

Watch:  **Deploy de Prometheus y Grafana en K8S**

05:20

Usamos Helm para realizar un deploy de Prometheus y Grafana en nuestro cluster K8S local. Además levantaremos una instancia de la App de ejemplo en Go.

```
helm install --name prometheus --set-file  
extraScrapeConfigs=prometheus/extraScrapeConfigs.yaml  
stable/prometheus  
helm install --name grafana stable/grafana  
helm install --name app app
```

Al hacer `kubectl get svc` deberías ver:

NAME	EXTERNAL-IP	PORT(S)	AGE	TYPE	CLUSTER-IP
app	<none>	8081/TCP	4m35s	ClusterIP	10.107.177.238
grafana	<none>	80/TCP	4m35s	ClusterIP	10.98.66.204
kubernetes				ClusterIP	10.96.0.1

<none>	443/TCP	58d		
prometheus-alertmanager			ClusterIP	10.110.111.190
<none>	80/TCP	4m38s		
prometheus-kube-state-metrics			ClusterIP	None
<none>	80/TCP	4m38s		
prometheus-node-exporter			ClusterIP	None
<none>	9100/TCP	4m38s		
prometheus-pushgateway			ClusterIP	10.100.225.138
<none>	9091/TCP	4m38s		
prometheus-server			ClusterIP	10.106.178.224
<none>	80/TCP	4m38s		

Haremos *port forward* para poder acceder a los servicios desde nuestro Host (tendrás que ejecutar cada comando en una terminal diferente ya que no se ejecuta de forma *detached*)

```
kubectl port-forward svc/prometheus-server 9090:80
kubectl port-forward svc/grafana 8080:80
kubectl port-forward svc/app 8081:8081
```

Para poder a Grafana necesitaremos el *password* del usuario admin, el cual se genera de forma aleatoria. Para conseguir el password hacemos:

```
kubectl get secret grafana -o jsonpath="{.data.admin-password}" | base64 --decode; echo
```