<https://blog.colinbreck.com/kubernetes-liveness-and-readiness-probes-how-to-avoid-shooting-yourself-in-the-foot/#fn1>

Liveness Probe -> Helps the kubelet when to restart the pod/container.

exec: Execute command for container status.

httpGet: HTTP GET Request for confirm container status.

tcpSocket: TCP Port check to confirm container status

Rediness Probe -> Help the kubelet when to allow the traffic to the PODs.

exec: Execute command for container status.

httpGet: HTTP GET Request for confirm container status.

tcpSocket: TCP Port check to confirm container status

There's no way to trigger pod restart within a readiness probe. As it was recommended in the comments, you should rely on liveness probe instead.

---

apiVersion: apps/v1

kind: Deployment

metadata:

name: nginx-deployment

labels:

app: nginx

spec:

replicas: 3

selector:

matchLabels:

app: nginx

template:

metadata:

labels:

app: nginx

spec:

containers:

- name: nginx

image: sreeharshav/testcontainer:v1

ports:

- containerPort: 80

readinessProbe:

initialDelaySeconds: 30

periodSeconds: 5

timeoutSeconds: 10

successThreshold: 1

failureThreshold: 3

httpGet:

path: /

port: 80

livenessProbe:

initialDelaySeconds: 60

periodSeconds: 5

timeoutSeconds: 10

successThreshold: 1

failureThreshold: 1

httpGet:

path: /

port: 80

---

kind: Service

apiVersion: v1

metadata:

name: myservice

labels:

app: nginx

spec:

selector:

app: nginx

type: NodePort

ports:

- name: name-of-the-port

port: 8000

targetPort: 80

root@ip-192-168-1-100:~# cat K8SB1-Redines-2.yml

apiVersion: apps/v1

kind: Deployment

metadata:

name: nginx-deployment

labels:

app: nginx

spec:

replicas: 3

selector:

matchLabels:

app: nginx

template:

metadata:

labels:

app: nginx

spec:

containers:

- name: nginx

image: sreeharshav/testcontainer:v1

ports:

- containerPort: 80

readinessProbe:

initialDelaySeconds: 30

periodSeconds: 5

timeoutSeconds: 10

successThreshold: 1

failureThreshold: 3

httpGet:

path: /

port: 80

---

kind: Service

apiVersion: v1

metadata:

name: myservice

labels:

app: nginx

spec:

selector:

app: nginx

type: NodePort

ports:

- name: name-of-the-port

port: 8000

targetPort: 80

Docekrfile:

FROM sreeharshav/rollingupdate:v3

CMD sleep 30 && /usr/sbin/nginx -g "daemon off;"

sreeharshav/k8srlp:v2

apiVersion: apps/v1

kind: Deployment

metadata:

labels:

run: nginx01

name: nginx01

spec:

replicas: 1

selector:

matchLabels:

run: nginx01

template:

metadata:

labels:

run: nginx01

spec:

containers:

- image: sreeharshav/k8srlp:v2

name: nginx01

readinessProbe:

exec: #exec or httpGet or tcpSocket

command:

- cat

- /usr/share/nginx/html/index.html

initialDelaySeconds: 5

periodSeconds: 10

failureThreshold: 2

successThreshold: 1

livenessProbe:

exec: #exec or httpGet or tcpSocket

command:

- cat

- /usr/share/nginx/html/index.html

initialDelaySeconds: 5

periodSeconds: 10

failureThreshold: 3

successThreshold: 1

---

apiVersion: v1

kind: Service

metadata:

name: nginx01

spec:

selector:

run: nginx01

ports:

- port: 8000

targetPort: 80

type: NodePort