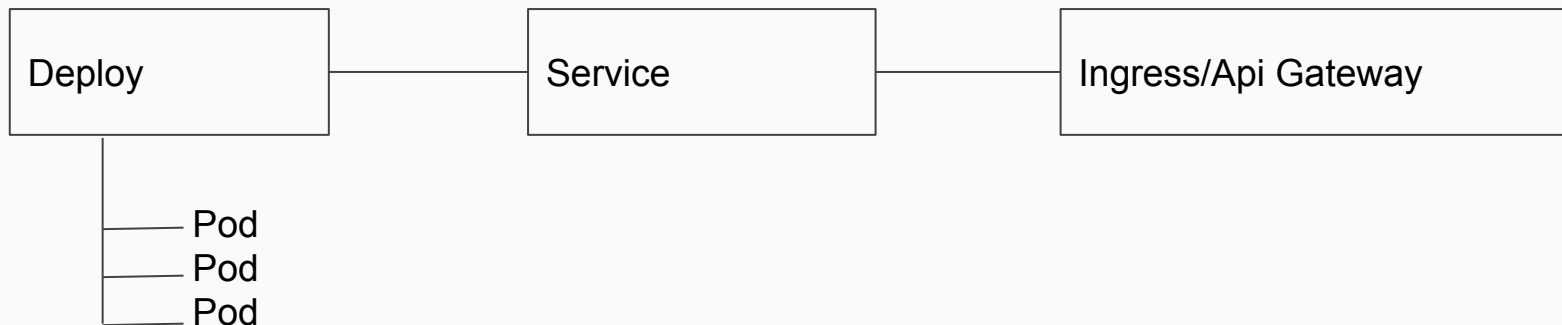


Usando readiness e liveness probes no kubernetes

George Tavares

Arquiteto de Software - Softplan - UNJ - PJ

Estrutura de um deploy no Kubernetes



Cada pod iniciando potencialmente pode fazer parte do serviço

Como saber se o container esta pronto para o serviço ?

Como saber se devemos reiniciar o pod?

Healthcheck DOCKER

```
HEALTHCHECK [OPTIONS] CMD command
```

- `--interval=DURATION` (default: 30s)
- `--timeout=DURATION` (default: 30s)
- `--start-period=DURATION` (default: 0s)
- `--retries=N` (default: 3)

Por que no Dockerfile? Dockerfile é para criar imagens, não para definir como é o ambiente para rodar a imagem.

- Liveness - é um probe que verifica se a aplicação não está quebrada. Após um período longo uma aplicação pode entrar em um estado quebrado por diversos motivos, e devemos forçar um reinício do serviço
- Readiness - é um probe que identifica do estado atual da aplicação em relação a poder receber tráfego. As vezes uma aplicação está sobrecarregada, ou às vezes está realizando uma rotina de manutenção/checagem, e deseja parar de receber dados até terminar a rotina.
- Em ambas as situações o k8s realiza essas checagem e atualiza a lista do SVC para balancear para os pods corretos.

- Falha no Readiness.
- Kubernetes para de rotear trafego para o pod.
- Falha do Liveness
- Kubernetes reinicia o container*.
- Sucesso no Readiness.
- Kubernetes começa a rotear trafego para o container novamente.

Tipos de checagem

A checagem pode se dar:

- Endpoint HTTP
- Endpoint TCP
- Comando dentro do container

Parametros

- `initialDelaySeconds`: Tempo depois do container iniciar para começar as checagem
- `periodSeconds`: Intervalo entre as checagem
- `timeoutSeconds`: Timeout da checagem
- `successThreshold`: Número de checagens consecutivas para marcar o container como Live/Ready
- `failureThreshold`: Número de checagens consecutivas para marcar o container como Unalive/Unready

No caso http check, ainda pode se marcar o verbo que queremos, `httpHeaders`, `path`, `port`

Exemplo

readinessProbe:

exec:

command:

- cat

- /tmp/ready

initialDelaySeconds: 5

periodSeconds: 5

livenessProbe:

httpGet:

path: /live

port: 8080

initialDelaySeconds: 5

initialDelaySeconds: 5

periodSeconds: 5

Perguntas

Obrigado

<https://github.com/tanquetav/liveready>