



Apasoft Training

[www.apasoft-training.com](http://www.apasoft-training.com)

# Kubeconfig

---

KUBERNETES



# Kubeconfig

---

## ☐ Kubeconfig

- ☐ Fichero en formato YAML que kubectl usa para obtener la configuración de acceso a los cluster de Kubernetes.
- ☐ La ubicación de este fichero se puede configurar mediante :
  - ☐ La opción - -kubeconfig con el comando kubectl
  - ☐ La variable de entorno \$KUBECONFIG
- ☐ Por defecto suele residir en el directorio del usuario actual, dependiendo del Sistema Operativo



# Kubeconfig

---

❑ ¿Dónde se suele encontrar el fichero?

❑ Linux

`~/.kube/config`

❑ Windows

`%USERPROFILE%\kube\config`

❑ Mac

`~/.kube/config`



# Kubeconfig

---

☐ ¿Qué contiene el fichero Kubeconfig para determinar la información de un cluster ?

☐ Certificado

☐ Servidor

☐ Nombre

☐ Usuario

☐ Token de usuario



# Kubeconfig

## □ Ejemplo de formato

```
apiVersion: v1
kind: Config
preferences: {}
```

```
clusters:
- cluster:
  name: desarrollo
- cluster:
  name: produccion
```

```
users:
- name: usu1
- name: desa1
```

```
contexts:
- context:
  name: desa
- context:
  name: desa1
- context:
  name: prod1
```



**Datos del cluster**

# Kubeconfig

## □ Ejemplo de formato

```
apiVersion: v1
kind: Config
preferences: {}
```

```
clusters:
- cluster:
  name: desarrollo
- cluster:
  name: produccion
```

```
users:
- name: usu1
- name: desa1
```

```
contexts:
- context:
  name: desa
- context:
  name: desa1
- context:
  name: prod1
```



Usuarios



# Kubeconfig

## □ Ejemplo de formato

```
apiVersion: v1  
kind: Config  
preferences: {}
```

```
clusters:  
- cluster:  
  name: desarrollo  
- cluster:  
  name: produccion
```

```
users:  
- name: usu1  
- name: desa1
```

```
contexts:  
- context:  
  name: desa  
- context:  
  name: desa1  
- context:  
  name: prod1
```



**Contextos-→ Cluster/usuario**



# Kubeconfig

## ❑ Cluster

- ❑ La clausula clúster contiene EndPoints de un clúster de kubernetes.
- ❑ Esto incluye la URL completa para el servidor apiserver de kubernetes, así como la autoridad de certificación del clúster
- ❑ También podemos configurar la propiedad “insecure-skip-tls-verify: true”, si el certificado de servicio del clúster no está firmado por una entidad de certificación de confianza del sistema.
- ❑ Tiene un nombre que actúa como clave de diccionario para el clúster dentro de este archivo kubeconfig.
- ❑ Se pueden agregar o modificar entradas de clúster mediante **kubectl config set-cluster**

```
clusters:  
- cluster:  
  certificate-authority: /fichero-ca  
  server: https://produccion.empresa:4443  
  name: produccion  
- cluster:  
  insecure-skip-tls-verify: true  
  server: https://desarrollo:443  
  name: desarrollo
```





# Kubeconfig

## ❑ Usuarios

- ❑ Un usuario define las credenciales de cliente para autenticarse en un clúster de kubernetes.
- ❑ Un usuario tiene un nombre que actúa como su clave.
- ❑ Las credenciales disponibles son certificado de cliente, clave de cliente, token y nombre de usuario/contraseña.
- ❑ El nombre de usuario/contraseña y el token son mutuamente excluyentes, pero los certificados y claves del cliente se pueden combinar con ellos.
- ❑ Se pueden agregar o modificar entradas de usuario mediante **kubectl config set-credentials**.

users:

- name: desarrollo1

user:

token: token1

- name: desarrollo2

user:

client-certificate: path/al certificado

client-key: path/a clave de cliente



# Kubeconfig

## Contextos

- Un contexto define un conjunto de clúster, usuario y namespace que se utiliza para enviar solicitudes al clúster.
- Cada uno de los tres es opcional; es válido especificar un contexto con solo un valor de clúster, usuario, espacio de nombres o no especificar ninguno.
- Los valores no especificados o los valores con nombre que no tengan las entradas correspondientes en el kubeconfig se reemplazarán por el valor predeterminado.
- Se puede agregar o modificar entradas de contexto con **kubectl config set-context**.

```
contexts:  
- context:  
  cluster: produccion  
  namespace: trabajo  
  user: desarrollo1  
  name: context-desarrollo
```



# Kubeconfig

---

## ☐ Contexto actual

- ☐ El contexto actual permite identificar el contexto concreto que se utilizará cuando nos indique ningún otro valor
- ☐ Se puede cambiar el contexto actual con el comando **kubectl config use-context**

## ☐ Ver la configuración actual

- ☐ Para ver la configuración que tenemos en este momento se utiliza el comando **kubectl config view**



# Kubeconfig

## ❏ Ejemplo de comando config

```
kubectl config set-credentials myself --username=admin --password=secret  
kubectl config set-cluster local-server --server=http://localhost:8080  
kubectl config set-context default-context --cluster=local-server --user=myself  
kubectl config use-context default-context  
kubectl config set contexts.default-context.namespace the-right-prefix  
kubectl config view
```

```
apiVersion: v1  
clusters:  
- cluster:  
  server: http://localhost:8080  
  name: local-server  
contexts:  
- context:  
  cluster: local-server  
  namespace: the-right-prefix  
  user: myself  
  name: default-context
```

```
current-context: default-context  
kind: Config  
preferences: {}  
users:  
- name: myself  
  user:  
    password: secret  
    username: admin
```