



Universidad de Guadalajara.
Centro Universitario de Ciencias Exactas e Ingenierías.



Ejercicio 10-1

Microservicios con Docker

Computación tolerante a fallas NRC 179961

Nombre del alumno: Loredó Padilla Orlando Javier

Código: 217560328

Departamento: Departamento de ciencias computacionales

Nombre del maestro: López Franco Michel Emanuel

Carrera: Ingeniería en computación

Sección: D06

Ciclo: 23B

Fecha: 20/11/2023.

Introducción

Genera una aplicación utilizando kubernetes.

Preguntas

¿Qué es Kubernetes?

Kubernetes es una plataforma portable y extensible de código abierto para administrar cargas de trabajo y servicios. Kubernetes facilita la automatización y la configuración declarativa. Tiene un ecosistema grande y en rápido crecimiento. El soporte, las herramientas y los servicios para Kubernetes están ampliamente disponibles.

¿Qué es Ingress?

Es un servicio que equilibra cargas de trabajo de tráfico de red en el clúster reenviando solicitudes públicas o privadas a sus apps. Puede utilizar Ingress para exponer varios servicios de app a la red privada o pública mediante un único dominio privado o público.

¿Qué es un LoadBalancer?

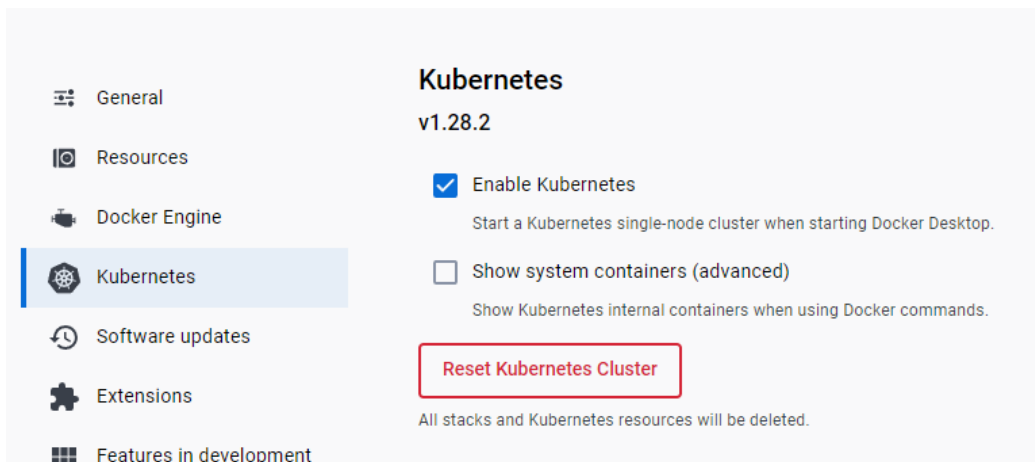
Es un balanceador de carga distribuye el tráfico entrante de aplicaciones entre varios destinos, tales como instancias EC2, en varias zonas de disponibilidad. Esto aumenta la disponibilidad de la aplicación. Puede agregar uno o varios agentes de escucha al balanceador de carga.

Programa

Para esta práctica utilizaremos el programa de prueba que nos ofrece la documentación de Docker, con el fin de agilizar el proceso. Este el código del Dockerfile.

```
Dockerfile > ...
1  FROM node:18-alpine
2
3  WORKDIR /app
4
5  COPY package*.json ./
6
7  COPY ./src ./src
8  COPY ./public ./public
9
10 RUN npm install \
11     && npm install -g serve \
12     && npm run build \
13     && rm -fr node_modules
14
15 EXPOSE 3000
16
17 CMD [ "serve", "-s", "build" ]
```

Ahora tendremos que ir a la aplicación de Docker a configuración y habilitar la aplicación de los kubernetes.



Con eso hecho deberemos de crear los dos siguientes archivos yaml para usar los kubernetes.

```
mi-aplicacion-deploymet.yaml
1  apiVersion: apps/v1
2  kind: Deployment
3  metadata:
4    name: mi-aplicacion-deployment
5  spec:
6    replicas: 3
7    selector:
8      matchLabels:
9        app: welcome-to-docker
10  template:
11    metadata:
12      labels:
13        app: welcome-to-docker
14    spec:
15      containers:
16      - name: welcome-to-docker-container
17        image: welcome-to-docker:latest
18        ports:
19        - containerPort: 3000
```

```
mi-aplicacion-service.yaml.lml
1  apiVersion: v1
2  kind: Service
3  metadata:
4    name: mi-aplicacion-service
5  spec:
6    selector:
7      app: welcome-to-docker
8    ports:
9      - protocol: TCP
10      port: 80
11      targetPort: 80
12    type: LoadBalancer
```

Y el último paso sería ejecutar los comandos para aplicar los kubernetes.

```
C:\Users\orlan\welcome-to-docker-main>kubectl apply -f mi-aplicacion-deployment.yaml
```

```
C:\Users\orlan\welcome-to-docker-main> kubectl apply -f mi-aplicacion-service.yaml
```

Conclusión

Los kubernetes son una manera interesante de expandir los contenedores que creamos de manera rápida y fácil, en este caso al ser únicamente ser para fines académicos no logramos aprovecharlos al máximo, pero cuando se pueda aplicar esto a gran escala será demasiado útil al poder tener una mayor seguridad en nuestros contenedores y evitar problemas como perder el servicio de estos cuando se caiga uno ya que estarán trabajando otros en paralelo que servirán de respaldo.

Referencias

- S/A (2022). ¿Qué es Kubernetes? En Kubernetes. Recuperado el 15/11/2023 en: <https://kubernetes.io/es/docs/concepts/overview/what-is-kubernetes/>