

Chapter 11 part 2

Accessing an application

`minikube ip` gets the IP address of the Minikube VM that the application is running on

Use the port for `web-service` with the ip to access the application



Alternatively, we can use minikube command:

```
minikube service web-service
```

Liveness and Readiness Probes

- Implementing Liveness and Readiness Probes allows the kubelet to control the health of the app running inside a Pod's container.
- When apps become unresponsive or delayed at startup, kubelet can force a container restart.
- When defining the probes, allow Readiness Probe to fail a few times before the liveness probe kicks in.
- If Readiness and Liveness probes overlap, there is a risk that the container never reaches ready state

Liveness

- Use a Liveness Probe to restart container with unresponsive application
- Liveness Probe checks on an application's health
- Upon health check fail, kubelet restarts the container automatically

```
apiVersion: v1
kind: Pod
metadata:
  labels:
    test: liveness
```

```

name: liveness-exec
spec:
  containers:
  - name: liveness
    image: k8s.gcr.io/busybox
    args:
    - /bin/sh
    - -c
    - touch /tmp/healthy; sleep 30; rm -rf /tmp/healthy; sleep 600
    livenessProbe:
      exec:
        command:
        - cat
        - /tmp/healthy
      initialDelaySeconds: 5
      periodSeconds: 5

```

- Checks health every 5 seconds using `periodSeconds` param
- if it cannot `cat /tmp/healthy` in 5 seconds, it will restart the container

```
kubelet create -f exec-liveness.yaml
```

```
kubelet get pods
```

```
kubelet describe pod liveness-exec
```

- After waiting for some time, it will have a warning-unhealthy status and restart the container

```

livenessProbe:
  httpGet:
    path: /healthz
    port: 8080
    httpHeaders:
    - name: X-Custom-Header
      value: Awesome
  initialDelaySeconds: 3
  periodSeconds: 3

```

- Sends HTTP GET request to the `/healthz` endpoint on port 8080
- If it returns a failure, it means that the application is unresponsive, therefore it kicks off a container restart

```
livenessProbe:
  tcpSocket:
    port: 8080
  initialDelaySeconds: 15
  periodSeconds: 20
```

- kubelet attempts to open the TCP socket to the container (that is running the app)

Readiness Probes

- Check pre-conditions before serving traffic
- A Pod with containers that do not report ready status will not receive traffic from K8s Services

```
readinessProbe:
  exec:
    command:
      - cat
      - /tmp/healthy
  initialDelaySeconds: 5
  periodSeconds: 5
```

- Set up similarly to the Liveness Probe