

Probes

Los **probes** son como comprobaciones que kubernetes nos suministra para que podamos comprobar entre otras cosas, si un pod está listo para recibir tráfico, si tienen algún problema, etc

Tenemos tres tipos:

- Readiness Probe → Se utiliza para saber cuando un contenedor está listo para aceptar tráfico
- Liveness Probe - > Se usa liveness probe para saber cuando hay que reiniciar un contenedor
- Startup Probe → Se utiliza para saber cuando la aplicación de un container se ha iniciado. (por ejemplo si es una aplicación que tarda mucho en cargar)

Readiness Probe

Liveness Probe

Con liveness probe revisamos el estado del contenedor y si el estado del contenedor no es el correcto Kubernetes lo reiniciará de forma automática

Ejemplo de liveness probe por http

```
apiVersion: v1
kind: Pod
metadata:
  labels:
    test: liveness
  name: liveness-http
spec:
  containers:
  - name: liveness
    image: k8s.gcr.io/liveness
    args:
    - /server
    livenessProbe: #hace una petición por el puerto 8080 en espera de
    obtener un valor de 200
      httpGet:
        path: /healthz
        port: 8080
        httpHeaders:
        - name: Custom-Header
          value: Awesome
      initialDelaySeconds: 3
      periodSeconds: 3
```

Ejemplo de liveness probe por comando

```
apiVersion: v1
kind: Pod
metadata:
  labels:
    test: liveness
  name: liveness-exec
spec:
  containers:
  - name: liveness
    image: k8s.gcr.io/busybox
    args:
    - /bin/sh
    - -c
      #creamos un fichero , espera 30 sg, lo borra y vuelve a esperar 600s
      touch /tmp/healthy; sleep 30; rm -rf /tmp/healthy; sleep 600
    livenessProbe:      # LA prueba se hace un cat del fichero que
      # creamos anteriormente cada 5 s
      exec:
        command:
        - cat
        - /tmp/healthy
      initialDelaySeconds: 5  #Tiempo de espera antes de empezar las
      # comprobaciones
      periodSeconds: 5      #Cada cuanto tiempo hace las comprobaciones
```



Un uso común de ejecutar un comando sería controlar el espacio disponible por si nuestro container se queda sin espacio se reinicie y kubernetes lo crea de nuevo

ejemplo de liveness probe por tcp

```
apiVersion: v1
kind: Pod
metadata:
  name: goproxy
  labels:
    app: goproxy
spec:
  containers:
  - name: goproxy
```

```
image: k8s.gcr.io/goproxy:0.1
ports:
- containerPort: 8080
readinessProbe:
  tcpSocket:
    port: 8080
  initialDelaySeconds: 5
  periodSeconds: 10
livenessProbe:
  tcpSocket:
    port: 8080
  initialDelaySeconds: 15
  periodSeconds: 20
```

Startup Probe

Referencias

- <https://github.com/ricardoandre97/k8s-resources/tree/master/probes>

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

Permanent link:
<http://wiki.intrusos.info/doku.php?id=virtualizacion:kubernetes:probes&rev=1604172887>

Last update: **2023/01/18 14:22**

