemplo) → **kubectl uncordon <nombre nodo>**
- Para que quite todos los pods etc que ya existieran en este nodo → **kubectl drain <nombre nodo>**
- Para volver a ponerlo como disponible → **kubectl cordon <nombre nodo>**

## Referencias

- <https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands>
- <https://www.albertcoronado.com/2021/06/08/tutorial-kubernetes-i/>

From:  
<http://wiki.intrusos.info/> - LCWIKI

Permanent link:  
<http://wiki.intrusos.info/doku.php?id=virtualizacion:kubernetes:comandos&rev=1627380295>

Last update: 2023/01/18 14:22

